

IBM IMS Fast Path Solution Pack for z/OS
2.1

*IMS High Performance Fast Path Utilities
User's Guide*



Note:

Before using this information and the product it supports, read the information in [“Notices” on page 1437.](#)

Third Edition (May 2021)

This edition applies to Version 2.1 of IBM IMS Fast Path Solution Pack for z/OS IMS High Performance Fast Path Utilities (program number 5698-FPP) and to any subsequent releases and modifications until otherwise indicated in new editions.

This edition replaces SC27-9536-01.

© **Copyright International Business Machines Corporation 1985, 2021.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

About this information..... xxv

Part 1. Overview of IMS HP Fast Path Utilities..... 1

- Chapter 1. What is IMS HP Fast Path Utilities?..... 3
 - What's new in IMS HP Fast Path Utilities..... 3
 - IMS HP Fast Path Utilities terminology..... 4
 - Managing IMS Data Entry Databases..... 7
 - Solutions to simplify and automate database maintenance tasks..... 9
 - IMS HP Fast Path Utilities features and benefits..... 11
 - IMS Fast Path Advanced Tool..... 11
 - IMS Fast Path Basic Tools..... 12
 - IMS Fast Path Online Tools..... 12
 - Online Space Management utility..... 12
 - Online Structure Change utility..... 13
 - Functional differences between the tools..... 13
 - Support for IMS Library Integrity Utilities..... 15
 - Support for IMS Tools Base and common tools..... 16
 - Support for IMS managed ACBs environment..... 18
 - Support for DEDB encryption..... 19
 - Service updates and support information..... 19
 - Functional changes..... 20
 - Functional changes in IMS HP Fast Path Utilities in IMS Fast Path Solution Pack 2.1..... 20
 - Functional changes in IMS HP Fast Path Utilities in IMS Fast Path Solution Pack 1.3..... 20
 - Functional changes in IMS HP Fast Path Utilities in IMS Fast Path Solution Pack 1.2..... 21
 - Functional changes in IMS HP Fast Path Utilities in IMS Fast Path Solution Pack 1.1..... 22
 - Functional changes in IMS High Performance Fast Path Utilities for z/OS 3.2..... 23
 - Functional changes in IMS High Performance Fast Path Utilities for z/OS 3.1..... 25
 - Compatibilities with former releases of the product and migration considerations..... 27
 - Product documentation and updates..... 28
 - IMS HP Fast Path Utilities accessibility features..... 30

Part 2. IMS Fast Path Advanced Tool..... 31

- Chapter 2. Overview of IMS Fast Path Advanced Tool..... 33
 - Functions of FPA..... 33
 - Benefits of FPA..... 35
 - Program overview of FPA..... 35
 - Restrictions of FPA..... 36
 - IMS Fast Path Advanced Tool compatibility..... 37
 - Compatibility between IMS Fast Path Advanced Tool and former releases of IMS Fast Path Advanced Tool..... 37
 - Compatibility between IMS Fast Path Advanced Tool and IMS Fast Path Basic Tools..... 37
 - Compatibility between IMS Fast Path Advanced Tool and IMS Fast Path Online Tools..... 38
- Chapter 3. Understanding FPA..... 39
 - Program structure of FPA..... 39
 - System flow of FPA..... 40
 - Example JCL statements for FPA 40
 - Checkpoint restart option 41

Scenario: Unloading all areas of a DEDB by using the checkpoint restart option.....	42
Scenario: Reloading all areas of a DEDB by using the checkpoint restart option.....	46
Keyword changes supported for restart mode.....	49
Chapter 4. Unloading a DEDB offline with FPA.....	51
Functions of Unload.....	51
Input and output for the Unload function	52
Running the Unload function.....	53
Considerations on using the Integrity Checker utility.....	54
Region size requirements for the Unload function.....	55
DD statements for the Unload function.....	56
Utility control statements for the Unload function.....	68
Scenario: Restructuring DEDB areas	72
Examples for the Unload function.....	73
Example 1: Unloading an area that is registered with DBRC.....	74
Example 2: Unloading all areas of a DEDB that is registered with DBRC.....	74
Example 3: Unloading multiple areas of a DEDB into one output unloaded segment record data set.....	75
Example 4: Unloading an area from the specified ADS of MADS area.....	76
Example 5: Unloading an area and changing the DEDB area definition.....	77
Example 6: Unloading all areas of a DEDB and changing some or all of the DEDB area definitions (IAREA=ALL, OAREA=ALL).....	77
Example 7: Unloading all areas of a DEDB and changing some or all of the DEDB area definitions (IAREA=ALL, OAREA=specific).....	78
Example 8: Unloading all areas of a DEDB and changing some or all of the DEDB area definitions (IAREA=specific, OAREA=ALL).....	79
Example 9: Unloading all areas of a DEDB and changing some or all of the DEDB area definitions (IAREA=specific, OAREA=specific).....	80
Example 10: Unloading all areas of a DEDB that are registered with DBRC and creating their image copies.....	81
Example 11: Unloading areas in user-specified format.....	82
Example 12: Unloading all areas of a DEDB that are registered with DBRC and creating their image copies with the checkpoint restart option.....	84
Example 13: Unloading multiple areas of a DEDB into one output unloaded segment record data set in SMS environment.....	84
Example 14: Unloading an area without activating the DBRC process.....	85
Example 15: Unloading an area and changing the DEDB area definition when IMS management of ACBs is enabled.....	86
Chapter 5. Reloading a DEDB offline with FPA.....	89
Functions of Reload.....	89
Input and output for the Reload function.....	90
Running the Reload function.....	90
Considerations on using the Integrity Checker utility.....	92
Region size requirements for the Reload function.....	93
DD statements for the Reload function.....	94
Utility control statements for the Reload function.....	108
Examples for the Reload function.....	115
Example 1: Reloading an area that is registered with DBRC.....	116
Example 2: Reloading all areas of a DEDB that are registered with DBRC.....	117
Example 3: Reloading multiple areas of a DEDB from one input unloaded segment record data set.....	117
Example 4: Reloading an area that is registered with DBRC as MADS area.....	118
Example 5: Reloading an area of which some of the parameters for the DEDB area definition are changed.....	119
Example 6: Reloading and analyzing an area that is registered with DBRC.....	120
Example 7: Reloading an area that is registered with DBRC by using the LOADCTL subcommand.....	120

Example 8: Reloading all areas by using the concatenated unloaded segment records data sets.....	121
Example 9: Reloading areas by using the concatenated unloaded segment records data sets.....	122
Example 10: Reloading an area that is registered with DBRC and creating its image copy.....	123
Example 11: Reloading an area from a user-specified format data set.....	124
Example 12: Reloading and analyzing an area and storing the sensor data in the IMS Tools KB repository.....	124
Example 13: Reloading multiple areas of a DEDB from one input unloaded segment record data set in SMS environment	125
Example 14: Reloading an area without activating the DBRC process.....	126
Example 15: Reloading an area of which some of the parameters for the DEDB area definition are changed when IMS management of ACBs is enabled.....	127
Chapter 6. Changing a DEDB offline with FPA.....	129
Functions of Change.....	129
Input and output for the Change function.....	130
Running the Change function.....	131
Considerations on using the Integrity Checker utility.....	132
Region size requirements for the Change function.....	133
DD statements for the Change function.....	134
Utility control statements for the Change function.....	149
Examples for the Change function.....	156
Example 1: Reorganizing an area that is registered with DBRC.....	157
Example 2: Restructuring all areas that are registered with DBRC.....	157
Example 3: Reorganizing multiple areas that are registered with DBRC while renaming the input ADS.....	158
Example 4: Reorganizing an area that is registered with DBRC and creating its image copy....	159
Example 5: Reorganizing an area, analyzing the reorganized area, and storing the sensor data in the IMS Tools KB repository.....	160
Chapter 7. Reorganizing a DEDB offline with FPA.....	163
Functions of Reorg.....	163
Reorganization of a ranked set of UOWs.....	163
The number of UOWs to be reorganized.....	165
Input and output for the Reorg function.....	166
Considerations for running the Reorg function.....	166
Recovery and restart.....	167
MADS considerations.....	167
Running the Reorg function.....	167
Considerations on using the Integrity Checker utility.....	169
Region size requirements for the Reorg function.....	169
DD statements for the Reorg function.....	170
Utility control statements for the Reorg function.....	183
Examples for the Reorg function.....	188
Example 1: Reorganizing all UOWs in an area.....	189
Example 2: Reorganizing UOWs that use IOVF CIs and generating an image copy.....	189
Example 3: Reorganizing some UOWs that satisfy one of the RBASEFS and the RDOVFFS conditions.....	190
Example 4: Reorganizing an area, analyzing the reorganized area, and storing the sensor data in the IMS Tools KB repository.....	191
Chapter 8. Analyzing a DEDB offline with FPA.....	193
Functions of Analyze.....	194
DEDB integrity verification.....	194
DEDB analysis and report.....	195
Secondary index database integrity verification.....	196
Input and output for the Analyze function.....	196
Running the Analyze function.....	197

Preprocess for the Analyze process: Initializing permanent data set.....	197
Running the Analyze process.....	198
Correcting integrity problems in DEDB areas.....	200
Correcting integrity problems in secondary index databases.....	200
Analyzing the complete analysis reports.....	200
Region size requirements for the Analyze function.....	200
DD statements for the Analyze function.....	203
Utility control statements for the Analyze function.....	218
Examples for the Analyze function.....	223
Example 1: Running checksum validation of pointers in an area.....	223
Example 2: Running cross-reference validation of pointers in an area.....	224
Example 3: Generating the analysis report of an area in the HPFRPTS data set.....	224
Example 4: Running checksum validation of pointers in an image copy.....	225
Example 5: Running cross-reference validation of pointers in all areas and validation of pointer segments in all secondary index databases.....	225
Example 6: Running cross-reference validation of pointers in an area and validation of pointer segments in all secondary index databases.....	226
Example 7: Running cross-reference validation of pointers in all areas and validation of pointer segments in a secondary index database.....	226
Example 8: Generating the analysis reports of an area and storing the sensor data in the IMS Tools KB repository.....	227
Chapter 9. Tuning a DEDB offline with FPA.....	229
Functions of Tune.....	229
Input and output for the Tune function.....	231
Tuning aid records.....	231
Running the Tune function.....	233
Region size requirements for the Tune function.....	236
DD statements for the Tune function.....	238
Utility control statements for the Tune function.....	247
Examples for the Tune function.....	253
Example 1: Evaluating the effect of the reorganization of a DEDB.....	254
Example 2: Simulating changes in area definitions.....	254
Example 3: Evaluating a randomizer.....	255
Example 4: Rerunning the simulation with different parameters.....	256
Example 5: Simulating changes in area definitions of online areas.....	257
Chapter 10. Extracting a DEDB offline with FPA.....	259
Functions of Extract.....	259
Segment selection.....	259
Output file format.....	260
Samples of SELECT and OUTPUT commands.....	260
Input and output for the Extract function.....	262
Running the Extract function.....	262
Region size requirements for the Extract function.....	263
DD statements for the Extract function.....	265
Utility control statements for the Extract function.....	272
Examples for the Extract function.....	276
Example 1: Extracting segments with selection criteria.....	277
Example 2: Changing the style of the segment records.....	277
Example 3: Creating a test database.....	278
Example 4: Extracting segments in user-defined format.....	279
Example 5: Extracting segments into designated output data sets.....	280
Chapter 11. Printing DMAC blocks in a DEDB offline with FPA.....	283
Functions of DMAC Print.....	283
Input and output for the DMAC Print function.....	283
Running the DMAC Print function.....	284

Region size requirements for the DMAC Print function.....	285
DD statements for the DMAC Print function.....	286
Utility control statements for the DMAC Print function.....	291
Examples for the DMAC Print function.....	292
Example 1: Printing a DMAC block in an area.....	293
Example 2: Printing DMAC blocks in all areas of a DEDB.....	293
Chapter 12. Building secondary index databases for a DEDB offline with FPA.....	295
Functions of Build Index.....	295
Input and output for the Build Index function.....	295
Build pointer segment records.....	296
Running the Build Index function.....	297
Region size requirements for the Build Index function.....	299
DD statements for the Build Index function.....	301
Utility control statements for the Build Index function.....	310
Examples for the Build Index function.....	314
Example 1: Building all the secondary index databases for the DEDB areas.....	315
Example 2: Building specific secondary index databases for the DEDB areas.....	315
Example 3: Building a secondary index database that contains only certain pointer segments.....	316
Example 4: Generating build pointer segment records and then concurrently building secondary indexes.....	317
Chapter 13. Synchronizing secondary index databases with a DEDB with FPA.....	321
Functions of Resync.....	321
Input and output for the Resync function.....	321
Resync work records.....	324
Running the Resync function.....	326
Synchronizing with the verify update method or the direct update method.....	328
Synchronizing with the two-step method.....	330
Scanning DEDB areas in area scan mode.....	331
Region size requirements for the Resync function.....	332
DD statements for the Resync function.....	333
Utility control statements for the Resync function.....	343
Examples for the Resync function.....	346
Example 1: Synchronization using the verify update method.....	346
Example 2: Synchronization using the two-step method.....	347
Example 3: Synchronization using the verify update method (running area scan and sort processes separately).....	349
Example 4: Synchronization using the verify update method (offline).....	352
Chapter 14. Reclaiming CIs in secondary index databases with FPA.....	355
Functions of Freespace Reclaim.....	355
Input and output for the Freespace Reclaim function.....	355
Running the Freespace Reclaim function.....	356
Region size requirements for the Freespace Reclaim function.....	357
DD statements for the Freespace Reclaim function.....	358
Utility control statements for the Freespace Reclaim function.....	363
Examples for the Freespace Reclaim function.....	364
Example 1: Reclaiming CIs in a secondary index database data set	365
Example 2: Reclaiming CIs in all secondary index database data sets of a DEDB and generating reports.....	365
Chapter 15. Initializing a DEDB offline with FPA.....	367
Functions of Initialize.....	367
Input and output for the Initialize function.....	367
Running the Initialize function.....	367
Region size requirements for the Initialize function.....	369

DD statements for the Initialize function.....	369
Utility control statements for the Initialize function.....	375
Examples for the Initialize function.....	377
Example 1: Initializing an area that is registered with DBRC.....	377
Example 2: Initializing multiple areas in SMS environment.....	378
Chapter 16. Command reference for FPA.....	381
Command set.....	381
FPA commands.....	382
ANALYZE command.....	382
CHANGE command.....	386
DMACPRT command.....	390
END command.....	391
EXTRACT command.....	391
GLOBAL command.....	394
INDEXBLD command (for the Build Index function).....	397
INDEXBLD command (for the Resync function).....	400
INDEXBLD command (for the Freespace Reclaim function).....	403
INITIALIZE command.....	404
RELOAD command.....	405
REORG command.....	409
TUNE command.....	413
UNLOAD command.....	415
FPA subcommands.....	419
ALLOCATE subcommand.....	419
AREADEF subcommand.....	421
BLDMFILECTL subcommand.....	422
FILECTL subcommand.....	425
IRENAME subcommand.....	428
LOADCTL subcommand.....	429
OUTPUT subcommand.....	431
REPORT subcommand.....	432
SELECT subcommand.....	433
SIMULATE subcommand.....	434
THRESHOLD subcommand.....	435
USER_RECORD subcommand.....	437
FPA keywords.....	438
ACBCHK keyword.....	438
ACBLIB keyword.....	439
ADXCFCGRP keyword.....	439
ALLOCONLY keyword.....	440
AMSOPT keyword.....	440
AREADSN keyword.....	441
AREASCAN keyword.....	442
AREA_INFORMATION_RECORD keyword.....	443
ATASKCTL keyword.....	443
ATTRIBUTE keyword.....	444
AVGREC keyword.....	445
BREAK keyword.....	446
BUILDDSNMASK keyword.....	446
BUILDMODE keyword.....	447
CISIZE keyword.....	448
CISNAP keyword.....	448
COMPRESS keyword.....	449
DATACLAS keyword.....	449
DBD keyword.....	450
DBRC keyword.....	451
DBREC_DISCARD_MSGLVL keyword.....	452

DBRLDIST keyword.....	452
DDNAME keyword.....	453
DISP keyword.....	453
DOVFFS_PCT keyword.....	455
DOVNULLPTR_MSGLVL keyword.....	455
DSNAME keyword.....	456
DUPKEYACT keyword.....	457
DUPKEYDSNMASK keyword.....	458
DUPKEYMAX keyword.....	458
EDITCOMP_SKIP keyword.....	459
EFORMAT keyword.....	460
EVERY keyword.....	461
EXITRTN keyword.....	461
EXPDT keyword.....	462
FIELDS keyword.....	463
FORMAT keyword.....	465
FS_DOVFIOVF keyword.....	466
FS_RAADOVF keyword.....	467
FS_RAADOVFIOVF keyword.....	467
FS_RAAIOVF keyword.....	468
FS_RECLAIM_REPORT keyword.....	468
HISTORYDD keyword.....	469
IAREA keyword.....	469
ICACHE keyword.....	472
ICHECK keyword.....	473
ICINCOMP keyword.....	474
ICNOTIFY keyword.....	475
ICOUT keyword.....	476
IDSNMASK keyword.....	476
IDXDUMP keyword.....	477
IDXMAXERROR keyword.....	478
IDXPROC keyword.....	478
IDXTASKCTL keyword.....	479
IEXITRTN keyword.....	479
IMSCATACB_INPUT keyword.....	480
IMSCATACB_OUTPUT keyword.....	481
IMSCATHLQ keyword.....	481
IMSID keyword.....	482
INDEXDBD keyword.....	482
INSERT_LIMIT_COUNT keyword.....	484
IOVFCI_RBA_REPORT keyword.....	484
IOVFFS_PCT keyword.....	485
IOVFUSED_PCT keyword.....	485
ITASKCTL keyword.....	486
ITKBDATA keyword.....	487
ITKBLOAD keyword.....	487
ITKBSRVR keyword.....	488
KEYCHGCHK keyword.....	489
KEYLABEL keyword.....	489
KEYSEQCHK keyword.....	490
LARGEREC_COUNT keyword.....	491
LARGEREC_REPORT keyword.....	491
LIKE keyword.....	492
LIUGOPT keyword.....	493
LOCATION keyword.....	494
LRECL keyword.....	494
MAXERROR keyword.....	495
MAXFAILS keyword.....	496

MGMTCLAS keyword.....	496
MSTIMESTAMP keyword.....	497
OAREA keyword.....	497
OCACHE keyword.....	499
OCHECK keyword.....	500
ODSNMASK keyword.....	500
OEXITRTN keyword.....	502
OTASKCTL keyword.....	503
PAD keyword.....	504
PROC_AFT_DUPKEY keyword.....	504
PTRCHK keyword.....	505
PTRCHKLVL keyword.....	506
RAAFS_PCT keyword.....	507
RAP_LONGSYN_PCT keyword.....	507
RAPOVFL_PCT keyword.....	508
RBASEFS keyword.....	509
RDOVFFS keyword.....	509
RECIOVF_PCT keyword.....	510
RECIO_AVG keyword.....	510
RECIO_MAX keyword.....	511
RECOVOPT keyword.....	511
RELOCATE_SDEP keyword.....	512
REORGLIM keyword.....	513
REPORTDD keyword.....	513
RESTART keyword.....	514
RESTART_COUNT keyword.....	515
RESYNCDSNMASK keyword.....	515
RESYNCDUMP keyword.....	516
RESYNCDUMPMAX keyword.....	517
RESYNCMODE keyword.....	517
RETPD keyword.....	519
RFSSCAN keyword.....	519
RFSSTAT keyword.....	520
RMOD keyword.....	521
RMODLIB keyword.....	521
ROOT keyword.....	522
ROOTIO_AVG keyword.....	522
ROOTIO_MAX keyword.....	523
SCAN keyword.....	523
SDEP keyword.....	524
SDEP_BYPASS_MSGLVL keyword.....	525
SDEPCHKLVL keyword.....	526
SDEP_ORPHAN_MSGLVL keyword.....	527
SDEP_PTRIOVF_MSGLVL keyword.....	528
SEGLDIST keyword.....	529
SEGLSTAT keyword.....	529
SEGMENT keyword.....	530
SEGREC_PREFIX keyword.....	531
SEGREC_SUFFIX keyword.....	533
SENSOR keyword.....	535
SENSOR_DBREC keyword.....	536
SENSOR_PTRERR keyword.....	537
SENSOR_UOW keyword.....	537
SENSOR_UOW_REPORT keyword.....	538
SITE_DSNAME keyword.....	539
SITE_MEMBER keyword.....	540
SKIP keyword.....	540
SORT keyword.....	541

SORTOPT_FILSZ keyword.....	542
SPACE keyword.....	543
SSP keyword.....	544
STARTUOW keyword.....	545
STOPAFT keyword.....	545
STOPUOW keyword.....	546
STORCLAS keyword.....	546
SYNC keyword.....	547
SYN_LEN_AVG keyword.....	547
SYN_MAX keyword.....	548
SYN_PCT_LEN keyword.....	548
TARDSNMASK keyword.....	549
TOICTL keyword.....	550
TOINUM keyword.....	551
TOITIME keyword.....	551
TOIXCFGRP keyword.....	552
UNIT keyword.....	552
UOW keyword.....	553
UOWDOVF_PCT keyword.....	553
UOWIOVF_AVG keyword.....	554
UOWIOVF_MAX keyword.....	554
UOWIOVF_PCT keyword.....	555
UOW_REPORT keyword.....	555
VOLCNT keyword.....	556
VOLSER keyword.....	556
WHERE keyword.....	557
Chapter 17. Sample reports of FPA.....	559
Reports generated by each process of FPA.....	560
Audit report.....	564
Processing report.....	565
DBD Definition report.....	569
Unloaded report.....	575
Reloaded report.....	578
Output File report.....	580
Input File report.....	583
Resolve Pointers report.....	585
Pointer Checksum report.....	586
Pointer Chain Reconstruction report.....	587
CI Map/CI Dump report.....	588
Process Summary of Analysis report.....	590
DEDB Area Analysis reports.....	591
Freespace Analysis report.....	592
DB Record Profile Analysis report.....	596
DB Record Placement Analysis report.....	599
Segment Placement Analysis report.....	600
Segment I/O Analysis report.....	600
Synonym Chain Analysis report.....	601
Largest Database Records report	603
Consolidated Largest Database Records report.....	604
UOW report.....	606
UOW Exception report.....	607
Image Copy Output report.....	608
Extracted report.....	609
DMACPRT report.....	612
Reorganization Processing Summary report.....	615
UOW Space Statistics report.....	616
Secondary Index Definition report.....	617

Secondary Index Processing report.....	620
Secondary Index Resync report.....	623
Secondary Index Analysis report.....	626
Secondary Index Duplicate Key report.....	627
Resync Pointer Segment Dump report.....	629
Secondary Index Freespace Reclaim report.....	630
Result of Access Method Services report.....	631
Initialization Processing Summary report.....	632
Pointer Segment Dump report.....	633
Site and System Default report.....	634
Process Summary report	636
Sensor Data Statistics report.....	637
Segment Length Distribution report.....	645
DB Record Length Distribution report.....	647
Area Tuning Summary report.....	649
Tuning Aid Record Data Set report.....	652
TUNED reports.....	653
RBA of Used IOVF CI report.....	656
Chapter 18. Site default options support for FPA.....	657
Supported keywords for the site default option.....	657
Enabling site default values.....	658
Registering and using site defaults in a site default module (HFPCZZ01).....	658
Registering and using site defaults in a site default table.....	661
Part 3. IMS Fast Path Online Tools.....	667
Chapter 19. Overview of IMS Fast Path Online Tools.....	669
Functions of FPO.....	669
Benefits of FPO.....	669
Program overview of FPO.....	671
IMS Fast Path Online Tools compatibility.....	672
Compatibility between former releases of IMS Fast Path Online Tools.....	672
Compatibility between IMS Fast Path Advanced Tool and IMS Fast Path Online Tools.....	672
Chapter 20. Analyzing a DEDB online with FPO.....	673
Functions of OPC.....	674
DEDB integrity verification.....	674
DEDB analysis and report.....	676
Root addressable area and IOVF processing	676
Input and output for OPC.....	677
Considerations for running OPC.....	679
Selecting the operating modes.....	679
Data sets required for each operating mode.....	680
Performance of OPC.....	682
DBRC SCI registration exit routine.....	682
Running OPC.....	682
Preprocess for OPC: Initializing permanent data set.....	682
Running the DEDB integrity verification process.....	684
Correcting integrity problems in DEDB areas.....	684
Analyzing the complete analysis reports.....	685
Region size requirements for OPC.....	685
DD statements for OPC.....	685
Option control statement for OPC.....	695
Utility control statements for OPC.....	696
Output for OPC.....	702
FPXRPTS DD data set.....	702

FPXSPRT DD data set.....	710
SYSPRINT/FPXMFILE DD data set	718
SNAPPIT DD data set	718
Examples for OPC.....	720
Example 1: Running fast scan of database integrity.....	720
Example 2: Running fast scan of database integrity and collecting pointer verification data ..	721
Example 3: Running cross-reference validation of pointers	722
Example 4: Running cross-reference validation of pointers and generating analysis reports...	722
Example 5: Collecting segment information records for database tuning.....	723
Example 6: Processing multiple areas and taking image copies.....	727
Example 7: Running cross-reference validation of pointers, generating analysis reports, and storing sensor data in the IMS Tools KB repository.....	727
Example 8: Generating reports that contain SDEP record information, prefix length, and data flags.....	728
Chapter 21. Extracting a DEDB online with FPO.....	729
Functions of ODE.....	729
Segment selection.....	729
Segment output.....	730
Run-level versus area-level control.....	730
Restrictions of ODE.....	731
Input and output for ODE.....	731
Considerations for running ODE.....	732
Running ODE.....	732
Region size requirements for ODE.....	733
DD statements for ODE.....	733
Option control statement for ODE.....	736
Utility control statements for ODE.....	737
Output for ODE.....	741
FPXRPTS DD data set.....	741
FPXOFFILE DD data set.....	742
SYSPRINT/FPXMFILE DD data set.....	742
Examples for ODE.....	744
Example 1: Running ODE with run-level control.....	744
Example 2: Running ODE with area-level control.....	745
Chapter 22. Extending a DEDB online with FPO.....	747
Functions of OAE.....	747
Restrictions of OAE.....	748
Input and output for OAE.....	748
Considerations for running OAE.....	749
Considerations on area extension.....	749
Considerations on running OAE.....	752
Considerations on image copy extension.....	752
Considerations on recovery and restart.....	753
APF authorization.....	754
DBRC SCI registration exit routine.....	754
Performance considerations.....	754
Running OAE.....	754
Step 1: Preparatory steps for OAE.....	754
Step 2: Coding and running JCL.....	756
Step 3: Post execution steps for OAE.....	757
Recovering and restarting OAE.....	758
Region size requirements for OAE.....	760
DD statements for OAE.....	761
Option control statement for OAE.....	763
Utility control statements for OAE.....	764
SYSIN DD data set (Shadow expansion).....	764

FPXIN DD data set (Original expansion).....	765
Output for OAE.....	767
SYSPRINT DD data set.....	768
FPXBIC01 and FPXBIC02 DD data sets	770
OAE Rename program.....	771
Running the OAE Rename program.....	771
DD statements for the OAE Rename program.....	771
Utility control statements for the OAE Rename program.....	771
Examples for OAE.....	772
Examples: Original expansion.....	772
Examples: Shadow expansion.....	776
Chapter 23. Reorganizing a DEDB online with FPO.....	779
Functions of OER.....	779
Reorganization of a ranked set of UOWs.....	779
The number of UOWs to be reorganized.....	781
Reports for reorganization result.....	782
Pointer check during online reorganization.....	782
Input and output for OER.....	782
Running OER.....	784
Region size requirements for OER.....	785
DD statements for OER.....	785
Option control statement for OER.....	795
Utility control statements for OER.....	796
Output for OER.....	803
FPXRPTS DD data set.....	803
FPXSPRT DD data set.....	808
SYSPRINT DD data set.....	816
FPXMFIL DD data set.....	816
Examples for OER.....	817
Example 1: Running OER without OPC.....	817
Example 2: Running OER with OPC.....	818
Example 3: Running OER with OPC for multiple areas and taking image copies.....	819
Example 4: Running OER with OPC and storing sensor data in the IMS Tools KB repository....	819
Example 5: Running OER with the segment shunting function.....	820
Chapter 24. Printing DMAC blocks in a DEDB online with FPO.....	821
Functions of ODM.....	821
Input and output for ODM.....	821
Running ODM.....	822
Region size requirements for ODM.....	823
DD statements for ODM.....	823
Option control statement for ODM.....	824
Utility control statement for ODM.....	824
Output for ODM.....	826
FPXRPTS DD data set.....	826
SYSPRINT DD data set.....	831
FPXMFIL DD data set.....	832
Examples for ODM.....	832
Example 1: Reporting only DMAC information in IMS storage.....	832
Example 2: Reporting two kinds of DMAC information	833
Chapter 25. Keyword reference for FPO.....	835
Syntax rules.....	835
Syntax rules for FPXIN.....	835
Syntax rules for DASYSIN.....	836
Syntax rules for FPXOPTS.....	836
Syntax rules for FPXCTL and <i>areaname</i>	836

Syntax rules for the OAE Rename program.....	837
FPO control statements.....	837
ANALYZE control statement.....	837
DMACPRT control statement.....	841
EXTEND control statement.....	842
EXTRACT control statement.....	843
FPXCTL control statement.....	844
GLOBAL control statement.....	847
OUTPUT control statement.....	848
OVERRIDE control statement.....	850
PTRCHK control statement.....	850
REORG control statement.....	854
REPORT control statement.....	856
SELECT control statement.....	857
THRESHOLD control statement.....	859
USER_RECORD control statement.....	861
FPO keywords	861
ADXCFGRP keyword.....	861
AFTERTIME keyword.....	862
ANDFIELD keyword.....	862
ARDIGIT keyword.....	863
AREA keyword.....	864
BREAK keyword.....	865
CASE keyword.....	866
CHKUOW keyword.....	866
CIPRT keyword.....	867
DATACLAS keyword.....	867
DBRC keyword (for GLOBAL).....	868
DBRLDIST keyword.....	868
DDNAME keyword.....	869
DOVFFS_PCT keyword.....	869
DOVNULLPTR_MSGLVL keyword	869
DSNAME keyword.....	870
ERRORACT keyword.....	870
EVERY keyword.....	871
EXC_RBASEFS keyword.....	872
EXC_RDOVFFS keyword.....	873
EXC_UDOVFFS keyword.....	873
EXPAND keyword.....	874
EXTEND_TYPE keyword.....	875
FIELD keyword.....	875
FIELDS keyword (for OUTPUT).....	876
FIELDS keyword (for USER_RECORD).....	877
FPBP64U keyword.....	879
FS_DOVFIOVF keyword.....	880
FS_RAADOVF keyword.....	880
FS_RAADOVFIOVF keyword.....	881
FS_RAAIOVF keyword.....	881
FULLSTEP keyword.....	882
HISTORYDD keyword.....	882
IC keyword.....	883
ICEXTEND keyword.....	884
ICERR_CONT keyword.....	884
ICNOTIFY keyword.....	885
IC1OUTDD keyword.....	886
IC2OUTDD keyword.....	886
INDOUBT keyword.....	887
IOVFCI_RBA_REPORT keyword.....	887

IOVFFS_PCT keyword.....	887
IOVFPOOL keyword.....	888
IOVFUSED_PCT keyword.....	888
IOVF_ROOT keyword.....	889
IOVF_UOW keyword.....	889
ITKBDATA keyword.....	890
ITKBLOAD keyword.....	890
ITKBSRVR keyword.....	891
KEYWORD_DUPCHK keyword.....	892
LARGEREC_COUNT keyword.....	893
LARGEREC_REPORT keyword.....	893
MAXBUFFS keyword.....	894
MAXERROR keyword.....	894
MAXFAILS keyword.....	895
MAXTWINCTR keyword.....	895
MAXVOLS keyword.....	896
MFILE keyword.....	896
MGMTCLAS keyword.....	897
MSG_LIM keyword.....	897
MSG_NUM keyword.....	898
OFFILE keyword.....	898
OLRECL keyword.....	899
OUTPUT keyword.....	899
PCHKONLY keyword.....	901
PROC_AFT_ERROR keyword.....	902
PTRCHK keyword.....	902
RAAFS_PCT keyword.....	903
RAPOVFL_PCT keyword.....	903
RBASEFS keyword.....	904
RCEXLOAD keyword.....	904
RDOVFFS keyword.....	905
RECIO_AVG keyword.....	905
RECIO_MAX keyword.....	906
RECIOVF_PCT keyword.....	906
REORGLIM keyword.....	906
REORGMSG keyword.....	907
REPORTDD keyword.....	907
RFILE keyword.....	908
RFSSCAN keyword.....	909
RFSSTAT keyword.....	909
RGROUP keyword.....	910
ROOTIO_AVG keyword.....	910
ROOTIO_MAX keyword.....	911
RUNTYPE keyword.....	911
SDEP keyword.....	912
SDEP_BYPASS_MSGLVL keyword.....	912
SDEP_CI keyword.....	913
SDEPHIER keyword.....	913
SDEP_ORPHAN_MSGLVL keyword.....	914
SDEP_PTRIOVF_MSGLVL keyword.....	915
SEG keyword.....	916
SEGLDIST keyword.....	918
SEGLSTAT keyword.....	918
SEGREC_PREFIX keyword.....	919
SEGREC_SUFFIX keyword.....	920
SENSOR keyword.....	922
SENSOR_DBREC keyword.....	923
SENSOR_PTRERR keyword.....	924

SENSOR_UOW keyword.....	925
SENSOR_UOW_REPORT keyword.....	925
SKIP keyword.....	926
SORT keyword.....	927
STARTUOW keyword.....	928
STAT keyword.....	928
STOPAFT keyword.....	929
STOPUOW keyword.....	930
STORCLAS keyword.....	930
SYN_LEN_AVG keyword.....	930
SYN_MAX keyword.....	931
SYN_PCT_LEN keyword.....	931
SYNTBL keyword.....	932
TARDSNMASK keyword.....	932
TARECGEN keyword.....	933
TIMECHECK keyword.....	934
TOITIME keyword.....	935
TOIXCFGR keyword.....	935
TYPRUN keyword.....	935
UOWDOVF_PCT keyword.....	937
UOWIOVF_AVG keyword.....	937
UOWIOVF_MAX keyword.....	938
UOWIOVF_PCT keyword.....	938
UOW_REPORT keyword.....	939
VOLUME keyword.....	939
VOLUME_EXT keyword.....	939
Chapter 26. FPO reports stored in IMS Tools KB.....	941
Chapter 27. Site default options support for FPO.....	943
Supported keywords for the site default option.....	943
How keyword values are determined.....	943
Setting site default values for FPO.....	944
Creating configuration member FPXCFG00.....	944
Setting members for site default options.....	945
Chapter 28. Return code user exit routine (FPXRCEX0).....	947
Chapter 29. RACF considerations for FPO.....	951
Part 4. Online Space Management utility.....	953
Chapter 30. Online Space Management utility overview.....	955
Online Space Management utility functions.....	955
Online Space Management utility benefits.....	956
Online Space Management utility input and output.....	956
Chapter 31. Using the Online Space Management utility in stand-alone mode.....	959
Running the Online Space Management utility job.....	959
Sample scenario for using the Online Space Management utility in stand-alone mode.....	960
Chapter 32. Using the Online Space Management utility in the active autonomics environment.....	963
Online Space Management utility in the active autonomics environment.....	963
Configuring the active autonomics environment.....	964
Verifying successful completion of Online Space Management utility jobs.....	968
Chapter 33. JCL requirements for the Online Space Management utility.....	971

Chapter 34. Utility control statements for the Online Space Management utility.....	977
GLOBAL command keywords.....	980
ADXCFGRP keyword.....	980
ICNOTIFY keyword.....	981
ICOUT keyword.....	981
IMSID keyword.....	982
ITKBDATA keyword.....	982
ITKBSRVR keyword.....	983
SCAN keyword.....	983
MANAGE command keywords.....	984
ACTION keyword.....	984
DBD keyword.....	985
IAREA keyword.....	985
REORGATIOVFEXT keyword.....	986
SCANDELT keyword.....	986
SETEXTENDPARM subcommand keywords.....	987
ACBLIB keyword.....	987
DATACLAS keyword.....	988
DDNAME keyword.....	988
DSNAME keyword.....	989
EXTEND_TYPE keyword.....	989
IOVF_UOW keyword.....	990
MAXVOLS keyword.....	990
MGMTCLAS keyword.....	990
SDEP_CI keyword.....	991
STORCLAS keyword.....	991
VOLUME keyword.....	992
VOLUME_EXT keyword.....	992
SETREORGPARM subcommand keywords.....	992
ERRORACT keyword.....	993
FPBP64U keyword.....	994
MAXBUFFS keyword.....	994
MAXFAILS keyword.....	995
RBASEFS keyword.....	995
RDOVFFS keyword.....	996
REORGLIM keyword.....	997
REORGMSG keyword.....	997
RFSSCAN keyword.....	998
RFSSTAT keyword.....	998
 Chapter 35. Online Space Management utility reports.....	 1001
Reports in HFPRPTS DD.....	1001
Reports in FPXRPTS DD.....	1004
Reports in FPXSPRT DD.....	1004

Part 5. Online Structure Change utility..... 1005

Chapter 36. Online Structure Change utility overview.....	1007
Online Structure Change utility benefits.....	1007
Online Structure Change utility functions.....	1007
Online Structure Change utility internal jobs.....	1010
Online Structure Change utility restrictions.....	1013
Online Structure Change utility considerations.....	1014
 Chapter 37. Restructuring an online DEDB.....	 1019
Running the Online Structure Change utility.....	1019

Region size requirements for the Online Structure Change utility.....	1021
Setting site default values for the Online Structure Change utility.....	1022
Troubleshooting Online Structure Change utility problems.....	1023
Chapter 38. JCL requirements for the Online Structure Change utility.....	1027
Chapter 39. Utility control statements for the Online Structure Change utility.....	1031
GLOBAL command keywords.....	1036
IMSCATHLQ keyword.....	1036
IMSID keyword.....	1037
ITKBSRVR keyword.....	1037
MSTIMESTAMP keyword.....	1037
SCAN keyword.....	1038
TOIXCFGRP keyword.....	1038
RESTRUCTURE command keywords.....	1039
ANALYZE keyword.....	1040
DBD keyword.....	1040
ERRORACT keyword.....	1041
IAREA keyword.....	1041
ITASKCTL keyword.....	1042
OAREA keyword.....	1042
ORIGADS_PROC keyword.....	1043
OTASKCTL keyword.....	1044
PAD keyword.....	1044
REPLRAND_AREALVL keyword.....	1045
RUNTYPE keyword.....	1045
SKIP_OLC keyword.....	1046
TAKEOVER_TIME keyword.....	1046
UPDATE_NUMBER keyword.....	1046
USR_MAX keyword.....	1047
ALLOCATE subcommand keywords.....	1047
ATTRIBUTE keyword.....	1048
OAREA keyword.....	1048
JCLPARM subcommand keywords.....	1049
COMPRESS keyword.....	1052
CONFIGLIB_DSN keyword.....	1052
DATACOMP_SUFFIX keyword.....	1052
DBDLIB_DSN keyword.....	1053
DBRCGRP keyword.....	1053
DBRCGRP_APPLY keyword.....	1054
DBR_NOFEOV keyword.....	1054
DBR_NOFEOV_APPLY keyword.....	1054
DCLASS_TEMP keyword.....	1055
DDNPREFIX_NEW keyword.....	1055
DDNPREFIX_SHADOW keyword.....	1056
DDNPREFIX_UIC keyword.....	1056
DSNMASK_NEW keyword.....	1056
DSNMASK_SHADOW keyword.....	1057
DSNMASK_UIC keyword.....	1057
ENQDSN keyword.....	1058
FPLIB_DSN keyword.....	1058
IMSID_APPLY keyword.....	1059
IMSLIB_DSN keyword.....	1059
IMSLIB_DSN_APPLY keyword.....	1059
IMSPLEX keyword.....	1060
IMSPLEX_APPLY keyword.....	1060
IMSRESLB_EDC_DSN keyword.....	1060
IMSRESLB_RND_DSN keyword.....	1061

INTRDR keyword.....	1061
JCLDSN keyword.....	1061
JOBCARDx keyword.....	1062
JOBN_PREFIX keyword.....	1062
LOGDSN keyword.....	1063
MCLASS_TEMP keyword.....	1063
NBA_APPLY keyword.....	1064
NEWDBDLIB_DSN keyword.....	1064
NEWRESLB_EDC_DSN keyword.....	1064
NEWRESLB_RND_DSN keyword.....	1065
OBA_APPLY keyword.....	1065
PROCLIB_DSN keyword.....	1066
PROCLIB_DSN_APPLY keyword.....	1066
PSBNAME_APPLY keyword.....	1066
RECONx_DSN keyword.....	1067
RECONx_DSN_APPLY keyword.....	1067
SCLASS_TEMP keyword.....	1067
SGLXLOAD keyword.....	1068
SPACE keyword.....	1068
SPOC_WAITTIME keyword.....	1069
SYSAFF keyword.....	1069
SYSAFF_APPLY keyword.....	1070
TOIXCFGRP_APPLY keyword.....	1070
TOSILOAD keyword.....	1070
UNIT_TEMP keyword.....	1071
USRHLQ keyword.....	1071
LOADCTL subcommand keywords.....	1071
INSERT_LIMIT_COUNT keyword.....	1072
LOCATION keyword.....	1072
SEGMENT keyword.....	1073
Chapter 40. Online Structure Change utility reports.....	1075
Audit report.....	1075
Processing report.....	1077
DBD Definition report.....	1079
Base Configuration report.....	1082
OSC Internal Jobs Scenario report.....	1082
Apply report.....	1083
Chapter 41. Examples for the Online Structure Change utility.....	1085
Example: Restructuring an online DEDB.....	1085
Example: Specifying input and output areas.....	1086
Example: Specifying the ALLOCATE subcommand.....	1087

Part 6. References..... 1089

Chapter 42. Performance tips and best practices for using IMS HP Fast Path Utilities.....	1091
Tips and best practices for IMS Fast Path Advanced Tool.....	1091
Tips and best practices for the FPA Unload function.....	1091
Tips and best practices for the FPA Reload function.....	1093
Tips and best practices for the FPA Change function.....	1094
Tips and best practices for the FPA Reorg function.....	1096
Tips and best practices for the FPA Analyze function.....	1098
Tips and best practices for the FPA Tune function.....	1099
Tips and best practices for the FPA Extract function.....	1100
Tips and best practices for the FPA DMAC Print function.....	1101
Tips and best practices for the FPA Build Index function.....	1102

Tips and best practices for the FPA Resync function.....	1103
Tips and best practices for the FPA Freespace Reclaim function.....	1105
Tips and best practices for the FPA Initialize function.....	1105
Tips and best practices for IMS Fast Path Online Tools.....	1106
Tips and best practices for the FPO Online Pointer Checker utility.....	1106
Tips and best practices for the FPO Online Data Extract utility.....	1107
Tips and best practices for the FPO Online Area Extender utility.....	1107
Tips and best practices for the FPO Online Expert Reorganization utility.....	1108
Tips and best practices for the FPO Online DMAC Print utility.....	1109
Tips and best practices for the Online Space Management utility.....	1109
Tips and best practices for the Online Structure Change utility.....	1110
Chapter 43. DEDB monitoring and tuning guide.....	1111
Summary of DEDB monitoring and tuning steps.....	1111
Collecting and storing statistical data for DEDB areas.....	1112
Storing data in the statistics history data set (History file).....	1113
Storing data in the statistics history data set (History2 file).....	1115
Storing data in the Sensor Data repository of IMS Tools KB.....	1117
Detecting exceptional states of DEDB areas.....	1120
Comparison of the exception detection methods.....	1120
Method 1: Using thresholds in FPA Analyze process.....	1123
Method 2: Using thresholds in DEDB Pointer Checker.....	1123
Method 3: Using Policy Services.....	1124
Method 4: Using thresholds in FPA Tune process.....	1125
Determining the actions to resolve the detected exceptions.....	1125
Generating the analysis reports.....	1126
JCL examples for monitoring and tuning DEDBs.....	1129
FPA Analyze JCL examples for monitoring and tuning DEDBs.....	1129
FPA Reload JCL examples for monitoring and tuning DEDBs.....	1135
FPA Change JCL examples for monitoring and tuning DEDBs.....	1141
FPA Reorg JCL examples for monitoring and tuning DEDBs.....	1148
DEDB Pointer Checker JCL examples for monitoring and tuning DEDBs.....	1154
Online Pointer Checker JCL examples for monitoring and tuning DEDBs.....	1157
Online Expert Reorganization JCL examples for monitoring and tuning DEDBs.....	1162
IMS HP Image Copy JCL examples for monitoring and tuning DEDBs.....	1167
Chapter 44. Viewing the reports stored in IMS Tools KB.....	1171
Viewing DEDB area space utilization trend.....	1171
Viewing specific DEDB management information.....	1171
Chapter 45. Record layouts.....	1173
Historical records layout.....	1173
History file records layout.....	1173
History2 file records layout.....	1175
Largest database records layout.....	1176
Unloaded segment records layout.....	1177
Total segments records layout.....	1183
Sequence error segment records layout.....	1183
DURDBDFN records layout.....	1185
ERRORSEG records layout.....	1185
Extracted records layout.....	1186
OUTFILE records layout.....	1190
UOW group definition record layout.....	1191
Build pointer segment records layout.....	1193
Resync pointer segment records and duplicate pointer segment records layout.....	1193
Chapter 46. Exit routine option and its interface.....	1197
Exit routine option overview.....	1197

Exit routine interface.....	1197
Exit routine FABRPCX.....	1199
Chapter 47. Segment edit/compression facility support.....	1201
Chapter 48. Determining how to process SDEP segments.....	1205
Chapter 49. How to read syntax diagrams.....	1207
Part 7. Troubleshooting.....	1209
Chapter 50. Messages and codes of IMS HP Fast Path Utilities.....	1211
Return codes.....	1211
Return codes of FPA.....	1211
Return codes of FPO.....	1211
Return codes of OSM.....	1212
Return codes of diagnostics aid (FABADIAG).....	1212
Abend codes.....	1212
Abend codes of FPA.....	1212
Abend codes of FPO.....	1212
Abend codes of OSM.....	1218
Messages.....	1218
FABU messages.....	1220
FPX messages.....	1223
HFP0 messages.....	1287
HFP1 messages.....	1294
HFP2 messages.....	1296
HFP4 messages.....	1299
HFPA messages.....	1301
HFPA messages.....	1308
HFPC messages.....	1315
HFPA messages.....	1340
HFPA messages.....	1343
HFPA message.....	1352
HFPA message.....	1355
HFPA messages.....	1355
HFPA messages.....	1357
HFPA messages.....	1366
HFPA messages.....	1372
HFPA messages.....	1373
HFPA messages.....	1377
HFPA messages.....	1389
HFPA messages.....	1391
HFPA messages.....	1392
HFPA messages.....	1394
HFPA messages.....	1398
HFPA messages.....	1405
HFPA messages.....	1409
HFPA messages.....	1413
Resolve Pointers report messages.....	1416
Process Summary of Analysis report messages.....	1422
Online Pointer Scan report messages.....	1425
Gathering diagnostic information.....	1431
Chapter 51. Diagnostics aid.....	1433
How to run diagnostics aid with JCL.....	1433
APAR status reports.....	1434

Load Module APAR Status report.....	1434
Macro APAR Status report.....	1434
Notices.....	1437
Index.....	1441

About this information

IBM® IMS Fast Path Solution Pack for z/OS® IMS High Performance Fast Path Utilities (also referred to as IMS HP Fast Path Utilities) improves performance and availability by streamlining database administrator (DBA) tasks.

To use the procedures in this information, you must first install IMS HP Fast Path Utilities as described in the *Program Directory for IMS Fast Path Solution Pack for z/OS 2.1, GI13-5905*, and then perform the post-installation steps as described in the *IMS Fast Path Solution Pack: Overview and Customization, GC27-9596*.

These topics are designed for database administrators and technical support personnel who are involved in database management, maintenance, and performance tuning, and require a knowledge of how to operate IMS HP Fast Path Utilities, and are specifically for those who manage the IMS Data Entry Databases (DEDBs). These topics help database administrators and technical support personnel to perform these tasks:

- Understand the functions of IMS HP Fast Path Utilities
- Run and use IMS HP Fast Path Utilities after it is installed
- Use DD statements to control how you use IMS HP Fast Path Utilities

IMS HP Fast Path Utilities includes all the features you need to manage your IMS Fast Path databases.

For information about IMS Fast Path Basic Tools and supplementary utilities, see the following information:

- *IMS Fast Path Solution Pack: IMS Fast Path Basic Tools User's Guide*
- *IMS Fast Path Solution Pack: Supplementary Utilities User's Guide*

To use these topics, you should have a working knowledge of:

- The z/OS operating system
- ISPF
- SMP/E

Always check the IMS Tools Product Documentation page for complete product documentation resources:

<https://www.ibm.com/support/pages/node/712955>

The IMS Tools Product Documentation page includes:

- Links to [IBM Documentation](#) for the user guides ("HTML")
- Links to the PDF versions of the user guides ("PDF")
- Program Directories for IMS Tools products
- Recent updates to the user guide, known as "Tech docs" ("See updates to this information!")
- Technical notes from IBM Software Support, known as "Tech notes"
- White papers that describe product business scenarios and solutions

Part 1. Overview of IMS HP Fast Path Utilities

The following topic introduces the tools of IBM IMS Fast Path Solution Pack for z/OS IMS High Performance Fast Path Utilities (also referred to as IMS HP Fast Path Utilities).

Topics:

- [Chapter 1, “What is IMS HP Fast Path Utilities?” on page 3](#)

Chapter 1. What is IMS HP Fast Path Utilities?

IBM IMS Fast Path Solution Pack for z/OS IMS High Performance Fast Path Utilities (also referred to as IMS HP Fast Path Utilities) provides extensive and easy-to-use utilities that include all the functions that you need to analyze, maintain, and tune IMS Data Entry Databases (DEDBs). It also provides features that can help increase system availability by enabling the operation of various key functions without taking your IMS databases offline.

IMS HP Fast Path Utilities includes the following main tools:

- IMS Fast Path Advanced Tool (FPA)
- IMS Fast Path Basic Tools (FPB)
- IMS Fast Path Online Tools (FPO)
- Online Space Management utility (OSM)
- Online Structure Change utility (OSC)

In addition to these tools and utilities, IMS HP Fast Path Utilities provides supplementary utilities.

Topics:

- [“What's new in IMS HP Fast Path Utilities” on page 3](#)
- [“IMS HP Fast Path Utilities terminology” on page 4](#)
- [“Managing IMS Data Entry Databases” on page 7](#)
- [“Solutions to simplify and automate database maintenance tasks” on page 9](#)
- [“IMS HP Fast Path Utilities features and benefits” on page 11](#)
- [“Support for IMS Library Integrity Utilities” on page 15](#)
- [“Support for IMS Tools Base and common tools” on page 16](#)
- [“Support for IMS managed ACBs environment” on page 18](#)
- [“Support for DEDB encryption” on page 19](#)
- [“Service updates and support information” on page 19](#)
- [“Functional changes” on page 20](#)
- [“Compatibilities with former releases of the product and migration considerations” on page 27](#)
- [“Product documentation and updates” on page 28](#)
- [“IMS HP Fast Path Utilities accessibility features” on page 30](#)

What's new in IMS HP Fast Path Utilities

IMS HP Fast Path Utilities of IMS Fast Path Solution Pack 2.1 provides the following new functions.

New and changed information is indicated by a vertical bar (|) to the left of a change. Editorial changes that have no technical significance are not noted.

SC27-9536-02 (May 2021)

Description	Related APARs
Resolve Pointers report message “0124” on page 1419 is updated.	PH36542

SC27-9536-01 (November 2020)

Description	Related APARs
IMS Fast Path Advanced Tool: IMS Fast Path Advanced Tool continues processing even when the SDEP keyword is specified and no SDEP segments are defined in the database description (DBD).	PH22928
Online Structure Change utility: The Online Structure Change utility (OSC) generates DBD Definition reports only when the job ends successfully. After the PTF for this APAR is applied, the utility generates DBD Definition reports even under certain error circumstances.	PH22928
Online Structure Change utility: When multiple OSC driver jobs are executed simultaneously in an IMSplex, the OSC internal job data set is used exclusively by one of the OSC driver jobs during its online change process, preventing other OSC internal jobs to be submitted by their OSC driver jobs. The new ENQDSN keyword specifies the ENQ data set, which the Online Structure Change utility uses to serialize the online change processes of multiple OSC driver jobs. For more information, see “ENQDSN keyword” on page 1058 .	PH25835
Online Structure Change utility: The Online Structure Change utility checks whether edit/compression exit routines are being used by other databases when a replacement of edit/compression exit routines is requested.	PH26632
Online Structure Change utility: If the OSC logger exit routine is configured, the Online Structure Change utility can be executed while the following IBM product jobs are running in the same IMS environment: <ul style="list-style-type: none">• IBM Security Guardium® S-TAP® for IMS on z/OS• IMS Online Reorganization Facility of IBM IMS Database Solution Pack for z/OS For information about configuring the OSC logger exit routine, see the topic "Configuring the OSC logger exit routine" in <i>IMS Fast Path Solution Pack: Overview and Customization</i> .	PH27204

SC27-9536-00 (March 2020)

Description	Related APARs
Online Structure Change utility: The Online Structure Change utility facilitates restructuring tasks for online IMS DEDBs. The utility provides advanced functions that help IMS database administrators change DBD descriptions, randomizing modules, and segment edit/compression routines without taking DEDB areas offline. For more information, see Part 5, “Online Structure Change utility,” on page 1005 . To use the Online Structure Change utility, IMS 15 is required. This utility cannot be used with IMS 14.	N/A

IMS HP Fast Path Utilities terminology

IMS HP Fast Path Utilities includes several unique terms that you should understand before you begin to use IMS HP Fast Path Utilities.

To make this information easier to read, the version and release levels of IMS are abbreviated, as follows:

- IMS 14 refers to IBM IMS 14.1 and IBM IMS Database Value Unit Edition 14.1.
- IMS 15 refers to IBM IMS 15.1 and later, and IBM IMS Database Value Unit Edition 15.1 and later.

The various versions of IMS are referred to simply as IMS, except where distinctions among them need to be made.

Subsections:

- [“IMS unique terms” on page 5](#)
- [“Short names for products, components, utilities, and functions” on page 5](#)

IMS unique terms

The following terms occur frequently in this information:

ACB

Represents application control block.

DBD

Represents database description.

DBRC

Represents IMS Database Recovery Control facility.

DEDB

Represents Data Entry Database.

DMAC

Represents DEDB Area Control list.

DMB

Represents data management block.

DMCB

Represents DEDB master control block.

MADS

Represents multiple area data set.

RECON

Represents Recovery Control.

Short names for products, components, utilities, and functions

The following short names are used for products, components, utilities, and functions.

Analyze function

Represents the Analyze function of FPA.

Autonomics Director or AD

Represents IBM IMS Tools Base for z/OS Autonomics Director.

Build Index function

Represents the Build Index function of FPA.

Change function

Represents the Change function of FPA.

DB Sensor

Represents the Database Sensor component in IBM IMS Fast Path Solution Pack for z/OS.

DBT

Represents IMS Database Tools.

DEDB Pointer Checker

Represents the DEDB Pointer Checker utility of FPB.

DEDB Tuning Aid

Represents the DEDB Tuning Aid utility of FPB.

DEDB Unload/Reload

Represents the DEDB Unload/Reload utilities of FPB.

DFSORT

Represents Data Facility Sort.

DMAC Print function

Represents the DMAC Print function of FPA.

Extract function

Represents the Extract function of FPA.

FPA

Represents the IMS Fast Path Advanced Tool of IMS HP Fast Path Utilities in IBM IMS Fast Path Solution Pack for z/OS.

FPB

Represents the IMS Fast Path Basic Tools of IMS HP Fast Path Utilities in IBM IMS Fast Path Solution Pack for z/OS.

FPO

Represents the IMS Fast Path Online Tools of IMS HP Fast Path Utilities in IBM IMS Fast Path Solution Pack for z/OS.

Freespace Reclaim function

Represents the Freespace Reclaim function of FPA.

IMS Database Recovery Facility

Represents IMS Database Recovery Facility of IBM IMS Recovery Solution Pack for z/OS.

IMS High Performance Image Copy, IMS HP Image Copy, or IMS HPIC

Represents IBM IMS High Performance Image Copy for z/OS.

IMS HP Fast Path Utilities or IMS HPFPU

Represents IMS High Performance Fast Path Utilities of IBM IMS Fast Path Solution Pack for z/OS.

IMS Fast Path Solution Pack

Represents the IBM IMS Fast Path Solution Pack for z/OS.

IMS Library Integrity Utilities or IMS LIU

Represents IBM IMS Library Integrity Utilities for z/OS.

IMS Recovery Solution Pack

Represents the IBM IMS Recovery Solution Pack for z/OS.

IMS Tools Base

Represents IBM IMS Tools Base for z/OS.

IMS Tools Catalog Interface

Represents IMS Tools Catalog Interface provided with Tools Base Common Services of IBM IMS Tools Base.

IMS Tools Knowledge Base or IMS Tools KB

Represents IMS Tools Knowledge Base of IBM IMS Tools Base for z/OS.

IMS Tools Online System Interface or TOSI

Represents the IMS Tools Online System Interface of IBM IMS Tools Base for z/OS.

Initialize function

Represents the Initialize function of FPA.

Integrity Checker

Represents the Integrity Checker function of IMS Library Integrity Utilities.

Management Console

Represents IBM Management Console for IMS and Db2® for z/OS.

Online Area Extender or OAE

Represents the DEDB Online Area Extender utility of FPO.

Online Data Extract or ODE

Represents the DEDB Online Data Extract utility of FPO.

Online DMAC Print or ODM

Represents the DEDB Online DMAC Print utility of FPO.

Online Expert Reorganization or OER

Represents the DEDB Online Expert Reorganization utility of FPO.

Online Pointer Checker or OPC

Represents the DEDB Online Pointer Checker utility of FPO.

Online Space Management or OSM

Represents the Online Space Management utility.

Online Structure Change utility or OSC

Represents the Online Structure Change utility.

Reload function

Represents the Reload function of FPA.

Reorg function

Represents the Reorg function of FPA.

Resync function

Represents the Resync function of FPA.

Policy Services

Represents IBM IMS Tools Base Policy Services for z/OS.

Tune function

Represents the Tune function of FPA.

Unload function

Represents the Unload function of FPA.

Managing IMS Data Entry Databases

IMS Fast Path Solution Pack helps you manage your IMS Data Entry Databases (DEDBs).

Using IMS Fast Path Solution Pack regularly helps you ensure that your DEDBs are healthy by alerting you when a database requires reorganization or tuning.

By using the IMS Fast Path Solution Pack, you can address the following data management tasks:

- [“Analyzing DEDB areas” on page 7](#)
- [“Managing space of online areas” on page 8](#)
- [“Tuning DEDB areas and randomizer” on page 8](#)
- [“Restructuring offline areas with high performance methods” on page 9](#)
- [“Restructuring online areas” on page 9](#)

Analyzing DEDB areas

As root segments and dependent segments for database records are added, updated, or deleted in a DEDB area, those segments can become scattered across a large number of separate CIs, or the free space in CIs can become fragmented into smaller pieces. The scattered segments and fragmentation of free space might cause the performance of applications to suffer and lead to inefficient use of space. Also, when available free space is used up, new segments cannot be added.

Reorganizing or restructuring DEDB areas is required to prevent degradation of application performance and out-of-space conditions.

IMS Fast Path Solution Pack provides both online and offline area analysis functions through IMS Fast Path Online Tools and IMS Fast Path Advanced Tool.

The online analysis function that is provided by the Online Pointer Checker (OPC) utility of IMS Fast Path Online Tools and the offline Analyze function of IMS Fast Path Advanced Tool verify the integrity of pointer values and analyze the space utilization, I/O efficiency of root segments and database records, and randomizing efficiency of a DEDB. Both online and offline analysis functions can process multiple areas of a DEDB in a single job step. These analysis functions can also reveal potential pointer errors.

These analysis functions generate area analysis reports that contain various statistics about the areas. The reports can help you reduce the amount of time that you spend diagnosing and solving DEDB space management issues and problems.

You can use the online analysis function if you want to analyze DEDB areas without taking the DEDB or areas offline. The online analysis function uses the buffering and locking services of IMS Fast Path system to read segments from online DEDB areas while maintaining integrity.

You can use the offline analysis function if you want to process multiple DEDB areas with high performance. The offline analysis function can process multiple areas in parallel in the job step by using its own high-performance I/O technology. The offline analysis function can also use image copy data sets as input instead of area data sets. By using image copy data sets as input, you can analyze DEDB areas without affecting the online applications service schedule.

Managing space of online areas

Taking an area offline only for the purpose of reorganizing free space is often not a viable option in DEDBs, especially in data centers where IMS environments are kept online 24 hours a day and 7 days a week. IMS provides the High-Speed DEDB Direct Reorganization utility (DBFUHDR0) to reorganize free space of online DEDB areas.

The Online Expert Reorganization (OER) utility of IMS Fast Path Online Tools extends the capability of DBFUHDR0. OER can evaluate UOWs in DEDB areas, rank the UOWs, and reorganize only highly ranked UOWs to reduce the volume of database update logs. OER can also collect statistics and generate area analysis reports of the reorganized area by calling the online area analysis function (OPC) after the reorganization process. The statistics data and analysis reports provide information about how effectively the free space was reorganized and whether any space utilization or I/O efficiency issues remain in the reorganized area.

Preventing an out-of-space condition is also crucial in operating environments that cannot be stopped. If space in the independent overflow (IOVF) portion or the sequential dependent (SDEP) portion is about to be used up, and if the database service cannot be stopped, the Online Area Extender (OAE) utility of IMS Fast Path Online Tools can help you dynamically extend the size of the DEDB areas. You can specify the portion of the area to extend: the IOVF portion, the SDEP portion, or both.

These utilities can be run by as stand-alone utilities, but they can also be run by using the Online Space Management utility. The Online Space Management utility integrates the online reorganization process of OER, the online area extension process of OAE, and the online area analysis process of OPC into one job step. When the Online Space Management utility is used in the active autonomic environment, it runs one or more of these processes based on the state of the area; whether the area requires reorganization, area extension, or both. This utility simplifies space management tasks and eliminates manual operations that are needed to individually run these utilities.

Tuning DEDB areas and randomizer

As database records and segments in an area increase, the space utilization of the area becomes inefficient to the point that tuning is required. A poorly tuned DEDB area might require excessive physical I/Os to read root segments and database records.

However, tuning a database often involves significant and time-consuming database administration tasks including tuning the randomizer, increasing the amount of free space, changing the database structure, or adding areas.

The Tune function of IMS Fast Path Advanced Tool can help ease these tuning tasks. The Tune function facilitates multiple iterations of the database tuning process, which allows you to select physical database attributes that meet your performance and space utilization requirements.

This function helps you determine the effective changes to database definition and randomizer by evaluating the potential benefits, or impact, of the changes to the area without the overhead of unloading and reloading the area. By ensuring that your new database definition is effective before reorganizing your database, you can avoid rerunning reorganization jobs that might be caused by inadequate database definitions.

Restructuring offline areas with high performance methods

When you have tuned the database definition or the randomizer, the subsequent steps are unloading the segment data from the existing area, reclaiming the space of the area based on the new database definition parameters, and reloading the unloaded data to the reclaimed space, possibly with the new randomizer. These tasks typically require you to take the database or the area offline for a short period of time.

By using the Unload, Reload, Change, and Reorg functions of IMS Fast Path Advanced Tool, you can perform the offline reorganization processing efficiently.

Unloading and reloading areas in multiple steps

Both the Unload and the Reload functions use their own high performance I/O technique to reduce CPU time and elapsed time. If you combine the unload step and the reload step, a DEDB area can be read and loaded faster than when the area is processed by IMS online applications.

This method creates an intermediate file (a sequential file that contains unloaded segment records) that can be used to obtain specific database statistics for data analysis or that can be used to create DEDB areas for testing purposes.

Changing an area in a single step

The Change function does not create an intermediate file. The unloaded segment data is passed to the reload process through the main storage buffers. The internal unload and reload processes use the same high performance I/O technique that is used by the Unload and Reload functions. Through the use of these techniques, the Change function further reduces the elapsed time of the entire reorganization process.

Integrating image copy process

The Unload, Reload, Change, and Reorg functions of IMS Fast Path Advanced Tool can internally call an image copy process. The Unload function supports unloading of segment data from image copies. This capability can reduce impact on online database applications because it does not read the area data sets again for batch application data processing or reorganization processing. You can take image copies of reloaded areas by using IMS High Performance Image Copy for z/OS within the Reload, Change, or Reorg function job step.

Restructuring online areas

If you use the Online Structure Change utility, you can restructure online DEDB or areas with minimal impact to database availability. With this utility, you can change DBD descriptions, randomizing modules, and segment edit/compression routines.

Taking systems offline for maintenance can affect your business. In today's market, systems must be highly available because the cost of outage can be too high. Online restructuring provides users with full access to the database during database restructure, improves the overall database availability, and reduces planned downtime.

The utility allocates shadow area data sets to perform unload and reload while capturing all database updates made by applications. When unload and reload is done, it uses an IMS DBCTL system to apply all the captured updates and switches the original area data sets with new restructured area data sets. A single /DBR command outage is necessary to switch the area data sets.

Solutions to simplify and automate database maintenance tasks

The utilities and functions of IMS Fast Path Solution Pack can be used individually. However, if used together with IBM IMS Tools Base for z/OS components, they can further simplify the database maintenance tasks by providing the following solutions.

- [“Managing utility reports in a central repository” on page 10](#)
- [“Automated DEDB space usage monitoring” on page 10](#)
- [“Automated DEDB space management” on page 10](#)
- [“Consolidation of DEDB space management information into a graphical interface” on page 10](#)

Managing utility reports in a central repository

The reports that are generated by the OPC utility, the OER utility, and the IMS Fast Path Advanced Tool functions can be stored in a repository. IBM IMS Tools Base IMS Tools Knowledge Base for z/OS (IMS Tools Knowledge Base) provides the repository infrastructure and provides centralized management of various types of repositories including the report repository. You can search historical copies of reports by database name, date and time, or other search criteria. IMS Tools Knowledge Base helps you find the reports faster to accelerate decision making and problem analysis.

Automated DEDB space usage monitoring

The online analysis function of the OPC utility and the OER utility and the offline analysis function of IMS Fast Path Advanced Tool support collection of some key statistics values for area space management. These values can be accumulated into sequential data sets that can be analyzed by user-written application programs.

By using IMS Fast Path Solution Pack together with IBM IMS Tools Base, you can have these key statistics values stored in the *Sensor Data repository* to support automated database space monitoring solutions.

You can use IBM provided templates to define a set of threshold criteria for detecting exceptions in DEDB area space usage and associated recommendations for reorganization and tuning. The set of threshold criteria and the mapping of exceptions to the recommended actions are stored in another repository as an exception detection policy.

IBM IMS Tools Base Autonomics Director for z/OS uses the information and data that is stored in IMS Tools Knowledge Base repositories to assist you in identifying and performing the tasks that pertain to maintaining the health of DEDB areas.

Automated DEDB space management

The Online Space Management utility extends and reorganizes online areas. When it is used with active autonomics features of Autonomics Director, an Online Space Management utility job runs automatically only when specific exceptions are detected by policy evaluation. For each area, Autonomics Director provides recommendations for resolving the exceptional state, which can be area extension, reorganization, or both. Based on the recommendations, the Online Space Management utility job runs the required functions to resolve the state of the area. For example, if an area is experiencing free space fragmentation and a specific exception that recommends reorganization is detected by policy evaluation, Autonomics Director initiates an Online Space Management utility job to reorganize the area.

Consolidation of DEDB space management information into a graphical interface

IBM Management Console for IMS and Db2 for z/OS consolidates key information and data that is stored in IMS Tools Knowledge Base repositories into a graphical interface.

You can aggregate and view the following information in an intuitive graphical web interface:

- Database exceptions that are detected by Autonomics Director and recommended resolutions
- Various reports that are generated by IMS Fast Path Solution Pack utilities
- Graphical charts that show historical sensor data that is collected by IMS Fast Path Solution Pack utilities

IMS HP Fast Path Utilities features and benefits

IMS HP Fast Path Utilities provides various utilities that help you manage your DEDBs.

IMS Fast Path Advanced Tool

IMS Fast Path Advanced Tool (also referred to as FPA) provides various key functions for managing offline DEDBs and Fast Path secondary index databases in just one step.

You can run the functions of this tool by using simple JCL statements that follow an easy-to-understand format. You can omit certain DD statements from the JCL statements. When these DD statements are omitted, the tool dynamically allocates the data sets. The tool enables you to use the same JCL statements for various functions by simply changing the control statements.

This tool improves performance (that is, elapsed time and CPU time) by using the data space for IOVF and SDEP parts, by using fast I/O call against DEDB areas, by concurrently processing multiple DEDB areas, and by reducing the time required for I/O with the use of internal sort.

With most of the functions, you can take image copies, validate database pointers, and obtain statistical information reports of the processed databases in a single job step. These functions can use image copies of a DEDB as their input. IMS Fast Path Advanced Tool helps you maintain the health of your databases, improve your productivity, and achieve a smaller maintenance window.

IMS Fast Path Advanced Tool provides the following functions:

- The Unload function unloads multiple DEDB areas to QSAM data sets. If necessary, a sort step can be included in the Unload process.
- The Reload function reloads multiple DEDB areas from the QSAM data sets. If necessary, a sort step and an analyze step can be included in the Reload process.
- The Change function restructures multiple DEDB areas. If necessary, an analyze step can be included in the Change process.
- The Reorg function reorganizes a set of UOWs in a DEDB area on the basis of a ranked set of UOWs.
- The Analyze function provides complete DEDB integrity verification of all IMS pointer values, free space element chains, VSAM control fields, and space utilization in DEDBs. It also provides complete integrity verification of pointer segments in secondary index databases. In addition, it supplies the comprehensive analysis information that is required to effectively manage the DEDB space utilization, performance characteristics, and physical attributes.
- The Tune function simulates changes in the DBD parameters and randomizers, and reports the effectiveness of the new database attributes.
- The Extract function extracts segment data from multiple DEDB areas and writes the output to QSAM data sets. The extracted data can be used for batch application processing.
- The DMAC Print function provides maps for DMAC blocks by using IMS mapping macro and the dump of DMAC in multiple DEDB areas.
- The Build Index function builds the secondary index databases from multiple DEDB areas.
- The Resync function synchronizes secondary index databases with the DEDB when they are out of synchronization.
- The Freespace Reclaim function reclaims the control intervals (CIs) in secondary index database data sets.
- The Initialize function efficiently initializes a single DEDB area. It can also concurrently initialize multiple DEDB areas.

IMS Fast Path Basic Tools

IMS Fast Path Basic Tools (also referred to as FPB) helps you analyze, maintain, tune, and migrate offline DEDBs.

You can use the tuning tool to evaluate the effectiveness of the changes in the DEDB area definition parameters without consuming time to unload and reload the database. IMS Fast Path Basic Tools assures the compatibility for your JCL streams and the input control statements that are used in the former releases of FPB.

IMS Fast Path Basic Tools provides the following functions:

DEDB Pointer Checker

Provides a fast and efficient mechanism for verifying the integrity of all IMS pointer values, free space element chains, VSAM control fields, and space utilization in IMS DEDBs. DEDB Pointer Checker provides the comprehensive analysis information that is required to effectively manage the space utilization, performance characteristics, and physical attributes of an IMS DEDBs.

DEDB Tuning Aid

An extension of DEDB Pointer Checker. This feature enables you to evaluate the DEDB definition in reorganization, or to reduce the impact of changes to an existing DEDB, without consuming time to unload and reload the database.

DEDB Unload and Reload

A set of high-performance utilities for efficient Unload and Reload process, or to reorganize IMS DEDBs.

For more information about IMS Fast Path Basic Tools, see the *IMS Fast Path Solution Pack: IMS Fast Path Basic Tools User's Guide*.

IMS Fast Path Online Tools

IMS Fast Path Online Tools (also referred to as FPO) helps you to boost system availability by performing various key functions without having to take your IMS database offline.

IMS Fast Path Online Tools provides the following functions:

DEDB Online Pointer Checker

Verifies the integrity of all IMS pointer values, free space element chains, VSAM control fields, and space utilization in a DEDB area while that area is online and being updated by users. It also supplies the comprehensive analysis information that is required to effectively manage the space utilization, performance characteristics, and physical attributes of an IMS DEDB.

DEDB Online Data Extract

Extracts segment data from an online DEDB area and writes the output to a sequential file.

DEDB Online Area Extender

Extends the size of the DEDB area dynamically, without taking the DEDB area offline. By providing an alternative to taking the DEDB offline, this utility is helpful in environments where database availability is crucial.

DEDB Online Expert Reorganization

Reorganizes a set of UOWs in a DEDB area on the basis of a ranked set of UOWs.

DEDB Online DMAC Print

Provides maps for storage that is used by the IMS control region by using the IMS mapping macro and the IMS storage dump of DMAC.

Online Space Management utility

The Online Space Management utility (also referred to as OSM) simplifies reorganization and extension of online DEDB areas by integrating and automating the database space management tasks.

It runs the Online Expert Reorganization utility, Online Area Extender utility, and Online Pointer Checker utility processes of IMS Fast Path Online Tools, and optionally the DEDB Sequential Dependent Scan

utility (DBFUMSCO) and DEDB Sequential Dependent Delete utility (DBFUMDL0) processes, in one job step to reorganize, extend, and verify the DEDB integrity of the area.

When the Online Space Management utility is used with active autonomics features of Autonomics Director, Autonomics Director periodically collects and evaluates sensor data, makes recommendations, and starts Online Space Management utility jobs to perform the recommended actions.

Online Structure Change utility

The Online Structure Change utility (also referred to as OSC) provides advanced functions that help IMS database administrators change the structure of DEDB areas without taking DEDB areas offline.

The Online Structure Change utility helps you restructure online DEDB or areas with minimal impact to database availability. The utility supports changing DBD descriptions, randomizing modules, and segment edit/compression routines.

The online restructuring capability provides users with full access to the database during database restructure, improves the overall database availability, and reduces planned downtime.

The utility allocates shadow area data sets to perform unload and reload while capturing all database updates made by applications. When unload and reload is done, it uses an IMS DBCTL system to apply all the captured updates and switches the original area data sets with new restructured area data sets. A single /DBR command outage is necessary to switch the area data sets.

To use the Online Structure Change utility, IMS 15 is required. This utility cannot be used with IMS 14.

Functional differences between the tools

The tools of IMS HP Fast Path Utilities provide similar functions. You can decide which tool to use according to your scenario.

The following table shows the functional differences and supporting tools.

Table 1. Functional differences between the tools

Function name	FPA	FPB	FPO
Integrity verification and analysis	ANALYZE command <ul style="list-style-type: none"> Allows the use of image copy data sets as input data sets Allows the use of image copy data sets as output data sets Provides complete integrity verification of pointer segments in secondary index databases 	DEDB Pointer Checker <ul style="list-style-type: none"> Allows the use of image copy data sets as input data sets 	Online Pointer Checker <ul style="list-style-type: none"> Allows the use of image copy data sets as output data sets

Table 1. Functional differences between the tools (continued)

Function name	FPA	FPB	FPO
Unload	UNLOAD command <ul style="list-style-type: none"> • Allows the use of image copy data sets as input data sets • Allows the use of image copy data sets as output data sets • Allows the use of the load control information with the RELOAD command 	DEDB Unload <ul style="list-style-type: none"> • Allows the use of the load control information 	Online Data Extract
Reload	RELOAD command <ul style="list-style-type: none"> • Allows the use of image copy data sets as output data sets • Allows the use of the load control information with the LOADCTL subcommand • Provides integrity verification and analysis for an area 	DEDB Reload <ul style="list-style-type: none"> • Allows the use of image copy data sets as output data sets • Allows the use of the load control information with DEDB Unload 	N/A
Change	CHANGE command <ul style="list-style-type: none"> • Allows the use of image copy data sets as output data sets • Provides integrity verification and analysis for an area 	Combination of DEDB Unload and Reload	N/A
Reorganize	REORG command <ul style="list-style-type: none"> • Allows the use of image copy data sets as output data sets • Provides integrity verification and analysis for an area 	Combination of DEDB Unload and Reload	Online Expert Reorganization <ul style="list-style-type: none"> • Allows the use of image copy data sets as output data sets • Provides integrity verification and analysis for an area
Extend	CHANGE command or combination of the UNLOAD and the RELOAD commands <ul style="list-style-type: none"> • Allows the use of image copy data sets as output data sets • Provides integrity verification and analysis for an area 	Combination of DEDB Unload and Reload	Online Area Extender

Table 1. Functional differences between the tools (continued)

Function name	FPA	FPB	FPO
Initialize DEDB areas	INITIALIZE command	N/A	N/A
Extract segments	EXTRACT command	Use Unload/Reload user exit routines	Online Data Extract
Print DMAC	DMACPRT command	N/A	Online DMAC Print
Build secondary index databases	INDEXBLD command	N/A	N/A
Synchronize secondary index databases	INDEXBLD command	N/A	N/A
Reclaim free space in secondary index database data sets	INDEXBLD command	N/A	N/A
Tuning Aid	TUNE command	DEDB Tuning Aid	Combination of Online Pointer Checker and the FPA TUNE command

Support for IMS Library Integrity Utilities

IMS Library Integrity Utilities is an IMS Tools product that aids you in managing data for the libraries, such as the DBD libraries, PSB libraries, ACB libraries, and RECON data sets that you use when referring to the IMS database.

Some typical data management functions are as follows:

- To prevent the database corruption that the use of an incorrect member of a library can cause.
- To check the consistency among each library.
- To check, compare, change, generate, and maintain the members of a library.

The Integrity Checker utility of IMS Library Integrity Utilities prevents database corruption caused when IMS applications use incorrect IMS control blocks to get access to a database. It verifies that the DMB that IMS is using to access to the database is the same as the one that IMS used to load it. To do this, Integrity Checker compares the contents of the DMB that IMS is using against the registered DMB information (registered DMB entries are referred to as RDEs), which is stored in the IMS LIU Integrity Control data set (LICON data set) before the verification. To learn more about DMB verification in both IMS batch and online environments, see the *IMS Library Integrity Utilities User's Guide*.

The following functions of IMS HP Fast Path Utilities support the creation of RDEs in which the DMB information is registered:

- FPA Unload
- FPA Reload
- FPA Change
- FPA Reorg
- FPA Build Index

Integrity Checker requires approximately 700 KB of extended storage.

For information about configuring IMS Library Integrity Utilities and activating the Integrity Checker utility, see *IMS Fast Path Solution Pack: Overview and Customization*.

Restrictions for using Integrity Checker

Only the Unload, Reload, Change, Reorg, and Build Index functions of FPA can create an RDE for the reorganized area. Thus, when you reorganize a DEDB area to change DBD definition by using the functions of FPB, you must manually create an RDE for the reorganized area before using the area, by using the LICON utility of IMS Library Integrity Utilities.

To work on RDEs, use the LICON utility. For more information, see the *IMS Library Integrity Utilities User's Guide*.

Support for IMS Tools Base and common tools

IBM IMS Tools Base for z/OS (also referred to as IMS Tools Base) is a collection of tools that provide a uniform infrastructure that other IMS Tools rely on. Many of the tools that are included in IMS HP Fast Path Utilities take advantage of the functionality that is provided by tools of IMS Tools Base.

Subsections:

- [“IMS Tools Knowledge Base” on page 16](#)
- [“IMS Tools Online System Interface” on page 16](#)
- [“Policy Services” on page 17](#)
- [“Autonomics Director” on page 17](#)
- [“IMS Tools Catalog Interface” on page 18](#)
- [“Management Console” on page 18](#)

For information about installing and configuring these common tools, see *IMS Fast Path Solution Pack: Overview and Customization*.

IMS Tools Knowledge Base

IMS Tools Base IMS Tools Knowledge Base provides a centralized information management environment for IMS Tools products and Policy Services.

IMS Tools Knowledge Base allows you to store, manage, and access reports, sensor data, policies, rules, and notification lists that are generated or used by any tool that has been enabled and registered to participate in an IMS Tools Knowledge Base information management environment.

If you specify the IMS Tools KB server XCF group name on the ITKBSRVR keyword and the ITKB load module library on the STEPLIB DD statement, the utility reports are stored in the IMS Tools KB Output repository. If you also specify the SENSOR keyword, sensor data is collected from DEDB areas and is stored in the IMS Tools KB Sensor Data repository.

You can find the lists of the reports that can be stored in IMS Tools KB in the following topics:

- [“Reports generated by each process of FPA” on page 560](#)
- [Chapter 26, “FPO reports stored in IMS Tools KB,” on page 941](#)
- [Chapter 35, “Online Space Management utility reports,” on page 1001](#)
- [Chapter 40, “Online Structure Change utility reports,” on page 1075](#)
- [Topic "Reference: Supplementary utility reports stored in IMS Tools KB" in the *IMS Fast Path Solution Pack: Supplementary Utilities User's Guide*](#)

IMS Tools Online System Interface

IMS Tools Base IMS Tools Online System Interface is a general-purpose command interface that allows IMS Tools to communicate with all supported IMS versions.

In IMS HP Fast Path Utilities, IMS Tools Online System Interface is used by the following functions:

- The FPA Resync function (INDEXBLD command with the IDXPROC=RESYNC option), to synchronize the primary DEDB database and its secondary index databases. IMS Tools Online System Interface is used

only when offline synchronization is requested. For more information about the Resync function, see [“Functions of Resync” on page 321](#).

- The OAE Rename program of the FPO Online Area Extender utility. For more information about the OAE Rename program, see [“OAE Rename program” on page 771](#).
- The Online Structure Change utility, to issue IMS commands to IMS systems and IMS DBCTL systems.

Note: If you are migrating from IMS High Performance Fast Path Utilities for z/OS 3.2, see the migration considerations for IMS Tools Online System Interface in [“Compatibilities with former releases of the product and migration considerations” on page 27](#).

Policy Services

IMS Tools Base Policy Services is a core IMS Tools component that can monitor a specific database state by evaluating the sensor data that is collected by an IMS Tools product, and that can provide a response to any conditions that exceed the threshold values specified for this state.

Policy Services supports conditional autonomic database health management functionality for participating IMS Tools products. Conditional autonomies can provide the following functionality:

- Evaluate the need for database maintenance operations
- Recommend corrective actions based on user-defined (policy-driven) requirements

You can use the following utilities to collect sensor data:

- DEDB analysis utilities:
 - The Analyze function of FPA
 - Online Pointer Checker of FPO
 - Online Space Management utility (Online Pointer Checker process)
- DEDB analysis functions in reorganization utilities:
 - The Change, Reload, and Reorg functions of FPA (with the Analyze process)
 - Online Expert Reorganization of FPO
- DEDB analysis function in an image copy utility:
 - IMS HP Image Copy utility

Autonomics Director

IMS Tools Base Autonomics Director can automate ongoing IMS database monitoring and maintenance based on a detailed understanding of the current state of IMS databases.

Autonomics Director assists you in identifying and performing the functions that are associated with maintaining the health of DEDB areas.

Autonomics Director along with the IMS Fast Path Solution Pack provides the following functionality:

- Automatic and consistent collection of sensor data related to state of DEDB areas
- Automatic and consistent analysis of area state against user-defined threshold criteria
- Generation of recommended actions for subject DEDB areas
- Execution of recommended actions for subject DEDB areas

If you specify the Autonomics Director server name on the ADXCFGRP keyword, a sensor data notification is sent to Autonomics Director, which triggers the follow-on action of evaluating sensor data. Also, in the active autonomies environment, a result notification is sent to Autonomics Director, which sets the result status of the Online Space Management utility job to Autonomics Director.

IMS Tools Catalog Interface

IMS Tools Catalog Interface is a common interface used by IMS Tools products to process the IMS catalog directory.

All the functions of FPA and the Online Structure Change utility can use IMS Tools Catalog Interface. If you specify the high-level qualifier of the bootstrap data set of the IMS directory on the IMSCATHLQ keyword, the utility uses IMS Tools Catalog Interface. IMS Tools Catalog Interface calls the IMS catalog API to read ACBs from the IMS directory, which causes the utility to use the ACBs in the IMS directory instead of the ACBs in ACB libraries.

To use IMS Tools Catalog Interface, IMS catalog must be set up and ACBs must be managed by IMS (IMS management of ACBs must be enabled). For more information, see *IMS Database Administration* and *IMS System Definition*.

Management Console

IBM Management Console for IMS and Db2 for z/OS (Management Console) consolidates key IMS statistics into an intuitive, graphical web interface.

You can view aggregate information for customizable groups of databases as well as the following types of detailed information for individual databases:

- Graphical charts that show historical sensor data that is collected by IMS Tools
- Database exceptions that are detected by Autonomics Director through the automated analysis of database state and recommended resolutions
- Various reports that are generated by IMS Tools
- Graphical view of the database structure map, program specifications, DBD and PSB macro source, and DBD and PSB XML document

Support for IMS managed ACBs environment

In an IMS managed ACBs environment, IMS can manage the runtime application control blocks (ACBs) for databases and program views for you. When IMS manages ACBs, IMS no longer requires DBD, PSB, and ACB libraries.

IMS Fast Path Advanced Tool

All the functions of FPA support IMS managed ACBs environment. To use IMS managed ACBs instead of ACBs in ACB libraries, you must specify the following keywords:

- [“IMSCATHLQ keyword” on page 481](#)
- [“IMSCATACB_INPUT keyword” on page 480](#)
- [“IMSCATACB_OUTPUT keyword” on page 481](#)

FPA uses the IMS Tools Catalog Interface to process the IMS catalog directory. To learn more about the interface, see [“IMS Tools Catalog Interface” on page 18](#).

IMS Fast Path Basic Tools

The following tools of FPB support IMS managed ACBs environment.

- DEDB Pointer Checker
- DEDB Unload
- DEDB Reload

For more information, see the *IMS Fast Path Solution Pack: IMS Fast Path Basic Tools User's Guide*.

IMS Fast Path Online Tools

All the tools of FPO can be executed in an IMS managed ACBs environment.

To extend the IOVF portion with the Online Area Extender (OAE) utility in an IMS managed ACBs environment, either the IOVF_ROOT=(*number2,overflow2*) keyword or the IOVF_UOW keyword is required.

Online Space Management utility

All the functions of the Online Space Management (OSM) utility can be executed in an IMS managed ACBs environment.

To extend the IOVF portion with the Online Space Management utility in an IMS managed ACBs environment, either the IOVF_ROOT=(*number2,overflow2*) keyword or the IOVF_UOW keyword is required.

Online Structure Change utility

The Online Structure Change (OSC) utility supports IMS managed ACB environment. To use IMS managed ACBs instead of ACBs in ACB libraries, you must specify the IMSCATHLQ keyword (see [“IMSCATHLQ keyword”](#) on page 1036).

The Online Structure Change utility uses the IMS Tools Catalog Interface to process the IMS catalog directory. To learn more about the interface, see [“IMS Tools Catalog Interface”](#) on page 18.

Supplementary utilities

The following supplementary utilities support IMS managed ACB environment:

- DEDB Reload Segment Data Set Create utility (FABCUR6)
- DEDB Unloaded Segment Data Set Retrieve utility (FABCUR7)
- IMS DEDB randomizing module interface module (FABCRMIF or FABDRMIF)

To use IMS managed ACBs instead of ACBs in ACB libraries, you must specify the IMSCATHLQ statement.

For more information, see the *IMS Fast Path Solution Pack: Supplementary Utilities User's Guide*.

Supplementary utilities use the IMS Tools Catalog Interface to process the IMS catalog directory. To learn more about the interface, see [“IMS Tools Catalog Interface”](#) on page 18.

Support for DEDB encryption

All the tools and utilities of IMS HP Fast Path Utilities support encrypted DEDB area data sets (ADSs).

To enable the tools and utilities to process DEDB ADSs that are encrypted with z/OS data set encryption, the PTF for APAR PI83756 must be applied to IMS 15.

You do not need to change the JCL streams to process encrypted DEDB ADSs.

For IMS Fast Path Advanced Tool, if you use the ALLOCATE subcommand to dynamically define a VSAM cluster that is used for the processing of a FPA command, you can also encrypt the VSAM cluster by specifying the encryption key label with the KEYLABEL keyword.

For more information, see [“KEYLABEL keyword”](#) on page 489.

Service updates and support information

Service updates and support information for this product, including software fix packs, PTFs, frequently asked questions (FAQs), technical notes, troubleshooting information, and downloads, are available from the web.

To find service updates and support information, see the following website:

[IBM Support: IMS Fast Path Solution Pack for z/OS](#)

Functional changes

The following topics summarize the major functional changes made to each release of IMS HP Fast Path Utilities.

Functional changes in IMS HP Fast Path Utilities in IMS Fast Path Solution Pack 2.1

This topic summarizes functional changes made to IMS HP Fast Path Utilities of IMS Fast Path Solution Pack 2.1.

Online Structure Change utility

The Online Structure Change utility facilitates restructuring tasks for online IMS DEDBs. The utility provides advanced functions that help IMS database administrators change DBD descriptions, randomizing modules, and segment edit/compression routines without taking DEDB areas offline. For more information, see [Part 5, “Online Structure Change utility,” on page 1005](#).

Supplementary utilities

The following supplementary utilities have been enhanced to support IMS managed ACBs:

- DEDB Reload Segment Data Set Create utility (FABCUR6)
- DEDB Unloaded Segment Data Set Retrieve utility (FABCUR7)
- IMS DEDB randomizing module interface module (FABCRMIF and FABDRMIF)

For more information, see [“Support for IMS managed ACBs environment” on page 18](#).

Functional changes in IMS HP Fast Path Utilities in IMS Fast Path Solution Pack 1.3

IMS HP Fast Path Utilities of IMS Fast Path Solution Pack 1.3 includes functional changes to IMS Fast Path Advanced Tool and IMS Fast Path Online Tools. Also, a new utility, Online Space Management utility, is introduced.

All the tools and utilities

DEDB encryption support

All the tools and utilities of IMS HP Fast Path Utilities support encrypted DEDB area data sets (ADSs). You do not need to change the JCL streams to process encrypted DEDB ADSs. For more information, see [“Support for DEDB encryption” on page 19](#).

For IMS Fast Path Advanced Tool, if you use the ALLOCATE subcommand to dynamically define a VSAM cluster that is used for the processing of a FPA command, you can also encrypt the VSAM cluster by specifying the encryption key label with the KEYLABEL keyword.

IMS Fast Path Advanced Tool

New function: Tune

The Tune function provides a fast and efficient mechanism for simulating the database with new database parameters. The function simulates changes in the DBD parameters and randomizers without loading the segments. This simulation helps you select the physical database attributes that satisfy your performance and space utilization requirements. The Tune function helps you simplify and reduce the amount of time required for database tuning activities. For more information, see [Chapter 9, “Tuning a DEDB offline with FPA,” on page 229](#).

New function: Resync

The Resync function synchronizes Fast Path secondary index databases. You can use the Resync function to synchronize a DEDB and its secondary index databases when they are out of

synchronization. Both online synchronization and offline synchronization are supported. For more information, see [“Functions of Resync” on page 321](#).

New function: Freespace Reclaim

The Freespace Reclaim function reclaims CIs in secondary index database data sets. For more information, see [Chapter 14, “Reclaiming CIs in secondary index databases with FPA,” on page 355](#).

IMS managed ACBs support

All the functions of FPA support IMS managed ACBs. FPA functions can use IMS managed ACBs instead of ACBs in ACB libraries. For more information, see [“Support for IMS managed ACBs environment” on page 18](#).

IMS Fast Path Online Tools

Online Pointer Checker (OPC) performance improvement

The elapsed time required for processing multiple areas with OPC or with OER with pointer check is reduced.

Online Space Management utility

The new Online Space Management (OSM) utility reorganizes, extends, and collects sensor data of online IMS DEDB areas in one job step. You can also use this utility together with active autonomics features of IMS Tools Base Autonomics Director to automate space management tasks. For more information, see [“Online Space Management utility functions” on page 955](#).

IMS Fast Path Basic Tools

IMS managed ACBs support

The DEDB Pointer Checker utility, the DEDB Unload utility, and the DEDB Reload utility support IMS managed ACBs. These utilities can use IMS managed ACBs instead of ACBs in ACB libraries. For more information, see [“Support for IMS managed ACBs environment” on page 18](#).

Functional changes in IMS HP Fast Path Utilities in IMS Fast Path Solution Pack 1.2

IMS HP Fast Path Utilities of IMS Fast Path Solution Pack 1.2 includes functional changes to IMS Fast Path Advanced Tool, IMS Fast Path Online Tools, and supplementary utilities.

The Database Sensor function is added to various tools to support the following new solutions that are provided by Tools Base for z/OS 1.3:

- Autonomics Director
- Administration Console

By using the Database Sensor function, you can store key statistics data of DEDB areas in the Sensor Data repository of IMS Tools Knowledge Base as sensor data.

Subsections:

- [“IMS Fast Path Advanced Tool” on page 21](#)
- [“IMS Fast Path Online Tools” on page 22](#)
- [“Supplementary utilities” on page 22](#)

IMS Fast Path Advanced Tool

Integrated Database Sensor function

New keywords, SENSOR and SENSOR_DBREC, are added to the ANALYZE, CHANGE, RELOAD, and REORG commands to support the Database Sensor function. For more information, see the following topics:

- [“SENSOR keyword” on page 535](#)

- [“SENSOR_DBREC keyword” on page 536](#)

Sensor data notification to Autonomics Director

A new keyword, ADXCFGRP, is added to the GLOBAL command to support the sensor data notification to an Autonomics Director server. For more information, see [“ADXCFGRP keyword” on page 439](#).

Initialize function support

A new function, FPA Initialize function, is added by APAR/PTF PM73363/UK83120. You can use the FPA Initialize function to initialize DEDB areas efficiently. For more information, see [Chapter 15, “Initializing a DEDB offline with FPA,” on page 367](#).

This APAR/PTF provides another enhancement. All FPA functions and FP Stand-alone DB Sensor support a new option to bypass the DBRC process of the database. This option is effective even when DBRC=FORCE is set for IMS. For more information, see [“DBRC keyword” on page 451](#).

IMS Fast Path Online Tools

Integrated Database Sensor function

New keywords, SENSOR and SENSOR_DBREC, are added to the ANALYZE statement of OPC and the PTRCHK statement of OER to support the Database Sensor function. For more information, see the following topics:

- [“SENSOR keyword” on page 922](#)
- [“SENSOR_DBREC keyword” on page 923](#)

Sensor data notification to Autonomics Director

A new keyword, ADXCFGRP, is added to the GLOBAL command to support the sensor data notification to an Autonomics Director server. For more information, see [“ADXCFGRP keyword” on page 861](#).

Supplementary utilities

Database Sensor function in HPFPU Hash Check support for IMS HP Image Copy

A new keyword, SENSOR, is added to the GLOBAL statement of the HPFPU Hash Check support for IMS HP Image Copy. For more information, see the topic "HPFPU Hash Check support for IMS HP Image Copy" in the *IMS Fast Path Solution Pack: Supplementary Utilities User's Guide*.

Sensor data notification to Autonomics Director

A new keyword, ADXCFGRP, is added to the GLOBAL command to support the sensor data notification to an Autonomics Director server. For more information, see the topic "HPFPU Hash Check support for IMS HP Image Copy" in the *IMS Fast Path Solution Pack: Supplementary Utilities User's Guide*.

Functional changes in IMS HP Fast Path Utilities in IMS Fast Path Solution Pack 1.1

IMS HP Fast Path Utilities of IMS Fast Path Solution Pack 1.1 provides the same functions that are provided in IBM IMS High Performance Fast Path Utilities for z/OS 3.2.

Subsections:

- [“All tools” on page 22](#)
- [“IMS Fast Path Advanced Tool” on page 23](#)
- [“IMS Fast Path Online Tools” on page 23](#)

All tools

The tools of IMS HP Fast Path Utilities are enabled for IMS 12 (APAR PM21939).

IMS Fast Path Advanced Tool

Build Index function

The Build Index function builds the secondary index databases from multiple DEDB areas. See [Chapter 12, “Building secondary index databases for a DEDB offline with FPA,” on page 295](#) to understand and use this function.

Checkpoint restart capability for the Unload function

The FPA Unload function supports the checkpoint restart option. You can use this option to reprocess only the failed areas without reprocessing successfully processed areas (APAR PM17824). See [“Checkpoint restart option” on page 41](#) for a description of this function.

Integrity Checker for the Unload function

The FPA Unload function supports the Integrity Checker utility of IMS Library Integrity Utilities to create RDEs (Registered DMB Entries) while unloading DEDB areas (APAR PM27942). See [“Support for IMS Library Integrity Utilities” on page 15](#) for a description of this function.

DBD consistency check for the Reload function

The Reload process can optionally check the consistency between the DBD information in the library that is specified on the IMSACB DD statement and the DBD information in the USR files that were created during the Unload process (APAR PM33340). See [“ACBCHK keyword” on page 438](#) for a description to use this option.

IMS Fast Path Online Tools

Multiple areas support for OPC and OER

OPC and OER can serially process multiple areas within a single step (APAR PM07840). See [“Example 6: Processing multiple areas and taking image copies” on page 727](#) and [“Example 3: Running OER with OPC for multiple areas and taking image copies” on page 819](#) for JCL examples.

Functional changes in IMS High Performance Fast Path Utilities for z/OS 3.2

This topic summarizes the functional changes made to IMS High Performance Fast Path Utilities for z/OS 3.2, which is the former product of IMS HP Fast Path Utilities that is provided in IMS Fast Path Solution Pack.

Subsections:

- [“IMS Fast Path Advanced Tool” on page 23](#)
- [“IMS Fast Path Basic Tools” on page 24](#)
- [“IMS Fast Path Online Tools” on page 24](#)
- [“Supplementary utilities” on page 25](#)

IMS Fast Path Advanced Tool

A new function introduced

The Reorg function reorganizes a set of UOWs in a DEDB area on the basis of a ranked set of UOWs. For more information, see [Chapter 7, “Reorganizing a DEDB offline with FPA,” on page 163](#).

IMS 11 support

FPA is enhanced to support IMS 11.

Sequence error segment data records support

The Unload function and the Change function enable you to store sequence error segment data records in an output data set. For more information, see the descriptions of the HFPSEQER DD statements in [“DD statements for the Unload function” on page 56](#) and [“DD statements for the Change function” on page 134](#).

Pointer error UOW analysis support

The Unload function is enhanced to analyze a UOW that has pointer errors during an unload process. For more information, see [“Unloaded report” on page 575](#) and [“PTRCHK keyword” on page 505](#).

IMS Tools KB support

You can store and manage the reports of IMS HP Fast Path Utilities in the Output repository of IMS Tools KB. For more information, see [“IMS Tools Knowledge Base” on page 16](#) and [Chapter 44, “Viewing the reports stored in IMS Tools KB,” on page 1171](#).

Report enhancement for PTRCHKLVL=QUICK

When the REPORT subcommand is specified with PTRCHKLVL=QUICK for the Analyze, Change, Reload, or Reorg command, FPA generates reports based on the information that is generated by the PTRCHKLVL=QUICK process. For more information, see [Chapter 17, “Sample reports of FPA,” on page 559](#).

User specified format support

- The Extract function enables you to extract data in a user specified format. For more information, see [“EFORMAT keyword” on page 460](#) and [“User format” on page 1189](#).
- The Unload function and the Reload function are enhanced to support user specified formats. For more information, see [“FORMAT keyword” on page 465](#) and [“User format” on page 1189](#).

IMS Library Integrity Utilities support

The Reload, Change, and Reorg functions are enhanced to support creation of an RDE in which the DMB information is registered. For more information, see [“Support for IMS Library Integrity Utilities” on page 15](#) and [“LIUGOPT keyword” on page 493](#).

Largest Database Records report support

The Analyze function generates the Largest Database Records report when the new keyword LARGEREC_REPORT=YES is specified with the REPORT subcommand and the PTRCHKLVL=FULL keyword is specified. For more information, see [“LARGEREC_REPORT keyword” on page 491](#).

IMS Fast Path Basic Tools

IMS 11 support

FPB is enhanced to support IMS 11.

Bypass option for validation process of sequential dependent segments

An option to bypass the validation process of sequential dependent segments is added to the DEDB Pointer Checker FABADA1 step. For more information, see TBLENTY= in the topic "FABADA1 SYSIN DD data set" in the *IMS Fast Path Solution Pack: IMS Fast Path Basic Tools User's Guide*.

DEDB Reload function enhancements (FABCUR3)

- Checks the validity of RAP data in the prefix part of the unloaded segment record by using the result of the randomizer call. For more information, see RAPERROR= in the topic "FABCUR3 SYSIN DD data set" in the *IMS Fast Path Solution Pack: IMS Fast Path Basic Tools User's Guide*.
- Utilizes the ACBLIB member instead of the DURDBDFN file to obtain the DEDB definition. For more information, see ACBLIB DD in the topic "FABCUR3 JCL" in the *IMS Fast Path Solution Pack: IMS Fast Path Basic Tools User's Guide*.

IMS Fast Path Online Tools

Diagnostics aid support

FPO is enhanced to support the diagnostics aid program HFPUDIAG.

Single FPXIN DD file support

Accepts a single FPXIN DD file, which contains Online Pointer Checker utility control statements as the input file instead of the SYSIN and DASYSIN DD files that have been supported up to IMS HP Fast Path Utilities 3.1. Compatibility is maintained with the former SYSIN and DASYSIN DD files.

Pointer cross-reference check and statistical analysis report support

The Online Pointer Checker utility and the Online Expert Reorganization utility are enhanced to check the cross-reference of pointers for an online DEDB area and produces various analysis reports in one step.

IMS 11 support

FPO is enhanced to support IMS 11.

IMS Tools KB support

You can store and manage the reports of IMS HP Fast Path Utilities in the Output repository of IMS Tools KB. For more information, see [“IMS Tools Knowledge Base” on page 16](#) and [Chapter 44, “Viewing the reports stored in IMS Tools KB,” on page 1171](#).

User specified format support

The Online Data Extract utility enables you to extract data in a user specified format. For more information, see [“OUTPUT keyword” on page 899](#) and [“User format” on page 1189](#).

DEDB Tuning Aid support

The Online Pointer Checker utility generates segment information records that can be used as input for DEDB Tuning Aid. For more information, see [“TYPRUN keyword” on page 935](#).

Supplementary utilities

Segment Length Distribution report support for DEDB Pointer Checker Hash Check option

DEDB Pointer Checker Hash Check option that is invoked by IMS HP Image Copy is enhanced to generate the Segment Length Distribution report and the Process Summary report. DEDB Pointer Checker Hash is also enhanced to store its reports to the IMS Tools KB repository. For more information, see the topic "REPORTS DD data set" in the *IMS Fast Path Solution Pack: Supplementary Utilities User's Guide*, [“IMS Tools Knowledge Base” on page 16](#), and [Chapter 44, “Viewing the reports stored in IMS Tools KB,” on page 1171](#).

Functional changes in IMS High Performance Fast Path Utilities for z/OS 3.1

IMS High Performance Fast Path Utilities for z/OS 3.1 includes functional changes to all the tools.

Subsections:

- [“IMS Fast Path Advanced Tool” on page 25](#)
- [“IMS Fast Path Basic Tools” on page 25](#)
- [“IMS Fast Path Online Tools” on page 26](#)
- [“Supplementary utilities” on page 27](#)

IMS Fast Path Advanced Tool

New functions introduced

- The Extract function retrieves segment data from multiple DEDB areas and writes that data into sequential files in a user specified format. For more information, see [Chapter 10, “Extracting a DEDB offline with FPA,” on page 259](#).
- The DMAC Print function prints the contents of the DMAC blocks for multiple DEDB areas. For more information, see [Chapter 11, “Printing DMAC blocks in a DEDB offline with FPA,” on page 283](#).

Image copy support

The Analyze, Change, DMAC Print, Extract, and Unload functions support image copy data sets as input data sets. Also, the Analyze, Change, Reload, and Unload functions are enhanced to generate image copies.

IMS 10 support

FPA is enhanced to support IMS 10.

XCI randomizer interface for batch support

The support for the XCI randomizer interface for batch is added to the Unload function and the Change function. The XCI randomizer interface is provided by IMS 10 APAR PK40256.

IMS Fast Path Basic Tools

DEDB Pointer checker DD statements enhancements

- FABADA3 supports new DDs DAR13S, DAR3CS, DAR13R, and SORT13R as substitutes for DAR13SI, DAR3CSI, DAR13O, and SORTR13.

- FABADA5 supports new DDs DAR12 and DAR13R as substitutes for DAR12I and DAR13I.

Area information record suppression support for unloaded segment records

- FABCUR1 supports the same LRECL value as BMC Trimar's unloaded FP segment records file. For more information, see the topic "DEDB Unload SYSIN DD data set control statements" in the *IMS Fast Path Solution Pack: IMS Fast Path Basic Tools User's Guide*.
- FABCUR1/FABCUR3 supports empty areas when AREC=NO is specified. For more information, see the AREC keyword in the *IMS Fast Path Solution Pack: IMS Fast Path Basic Tools User's Guide*:
 - Topic "DD statements for DEDB Unload"
 - Topic "DD statements for DEDB Reload"
- FABCUR6 supports the same LRECL value as BMC Trimar's unloaded FP segment records file. For more information, see the LRECL keyword in the topic "UR6CTL DD data set" in the *IMS Fast Path Solution Pack: Supplementary Utilities User's Guide*.
- FABCUR6/FABCUR7 supports an empty area when AREA_INFORMATION_RECORD=NO is specified. For more information, see the AREA_INFORMATION_RECORD keyword in the *IMS Fast Path Solution Pack: Supplementary Utilities User's Guide*:
 - Topic "UR6CTL DD data set"
 - Topic "UR7CTL DD data set"

Site default options support

Programs FABCUR6 and FABCUR9 support site default options.

HD Unload to DEDB Unloaded Segment Data Set Conversion utility support

HD Unload to DEDB Unloaded Segment Data Set Conversion utility (FABCUR8) is introduced, which can convert an HD unload data set to a DEDB unloaded segment records data set. For more information, see the topic "HD To DEDB Unload Data Set Conversion utility" in the *IMS Fast Path Solution Pack: Supplementary Utilities User's Guide*.

IMS 10 support

FPB is enhanced to support IMS 10.

FABCUR6 and FABCUR7 enhancements

- FABCUR6 and FABCUR7 support suppression of area information record.
- FABCUR6 supports the OUTDD=YES|NO control statement to make output DD, UR6DBDFN, DURDzzzO (or XDzzzzzO), DURDzzzE (or XDzzzzzE), DURSzzzO (or XSzzzzzO) DD essential or optional.
- FABCUR6 supports EMPTY=YES for areas that are specified in the FILECTL control statements.
- FABCUR7 can be used to specify ACBLIB DD instead of UR7DBDFN DD.

For more information, see the following topics in the *IMS Fast Path Solution Pack: Supplementary Utilities User's Guide*:

- "DEDB Reload Segment Data Set Create utility"
- "DEDB Unloaded Segment Data Set Retrieve utility"

XCI randomizer interface for batch support

The support for the XCI randomizer interface for batch is added to DEDB Tuning Aid, DEDB Unload, DEDB Reload. The XCI randomizer interface is provided by IMS 10 APAR PK40256.

IMS Fast Path Online Tools

New tools introduced

- The Online Expert Reorganization utility reorganizes a set of UOWs in a DEDB area on the basis of the RBASEFS and RDOVFFS criteria. For more information, see [Chapter 23, "Reorganizing a DEDB online with FPO," on page 779](#)

- The Online DMAC Print utility provides maps for the storage that is used by the IMS control region by using the IMS mapping macro and the IMS storage dump of DMAC. For more information, see Chapter 24, “[Printing DMAC blocks in a DEDB online with FPO,](#)” on page 821.

IMS 10 support

FPO is enhanced to support IMS 10.

Report enhancements

The Online Pointer Checker utility and the Online Data Extract utility provide breakdown of segment types in the SDEP segment details (For Segment Read) field of the statistics reports.

New method for extending ADSs

The Online Area Extender utility is enhanced to support new method for extending ADSs. For more information, see “[OAE Rename program](#)” on page 771 and “[EXTEND_TYPE keyword](#)” on page 875.

Online Expert Reorganization enhancements

- Supports a new keyword REORGMSG to control the amount of reorganization messages. For more information, see “[REORGMSG keyword](#)” on page 907.
- Prints the number of IOVF CIs used and IOVF CIs reclaimed in the summary report.
- Supports five-digit page number.

XCI randomizer interface for batch support

The support for the XCI randomizer interface for batch is added to the Online Data Extract utility. The XCI randomizer interface is provided by IMS 10 APAR PK40256.

Supplementary utilities

XCI randomizer interface for batch support

The support for the XCI randomizer interface for batch is added to the FABCUR6 and FABEUR6 programs and to the IMS DEDB randomizing module. The XCI randomizer interface is provided by IMS 10 APAR PK40256. For more information about IMS DEDB randomizing module, see the topic "IMS DEDB randomizing module" in the *IMS Fast Path Solution Pack: Supplementary Utilities User's Guide*.

Support for DEDB Hash pointer check under IMS Database Recovery Facility

The single step DEDB Hash pointer checking function can be invoked under IMS Database Recovery Facility. For more information, see the topic "HPFPU Hash Check support for IMS Database Recovery Facility" in the *IMS Fast Path Solution Pack: Supplementary Utilities User's Guide*.

Compatibilities with former releases of the product and migration considerations

All the utilities and tools provided in this release of IMS Fast Path Solution Pack are compatible with those provided in earlier releases of IMS Fast Path Solution Pack.

Compatibilities

JCL streams and the input control statements that are used in an earlier release of IMS Fast Path Solution Pack are compatible with this release of IMS HP Fast Path Utilities.

For information about compatibilities, see the following topics:

- “[IMS Fast Path Advanced Tool compatibility](#)” on page 37
- “[IMS Fast Path Online Tools compatibility](#)” on page 672

For IMS Fast Path Basic Tools, see the topic "IMS Fast Path Basic Tools compatibility" in the *IMS Fast Path Solution Pack: IMS Fast Path Basic Tools User's Guide*.

Migration considerations

To migrate from an earlier release of IMS FP Solution Pack to this release, see the topic "Migration considerations" in *IMS Fast Path Solution Pack: Overview and Customization*.

Product documentation and updates

IMS Tools information is available at multiple places on the web. You can receive updates to IMS Tools information automatically by registering with the IBM My Notifications service.

Information on the web

Always refer to the IMS Tools Product Documentation web page for complete product documentation resources:

<https://www.ibm.com/support/pages/node/712955>

The IMS Tools Product Documentation web page includes:

- Links to [IBM Documentation](#) for the user guides ("HTML")
- PDF versions of the user guides ("PDF")
- Program Directories for IMS Tools products
- Recent updates to the user guides, referred to as "Tech docs" ("See updates to this information!")
- Technical notes from IBM Software Support, referred to as "Tech notes"
- White papers that describe product business scenarios and solutions

IBM Redbooks® publications that cover IMS Tools are available from the following web page:

<http://www.redbooks.ibm.com>

The IBM Information Management System website shows how IT organizations can maximize their investment in IMS databases while staying ahead of today's top data management challenges:

<https://www.ibm.com/software/data/ims/>

Receiving documentation updates automatically

To automatically receive automated emails that notify you when new technote documents are released, when existing product documentation is updated, and when new product documentation is available, you can register with the IBM My Notifications service. You can customize the service so that you receive information about only those IBM products that you specify.

To register with the My Notifications service:

1. Go to <http://www.ibm.com/support/mysupport>
2. Enter your IBM ID and password, or create one by clicking **register now**.
3. When the My Notifications page is displayed, click **Subscribe** to select those products that you want to receive information updates about. The IMS Tools option is located under **Software > Information Management**.
4. Click **Continue** to specify the types of updates that you want to receive.
5. Click **Submit** to save your profile.

How to send your comments

Your feedback helps IBM to provide quality information. Send any comments that you have about this book or other IMS Tools documentation to comments@us.ibm.com. Include the name and version number of the product and the title and number of the book. If you are commenting on specific text, list the location of the text (for example, a chapter, topic, or section title).

Prerequisite knowledge and publications

Before using this information, you should understand basic IMS concepts, the IMS environment, and your installation's IMS system.

The IMS publications are prerequisite for all IMS HP Fast Path Utilities components.

Related publications

For information about IMS Fast Path Basic Tools and supplementary utilities, see the following information:

- *IMS Fast Path Solution Pack: IMS Fast Path Basic Tools User's Guide* (SC27-9597)
- *IMS Fast Path Solution Pack: Supplementary Utilities User's Guide* (SC27-9598)

This information refers to information in other guides using shortened versions of the information titles. The following table contains a list of information referred to by their short titles:

Short title used in this information	Title	Order number
<i>IMS Database Repair Facility User's Guide</i>	<i>IBM IMS Database Repair Facility for IMS Solution Packs User's Guide</i>	SC19-2916
<i>IMS Fast Path Solution Pack: Overview and Customization</i>	<i>IBM IMS Fast Path Solution Pack for z/OS 2.1: Overview and Customization</i>	GC27-9596
<i>IMS High Performance Image Copy User's Guide</i>	<i>IBM IMS High Performance Image Copy for z/OS4.2 User's Guide</i>	SC19-2756
<i>IMS Library Integrity Utilities User's Guide</i>	<i>IBM IMS Library Integrity Utilities for z/OS 2.2 User's Guide</i>	SC19-3979
<i>IMS Recovery Solution Pack IMS Database Recovery Facility User's Guide</i>	<i>IBM IMS Recovery Solution Pack for z/OS 2.1 IMS Database Recovery Facility User's Guide</i>	SC27-8441
<i>IMS Solution Packs Data Sensor User's Guide</i>	<i>IBM IMS Solution Packs Data Sensor User's Guide</i>	SC19-3283
<i>IMS Solution Packs IMS Database Space Tuning Guide</i>	<i>IBM IMS Solution Packs IMS Database Space Tuning Guide</i>	SC19-3395
<i>Management Console User's Guide</i>	<i>IBM Management Console for IMS and Db2 for z/OS 1.1 User's Guide</i>	SC27-6714
<i>Tools Base Autonomics Director User's Guide</i>	<i>IBM Tools Base for z/OS 1.6 Autonomics Director User's Guide</i>	SC19-4376
<i>Tools Base Configuration Guide for IMS</i>	<i>IBM Tools Base for z/OS 1.6 Configuration Guide for IMS</i>	SC19-4370
<i>Tools Base IMS Tools Common Services User's Guide</i>	<i>IBM Tools Base for z/OS 1.6 IMS Tools Common Services User's Guide</i>	SC19-4371
<i>Tools Base IMS Tools Knowledge Base User's Guide</i>	<i>IBM Tools Base for z/OS 1.6 IMS Tools Knowledge Base User's Guide</i>	SC19-4372
<i>Tools Base Policy Services User's Guide</i>	<i>IBM Tools Base for z/OS 1.6, Policy Services User's Guide</i>	SC19-4374

IMS HP Fast Path Utilities accessibility features

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use a software product successfully.

Accessibility features

The major accessibility feature in IMS HP Fast Path Utilities is the keyboard-only operation for ISPF editors. It uses the standard TSO/ISPF interface.

Keyboard navigation

You can access the information center and IMS ISPF panel functions by using a keyboard or keyboard shortcut keys.

For information about navigating the IMS ISPF panels using TSO/E or ISPF, refer to the following publications

- *z/OS ISPF User's Guide, Volume 1*
- *z/OS TSO/E Primer*
- *z/OS TSO/E User's Guide*

These guides describe how to use ISPF, including the use of keyboard shortcuts or function keys (PF keys), include the default settings for the PF keys, and explain how to modify their functions.

IBM and accessibility

See the IBM Human Ability and Accessibility Center at www.ibm.com/able for more information about the commitment that IBM has to accessibility.

Part 2. IMS Fast Path Advanced Tool

The following topics introduce IMS Fast Path Advanced Tool and explain how to use the functions that are provided by the tool.

Topics:

- [Chapter 2, “Overview of IMS Fast Path Advanced Tool,” on page 33](#)
- [Chapter 3, “Understanding FPA,” on page 39](#)
- [Chapter 4, “Unloading a DEDB offline with FPA,” on page 51](#)
- [Chapter 5, “Reloading a DEDB offline with FPA,” on page 89](#)
- [Chapter 6, “Changing a DEDB offline with FPA,” on page 129](#)
- [Chapter 7, “Reorganizing a DEDB offline with FPA,” on page 163](#)
- [Chapter 8, “Analyzing a DEDB offline with FPA,” on page 193](#)
- [Chapter 9, “Tuning a DEDB offline with FPA,” on page 229](#)
- [Chapter 10, “Extracting a DEDB offline with FPA,” on page 259](#)
- [Chapter 11, “Printing DMAC blocks in a DEDB offline with FPA,” on page 283](#)
- [Chapter 12, “Building secondary index databases for a DEDB offline with FPA,” on page 295](#)
- [Chapter 13, “Synchronizing secondary index databases with a DEDB with FPA,” on page 321](#)
- [Chapter 14, “Reclaiming CIs in secondary index databases with FPA,” on page 355](#)
- [Chapter 15, “Initializing a DEDB offline with FPA,” on page 367](#)
- [Chapter 16, “Command reference for FPA,” on page 381](#)
- [Chapter 17, “Sample reports of FPA,” on page 559](#)
- [Chapter 18, “Site default options support for FPA,” on page 657](#)

Chapter 2. Overview of IMS Fast Path Advanced Tool

The following topics provide you with an overview of IMS Fast Path Advanced Tool (also referred to as FPA).

Topics:

- [“Functions of FPA” on page 33](#)
- [“Benefits of FPA” on page 35](#)
- [“Program overview of FPA” on page 35](#)
- [“Restrictions of FPA” on page 36](#)
- [“IMS Fast Path Advanced Tool compatibility” on page 37](#)

Functions of FPA

IMS Fast Path Advanced Tool provides various functions for managing IMS data entry databases (DEDBs).

Unload

The Unload function of FPA efficiently unloads a single DEDB area. It can also concurrently unload multiple DEDB areas without impacting all areas of a database. Specifically, the Unload function unloads multiple DEDB areas to QSAM data sets. If necessary, a sort step can be included in the Unload process. Optionally an image copy can be generated after completing the Unload process.

Reload

The Reload function of FPA efficiently reloads a single DEDB area. It can also concurrently reload multiple DEDB areas without impacting all areas of a database. Specifically, the Reload function reloads multiple DEDB areas from the QSAM data sets. If necessary, a sort step and an analyze step can be included in the Reload process. Optionally an image copy can be generated after completing the Reload process.

Change

The Change function of FPA efficiently reorganizes or restructures a single DEDB area. It can also concurrently reorganize or restructure multiple DEDB areas without impacting all areas of a database. Specifically, the Change function reorganizes or restructures multiple DEDB areas in one step. If necessary, an analyze step can be included in the Change process. Optionally an image copy can be generated after completing the Change process.

Reorg

The Reorg function reorganizes a set of UOWs in a DEDB area on the basis of a ranked set of UOWs. The Reorg function includes the UOW Evaluation phase where the reorganization free space score (RFS score) for a set of UOWs of the DEDB area is determined. In the UOW Ranking phase that follows the UOW Evaluation phase, the Reorg function ranks the set of UOWs according to the RFS score. Finally, the highly ranked UOWs are reorganized up to the user-specified number of UOWs.

The Reorg function reduces the possibility of having too many or too few UOWs reorganized by ranking the set of UOWs in the DEDB area, and allows the user to select how the subset of the ranked set of UOWs is chosen. Consequently, the user can limit the size of the subset of UOWs that will be reorganized before the reorganization is done.

Analyze

The Analyze function provides a fast, efficient mechanism for verifying the integrity of all IMS pointer values, free space element chains, VSAM control fields, and space utilization in IMS DEDBs. It also provides simple integrity verification of pointer segments in secondary index databases. The Analyze function provides the comprehensive analysis information required to effectively manage the space utilization, performance characteristics, and physical attributes of IMS DEDBs. The Analyze function uses a VSAM ESDS or an image copy as input. The Analyze function can be an aid for database administrators and technical support personnel in database management and maintenance. Optionally an image copy can be generated after completing the Analyze process.

Tune

The Tune function provides a fast and efficient mechanism for tuning databases. The function simulates changes in the DBD parameters and randomizers without loading the segments letting you select the physical database attributes that satisfy the performance and space utilization requirements.

The Tune function provides a comprehensive tuning facility for managing databases and optimal performance.

Extract

The Extract function provides an easy-to-use, flexible method to extract segment data from one or more offline DEDB areas and writes the output to QSAM data sets.

The Extract function can be used as an alternative to writing your own application to extract DEDB data. Other potential uses of this function include:

- Record reporting
- Extraction of records to populate a test database

DMAC Print

The DMAC Print function provides maps of the contents of the DMAC blocks for one or more areas by using the IMS mapping macro and the DMAC dump. DMAC is an important CI that has the information for ROOT, UOW definition, and SDEP LB/LE values, and so on, for a DEDB area data set. The DMAC Print function maps the global section and the local section with offset, label, and values. It also provides the dump that is specified by the DMAC length, EQE list, CUSN, and RBA.

Build Index

The Build Index function builds the secondary index databases from multiple DEDB areas.

Resync

The Resync function synchronizes a DEDB and its secondary index databases when they are out of synchronization. This function runs against online areas.

Freespace Reclaim

The Freespace Reclaim function reclaims the control intervals (CIs) in secondary index database data sets.

Initialize

The Initialize function of FPA efficiently initializes a single DEDB area. It can also concurrently initialize multiple DEDB areas.

Benefits of FPA

IMS Fast Path Advanced Tool offers a number of benefits for database administrators and technical support personnel who are involved in database management, maintenance, and performance tuning.

IMS Fast Path Advanced Tool provides the following benefits:

- Provides JCL ease of use
- Increases application availability
- Increases the productivity of database support personnel
- Reduces maintenance and database conversion costs significantly
- Eliminates the manual steps for unloading, reloading, restructuring, and analyzing databases
- Reduces the amount of time that unloads, reloads, and restructures traditionally require for DEDB to reclaim space
- Reduces the amount of time required for DEDB structure changes
- Significantly reduces I/O through concurrent unloads, reloads, and restructures
- Simplifies the run of required function by using command language
- Provides a site default option to override system default values and saves the effort of specifying keywords in HFPSYSIN DD

Program overview of FPA

IMS Fast Path Advanced Tool makes it easy to develop standardized JCL (that is, a single driver) for unloading, reloading, restructuring, reorganizing, analyzing, tuning, extracting, and initializing DEDBs.

It also provides the following functions:

- Reorganizes or restructures a DEDB in only one step, without running the sort step.
- Reloads, reorganizes, or restructures multiple DEDB areas, and the process of analyzing the DEDB areas in one step.
- Analyzes, unloads, reloads, reorganizes, or restructures multiple DEDB areas, and supports the process of generating the image copies by using IMS HP Image Copy utility in one step.
- Simulates changes in the DBD parameters and randomizers without loading the segments.

FPA offers the following features:

- By specifying control statements, a single program can run multiple functions.
- The following input and output data sets can be allocated dynamically to minimize JCL DD statements.
 - DEDB area data sets for:
 - The input of the Analyze process
 - The input and output of the Change process
 - The input of the DMAC Print process
 - The input of the Extract process
 - The input of the Unload process
 - The input of the Tune process
 - The output of the Reload process
 - The output of the Initialize process
 - The renamed area data set of the Change process
 - The Reorg process
 - The input of the Build Index process
 - Secondary index database data sets for:

- The input of the Analyze process
- The output of the Build Index process
- ACB libraries
- DBRC RECON data sets
- Data sets of unloaded segment records for the output of the Unload process and the input of the Reload process
- Image copy data sets for:
 - The input of the Analyze process
 - The input of the Change process
 - The input of the DMAC Print process
 - The input of the Extract process
 - The input of the Unload process
 - The input of the Tune process
 - The input of the Build Index process
- Output data set for the messages
- Output data set for the reports
- SORT work data sets
- Site default table
- A site default table overrides system default values and saves the effort of specifying keywords in HFPSYSIN DD. To create the site default table, run the FPA program by using the same JCL and control statements.
- Multiple areas of a DEDB can be analyzed, extracted, initialized, unloaded, reloaded, restructured (changed), simulated (tuned), or reorganized concurrently in multiple subtasks as long as system resources are available. The number of concurrent processes can be controlled by an optional parameter of a control statement.
- The DBRC environment that is supported is the same as that supported in IMS standard utilities. Before running the Unload or the Reload process, an area authorization is obtained from DBRC to keep the integrity of the area.
- Multiple area data sets (ADSs) are supported except for the Reorg process. One of the available error-free ADSs that are registered with DBRC will be selected for unloading, analyzing, or extracting. All or selected ADSs registered with DBRC can be made available after reloading. The ADSs for input are not checked with DBRC for changing.
- In the Build Index process, multiple areas of a DEDB can be scanned concurrently and multiple secondary index databases can be loaded concurrently in multiple subtasks as long as system resources are available. The number of concurrent processes can be controlled by the optional parameter of the control statement.

Restrictions of FPA

Before using IMS Fast Path Advanced Tool, you need to understand the restriction of IMS Fast Path Advanced Tool.

When a randomizer is invoked by IMS, register 10 contains the addresses of the EPST, and register 11 contains the addresses of the ESCD. Some user-written or user-customized randomizers are designed to use these addresses to gain access to IMS control blocks other than those passed as part of the published interface. To show that the call is not being issued in an IMS online environment, when one of the IMS Fast Path Advanced Tool functions issues a call to a randomizing module, register 10 is set to -1 and register 11 is set to 0.

However, if the IMSID parameter is specified on the EXEC statement, FPA generates the buffers for temporary ESCD and SCD. Register 11 contains the address of the temporary ESCD when FPA issues a call

to a randomizing module. FPA sets only the ESCDSCD field, which contains the address of SCD, in the temporary ESCD. Also, FPA sets only the SSCDIMID field, which contains the IMS system ID, in the temporary SCD. Other fields of temporary ESCD and temporary SCD cannot be referred to.

IMS Fast Path Advanced Tool compatibility

The following topics describe the compatibilities of the tools within IMS HP Fast Path Utilities.

Compatibility between IMS Fast Path Advanced Tool and former releases of IMS Fast Path Advanced Tool

The compatibility between IMS Fast Path Advanced Tool (FPA) and the former releases of FPA is described in this topic.

The JCL streams and the input control statements that are used in the former releases of FPA are compatible with this release of FPA.

The following items that are created in the former releases of FPA can be used in this release of the product:

- The unloaded segment records data set that is created by the FPA Unload function. You can use the unloaded segment records data set that is created by the former release of FPA Unload function with the FPA Reload function of this release.
- The total segments records file that is created by the FPA Unload function.
- The History file records that are created by the FPA Analyze function.
- The History2 file records that are created by the FPA Analyze function.

Site default options that are used in the former releases of the product can be used in this release of the product. However, if the control statements or the keywords that are introduced in this release are used as site default options, those site default options cannot be used in the former releases.

When the existing RELOAD command JCL is invoked, the Analyze function is automatically processed with PTRCHKLVL=QUICK after the Reload function because IBM IMS High Performance Fast Path Utilities for z/OS 2.2 (product number 5655-R05), provides the Analyze function. To keep the IBM IMS High Performance Fast Path Utilities for z/OS 2.1 Reload JCL compatible, you need to run the HFPCZZJ JCL and prepare the HFPCZZ00 module (migration module) in IBM IMS High Performance Fast Path Utilities for z/OS 2.2. In this version of IBM IMS Fast Path Solution Pack, IMS High Performance Fast Path Utilities, you can use the site default option and change the PTRCHKLVL option. FPA can still process the HFPCZZ00 module, but the member HFPCZZJ is not provided.

Note: When the site default options are used, the HFPCZZ00 module (migration module) is ignored.

Compatibility between IMS Fast Path Advanced Tool and IMS Fast Path Basic Tools

There are compatibilities and incompatibilities between the IMS Fast Path Advanced Tool (FPA) functions and the IMS Fast Path Basic Tools (FPB) utilities.

Product material

The following items are compatible:

- The unloaded segment records data set that is created by the FPA Unload function and the FPB DEDB Unload utility. You can use the unloaded segment records data set that is created by FPB DEDB Unload utility for FPA Reload function.
- The total segments records file that is created by the FPA Unload, Reload, Change functions, and the FPB DEDB Unload utility.
- The History file records that are created by the FPA Analyze function and the FPB DEDB Pointer Checker utility.

The following items are not compatible:

- The RBA values for segment allocation differ due to design difference when the LOADCTL subcommand is specified for the FPA Reload function and when the LOADCTL control statement is specified for the FPB Unload utility.
- The History2 file records that are created by the FPA Analyze function and the FPB DEDB Pointer Checker utility. The logic to round off the values has changed. You might see a slight difference in some fields.
- The tuning aid record data sets, which are used by the FPA Tune function, are not compatible with the FPB DEDB Tuning Aid utility.

DD statements

DD statements are not compatible.

Utility control statements

The utility control statements, keywords, and parameters are not compatible.

Size of intermediate storage data sets

The size of the following intermediate storage data sets that are used by DFSORT are compatible:

- In FPA, the size of *itttWKnn* DD data set that is used when PTRCHKLVL=FULL is specified, is similar to the size of SORTWKnn DD data set that is used in the FPB DEDBPC SORT3CS step.
- In FPA, the size of *itttWKnn* DD data set that is used when CISNAP=YES is specified, is similar to the size of SORTWKnn DD data set that is used in the FPB DEDBPC SORT4 step.
- In FPA, the size of *itttWKnn* DD data set that is used when the REPORT subcommand is specified, is similar to the total size of SORTWKnn DD data sets that are used in the FPB DEDBPC SORT12 step and in the FPB DEDBPC SORTR13 step.

Compatibility between IMS Fast Path Advanced Tool and IMS Fast Path Online Tools

There are compatibilities and incompatibilities between the IMS Fast Path Advanced Tool (FPA) functions and the IMS Fast Path Online Tools (FPO) utilities.

Product material

The following items are compatible:

- The Extracted records data set that is created by the FPA Extract function and the FPO ODE utility.
- The History and History 2 file records that are created by the FPA Analyze function and the FPO OPC utility.
- The sensor data that is collected from DEDB areas and that is stored in the IMS Tools KB Sensor Data repository by the FPA Analyze function and the FPO OPC utility.
- The tuning aid record data sets that are created by the FPA Tune function, FPO OPC utility, and the FPO OER utility.

DD statements

DD statements are not compatible.

Utility control statements

The utility control statements, keywords, and parameters are not compatible.

Chapter 3. Understanding FPA

Before using FPA, you need to understand how FPA works.

Topics:

- [“Program structure of FPA” on page 39](#)
- [“System flow of FPA” on page 40](#)
- [“Example JCL statements for FPA ” on page 40](#)
- [“Checkpoint restart option ” on page 41](#)

Program structure of FPA

To run the required functions effectively, for multiple areas concurrently, FPA is composed of various internal function tasks.

FPA operates independently of the IMS control region.

The internal function tasks include:

Main task

FPA main program controls the internal function tasks that are required to run the user-requested functions that are specified by control statements.

Command subtask

Commands, subcommands, their associating parameters specified in the control statements (HFPSYSIN DD), and a site default table are analyzed to prepare internal control blocks to perform the requested function.

Function subtask

The major function for each command is controlled by this task. FPA can run multiple functional tasks concurrently.

Thread subtask

One or multiple thread tasks are controlled by one function task. Multiple thread tasks can run the Unload or the Reload process concurrently to process multiple areas that belong to a DEDB. There are two types of thread tasks, the input and the output thread tasks:

Input thread subtask

The input thread subtask performs the input area process.

Output thread subtask

The output thread subtask performs the write process for the area Reload process.

Service subtask

FPA system resources, such as the internal control blocks and I/O buffers, are controlled as resource pools to make use of the limited system resources efficiently.

Report subtask

Reports, which are the results of the function process, are produced for each command. Report process and function process can be run concurrently when multiple commands are requested.

DBRC subtask

FPA communicates with IMS DBRC to obtain the authorization for an area, and to update the information about the area that is registered to a RECON.

System flow of FPA

The following figure shows the system flow of FPA.

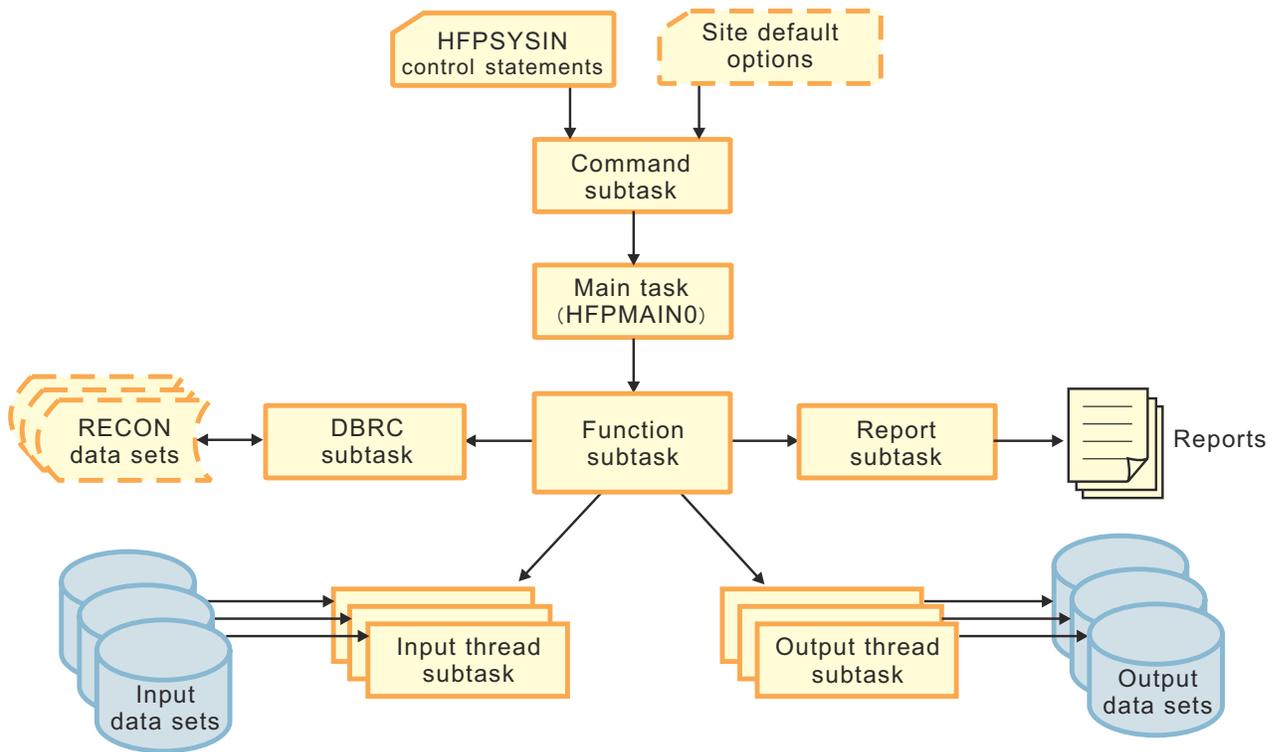


Figure 1. System flow of FPA

Example JCL statements for FPA

The following figure shows example JCL statement for verifying the syntax of the UNLOAD command.

This JCL verifies the syntax of the UNLOAD command of “[Example 1: Unloading an area that is registered with DBRC](#)” on page 74.

```
//HFP EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES,
  SCAN=YES
UNLOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  OAREA=(DB22AR0),
  ODSNMASK='HPFP.USRFILE.&AREA'
/*
```

Figure 2. Verifying the syntax of the UNLOAD command

Checkpoint restart option

The Unload and the Reload functions of FPA support checkpoint restart capability. You can enable the checkpoint restart option to reprocess only the failed areas, without reprocessing previously successfully processed areas, by rerunning the same JCL stream that was used in the previous run.

When this option is enabled, the job runs in checkpoint mode for the first run. During the run, the function creates a checkpoint file and, for each area, records the result of each processing. In subsequent runs of the same job, which run in restart mode, the function uses the checkpoint file to identify the failed areas, and runs the remaining processing for those areas. The checkpoint file is updated every time the same job is run, so you can continue to use the same JCL until all the areas are successfully processed.

Important: For a restart to begin, the FPA Unload or the FPA Reload environment must be identical to the one that existed at the time of the checkpoint. If any areas have been added or subtracted, the restart is invalidated.

Checkpoint restart for the Unload function

When the checkpoint restart option is enabled, the Unload function works as follows:

- When the Unload function fails to unload some areas, only the failed areas are unloaded in the next run.
- When the Unload function successfully unloads the areas but fails to create image copies for some areas, image copies for the failed areas are created in the next run (areas are not unloaded again in the subsequent runs.)
- When the Unload function successfully unloads and creates the image copies of the areas but fails to issue the NOTIFY.IC command to register the image copies to DBRC for some areas, the NOTIFY.IC commands for the areas are issued in the next run (areas are not unloaded and image copies are not created again in the subsequent runs.)

The checkpoint restart option reprocesses the areas that are in the following statuses:

- For unload processing that runs in REORG mode, the area with a return code of 8 or higher or the area that ended abnormally is reprocessed. All other areas that are to be unloaded to the same unload data set are also reprocessed.
- For unload processing that runs in non-REORG mode (restructure), if processing for an area ended with a return code of 8 or higher or ended abnormally, all the areas that are specified in the JCL are reprocessed.
- The area that failed during image copy processing is reprocessed. All the other areas whose image copies are to be created on the same stack are also reprocessed. Because the unload processing for the areas succeeded in the previous run, the unload processing for the areas is skipped in the restart processing.
- The area that failed or was skipped while the NOTIFY.IC command was being issued is reprocessed. Because the unload processing and the image copy processing for this area succeeded in the previous run, the unload processing and the image copy processing are skipped in the restart processing.

Checkpoint restart for the Reload function

When the checkpoint restart option is enabled, the Reload function works as follows:

- When the Reload function fails to reload some areas, all areas are reloaded again in the next run.
- When the Reload function successfully reloads all the areas but fails to analyze some areas with a return code of 12, all areas are analyzed again in the next run (areas are not reloaded again in the subsequent runs.)
- When the Reload function successfully reloads all the areas but fails to analyze some areas with a return code of 8, which indicates that pointer errors were found in the areas, the failed areas are not analyzed in the next run.

- When the Reload function successfully reloads all the areas but fails to create image copies for some areas, image copies for the failed areas are created in the next run (areas are not reloaded again in the subsequent runs.)
- When the Reload function successfully reloads and analyzes all the areas, and then creates image copies of an area, it issues NOTIFY.IC commands to register the image copies of the area to DBRC. If the function fails to issue NOTIFY.IC commands for some areas, the NOTIFY.IC commands for the areas are issued in the next run (areas are not reloaded, analyzed, and image copies are not created again in the subsequent runs.)

The checkpoint restart option reprocesses the areas that are in the following statuses:

- For reload processing, if processing for an area ended with a return code of 8 or higher or ended abnormally, all the areas that are specified in the JCL are reprocessed.
- For analyze processing, if processing for an area ended with a return code of 12 or ended abnormally, all the areas that are specified in the JCL are reprocessed. Because the reload processing for the areas succeeded in the previous run, the reload processing for the areas is skipped in the restart processing. If the image copy processing for the areas also succeeded in the previous run, the image copy processing for the areas is skipped in the restart processing.
- The area that failed during image copy processing is reprocessed. All the other areas whose image copies are to be created on the same stack are also reprocessed. Because the reload processing for the areas succeeded in the previous run, the reload processing for the areas is skipped in the restart processing. If the analyze processing for the areas also succeeded in the previous run, the analyze processing for the areas is skipped in the restart processing.
- The area that failed or was skipped while the NOTIFY.IC command was being issued is reprocessed. Because the reload processing, analyze processing, and the image copy processing for this area succeeded in the previous run, the reload processing, analyze processing, and the image copy processing are skipped in the restart processing.

When all areas are processed successfully in the checkpoint processing or in the restart processing, the contents of the checkpoint file is cleared. If the same job is resubmitted after the contents of the checkpoint file has been cleared, the job runs in checkpoint mode, which means that all the areas that are specified for the job are processed.

Note: When RESTART=NO, the NOTIFY.IC command is issued to DBRC even when an error is detected either in the primary image copy data set or in the secondary image copy data set. However, when RESTART=YES, the NOTIFY.IC command is not issued to DBRC during that run of the job for image copy data sets that contain errors.

To enable the checkpoint restart option, make the following specifications to your JCL:

- Specify the checkpoint file on the HFPCHKPT DD statement.
- Specify the RESTART=YES keyword.

Scenario: Unloading all areas of a DEDB by using the checkpoint restart option

To unload all areas of a DEDB by using the checkpoint restart option, modify your unload JCL accordingly. Then, run the job and check the reports. If processing fails for one or more areas, correct the error conditions and repeat the steps until all areas are processed.

About this task

The following steps describe how to use the checkpoint restart option to unload all areas of a DEDB, create image copies of the areas, and register the image copy data sets to DBRC.

In these steps, the following conditions are assumed:

- DEDB, DEDBJN22, has nine areas.
- In DBRC, all ADSs are in the AVAILABLE status.
- The following figure shows the flow of this scenario:

- In the first run of the job, five areas ended with errors.
- In the second run, two areas ended with errors.
- In the third run, the remaining two areas were processed successfully.

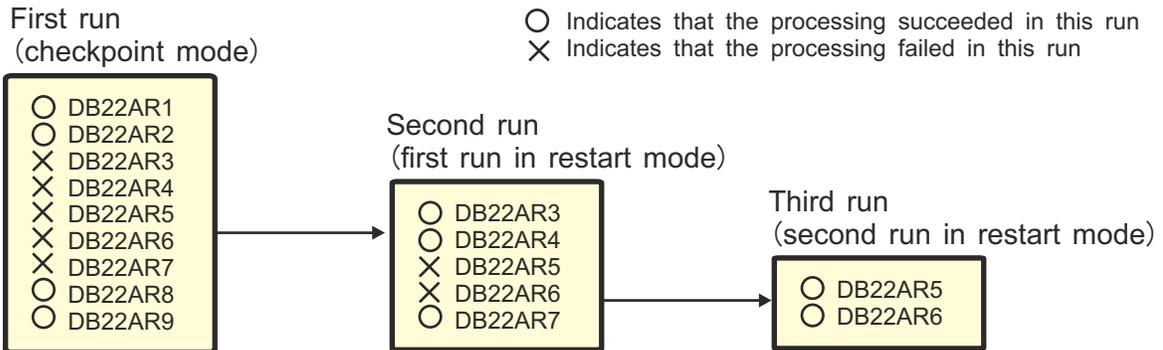


Figure 3. Flow of the checkpoint restart option (FPA Unload function)

Procedure

1. Modify the JCL stream for the Unload function to enable the checkpoint restart option by including the HFPCHKPT DD statement and RESTART parameters, as shown in the following figure:

```

//HFPEXE EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=HPS.SHPSLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPPRINT DD SYSOUT=*
//HFPRPTS DD SYSOUT=*
//ICEPRINT DD SYSOUT=*
//HFPCHKPT DD DISP=SHR,DSN=HPFP.HFPCHKPT
//STACK1 DD UNIT=TAPE,DSN=HFP001.ICDS,
// DISP=OLD,VOL=SER=HFP001
//STACK2 DD UNIT=TAPE,DSN=HFP002.ICDS,
// DISP=OLD,VOL=SER=HFP002
//STACK3 DD UNIT=TAPE,DSN=HFP003.ICDS,
// DISP=OLD,VOL=SER=HFP003
//STACK4 DD UNIT=TAPE,DSN=HFP004.ICDS,
// DISP=OLD,VOL=SER=HFP004
//HFPSYSIN DD *
GLOBAL DBRC=YES,
RESTART=YES
UNLOAD
DBD=DEDBJN22,
IAREA=ALL,
OAREA=ALL,
ODSNMASK='HPFP.USERFILE.&AREA',
KEYSEQCHK=ABEND,
ICOUT=YES
/*
//ICEIN DD *
GLOBAL ICHLQ=HFPLICDS,UNIT=TAPE
IC DBD=DEDBJN22,AREA=DB22AR1,STACK=STACK1
IC DBD=DEDBJN22,AREA=DB22AR2,STACK=STACK1
IC DBD=DEDBJN22,AREA=DB22AR3,STACK=STACK2
IC DBD=DEDBJN22,AREA=DB22AR4,STACK=STACK3
IC DBD=DEDBJN22,AREA=DB22AR5,STACK=STACK4
IC DBD=DEDBJN22,AREA=DB22AR6,STACK=STACK4
IC DBD=DEDBJN22,AREA=DB22AR7,STACK=STACK2
IC DBD=DEDBJN22,AREA=DB22AR8,STACK=STACK1
IC DBD=DEDBJN22,AREA=DB22AR9,STACK=STACK1
/*

```

Figure 4. Sample JCL stream for checkpoint restart (FPA Unload function)

2. Submit the JCL and check the job log for errors.

- If the job log contains no errors, all the areas were processed successfully. The checkpoint restart processing is not necessary.
 - If the job ended with a return code of 8, continue with Step “3” on page 44.
3. View the Process Summary report that is generated in the HFPRPTS DD.

The following figure shows an example Process Summary report that contains errors.

```

IMS HPFP UTILITIES - FPA UNLOAD          "Process Summary Report"          PAGE: 1
5698-FPP V2R1                          2019-11-22 18:55:26

```

DBDNAME	AREANAME	AREA NO	UL RC	IC RC	IC REG	STATUS (If not blank, see messages in HFPPRINT/ICEPRINT.)
DEDBJN22	DB22AR1	1	00	00	REG	
DEDBJN22	DB22AR2	2	00	00	REG	
DEDBJN22#	DB22AR3	3	08			Processing ended with errors.
DEDBJN22#	DB22AR4	4	08			Processing ended with errors.
DEDBJN22#	DB22AR5	5	00	08		Processing ended with errors.
DEDBJN22#	DB22AR6	6	00	08		Processing ended with errors.
DEDBJN22#	DB22AR7	7	08			Processing ended with errors.
DEDBJN22	DB22AR8	8	00	00	REG	
DEDBJN22	DB22AR9	9	00	00	REG	

Figure 5. Sample Process Summary report (FPA Unload function - checkpoint mode)

From this report, you can identify the status of the job as follows:

- From the unload return code column (UL RC), unload processing did not complete for areas DB22AR3, DB22AR4, and DB22AR7.
- From the image copy return code column (IC RC), image copy processing ended with errors for area DB22AR5 and DB22AR6.
- From the status column (STATUS), processing completed successfully for DB22AR1, DB22AR2, DB22AR8, and DB22AR9.

Based on this report, five areas that have a number sign (#) at the beginning of AREANAME need to be reprocessed in the restart processing.

4. Check the error messages in the HFPPRINT DD data set to obtain information about the cause of errors.

Locate the error messages for areas DB22AR3, DB22AR4, DB22AR5, DB22AR6, and DB22AR7.

Assume that the following errors were recorded:

- HFPJ0008E for DB22AR3 and DB22AR7: This message indicates that pointer errors were detected in the target areas.
- HFPT0005E for DB22AR4: This message indicates that a key sequence error was detected in the target area.
- HFPI0005E for DB22AR5 and DB22AR6: This message indicates that errors occurred in the image copy processing.

To resolve the errors, correct the pointer errors for area DB22AR3 and DB22AR7, and the key sequence error for area DB22AR4.

For areas DB22AR5 and DB22AR6, check the error messages in the ICEPRINT DD data set to obtain information about the cause of image copy processing errors.

Locate the messages for areas DB22AR5 and DB22AR6 in the ICEPRINT DD data set.

Assume that the following messages were recorded:

- FABJ4131I for DB22AR5, and DFS319A: These messages indicate that an I/O error occurred while the image copy data set was being accessed.

- FABJ4130I for DB22AR6: This message indicates that the processing for the area was skipped because the specified output image copy data set is a stacked data set that is written to the same stack as DB22AR5.

To resolve the image copy processing errors, identify the cause of the I/O error, and correct the error situation for DB22AR5.

5. Submit the same JCL and check the job log for errors.

- If the job log contains no errors, the remaining areas were processed successfully. You do not need to rerun the checkpoint restart processing.
- If the job ended with a return code of 8, continue with Step “6” on page 45.

6. View the Process Summary report that is generated in the HFPRPTS DD.

The following figure shows an example Process Summary report that still contains errors.

```

IMS HPFP UTILITIES - FPA UNLOAD                                "Process Summary Report"                                PAGE: 1
5698-FPP V2R1                                                2019-11-22 19:08:38
-----+-----+-----+-----+-----+-----+-----+-----+-----+
| RESTART MODE ( 1ST RUN IN RESTART MODE)                    |
| # indicates the area that is to be reprocessed in the next restart processing |
|-----+-----+-----+-----+-----+-----+-----+-----+-----+
| UL: UNLOAD  IC: HPIC  IC REG: NOTIFY.IC                    |
|-----+-----+-----+-----+-----+-----+-----+-----+-----+
DBDNAME  AREANAME  AREA NO  UL  IC  IC  STATUS (If not blank, see messages in HFPPRINT/ICEPRINT.)
-----+-----+-----+-----+-----+-----+-----+-----+-----+
DEDBJN22 DB22AR3    3  00  00  00 REG
DEDBJN22 DB22AR4    4  00  00  00 REG
DEDBJN22#DB22AR5    5  NA  08  Processing ended with errors.
DEDBJN22#DB22AR6    6  NA  08  Processing ended with errors.
DEDBJN22 DB22AR7    7  00  00  00 REG

```

Figure 6. Sample Process Summary report (FPA Unload function - restart mode with errors)

The Process Summary report that is generated in restart mode contains the area information for the areas that were reprocessed in the restart processing.

From this report, you can identify the status of the job as follows:

- From the image copy return code column (IC RC), the image copy processing ended with errors for area DB22AR5 and DB22AR6.
- From the status column (STATUS), processing completed successfully for DB22AR3, DB22AR4, and DB22AR7.

Based on this report, two areas that have a number sign (#) at the beginning of AREANAME need to be reprocessed in the restart processing.

7. Check the error messages in the HFPPRINT DD data set to obtain information about the cause of errors.

Locate the error messages for areas DB22AR5 and DB22AR6.

Assume that message HFPI0005E was recorded for DB22AR5 and DB22AR6. This message indicates that errors occurred in the image copy processing.

Check the error messages in the ICEPRINT DD data set to obtain information about the cause of image copy processing errors.

Locate the messages for areas DB22AR5 and DB22AR6 in the ICEPRINT DD data set.

Assume that the following messages were recorded:

- FABJ4131I for DB22AR5, and DFS319A: These messages indicate that an I/O error occurred while the image copy data set was being accessed.
- FABJ4130I for DB22AR6: This message indicates that the processing for the area was skipped because the specified output image copy data set is a stacked data set that is written to the same stack as DB22AR5.

To resolve the image copy processing errors, identify the cause of the I/O error, and correct the error situation.

8. Submit the same JCL and check the job log for errors.

- If the job log contains no errors, the remaining areas were processed successfully. You do not need to rerun the checkpoint restart processing.
- If the job ended with a return code of 8, repeat from Step “6” on page 45.

Assume that, this time, no errors were recorded in the job log.

9. View the Process Summary report that is generated in the HFPRPTS DD.

The following figure shows an example Process Summary report that does not contain any errors. This report indicates that all areas were processed successfully.

```

IMS HPFP UTILITIES - FPA UNLOAD                "Process Summary Report"
5698-FPP V2R1                                PAGE: 1
                                              2019-11-22 19:21:38
-----
| RESTART MODE ( 2ND RUN IN RESTART MODE)
| # indicates the area that is to be reprocessed in the next restart processing
|-----
| UL: UNLOAD  IC: HPIC  IC REG: NOTIFY.IC
|-----
DBDNAME  AREANAME  AREA NO  UL  IC  IC  STATUS (If not blank, see messages in HFPPRINT/ICEPRINT.)
-----  -
DEDBJN22 DB22AR5   5      NA  00  REG
DEDBJN22 DB22AR6   6      NA  00  REG
  
```

Figure 7. Sample Process Summary report (FPA Unload function - restart mode with no errors)

Scenario: Reloading all areas of a DEDB by using the checkpoint restart option

To reload all areas of a DEDB by using the checkpoint restart option, modify your reload JCL accordingly. Then, run the job and check the reports. If processing fails for one or more areas, correct the error conditions and repeat the steps until all areas are processed.

About this task

The following steps describe how to use the checkpoint restart option to reload all areas of a DEDB, analyze the areas, create image copies of the areas, and register the image copy data sets to DBRC.

In these steps, the following conditions are assumed:

- DEDB, DEDBJN23, has four areas.
- In DBRC, all ADSs are in the UNAVAILABLE status.
- The following figure shows the flow of this scenario:
 - In the first run of the job, one area ended with errors and all the other areas were skipped because of the errors.
 - In the second run, two areas ended with errors.
 - In the third run, the remaining two areas were processed successfully.

- Indicates that the processing succeeded in this run
- ✗ Indicates that the processing failed in this run

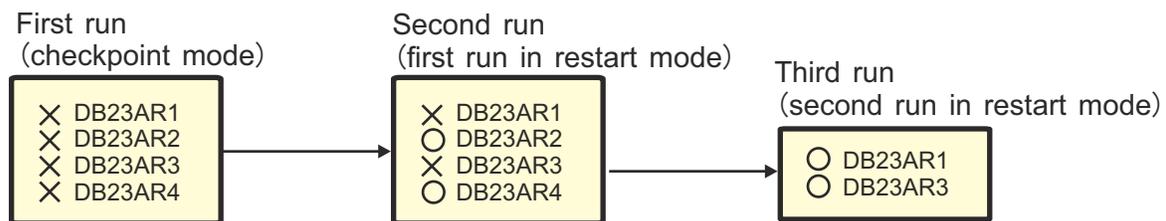


Figure 8. Flow of the checkpoint restart option (FPA Reload function)

Procedure

1. Modify the JCL stream for the Reload function to enable the checkpoint restart option by including the HFPCHKPT DD statement and RESTART parameters, as shown in the following figure:

```
//HFPEXE EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=HPS.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//        DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPPRINT DD SYSOUT=*
//HFPRPTS DD SYSOUT=*
//ICEPRINT DD SYSOUT=*
//HFPCHKPT DD DISP=SHR,DSN=HPFP.HFPCHKPT
//HFPSYSIN DD *
GLOBAL DBRC=YES,
RESTART=YES
RELOAD
  DBD=DEDBJN23,
  IAREA=ALL,
  OAREA=ALL,
  IDSNMASK='HPFP.USRFILE.&AREA',
  ODSNMASK='HPFP.&AREA.ADS1',
  PTRCHKLVL=FULL,
  ICOUT=YES
/*
//ICEIN DD *
GLOBAL DBRC=Y,
  ICHLQ=HFPICD,
  DSN=&ICHLQ..&DDN..IC1
IC DBD=DEDBJN23,AREA=DB23AR1,ICOUT=(*)
IC DBD=DEDBJN23,AREA=DB23AR2,ICOUT=(*)
IC DBD=DEDBJN23,AREA=DB23AR3,ICOUT=(*)
IC DBD=DEDBJN23,AREA=DB23AR4,ICOUT=(*)
/*
```

Figure 9. Sample JCL stream for checkpoint restart (FPA Reload function)

2. Submit the JCL and check the job log for errors.
 - If the job log contains no errors, all the areas were processed successfully. The checkpoint restart processing is not necessary.
 - If the job ended with a return code of 8 or higher, or ended abnormally, continue with Step “3” on page 47.
3. View the Process Summary report that is generated in the HFPRPTS DD.

The following figure shows an example Process Summary report that contains errors.

IMS HPFP UTILITIES - FPA RELOAD		"Process Summary Report"		PAGE: 1
5698-FPP V2R1				2019-07-13 19:02:33
+-----+ CHECKPOINT MODE # indicates the area that is to be reprocessed in the next restart processing +-----+ RL: RELOAD AN:ANALYZE IC: HPIC IC REG: NOTIFY.IC +-----+				
DBDNAME	AREANAME	AREA NO	RL AN IC IC RC RC REG	STATUS (If not blank, see messages in HFPPRINT/ICEPRINT.)
DEDBJN23#DB23AR1		1		Processing was skipped.
DEDBJN23#DB23AR2		2	12	Processing did not complete.
DEDBJN23#DB23AR3		3		Processing was skipped.
DEDBJN23#DB23AR4		4		Processing was skipped.

Figure 10. Sample Process Summary report (FPA Reload function - checkpoint mode)

From the reload return code column (RL RC) and the status column (STATUS), you can identify that reload processing did not complete for area DB23AR2, and DB23AR1, DB23AR3, and DB23AR4 were skipped because of the error in DB23AR2.

Based on this report, four areas that have a number sign (#) at the beginning of AREANAME need to be reprocessed in the restart processing.

4. Check the error messages in the HFPPRINT DD data set to obtain information about the cause of errors.

Locate the error messages for area DB23AR2.

In this scenario, assume that message HFPK3205E was recorded for DB23AR2. This message indicates that a key sequence error was detected in the USR file.

To resolve the error, correct the key sequence error in area DB23AR2 in the USR file. The key sequence error can be resolved by sorting the USR file in some cases.

5. Submit the same JCL and check the job log for errors.

Important - Before you submit the JCL: If the job restarts from the reload processing, you must delete and redefine all reloaded ADSs, and reset the statuses of area data sets in the RECON data sets so that the statuses are changed from AVAILABLE to UNAVAIL.

- If the job log contains no errors, the remaining areas were processed successfully. You do not need to rerun the checkpoint restart processing.
- If the job ended with a return code of 8 or higher, continue with Step “6” on page 48.

6. View the Process Summary report that is generated in the HFPRPTS DD.

The following figure shows an example Process Summary report that still contains errors.

```
IMS HFPF UTILITIES - FPA RELOAD                "Process Summary Report"                PAGE: 1
5698-FPP V2R1                                2019-07-13 19:03:00
```

DBDNAME	AREANAME	AREA	NO	RL	AN	IC	IC	STATUS (If not blank, see messages in HFPPRINT/ICEPRINT.)
				RC	RC	RC	REG	
DEDBJN23#DB23AR1			1	00	00	08		Processing ended with errors.
DEDBJN23 DB23AR2			2	00	00	00	REG	
DEDBJN23#DB23AR3			3	00	00	08		Processing ended with errors.
DEDBJN23 DB23AR4			4	00	00	00	REG	

Figure 11. Sample Process Summary report (FPA Reload function - restart mode with errors)

The Process Summary report that is generated in restart mode contains the area information for the areas that were reprocessed in the restart processing.

From this report, you can identify the status of the job as follows:

- From the image copy return code column (IC RC), the image copy processing ended with errors for areas DB23AR1 and DB23AR3.
- From the status column (STATUS), processing completed successfully for areas DB23AR2 and DB23AR4.

Based on this report, two areas that have a number sign (#) at the beginning of AREANAME need to be reprocessed in the restart processing.

7. Check the error messages in the HFPPRINT DD data set to obtain information about the cause of errors.

Locate the error messages for areas DB23AR1 and DB23AR3.

Assume that message HFPI0005E was recorded for DB23AR1 and DB23AR3. This message indicates that errors occurred in the image copy processing.

Check the error messages in the ICEPRINT DD data set to obtain information about the cause of image copy processing errors.

Locate the messages for areas DB23AR1 and DB23AR3 in the ICEPRINT DD data set.

Assume that messages FABJ4131I were recorded for DB23AR1 and DB23AR3, and message DFS319A was recorded. These messages indicate that an I/O error occurred while the image copy data set was being accessed.

To resolve the image copy processing errors, identify the cause of the I/O error, and correct the error situation.

8. Submit the same JCL and check the job log for errors.

- If the job log contains no errors, the remaining areas were processed successfully. You do not need to rerun the checkpoint restart processing.
- If the job ended with a return code of 8 or higher, repeat from Step “6” on page 48.

Assume that, this time, no errors were recorded in the job log.

9. View the Process Summary report that is generated in the HFPRPTS DD.

The following figure shows an example Process Summary report that does not contain any errors. This report indicates that all areas were processed successfully.

```

IMS HPFP UTILITIES - FPA RELOAD                                "Process Summary Report"
5698-FPP V2R1                                                PAGE: 1
                                                            2019-07-13 19:04:09
+-----+
| RESTART MODE ( 2ND RUN IN RESTART MODE)                   |
| # indicates the area that is to be reprocessed in the next restart processing |
+-----+
| RL: RELOAD  AN:ANALYZE  IC: HPIC  IC REG: NOTIFY.IC      |
+-----+
DBDNAME  AREANAME  AREA  NO  RL AN IC IC  STATUS (If not blank, see messages in HFPPRINT/ICEPRINT.)
-----  -
DEDBJN23 DB23AR1   1    NA NA 00 REG
DEDBJN23 DB23AR3   3    NA NA 00 REG

```

Figure 12. Sample Process Summary report (FPA Reload function - restart mode with no errors)

Keyword changes supported for restart mode

You can change some keywords before submitting the job to run in restart mode.

You can add, delete, or update the following keywords. If you make changes to any of the keywords that are not in the following list, the job ends abnormally with message HFPP0005E.

GLOBAL command

- RESTART_COUNT

UNLOAD command

- SEGREC_PREFIX
- SEGREC_SUFFIX
- SORTOPT_FILSZ
- ODSNMASK

RELOAD command

- DOVFNULPTR_MSGLVL
- KEYSEQCHK
- MAXERROR
- ODSNMASK
- SEGREC_PREFIX
- SEGREC_SUFFIX
- SORTOPT_FILSZ

ALLOCATE subcommand

All keywords supported for the ALLOCATE subcommand except for the OAREA keyword.

FILECTL subcommand

All keywords supported for the FILECTL subcommand except for the OAREA keyword.

Chapter 4. Unloading a DEDB offline with FPA

You can unload a DEDB by using the FPA Unload function.

Topics:

- [“Functions of Unload” on page 51](#)
- [“Input and output for the Unload function ” on page 52](#)
- [“Running the Unload function” on page 53](#)
- [“Region size requirements for the Unload function” on page 55](#)
- [“DD statements for the Unload function” on page 56](#)
- [“Utility control statements for the Unload function” on page 68](#)
- [“Scenario: Restructuring DEDB areas ” on page 72](#)
- [“Examples for the Unload function” on page 73](#)

Functions of Unload

The Unload function of FPA efficiently unloads a single DEDB area. It can also concurrently unload multiple DEDB areas without impacting all areas of a database.

Specifically, the Unload function unloads multiple DEDB areas to QSAM data sets. If necessary, a sort step can be included in the Unload process. Additionally, the Unload function has the following characteristics:

- It operates independently of the IMS control region. Hence, one or more areas of a multi-area database can be unloaded while the application continues to use the areas that are not included in the maintenance process. This feature can significantly increase application availability.

Note: This requires specific control of area selection and application code capable of handling 'FH' status codes.

- It can concurrently process multiple database areas with no database contention. This concurrent processing capability significantly decreases the time that is required for database maintenance, and it further increases application availability.
- During the Unload process, you can change any or all of the following database specifications:
 - Number of database areas
 - Randomizing module
 - Segment edit/compression routine
 - UOW parameter values
 - ROOT parameter values
 - CI size
 - Size of the VSAM data set
 - Pointer options
 - Addition of new segment
 - Change of existing segment hierarchical structure within the same parent

Restrictions:

- The names of the existing segments cannot be changed.
- The existing segments cannot be deleted.
- The maximum length of the existing segments cannot be reduced.
- During the Unload process, an empty area unloaded by the Unload function is clearly identified with an information message. The Audit report also shows the empty DEDB area.

- The new area and RAP values for the database record are determined during unload processing. This allows all records for an area to be written to the same output data set, avoids an extra pass of the file, and simplifies the Reload process.
- There are no source code modifications made to any user-written or IMS program or control block. Because IMS program integrity is maintained, new exposures are not introduced.
- The Unload process enables you to produce an expanded-format unloaded data set from compressed segments.
- The checkpoint restart option automatically identifies the areas that failed in the previous run of the job and reprocesses only the failed areas from the process in which the failure occurred.

Input and output for the Unload function

The Unload function is invoked by the UNLOAD command in the HFPSYSIN DD control statement to unload multiple DEDB areas to QSAM data sets.

To reorganize areas of a DEDB, only the IMSACB library is required. If you want to change the definition of a DEDB, for example to extend an area by increasing any of the sizes of UOW, IOVF, and CI, you also need to specify the NEWACB library that has the DMCB member of the new DEDB definition. The SORT program can be invoked internally to achieve better SORT performance by eliminating the SORTIN and the SORTOUT file I/O operation. When the unload process is completed, IMS HP Image Copy can be invoked internally to create the image copies.

If you enable the IMS catalog and the IMS management of ACBs, ACB definitions are retrieved from the IMS directory instead of IMSACB and NEWACB libraries.

The following figure shows the input and the output for the Unload function.

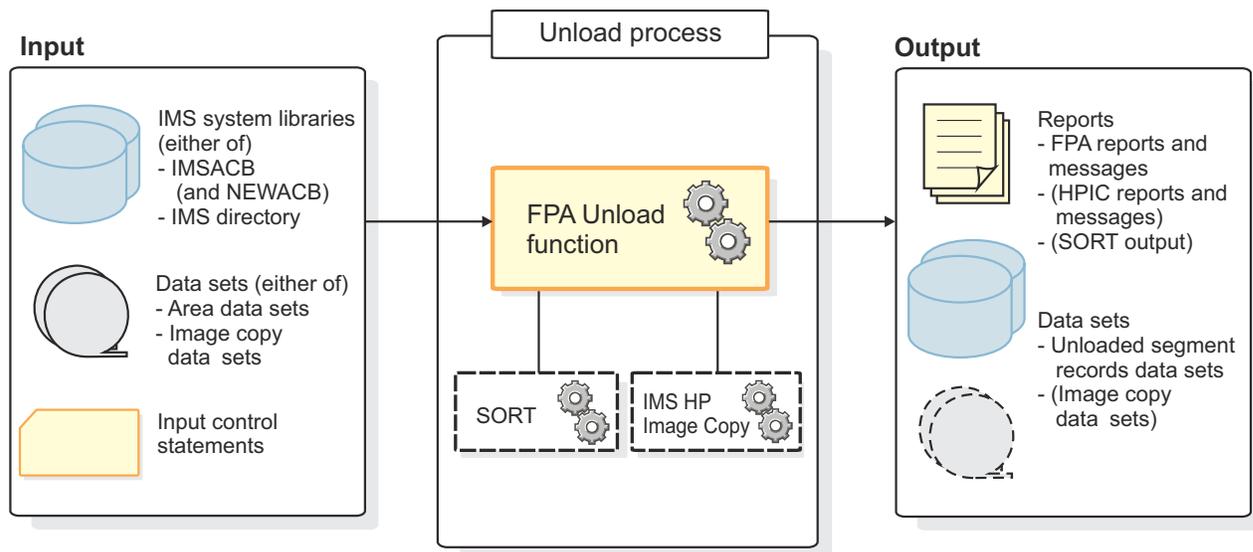


Figure 13. Input and output for the Unload function

The Unload function can read and unload the following image copies of a DEDB:

- Standard image copies that are created by IMS Image Copy utility (DFSUDMP0) or IMS HP Image Copy
- Compressed image copies that are created by IMS HP Image Copy

When image copies are provided as input data sets for the Unload function, new image copies cannot be created after the completion of the unload process.

Running the Unload function

FPA runs the Unload function as a standard z/OS batch job. You must specify an EXEC statement and DD statements that define the input and output data sets in your JCL.

Procedure

1. Optional: Estimate the region size that is required to run the job.
See [“Region size requirements for the Unload function”](#) on page 55.
2. Specify the EXEC statement in the following format:

```
//UNLOAD EXEC PGM=HFPMAIN0,REGION=rrrrM,  
// PARM='DBRCGRP=dbrcgrp,GSGNAME=gsgname,IMSPLEX=imsplex,  
IMSID=imsid'
```

Parameter	Description
DBRCGRP=	DBRCGRP is an optional 1- to 3-character parameter that specifies the identifier assigned to a group of DBRC instances (DBRC sharing group) that access the same RECON data set in the IMSplex.
GSGNAME=	GSGNAME is an optional 1- to 8-character parameter that specifies the global service group.
IMSPLEX=	IMSPLEX is an optional 1- to 5-character parameter that specifies the names of the IMSplex groups that share databases or message queues.
IMSID=	IMSID is an optional 1- to 8-character parameter that specifies the IMS system identification name. If you specify this parameter, FPA generates the buffers for temporary ESCD and SCD. When FPA issues a call to a randomizing module, register 11 contains the address of the temporary ESCD. FPA sets only the ESCDSCD field, which contains the address of SCD, in the temporary ESCD. Also, FPA sets only the SSCDIMID field, which contains the IMS system ID, in the temporary SCD. Other fields of temporary ESCD and temporary SCD cannot be referred to.

3. Specify the DD statements to define input data sets and output data sets.
4. Optional step. If you want to create an RDE during the FPA process, follow these steps:
 - a) Activate Integrity Checker.
To create a new RDE by using the Integrity Checker utility of IMS Library Integrity Utilities, follow the instructions in "Activating Integrity Checker" in the *IMS Library Integrity Utilities User's Guide*.
 - b) Request that FPA creates an RDE by using the Integrity Checker utility of IMS Library Integrity Utilities.
 - i) If you are to use a specific LIU global option module, specify the name of the LIU global option module in the LIUGOPT parameter of the FPA control statement (HFPSYSIN). You also need to specify DBRC=YES in the FPA JCL because Integrity Checker requires that DBRC be active.
 - ii) In the JOBLIB/STEPLIB DD statement, concatenate the following load module libraries.
 - The IMS Library Integrity Utilities load module library
 - The LIU global option modules (LIU@INST, LIU@*imsi*, or both)
5. Run the JCL job.

Example

The following figure shows a JCL example for the UNLOAD command.

```

//*****
//** UNLOAD THE AREA(S) **
//*****
//UNLOAD EXEC PGM=HFPMAIN0
//*
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSRESLB DD DISP=SHR,DSN=IMSVS.SDFSRESL
//HFPPRINT DD SYSOUT=A
//HFPRPTS DD SYSOUT=A
//*
//HFPSYSIN DD *
GLOBAL DBRC=NO
UNLOAD
DBD=IVPDB3
/*
//DFSIVD3A DD DISP=OLD,DSN=IMSVS.DFSIVD31
//DFSIVD3B DD DISP=OLD,DSN=IMSVS.DFSIVD33
//OAREA001 DD DISP=(NEW,PASS),DSN=&OAREA001,
// UNIT=SYSDA,SPACE=(CYL,(7,2),RLSE)
//OAREA002 DD DISP=(NEW,PASS),DSN=&OAREA002,
// UNIT=SYSDA,SPACE=(CYL,(7,2),RLSE)
//*

```

Figure 14. JCL example for the UNLOAD command

Related concepts

Region size requirements for the Unload function

The Unload function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Related tasks

Scenario: [Unloading all areas of a DEDB by using the checkpoint restart option](#)

To unload all areas of a DEDB by using the checkpoint restart option, modify your unload JCL accordingly. Then, run the job and check the reports. If processing fails for one or more areas, correct the error conditions and repeat the steps until all areas are processed.

Related reference

[LIUGOPT keyword](#)

Use the LIUGOPT keyword to specify the LIU global option module that is to be used by IMS Library Integrity Utilities. The Integrity Checker utility of IMS Library Integrity Utilities requires that DBRC be active.

[Examples for the Unload function](#)

Review examples and learn techniques for unloading DEDB areas.

[Tips and best practices for the FPA Unload function](#)

The FPA Unload function supports several keywords that improve the performance of FPA jobs and that further simplify management of DEDBs.

Considerations on using the Integrity Checker utility

Some considerations apply when using the Integrity Checker utility of IMS Library Integrity Utilities.

- When you change a DBD and reorganize DEDB areas by using FPA, Integrity Checker creates new RDEs for the reorganized areas. If reorganization completes successfully, but Integrity Checker fails to create new RDEs, you must take one of the following actions:
 - Rerun the reorganization job to create new RDEs for the reorganized areas.
 - If you decide not to rerun the reorganization job and to use the original areas, you must delete the RDEs that were created during the reorganization by using the DELETE.DB command of the LICON utility before using the areas.
 - If you decide not to rerun the reorganization job and to use the reorganized areas, you must create the RDEs whose creation failed during the reorganization by using the INIT.DB command of the LICON utility before using the areas.

- When any of the following error conditions were detected in FPA or Integrity Checker processing, RDEs are not created:
 - Either the return code is equal to or greater than 8 or the processing ends abnormally for one or more areas in the Reload, Change, or Reorg processing.
 - The return code is equal to or greater than 8 for the Analyze function even when Reload, Change, or Reorg function completes successfully.
 - In checkpoint restart mode, the RDE for the area whose return code from the unload processing is 8 or higher.
 - Integrity Checker fails to create an RDE.
 - When a difference is found between the DBD library and the stored RDE in the following functions:
 - Unload function
 - Change function that does not change the DBD definition
 - Reorg function
- When the randomizer routine, edit/compression routine, or both are defined in the DBD library, checksum values of the routines are stored in the RDE. You must make sure that the libraries that contain the correct routines are specified in one of the following DD statements:
 - Unload function
 - IMSRESLB
When IMSRESLB is not specified, JOBLIB/STEPLIB
 - Reload and Change functions
 - IMSRESLB
When IMSRESLB is not specified, JOBLIB/STEPLIB
 - Reorg function
 - JOBLIB/STEPLIB
If the library that contains the Edit/Compression routine is not concatenated to JOBLIB/STEPLIB, FABLO212W message is issued by Integrity Checker.

Region size requirements for the Unload function

The Unload function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Region size for the Unload function depends on the size of a UOW, the number of areas being unloaded, the number of output files, and the ITASKCTL specifications. Typically, database maintenance is a relatively important function. To increase the performance of the unload program, it would be advisable to run it in a region that has a relatively high dispatching priority (that is, at least equal to the dispatching priority of a message processing region).

Estimating the region size for unload

The region size required for the Unload function is a function of the size of a UOW, the number of areas being unloaded, the number of output files, and the ITASKCTL specifications.

In most cases, the specification on the EXEC statement for HFPMAIN0 of a region that is too small, is detected during the initialization processing of the main task. This causes processing to be terminated at that point. Under certain conditions, the region size shortfall does not become evident until the actual unload has begun. The job must be completely rerun. Thus, the region size calculations must always be performed prior to initiating unload processing.

If SORT=YES is specified, the sort program also requires region space. For how to estimate the required region size, see the *DFSORT Application Programming Guide*.

The following worksheet is for estimating the region size that is required for a typical unload job. The actual region size depends on the keywords that are specified for the job. Therefore, when you specify the region size in your JCL, use a higher value than the value calculated from this worksheet.

FPA programs, internal tables, control blocks	3,020,000	
Number of areas defined in input DBD * 1,300	-----	
Number of areas defined in output DBD * 1,300	-----	(If restructure mode)
Internal tables, control blocks	21,000	
Maximum Segment length * 1,000	-----	
Maximum CISZ * 4	-----	
Internal I/O buffer	1,000,000	(If OCHECK=DEFER)
subtotal	-----	
Value specified on ITASKCTL * subtotal	-----	
Internal tables, control blocks	21,000	
Internal I/O buffer	1,000,000	(If OCHECK=NODEFER)
Storage used by Sort program	-----	(If SORT=YES)
subtotal	-----	
Number of output USR files * subtotal	-----	
Number of unloaded areas * 3,700	-----	
Total of CISZ * UOW1 of each area * 2	-----	
Total bytes of unload storage	-----	

Figure 15. Worksheet for estimating the region size for the Unload function

If ICACHE=YES is specified, the Unload function preloads IOVF and SDEP parts into the data space. The data space size required for the Unload function is a total of the size of IOVF and SDEP parts in each area. In addition, if the input data is an image copy data set, all image copy records are stored into the data space. In this case, the data space for the image copy data set must be added to the size that is calculated by the preceding worksheet.

The following figure shows a worksheet for estimating the data space size for the Unload function.

Total size of IOVF parts in all areas	-----	
Total size of SDEP parts in all areas	-----	(If SDEP=LOGICAL or PHYSICAL)
Total of CISZ * UOW1 * (ROOT1 - ROOT2) in each area	-----	(If input data is IC)
Total bytes of a data space storage	-----	

Figure 16. Worksheet for estimating data space size for the Unload function

DD statements for the Unload function

DD statements of the FPA Unload function determine the input and output data sets and specify how to run the Unload function.

Subsections:

- [“List of DD statements” on page 57](#)
- [“Input DD statements” on page 58](#)
- [“Output DD statements” on page 62](#)
- [“Input/Output DD statements” on page 66](#)

List of DD statements

The following table summarizes the DD statements for the Unload function.

Table 2. DD statements for the Unload function

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
JOBLIB or STEPLIB	Input	PDS	No	Required
IMSACB, IMSACBA, or IMSACBB	Input	PDS	No	Required when IMSCATHLQ=*NO
IMSDALIB	Input	PDS	No	Optional
MODSTAT	Input	PS	Yes	Optional
MODSTAT2	Input	PS	Yes	Optional
OLCSTAT	Input	PS	Yes	Optional
EXITLIB	Input	PDS	No	Optional
IMSRESLB	Input	PDS	No	Optional
NEWACB	Input	PDS	Yes	Optional
NEWRESLB	Input	PDS	No	Optional
<i>addnname</i> or <i>areaname</i>	Input	VSAM ESDS cluster or PS (Note)	Yes	Required under certain conditions (for details, see addnname/areaname DD)
OAREAnn or OARnnnnn	Output	PS	Yes	Required when neither the ODSNMASK keyword nor the FILECTL subcommand is specified
HFPSYSIN	Input	LRECL=80	No	Required
HFPCSITE	Input	PDS	Yes	Optional
HFPPRINT	Output	SYSOUT	Yes	Optional
HFPRPTS	Output	LRECL=133	Yes	Optional
HFPASNAP	Output	LRECL=133	No	Required when CISNAP=YES
HFPSEQER	Output	PS	No	Required when KEYSEQCHK=BYPASS or KEYSEQCHK=WARNING
HFPTOTAL	Output	PS	No	Optional
HFPABEND	Output	LRECL=125	Yes	Optional
HFPCHKPT	Input and output	PS	No	Required when RESTART=YES
RECONn	Input and output	KSDS	Yes	Required when both of the following conditions are met: <ul style="list-style-type: none"> • DBRC=YES • DFSMDA members for RECONn data sets are not used
<i>ctttWKnn</i>	Input and output		Yes	Optional

Table 2. DD statements for the Unload function (continued)

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
<i>itttWKnn</i>	Input and output		Yes	Optional
<i>ctttSMMSG</i>	Output	LRECL=133	Yes	Optional
<i>itttSMMSG</i>	Output	LRECL=133	Yes	Optional
DFSPRINT	Output	LRECL=122	Yes	Optional
ICEIN	Input	LRECL=80	No	Optional
ICEPRINT	Output	LRECL=133	Yes	Optional
ICERPRT	Output	LRECL=133	No	Optional
<i>dataout1</i>	Output	PS	Yes	Optional
<i>dataout2</i>	Output	PS	Yes	Optional

Note: If image copy is used as an input data set.

Input DD statements

The following input DD statements are used for the Unload function.

JOBLIB/STEPLIB DD

This DD statement is a required DD statement. This DD statement defines the input data sets as follows:

- The FPA load module library (required).
- The library that contains DFSMDA dynamic allocation members for the RECON data set. When DBRC=YES is specified and the RECONn DD statement is omitted, this DD statement is required.
- The library that contains randomizer/edit compression members that were specified in the NEWACB data set if NEWRESLB DD statement is not specified.
- The library that contains a user exit routine member that was specified in the EXITRTN= keyword if EXITLIB DD statement is not specified.
- The library that contains a default site default member (HFPCSITE) to determine the data set name and the member of the site default table.
- The IMS load module library (required).
- The IMS Tools Base library (SHKTLOAD), if you specify the ITKBSRVR keyword.
- The IMS Tools Base library (SGLXLOAD), if you specify the IMSCATHLQ keyword.
- The IMS HP Image Copy library, if you specify the ICOUT=YES option.
- The IMS Library Integrity Utilities library, if you want to activate Integrity Checker.

FPA must be run as an APF-authorized program. All of the data sets that are listed within the JOBLIB or the STEPLIB concatenation must also be APF-authorized. If you use DFSMDA members and the DFSMDA members are in a non-APF authorized library, you can specify the DFSMDA library on the IMSDALIB DD statement.

IMSACB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. The IMSACB DD statements (IMSACB, IMSACBA, IMSACBB, MODSTAT, MODSTAT2, and

OLCSTAT) can be used to identify the active ACB library. When both OLCSTAT or MODSTAT and MODSTAT2 are found, FPA checks the time stamp and determines the latest one.

The active IMSACB library is determined according to the following rules:

1. Check IMSACB DD statement in the JCL statement.
2. Check OLCSTAT DD statement in the JCL statement.
3. Check MODSTAT and MODSTAT2 DD statement in the JCL statement.
4. Allocate OLCSTAT dynamically and check the OLCSTAT member.
5. Allocate MODSTAT and MODSTAT2 dynamically and check the MODSTAT and MODSTAT2 members.

If you specify the IMSCATHLQ keyword, IMSACB DD statements are not used. The IMS directory is used instead of the ACB library.

IMSACBA DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSACBB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSDALIB DD

This DD statement is an optional DD statement that defines the IMSDALIB data set. This DD statement is useful when the data set for dynamic allocation members is not APF-authorized.

MODSTAT DD

This DD statement is an optional DD statement that defines the MODSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

MODSTAT2 DD

This DD statement is an optional DD statement that defines the MODSTAT2 data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT2 library by using a TYPE=DATABASE statement of the DFSMDA macro.

OLCSTAT DD

This DD statement is an optional DD statement that defines the OLCSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the OLCSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

EXITLIB DD

This DD statement is an optional DD statement. This statement defines the user exit routine library. This DD statement is used when the EXITRTN keyword is specified for the UNLOAD command. When the EXITLIB DD statement is specified in the JCL statement, user exit routines are loaded directly only from the EXITLIB DD statement. When the EXITLIB DD statement is not specified in the JCL statement, the routines are loaded from the JOBLIB or the STEPLIB DD statement.

All of the data sets that are listed within the EXITLIB concatenation must be APF-authorized.

IMSRESLB DD

This DD statement is an optional DD statement that defines the libraries that contain the randomizer routines and compression routines that are specified in the ACB member in the IMSACB DD statements. If the IMSRESLB DD statement is specified in the JCL statement, only the randomizer and compression routines are loaded directly from the IMSRESLB DD statement. If the IMSRESLB DD statement is not specified in the JCL statement, they are loaded from the JOBLIB or the STEPLIB DD statement.

The IMS version of IMSRESLB DD statement must be the same as those of IMSACB DD.

All the data sets that are listed within the IMSRESLB concatenation must be APF-authorized.

NEWACB DD

This DD statement is an optional DD statement. This statement defines the library that contains the new ACB library for the database.

If this statement is specified, Unload is processed in non-REORG mode. However, if the ACB member in the NEWACB DD library and the ACB member in the IMSACB DD library are the same, and NEWRESLB DD statement is not specified in the JCL statement, UNLOAD is processed in REORG mode. Do not specify this DD statement and do not allocate the data set dynamically if you want to do Unload in REORG mode.

If you specify the IMSCATHLQ keyword, NEWACB DD statement is not used. The IMS directory is used instead of the ACB library.

Note on dynamic allocation:

When this statement is not specified in the JCL statement, the data set is allocated dynamically by using a DFSMDA member. This DD library is allocated dynamically even if you do not specify in the JCL statement or a DFSMDA member. You can build the DFSMDA member for the NEWACB library by using a TYPE=DATABASE statement of the DFSMDA macro.

NEWRESLB DD

This DD statement is an optional DD statement. This statement is effective when the NEWACB DD statement is specified in JCL statement or the IMSCATACB_OUTPUT keyword is specified. This statement defines the libraries in which randomizer routines and compression routines for NEWACB DD statements or the IMSCATACB_OUTPUT keyword are contained.

If the NEWACB DD statement or the IMSCATACB_OUTPUT keyword is specified, the randomizer and compression routines are loaded from the NEWRESLB DD statement. If the NEWRESLB DD statement is not specified when the NEWACB DD statement or the IMSCATACB_OUTPUT keyword is specified, the randomizer and compression routines are loaded from the JOBLIB or the STEPLIB DD statement.

All of the data sets that are listed within the NEWRESLB concatenation must be APF-authorized.

addname/areaname DD

This DD statement is a required DD statement. This statement defines an input data set, which is the VSAM ESDS or image copy for an area to be unloaded. The ddname used must be the ddname for the area that is defined in the input ACB library (IMSACB DD). There can be multiple *addname* DD statements for each area included in the unload request.

The input data set that is used by FPA is determined by the following elements:

- The IDSNMASK keyword parameter of the UNLOAD command
- When the GLOBAL DBRC=YES option is specified, the ADS information in the DBRC RECON data sets (only for VSAM ESDS data sets)
- The area name member in the IMS DFSMDA library (only for VSAM ESDS data sets)

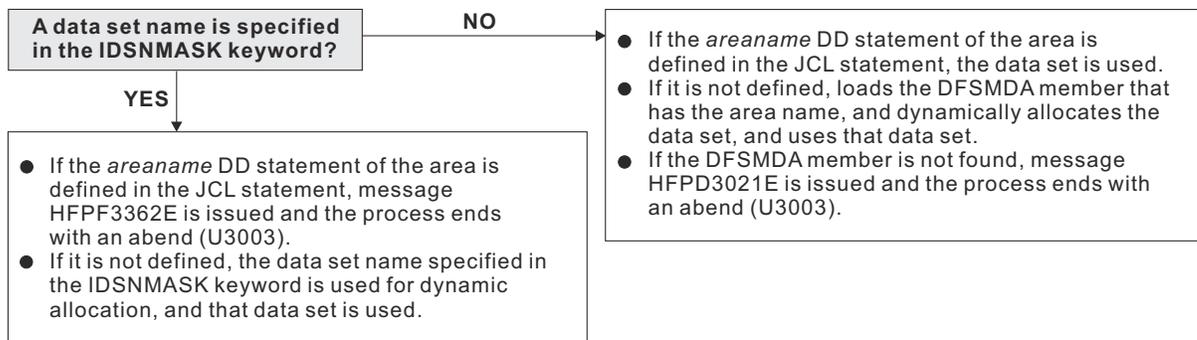
Or it can be specified in one of the following DD statements explicitly:

- *//addname* DD JCL statement
- *//areaname* DD JCL statement

If an area data set is used for the Unload command input, it must be error free (that is, it has no EQEs). And if the area is registered with DBRC, the data set must have no EEQEs.

The following figure shows how FPA determines area data set names.

When DBRC=NO:



When DBRC=YES:

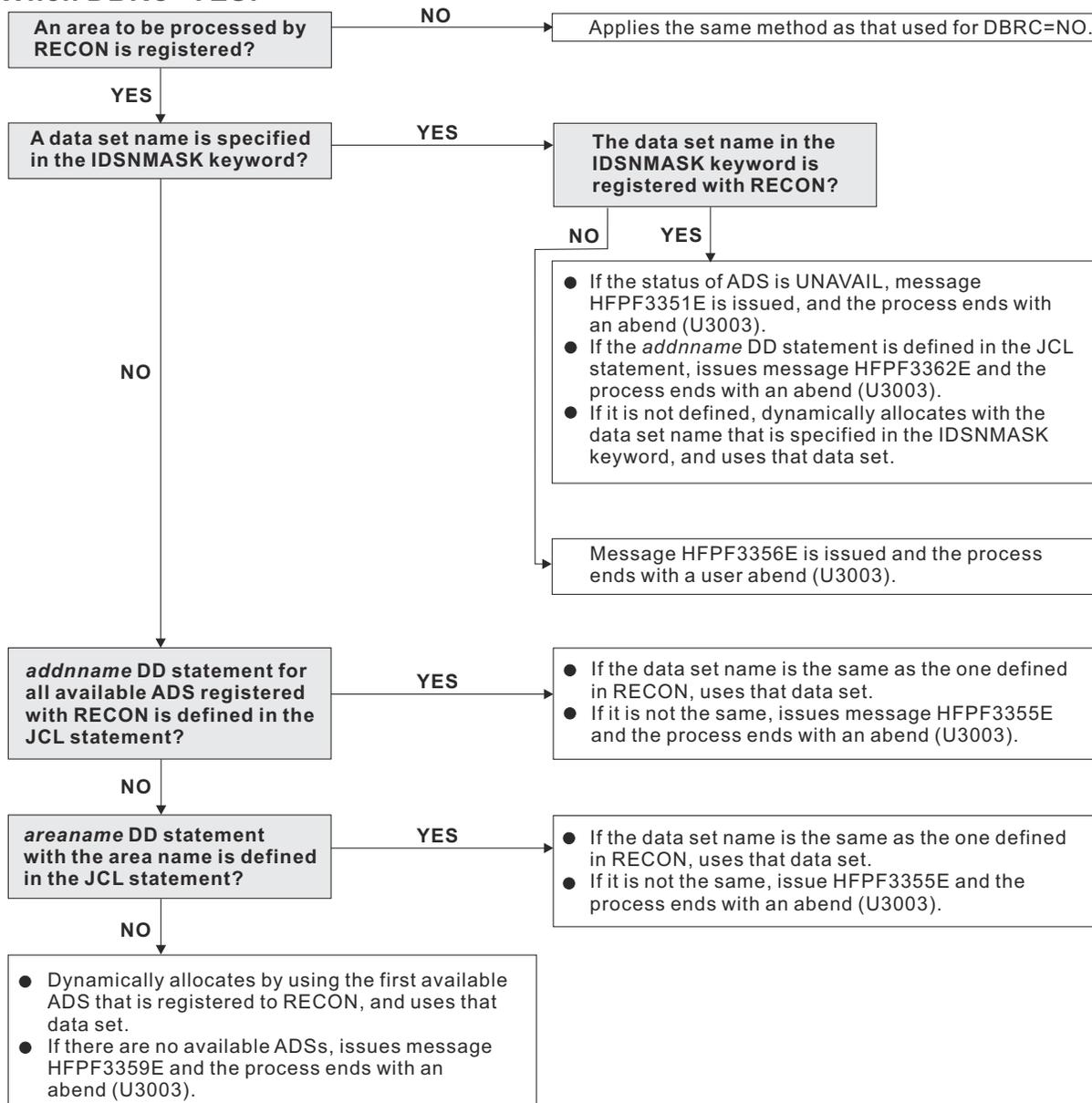


Figure 17. Input data set names for the UNLOAD command

HFPSYSIN DD

This DD statement is a required DD statement that defines the input control statement that specifies the FPA functions.

The HFPSYSIN DD statement can be coded as a standard SYSIN file, a sequential data set, or a PDS member. LRECL=80 is required for the DCB of this data set. Only one command, besides the GLOBAL command, can be specified in HFPSYSIN. FPA analyzes the command syntax in columns 1 - 72. Columns 73 - 80 are regarded as comment.

For details about coding the HFPSYSIN DD statement, see [“Utility control statements for the Unload function”](#) on page 68.

HFPCSITE DD

This DD statement is an optional DD statement. The HFPCSITE DD statement can be used to define the data set name of the site default table. When the SITE_DSNAME keyword is specified in the HFPSYSIN DD statement, the HFPCSITE DD statement is allocated dynamically. HFPCSITE DD in the JCL is mutually exclusive with the SITE_DSNAME keyword in the HFPSYSIN DD statement. The data set attribute of HFPCSITE DD is PDS.

ICEIN DD

This DD statement is an optional DD statement. This statement specifies the input data set that contains the control statements for IMS HP Image Copy. The data set can reside on a tape, a direct-access volume, or it can be routed through the input stream (DD * or DD DATA).

This data set is used only when you specify ICOUT=YES. If the DD statement is not coded and if ICOUT=YES is specified, the default control statements are used. For details about the ICEIN control statements, see *IMS High Performance Image Copy User's Guide*.

Output DD statements

The following output DD statements are used for the Unload function.

OAREAnnn/OARnnnnn DD

This DD statement is a required DD statement when neither the ODSNMASK keyword nor the FILECTL subcommand is specified.

This statement defines the output data set that contains the unloaded segment data records, each of which is composed of a prefix and segment data, for one or more areas. This statement defines an output data set for all of the database segment records that are produced for one or more of the areas defined in the "output" ACB library (NEWACB DD statement or IMSACB DD statement in Reorg mode). This data set will be used for an input of the RELOAD command to reload areas.

This data set can be allocated dynamically by one of following methods:

- The ODSNMASK keyword parameter of the UNLOAD command
- The DSNAME keyword parameter of the FILECTL subcommand

Or it can be specified with one of the following DD statements explicitly:

- //OAREAnnn DD JCL statement
- //OARnnnnn DD JCL statement
- //ddname DD (The ddname is specified by the DDNAME keyword parameter of the FILECTL subcommand.)

This data set can be also defined by the FILECTL subcommand with the UNLOAD command.

An OAREAnnn DD statement is for areas in the range of 1 - 999, and an OARnnnnn DD statement is for areas in the range of 1 - 2048. If the area number of the unloaded area is greater than 999, you must use the OARnnnnn DD statement. The value of nnn or nnnnn must be right-aligned, with leading zeros if necessary. Do not specify DISP=MOD for these DD statements.

If the unloaded data set, which was unloaded in a user-specified format with the FORMAT=USER keyword, satisfies all of the following conditions, the data set can be used as an input for the Reload function.

- The USER_RECORD subcommand is not specified.
- SEGMENT_NAME or SEGMENT_CODE is specified on the SEGREC_PREFIX keyword.
- When COMPRTN is specified in the DBD definition, SEGMENT_ATTR is specified on the SEGREC_PREFIX keyword.
- When SEGMENT_CKEY is specified on the SEGREC_PREFIX keyword, SEGMENT_CKEY_LENGTH is specified before SEGMENT_CKEY.
- When KEY or SEGMENT_KEY is specified on the SEGREC_PREFIX keyword, SEGMENT_KEY_LENGTH is specified before KEY or SEGMENT_KEY.

For the format of an unloaded segment record, see [“Unloaded segment records layout” on page 1177](#). For the format of the unloaded segment record when FORMAT=USER is specified, see [“Extracted records layout” on page 1186](#).

Note on dynamic allocation:

When this statement is not specified in the JCL statement, the data set is allocated dynamically by using the ODSNMASK keyword.

HFPPRINT DD

This DD statement is an optional DD statement that defines the processing log output data set. The data set contains the processing messages that are issued by FPA.

The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set. You cannot specify DUMMY.

The HFPPRINT data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPPRINT DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*.

The following figure shows an example of the processing messages that are issued for the UNLOAD command.

```
IMS HFPF UTILITIES                                "Processing messages"                                PAGE: 1
5698-FPP V2R1                                     2019-11-22 13:03:37
HFPC0001I COMMAND ANALYZE PROCESSING STARTED.
HFPC0005I DBRC=NO IS SPECIFIED.
- EEQE DETECTION IS NOT PERFORMED.
HFPD0001I NEWACB DYNAMICALLY ALLOCATED TO
- DSNNAME: IMSVS.ACBLIB
HFPC0002I COMMAND ANALYZE PROCESSING COMPLETED.
HFPF0001I UNLOAD PROCESSING STARTED FOR DATABASE: DEDBJN21
HFPJ0011I UNLOAD PROCESSING STARTED FOR UNLOADED SEGMENT RECORD FILE.
- DDNAME: OAREA001, DSNNAME: HFPF.HFPUSR.DB21AR0
HFPJ0002I UNLOAD PROCESSING COMPLETED FOR AREA NO: 1, AREANAME: DB21AR0
HFPJ0012I UNLOAD PROCESSING COMPLETED FOR UNLOADED SEGMENT RECORD FILE.
- DDNAME: OAREA001, DSNNAME: HFPF.HFPUSR.DB21AR0
HFPF0002I UNLOAD PROCESSING COMPLETED FOR DATABASE: DEDBJN21
HFPL0001I REPORT STARTED FOR COMMAND: UNLOAD, DATABASE: DEDBJN21
HFPL0002I REPORT COMPLETED FOR COMMAND: UNLOAD, DATABASE: DEDBJN21
```

Figure 18. Processing messages issued for the UNLOAD command

HFPRPTS DD

This DD statement is an optional DD statement that defines the report output data set. The data set contains the reports that can be used to verify the process of the command. The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133

- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

The following reports are generated in this data set:

- Audit report
- Processing report
- DBD Definition report
- Unloaded report
- Reloaded report
- Output File report

The HFPRPTS data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPRPTS DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*.

HFPASNAP DD

This DD statement is a required DD statement when CISNAP=YES. This statement defines the output data set for the CI Map/CI Dump report and the Pointer Chain Reconstruction report. This DD statement is effective when PTRCHK=DETAIL and CISNAP=YES are specified.

The following DCB attributes of the file are set by the Analyze process:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement.

The HFPASNAP data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPASNAP DD SYSOUT=A
```

HFPSEQER DD

This DD statement is a required DD statement when KEYSEQCHK=BYPASS or KEYSEQCHK=WARNING. This statement defines the output data set that contains the sequence error segment data records for one or more areas.

This statement defines an output data set for the sequence error segment data records that are produced for one or more of the areas defined in the "output" ACB library (NEWACB DD statement or IMSACB DD statement in Reorg mode).

The format of this record is specified in the HFPZSER macro, which is provided as a product-sensitive user interface. For details of the record format, see [“Sequence error segment records layout”](#) on page 1183.

HFPTOTAL DD

This DD statement is an optional DD statement. This statement defines an output data set that contains the total number of unloaded segments in an area. When this DD statement is present, the Unload process writes the total number of unloaded segments of each segment type in an area.

The following DCB attributes of the file are set by the FPA main task:

- DSORG=PS
- LRECL=60
- BLKSIZE=6000

Do not code the DCB information in your JCL statement.

The format of this record is specified in the HFPZTOT macro, which is provided as a product-sensitive user interface. For details of the record format, see [“Total segments records layout” on page 1183](#).

HFABEND DD

This DD statement is an optional DD statement. It defines a snap dump output data set that is used for diagnostic purposes. This data set might be used when the FPA job ends abnormally. Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

ctttMSG DD

This DD statement is an optional DD statement. However, this DD statement is required when DFSORT is used. This statement defines an output data set for storing DFSORT messages. For the condition for running DFSORT and the naming rules for the DD names, see the description of *ctttWKnn* DD.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

itttMSG DD

This DD statement is an optional DD statement. However, this DD statement is required when DFSORT is used. This statement defines an output data set for storing DFSORT messages. For the condition for running DFSORT and the naming rules for the DD names, see the description of *itttWKnn* DD.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

DFSPRINT DD

This DD statement is an optional DD statement. This statement specifies the output data set that contains the messages that are produced by IMS HP Image Copy. The data set can reside on a tape, a direct-access volume, or a printer, or it can be routed through the output stream (SYSOUT).

This data set is used only when you specify ICOUT=YES. If the DD statement is not coded and ICOUT=YES is specified, a SYSOUT stream is dynamically allocated for the DD.

ICEPRINT DD

This DD statement is an optional DD statement. This statement specifies the output data set that contains the reports that are produced by IMS HP Image Copy. The data set can reside on a tape, a direct-access volume, or a printer, or it can be routed through the output stream (SYSOUT).

This data set is used only when you specify ICOUT=YES. If the DD statement is not coded and ICOUT=YES is specified, a SYSOUT stream is dynamically allocated for the DD.

ICERPRT DD

This DD statement is an optional DD statement. This statement specifies the output data set that contains the Image Copy Dump Process report, which is produced by IMS HP Image Copy. It can reside on a tape, a direct-access volume, or a printer, or it can be routed through the output stream (SYSOUT).

This data set is used only when you specify ICOUT=YES.

dataout1 DD

This DD statement is an optional DD statement. This statement defines the primary image copy data set. One DD statement is required for each data set that is to be dumped. The DD name can be any 1- to 8-character string. The data set specified by the DD statement must be on either a direct-access device or a tape device.

This data set is used only when you specify ICOUT=YES. For details of the format and the characteristics of the image copy data set, see *IMS High Performance Image Copy User's Guide*. You can request IMS HP Image Copy to allocate the image copy data set dynamically by omitting this DD

statement and specifying the allocation parameters that are associated with the data set in the ICEIN data set.

dataout2 DD

This DD statement is an optional DD statement. This statement defines the secondary image copy data set. This statement is required only if the associated ICEIN control statement requests the secondary image copy. The data set specified by the DD statement must be on either a direct-access device or a tape device.

This data set is used only when you specify ICOUT=YES. For details of the format and the characteristics of the image copy data set, see *IMS High Performance Image Copy User's Guide*.

Input/Output DD statements

The following input/output DD statements are used for the Unload function.

HFPCHKPT DD

This DD statement is a function-unique DD statement for the checkpoint restart option. The DD statement is required when RESTART=YES is specified. This statement defines the checkpoint file. It must be the same file used both for a particular checkpoint and for the associated attempted subsequent restarts.

If RESTART=YES is specified, the checkpoint file (HFPCHKPT) must be specified in the job step. It must not be specified as DUMMY or NULLFILE, otherwise, the checkpoint restart capability is disabled. The checkpoint file must be allocated with the following attributes, otherwise, the job ends with error.

- DISP=MOD is not specified
- RECFM=VB is specified
- LRECL is larger than or equal to 22000

For more information, see [“RESTART keyword” on page 514](#).

RECON1/RECON2/RECON3 DD

These statements define the DBRC RECON data sets. These DD statements are required DD statements when both of the following conditions are met:

- DBRC=YES is specified for the GLOBAL command
- DFSMDA members for RECON n data sets are not used

The RECON n data set must be the same data set that is used in the IMS control region. Because FPA issues a signon, a signoff, an authorization call, and an unauthorization call to DBRC, it updates RECON n data sets.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is necessary, the data set is allocated dynamically by using a DFSMDA member.

ctttWKnn DD

These DD statements are optional DD statements. These data sets are intermediate storage data sets that are used by DFSORT. For more information about coding the SORTWK nn DD statements, see *DFSORT Application Programming Guide*.

The adequate size of storage data sets is 1.5 to 2.5 times the input file size. In the Unload function, the input file is the unloaded segment record file. The size of the file is computed by multiplying the average length of the unloaded segment record by the number of segments. Allocating twice the space used by the SORTIN data set is usually adequate for each work data set. This data set is allocated dynamically by OS. When the SORTIN is too large for sort, a B37 abend occurs. In this case, this DD is a required DD.

Do not specify DISP=MOD for these DD statements.

The naming rules of the ddnames are as follows:

c: The thousands digit of *n* for creating *n*-th output USR file.

U: If *n* is 999 or less, U must be specified.

V: If *n* is 1000 - 1999, V must be specified.

W: If *n* is 2000 - 2048, W must be specified.

ttt: The hundreds, tenths, and units digits of *n*.

nn: Specifies the number of the SORT work data sets. This value must be 01 - 03.

Note on dynamic allocation:

If this statement is not specified in the JCL statement when these data sets are necessary, FPA allocates them dynamically.

itttWKnn DD

These DD statements are optional DD statements. These are intermediate storage data sets that are used by DFSORT. For more information about coding the SORTWKnn DD statements, see the *DFSORT Application Programming Guide*.

Do not specify DISP=MOD for these DD statements.

The adequate size of storage data sets is 1.5 to 2.5 times the input file size. In the unload step, the size of each input file is as follows:

PTRCHK=DETAIL

30 x the number of pointers in the area

CISNAP=YES

112 x the number of pointers in which pointer errors are detected

Allocating twice the space used by the SORTIN data set is usually adequate for each work data set.

The naming rules of the ddnames are as follows:

i:

A, B, and C:

If PTRCHK=DETAIL is specified.

G, H, and I:

If CISNAP=YES is specified.

The thousands digit of the area number of the area to be analyzed, or the thousands digit of *n* for *n*-th input thread subtask.

A, G:

If the number of areas to be analyzed or *n* is 999 or less.

B, H:

If the number of areas to be analyzed or *n* is 1000 - 1999.

C, I:

If the number of areas to be analyzed or *n* is 2000 - 2048.

Note: When a subtask ends processing an area, it starts processing the next area. For this reason, the *n*-th input thread subtask does not always process the *n*-th input data set.

ttt:

The hundreds, tenths, and units digits of the area number of the area to be analyzed, or those of *n*.

nn:

The number of the SORT work data sets. This value must be 01 - 03.

Note on dynamic allocation:

If this statement is not specified in the JCL statement when these data sets are necessary, FPA allocates them dynamically.

Utility control statements for the Unload function

The utility control statements for the Unload function define the user description of the GLOBAL command, the UNLOAD command, the optional FILECTL subcommand, and the optional USER_RECORD subcommand, all of which control the Unload process.

The utility control statements must be specified in the HFPSYSIN data set. This control statement data set usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte, fixed-length records. Block size, if coded, must be a multiple of 80.

The following figure shows a sample of how to code the HFPSYSIN data set.

```
//HFPSYSIN DD *
GLOBAL
DBRC=YES
UNLOAD
DBD=dbdname
/*
```

Figure 19. UNLOAD HFPSYSIN data set

Subsections:

- [“GLOBAL command keywords” on page 68](#)
- [“UNLOAD command keywords” on page 69](#)
- [“FILECTL subcommand keywords” on page 71](#)
- [“USER_RECORD subcommand keywords” on page 72](#)

GLOBAL command keywords

For the GLOBAL command, you can specify the keywords that are listed in the following table. For the format of the GLOBAL command, see [“GLOBAL command” on page 394](#).

Table 3. GLOBAL command keywords

Keyword	Required or optional	Description	Topic
DBRC	Optional	Specifies to request or bypass the database recovery control (DBRC) facility process for the database that is specified by the DBD.	“DBRC keyword” on page 451
ICINCOMP	Optional	Specifies, when an image copy is used as an input data set, whether the image copy records are to be compressed by using the CSRCESTRV MVS™ macro when they are stored into a data space storage.	“ICINCOMP keyword” on page 474
ICNOTIFY	Optional	Specifies whether to issue the NOTIFY.IC command to DBRC.	“ICNOTIFY keyword” on page 475
ICOUT	Optional	Specifies whether to generate one or more image copies.	“ICOUT keyword” on page 476
IMSCATACB_INPUT	Optional	Specifies whether to retrieve the currently active ACB definition or the pending ACB definition from the IMS directory.	“IMSCATACB_INPUT keyword” on page 480

Table 3. GLOBAL command keywords (continued)

Keyword	Required or optional	Description	Topic
IMSCATHLQ	Optional	Specifies the high-level qualifier of the bootstrap data set of the IMS directory.	“IMSCATHLQ keyword” on page 481
ITKBLOAD	Optional	Specifies the library name of IMS Tools KB.	“ITKBLOAD keyword” on page 487
ITKBSRVR	Optional	Specifies the name of the IMS Tools KB server XCF group.	“ITKBSRVR keyword” on page 488
LIUGOPT	Optional	Specifies the LIU global option module that is to be used by IMS Library Integrity Utilities.	“LIUGOPT keyword” on page 493
MSTIMESTAMP	Optional	Specifies whether to include a time stamp in the messages.	“MSTIMESTAMP keyword” on page 497
RESTART	Optional	Enables the checkpoint restart option for reprocessing failed areas.	“RESTART keyword” on page 514
RESTART_COUNT	Optional	Specifies the maximum number of restart processing runs for the checkpoint restart option.	“RESTART_COUNT keyword” on page 515
SCAN	Optional	Verifies the syntax of command, keywords, and parameters in command sets without running FPA. The Scan function also allocates some (but not all) internal resources for running the process function.	“SCAN keyword” on page 523
SITE_DSNAME	Optional	Specifies the data set name of the site default table.	“SITE_DSNAME keyword” on page 539
SITE_MEMBER	Optional	Specifies the member name of the site default table.	“SITE_MEMBER keyword” on page 540

UNLOAD command keywords

For the UNLOAD command, you can specify the keywords that are listed in the following table. For the format of the UNLOAD command, see [“UNLOAD command” on page 415](#).

Table 4. UNLOAD command keywords

Keyword	Required or optional	Description	Topic
ACBCHK	Optional	Specifies whether to write the DBD information into the USR files.	“ACBCHK keyword” on page 438
AREA_INFORMATION_RECORD	Optional	Specifies whether to include the area information record in the unloaded segment record data set. AIR is an alias of this keyword.	AREA_INFORMATION_RECORD keyword
CISNAP	Optional	Requests the CI Map/CI Dump report and the Pointer Chain Reconstruction report when the unloaded area is analyzed concurrently.	“CISNAP keyword” on page 448
COMPRESS	Optional	Specifies whether the unloaded record should contain compressed segment if segment edit/compression routine is defined for the segment in DBDGEN. If COMPRTN is not specified for SEGM in the DBD statement, this keyword is ignored.	“COMPRESS keyword” on page 449

Table 4. UNLOAD command keywords (continued)

Keyword	Required or optional	Description	Topic
DBD	Required	Specifies the database description (DBD) that contains the areas to be processed.	“DBD keyword” on page 450
DBREC_DISCARD_MSGLVL	Optional	Specifies the severity level of the message when a randomized database record is discarded due to a USR file not being specified.	DBREC_DISCARD_MSGLVL keyword
DOVFNULPTR_MSGLVL	Optional	Specifies that FPA issues a warning message if the first DOVF CI in any UOW contains a null pointer for the current overflow CI.	DOVFNULPTR_MSGLVL keyword
EDITCOMP_SKIP	Optional	Specifies whether to skip the segment edit/compression routine process.	“EDITCOMP_SKIP keyword” on page 459
EXITRTN	Optional	Specifies the name of the user exit routine to be invoked prior to the write process of a Unload function.	“EXITRTN keyword” on page 461
FORMAT	Optional	Specifies the format of the unloaded file.	“FORMAT keyword” on page 465
IAREA	Optional	Specifies one or more input areas for the Unload process.	“IAREA keyword” on page 469
ICACHE	Optional	Specifies the type of control intervals that are to be preloaded into a data space storage for processing.	“ICACHE keyword” on page 472
ICHECK	Optional	Controls the concurrent number of input devices to access by using the ITASKCTL keyword.	“ICHECK keyword” on page 473
ICINCOMP	Optional	Specifies, when an image copy is used as an input data set, whether the image copy records are to be compressed by using the CSRCSRV MVS macro when they are stored into a data space storage.	“ICINCOMP keyword” on page 474
ICNOTIFY	Optional	Specifies whether to issue the NOTIFY.IC command to DBRC.	“ICNOTIFY keyword” on page 475
ICOUT	Optional	Specifies whether to generate one or more image copies.	“ICOUT keyword” on page 476
IDSNMASK	Optional	Specifies the data set name or the mask for dynamically allocating cataloged pre-existing input data sets.	“IDSNMASK keyword” on page 476
IMSCATACB_OUTPUT	Optional	Specifies whether to retrieve the currently active ACB definition or the pending ACB definition from the IMS directory. This keyword specifies the new ACB definition, which is used when an ACB definition change is requested.	“IMSCATACB_OUTPUT keyword” on page 481
ITASKCTL	Optional	Specifies the maximum number of input areas to be processed concurrently.	“ITASKCTL keyword” on page 486
KEYCHGCHK	Optional	Specifies whether to check that the key sequence field was not changed by a user exit routine.	“KEYCHGCHK keyword” on page 489
KEYSEQCHK	Optional	Specifies what to do if a key sequence error is detected.	“KEYSEQCHK keyword” on page 490
LRECL	Optional	Specifies how to determine the LRECL of an unloaded segment records file when it is not specified in the DD control statement explicitly.	“LRECL keyword” on page 494
OAREA	Optional	Specifies one or more output areas for the process.	“OAREA keyword” on page 497
OCHECK	Optional	Controls the concurrent number of output devices to access.	“OCHECK keyword” on page 500
ODSNMASK	Optional	Specifies the data set name or the mask for dynamically allocating cataloged pre-existing output data sets.	“ODSNMASK keyword” on page 500
PAD	Optional	Specifies the padding character to be used when the segment minimum length is increased due to DBD definition change. The PAD keyword is effective for fixed-length segments and variable-length segments.	“PAD keyword” on page 504

Table 4. UNLOAD command keywords (continued)

Keyword	Required or optional	Description	Topic
PTRCHK	Optional	Specifies what to do when a pointer error is detected.	“PTRCHK keyword” on page 505
RELOCATE_SDEP	Optional	Specifies that SDEP information will be relocated.	“RELOCATE_SDEP keyword” on page 512
SDEP	Optional	Specifies how the SDEP segment data is to be processed.	“SDEP keyword” on page 524
SDEP_BYPASS_MSGLVL	Optional	Specifies the severity level of the message that FPA issues when the SDEP is defined but SDEP segment data is not unloaded.	“SDEP_BYPASS_MSGLVL keyword” on page 525
SEGREC_PREFIX	Optional	Specifies the contents of the fields within the prefix portion of the segment records.	“SEGREC_PREFIX keyword” on page 531
SEGREC_SUFFIX	Optional	Specifies the contents of the fields within the suffix portion of the segment records.	“SEGREC_SUFFIX keyword” on page 533
SORT	Optional	Specifies whether to sort the unloaded segment records based on RAP (root anchor point).	“SORT keyword” on page 541
SORTOPT_FILSZ	Optional	Specifies that FPA adds the FILSZ parameter to the SORT control statement.	“SORTOPT_FILSZ keyword” on page 542
SSP	Optional	Specifies whether subset pointers should be retained or cleared.	“SSP keyword” on page 544

FILECTL subcommand keywords

The FILECTL subcommand defines the output files to be created during the Unload process. You can specify this subcommand to group the data targeted to one or more output areas into a single unloaded data set. One or more unload data sets are produced by each FILECTL subcommand that is specified. Do not specify the OAREA or the ODSNMASK keyword on both the UNLOAD command and the FILECTL subcommand. Specify the OAREA keyword on the FILECTL subcommand to specify the output areas to be placed into the unload data set.

For the FILECTL subcommand, you can specify the keywords that are listed in the following table. For the format of the FILECTL subcommand, see [“FILECTL subcommand” on page 425](#).

Table 5. FILECTL subcommand keywords

Keyword	Required or optional	Description	Topic
AVGREC	Optional	Identifies the space requirements of the output data set that is created by dynamic allocation.	“AVGREC keyword” on page 445
DATACLAS	Optional	Specifies the SMS data class of an output data set that is created by dynamic allocation.	“DATACLAS keyword” on page 449
DDNAME	Optional	Specifies the ddnames for the data sets to be processed by the subcommand.	“DDNAME keyword” on page 453
DISP	Optional	Specifies the status and the disposition of the data set that is accessed by dynamic allocation.	“DISP keyword” on page 453
DSNAME	Optional	Specifies a mask for the data set name that is used for dynamic allocation of the data set to be processed by the subcommand.	“DSNAME keyword” on page 456
EXPDT	Optional	Specifies the expiration date of an output data set that is created by dynamic allocation. Do not specify both the EXPDT keyword and the RETPD keyword because they are mutually exclusive.	“EXPDT keyword” on page 462
LIKE	Optional	Specifies the SMS allocation attributes of an output data set that is created by dynamic allocation. The model data set specified must be an existing SMS data set.	“LIKE keyword” on page 492

Table 5. FILECTL subcommand keywords (continued)

Keyword	Required or optional	Description	Topic
MGMTCLAS	Optional	Specifies the SMS management class of an output data set that is created by dynamic allocation.	“MGMTCLAS keyword” on page 496
OAREA	Optional	Specifies one or more output areas for the process.	“OAREA keyword” on page 497
RETPD	Optional	Specifies the retention period of an output data set that is created by dynamic allocation. Do not specify both the EXPDT keyword and the RETPD keyword because they are mutually exclusive.	“RETPD keyword” on page 519
SPACE	Optional	Specifies the space requirements of the output data set that is created by dynamic allocation.	“SPACE keyword” on page 543
STORCLAS	Optional	Specifies the SMS storage class of the output data set that is created by dynamic allocation.	“STORCLAS keyword” on page 546
UNIT	Optional	Specifies the physical device type to be used for a data set that is accessed by dynamic allocation. Use this keyword also to specify the number of physical devices to be allocated.	“UNIT keyword” on page 552
VOLCNT	Optional	Specifies the maximum number of volumes that will be required by an output data set that is created by dynamic allocation that uses nonspecific volumes.	“VOLCNT keyword” on page 556
VOLSER	Optional	Specifies one or more volume-serial identifiers for the data set that is accessed by dynamic allocation. If this keyword is omitted for an output data set, nonspecific (private) volumes are required.	“VOLSER keyword” on page 556

USER_RECORD subcommand keywords

The USER_RECORD subcommand specifies the user-defined header and the trailer records that are to be written to an unloaded file.

For the USER_RECORD subcommand, you can specify the keywords that are listed in the following table. For the format of the USER_RECORD subcommand, see [“USER_RECORD subcommand” on page 437](#).

Table 6. USER_RECORD subcommand keywords

Keyword	Required or optional	Description	Topic
BREAK	Optional	Specifies the level and the timing for the trigger event for user-defined header and trailer of an unloaded file.	“BREAK keyword” on page 446
FIELDS	Required	Specifies the content of the header and trailer records that are to be written to the output file.	“FIELDS keyword” on page 463

Scenario: Restructuring DEDB areas

Restructuring DEDB areas by using the Unload and the Reload functions of FPA includes several steps.

Procedure

The following steps show typical scenario to restructure all areas of a DEDB.

1. Determine the new definition of the DEDB.

You can use the FPA Tune function or DEDB Tuning Aid to determine the ideal definition.

Related reading:

- See [Chapter 9, “Tuning a DEDB offline with FPA,” on page 229](#) for information about using the FPA Tune function.
- See the topic "Running DEDB Tuning Aid" in the *IMS Fast Path Solution Pack: IMS Fast Path Basic Tools User's Guide* for information about using DEDB Tuning Aid

2. Set up a temporary ACB library and generate the new DBD and ACB.

If you enabled the IMS catalog and the IMS management of ACBs, generate a new ACB in the staging data set of the IMS directory.

If necessary, set up a temporary library and generate new randomizer routines and new compression routines.

3. Set up and tailor the unload JCL.

Related reading: Example JCL streams are provided in the following topics:

- [“Example 6: Unloading all areas of a DEDB and changing some or all of the DEDB area definitions \(IAREA=ALL, OAREA=ALL\)” on page 77](#)
- [“Example 15: Unloading an area and changing the DEDB area definition when IMS management of ACBs is enabled” on page 86](#)

4. Set up and tailor the reload JCL.

Related reading: Example JCL streams are provided in the following topics:

- [“Example 3: Reloading multiple areas of a DEDB from one input unloaded segment record data set” on page 117](#)
- [“Example 10: Reloading an area that is registered with DBRC and creating its image copy” on page 123](#)
- [“Example 15: Reloading an area of which some of the parameters for the DEDB area definition are changed when IMS management of ACBs is enabled” on page 127](#)

5. Stop all areas (that is, deallocate from the IMS control region) involved in the maintenance process. If the application does not handle "FH" status codes, all transactions that access the database must be stopped.

6. Create image copies of all the areas.

7. Run the FPA unload job.

8. Check for any messages generated by the unload job and the total number of segments that were unloaded from the database in the Unloaded report.

9. Run the FPA reload job.

10. Check for any messages generated by the reload job and the total number of segments that were reloaded to the database in the Reloaded report. Ensure that the number of segments processed matches the number of segments that were unloaded in Step [“8” on page 73](#).

11. Migrate the new DMB to the production ACB library. If necessary, migrate the new randomizer routines and the new compression routines to the production library.

If you enabled the IMS catalog and the IMS management of ACBs, activate the new ACB in the online IMS system by issuing the `IMPORT DEFN SOURCE(CATALOG)` command.

12. Start the reloaded areas.

Examples for the Unload function

Review examples and learn techniques for unloading DEDB areas.

In these examples, the following DEDB is assumed:

```

DEDB name:  DEDBJN22 has four areas
  area name:  DB22AR0 (area number is 1 and has three ADSs)
    ADS DD name:  DB22AR01,      DS name:  HPFP.DB22AR0.ADS1
    ADS DD name:  DB22AR02,      DS name:  HPFP.DB22AR0.ADS2
    ADS DD name:  DB22AR03,      DS name:  HPFP.DB22AR0.ADS3
  area name:  DB22AR1 (area number is 2 and has two ADSs)
    ADS DD name:  DB22AR11,      DS name:  HPFP.DB22AR1.ADS1
    ADS DD name:  DB22AR12,      DS name:  HPFP.DB22AR1.ADS2
  area name:  DB22AR2 (area number is 3 and has two ADSs)
    ADS DD name:  DB22AR21,      DS name:  HPFP.DB22AR2.ADS1
    ADS DD name:  DB22AR22,      DS name:  HPFP.DB22AR2.ADS2
  area name:  DB22AR3 (area number is 4 and has one ADS)
    ADS DD name:  DB22AR31,      DS name:  HPFP.DB22AR3.ADS1

```

In DBRC, all ADSs are in the AVAILABLE status.

Related tasks

Scenario: Unloading all areas of a DEDB by using the checkpoint restart option

To unload all areas of a DEDB by using the checkpoint restart option, modify your unload JCL accordingly. Then, run the job and check the reports. If processing fails for one or more areas, correct the error conditions and repeat the steps until all areas are processed.

Example 1: Unloading an area that is registered with DBRC

The following figure shows example JCL for unloading an area that is registered with DBRC.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS to be unloaded is dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC.
- The output unloaded segment records data set for the area is HPFP.USRFILE.DB22AR0, which is specified by the ODSNMASK='HPFP.USRFILE.&AREA' parameter. This data set is allocated dynamically. The data set must be preallocated and cataloged.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//        DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
UNLOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  OAREA=(DB22AR0),
  ODSNMASK='HPFP.USRFILE.&AREA'
/*

```

Figure 20. Unloading an area that is registered with DBRC

Example 2: Unloading all areas of a DEDB that is registered with DBRC

The following figure shows example JCL for unloading all areas of a DEDB that are registered with DBRC.

In this example:

- The input ADSs to be unloaded for all areas are dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC.
- The output unloaded segment records data set for each area is HPFP.USRFILE. *areaname*, which is specified by the ODSNMASK='HPFP.USRFILE.&AREA' parameter. This data set is allocated dynamically. The data set must be preallocated and cataloged.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//         DD DISP=SHR,DSN=IMSVS.SDFSRESL
//         DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//RECON1  DD DISP=SHR,DSN=IMSVS.RECON1
//RECON2  DD DISP=SHR,DSN=IMSVS.RECON2
//RECON3  DD DISP=SHR,DSN=IMSVS.RECON3
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
UNLOAD
  DBD=DEDBJN22,
  IAREA=ALL,
  OAREA=ALL,
  ODSNMASK='HPFP.USRFILE.&AREA'
/*

```

Figure 21. Unloading all areas of a DEDB that is registered with DBRC

Example 3: Unloading multiple areas of a DEDB into one output unloaded segment record data set

The following figure shows example JCL for unloading multiple areas of a DEDB into one output unloaded segment record data set.

In this example:

- The DBRC=NO option specifies not to check the area status in DBRC, so RECON data sets are not needed.
- The input ADSs to be unloaded for specified areas are dynamically allocated by using the IDSNMASK parameter.

The following ADSs are allocated:

Area	Allocated ADS
DB22AR0	HPFP.DB22AR0.ADS1
DB22AR1	HPFP.DB22AR1.ADS1
DB22AR2	HPFP.DB22AR2.ADS1
DB22AR3	HPFP.DB22AR3.ADS1

- The output unloaded segment records data set for area DB22AR0 and DB22AR1 is HPFP.USRFILE1 that is specified by DDNAME=USRFILE1 in the FILECTL subcommand.
- The output unloaded segment records data set for area DB22AR2 and DB22AR3 is HPFP.USRFILE2, which is specified by DSNAME='HPFP.USRFILE2' in the FILECTL subcommand. This data set is allocated dynamically.
- By the SORT=YES parameter, the two output unloaded segment records data sets are sorted internally by area number, RAP RBA, and the root key value. Because the work data set for areas DB22AR0 and DB22AR1 that is allocated dynamically by OS is too small to sort the unloaded segments records, data sets U001WK01 through U001WK03 are specified on the JCL statement.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//        DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//USRFIL1 DD DISP=(NEW,CATLG),DSN=HPFP.USRFIL1,SPACE=(CYL,(5,1)),
//        UNIT=SYSDA,VOL=SER=HFPVOL
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=NO
UNLOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1,DB22AR2,DB22AR3),
  IDSNMASK='HPFP.&AREA.ADS1',
  SORT=YES
FILECTL
  OAREA=(DB22AR0,DB22AR1),
  DDNAME=USRFIL1
FILECTL
  OAREA=(DB22AR2,DB22AR3),
  DSNNAME='HPFP.USRFIL2',
  DISP=(NEW,CATLG),
  SPACE=(CYL,5,1),
  UNIT=SYSDA,
  VOLSER=HFPVOL
/*
//U001WK01 DD DISP=NEW,UNIT=SYSDA,VOL=SER=HPFP,SPACE=(CYL,(50,10))
//U001WK02 DD DISP=NEW,UNIT=SYSDA,VOL=SER=HPFP,SPACE=(CYL,(50,10))
//U001WK03 DD DISP=NEW,UNIT=SYSDA,VOL=SER=HPFP,SPACE=(CYL,(50,10))

```

Figure 22. Unloading multiple areas of a DEDB into one output unloaded segment record data set

Example 4: Unloading an area from the specified ADS of MADS area

The following figure shows example JCL for unloading an area from the specified ADS, which is registered with DBRC as MADS area.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSVS.MDALIB data set. Because IMSVS.MDALIB is an APF-registered data set, it is concatenated in STEPLIB.
- The input ADS to be unloaded for MADS area is specified in JCL DB22AR01 DD statement. DD statement name and DS name of the ADS are verified that the ADS is registered with DBRC.
- The output unloaded segment records data sets for the area are dynamically allocated by the ODSNMASK='HPFP.USRFIL1' parameter. The data set must be preallocated and cataloged.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//        DD DISP=SHR,DSN=IMSVS.PGMLIB
//        DD DISP=SHR,DSN=IMSVS.MDALIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//DB22AR01 DD DISP=SHR,DSN=HPFP.DB22AR0.ADS1
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
UNLOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  OAREA=(DB22AR0),
  ODSNMASK='HPFP.USRFIL1'
/*

```

Figure 23. Unloading an area from the specified ADS of MADS area

Example 5: Unloading an area and changing the DEDB area definition

The following figure shows example JCL that is used to unload an area and to change some of the DEDB area definitions.

In this example, it is assumed that an area-specific randomizer is used so that a DEDB record is randomized to its original area. Also, in this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS to be unloaded is dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC.
- The output unloaded segment records data set is specified in OAREA001 DD statement.
- The ACBLIB member for new DEDB definition is in the NEWACB DD data set.
- All database records are randomized by the randomizer, which is loaded from NEWRESLB DD data set.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//        DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//NEWACB  DD DISP=SHR,DSN=IMSVS.ACBLIB.NEW
//NEWRESLB DD DISP=SHR,DSN=IMSVS.PGMLIB.NEW
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//OAREA001 DD DISP=OLD,DSN=HPFP.USRFILE
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
UNLOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  OAREA=(DB22AR0)
/*
```

Figure 24. Unloading an area and changing the DEDB area definition

Example 6: Unloading all areas of a DEDB and changing some or all of the DEDB area definitions (IAREA=ALL, OAREA=ALL)

The following figure shows example JCL that is used to unload all areas of a DEDB and to change some or all of the DEDB area definitions.

In this example, it is assumed that a general mode randomizer is used, and that a DEDB record might or might not be randomized to its original area. Also, in this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS to be unloaded is allocated dynamically. The allocation information (DD statement name and DS name of the ADS) is obtained from the DBRC.
- The output unloaded segment records data set for each area is HPFP.USRFILE.*areaname*. The data set is specified by the ODSNMASK='HPFP.USRFILE.&AREA' parameter, and is allocated dynamically. The data set must be preallocated and cataloged.
- The ACBLIB member for the new DEDB definition is in the NEWACB DD data set.
- All database records are randomized by the randomizer, which is loaded from the NEWRESLB DD data set.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//NEWACB  DD DISP=SHR,DSN=IMSVS.ACBLIB.NEW
//NEWRESLB DD DISP=SHR,DSN=IMSVS.PGMLIB.NEW
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
UNLOAD
  DBD=DEDBJN22,
  IAREA=ALL,
  OAREA=ALL,
  ODSNMASK='HPFP.USRFILE.&AREA'
/*

```

Figure 25. Unloading all areas of a DEDB and changing some or all of the DEDB area definitions (IAREA=ALL, OAREA=ALL)

For how to reload a DEDB, and how to reload all areas with these unloaded segment records file, see [“Example 2: Reloading all areas of a DEDB that are registered with DBRC” on page 117.](#)

Example 7: Unloading all areas of a DEDB and changing some or all of the DEDB area definitions (IAREA=ALL, OAREA=specific)

The following figure shows example JCL that is used to unload all areas of a DEDB and to change some or all of the DEDB area definitions.

In this example, it is assumed that a general mode randomizer is used, and that a DEDB record might or might not be randomized to its original area.

Unloading is done in two job steps; the first job step writes records that are reloaded to areas DB22AR0 and DB22AR1, and the second job step writes records that are reloaded to areas DB22AR2 and DB22AR3.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS to be unloaded is allocated dynamically. The allocation information (DD statement name and DS name of the ADS) is obtained from the DBRC.
- The output unloaded segment records data set of the first job step, which is for areas DB22AR0 and DB22AR1, and the second job step, which is for areas DB22AR2 and DB22AR3 is HPFP.USRFILE. *areaname*. The data set is specified by the ODSNMASK='HPFP.USRFILE.&AREA' parameter, and is allocated dynamically. The data set must be preallocated and cataloged.
- The ACBLIB member for the new DEDB definition is in the NEWACB DD data set.
- All database records are randomized by the randomizer, which is loaded from the NEWRESLB DD data set.

```

//HFP01 EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//NEWACB DD DISP=SHR,DSN=IMSVS.ACBLIB.NEW
//NEWRESLB DD DISP=SHR,DSN=IMSVS.PGMLIB.NEW
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
UNLOAD
  DBD=DEDBJN22,
  IAREA=ALL,
  OAREA=(DB22AR0,DB22AR1),
  ODSNMASK='HPFP.USRFILE.&AREA'
/*
//HFP02 EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//NEWACB DD DISP=SHR,DSN=IMSVS.ACBLIB.NEW
//NEWRESLB DD DISP=SHR,DSN=IMSVS.PGMLIB.NEW
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
UNLOAD
  DBD=DEDBJN22,
  IAREA=ALL,
  OAREA=(DB22AR2,DB22AR3),
  ODSNMASK='HPFP.USRFILE.&AREA'
/*

```

Figure 26. Unloading all areas of a DEDB and changing some or all of the DEDB area definitions (IAREA=ALL, OAREA=specific)

For how to reload a DEDB, and how to reload all areas with the type of unloaded segment records file in this example, see [“Example 2: Reloading all areas of a DEDB that are registered with DBRC”](#) on page 117.

Example 8: Unloading all areas of a DEDB and changing some or all of the DEDB area definitions (IAREA=specific, OAREA=ALL)

The following figure shows example JCL that is used to unload all areas of a DEDB and to change some or all of the DEDB area definitions.

In this example, it is assumed that a general mode randomizer is used, and that a DEDB record might or might not be randomized to its original area.

Unloading is done in two unload job steps; the first job step unloads DB22AR0 and DB22AR1 and writes records that are reloaded to all areas, and the second job step unloads DB22AR2 and DB22AR3, and writes records that are reloaded to all areas.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS to be unloaded is allocated dynamically. The allocation information (DD statement name and DS name of the ADS) is obtained from the DBRC.
- The output unloaded segment records data set of the first job step for each reload area is HPFP.USRFILE. *areaname*.RUN1. The data set is specified by the ODSNMASK='HPFP.USRFILE.&AREA.RUN1' parameter, and is allocated dynamically. The data set must be preallocated and cataloged.

- The output unloaded segment records data set of the second job step for each reload area is HPFP.USRFILE. *areaname*.RUN2. The data set is specified by the ODSNMASK='HPFP.USRFILE.&AREA.RUN2' parameter, and is allocated dynamically. The data set must be preallocated and cataloged.
- The ACBLIB member for the new DEDB definition is in the NEWACB DD data set.
- All database records are randomized by the randomizer, which is loaded from the NEWRESLB DD data set.

Notes:

1. If you want to unload with the SDEP=LOGICAL or with the SDEP=NO option so as to be able to reproduce the same SDEP logical begin time stamp value for the reload operation, specify FORMAT=DBT (the system default).
2. Unloaded segment records for an area might exist both in HPFP.USRFILE. *areaname*.RUN1 and in HPFP.USRFILE. *areaname*.RUN2. You need to concatenate the two data sets to use it for the reload input. For how to reload a DEDB, and how to reload all areas with these unloaded segment records files, see [“Example 8: Reloading all areas by using the concatenated unloaded segment records data sets”](#) on page 121.

```
//HFP01 EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//NEWACB DD DISP=SHR,DSN=IMSVS.ACBLIB.NEW
//NEWRESLB DD DISP=SHR,DSN=IMSVS.PGMLIB.NEW
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
UNLOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  OAREA=ALL,
  ODSNMASK=' HPFP . USRFILE . &AREA . RUN1 '
/*
//HFP02 EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//NEWACB DD DISP=SHR,DSN=IMSVS.ACBLIB.NEW
//NEWRESLB DD DISP=SHR,DSN=IMSVS.PGMLIB.NEW
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
UNLOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR2,DB22AR3),
  OAREA=ALL,
  ODSNMASK=' HPFP . USRFILE . &AREA . RUN2 '
/*
```

Figure 27. Unloading all areas of a DEDB and changing some or all of the DEDB area definitions (IAREA=specific, OAREA=ALL)

Example 9: Unloading all areas of a DEDB and changing some or all of the DEDB area definitions (IAREA=specific, OAREA=specific)

The following figure shows example JCL that is used to unload all areas of a DEDB and to change some or all of the DEDB area definitions.

In this example, it is assumed that a general mode randomizer is used, and that a DEDB record might or might not be randomized to its original area.

Unloading is done in 16 (four times four) unload job steps. One job step unloads an area, and writes records for an area.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS to be unloaded is allocated dynamically. The allocation information (DD statement name and DS name of the ADS) is obtained from the DBRC.
- The output unloaded segment records data set of one job step for each reload area is HPFP.USRFILE.area-m.RUNr. The data set is specified by the ODSNMASK='HPFP.USRFILE.&AREA.RUNr' parameter, and is allocated dynamically. The data set must be preallocated and cataloged.
- When an area is unloaded, DEDB records in the area might be randomized to all areas. Because database DEDBJN22 has four areas (DB22AR0, DB22AR1, DB22AR2, and DB22AR3), four unload processes are run for each input area. Ultimately 16 (four times four) unload job steps are run.
- The ACBLIB member for the new DEDB definition is in the NEWACB DD data set.
- All database records are randomized by the randomizer, which is loaded from the NEWRESLB DD data set.

Notes:

1. If you want to unload with the SDEP=LOGICAL or with the SDEP=NO option so as to be able to reproduce the same SDEP logical begin time stamp value for the reload operation, specify FORMAT=DBT (the system default).
2. There might be unloaded segment records for area-n in every HPFP.USRFILE.area-n.RUNr (r = 1 - 4). You need to concatenate all data sets to use it for the reload input. For how to reload a DEDB, and how to reload all areas with the type of unloaded segment records files described in this example, see [“Example 8: Reloading all areas by using the concatenated unloaded segment records data sets” on page 121.](#)

```
//HFPT EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//NEWACB DD DISP=SHR,DSN=IMSVS.ACBLIB.NEW
//NEWRESLB DD DISP=SHR,DSN=IMSVS.PGMLIB.NEW
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
UNLOAD
  DBD=DEDBJN22,
  IAREA=area-n,
  OAREA=area-m,
  ODSNMASK=' HPFP.USRFILE.&AREA.RUNr '
/*
```

Figure 28. Unloading all areas of a DEDB and changing some or all of the DEDB area definitions (IAREA=specific, OAREA=specific)

Example 10: Unloading all areas of a DEDB that are registered with DBRC and creating their image copies

The following figure shows example JCL that is used to unload all areas of a DEDB that are registered with DBRC and to create their image copies.

In this example:

- The input ADSs that are to be unloaded for all areas are dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC.

- The output unloaded segment records data set for each area is HPFP.USRFILE.*areaname* that is specified by the ODSNMASK='HPFP.USRFILE.&AREA' parameter, and is allocated dynamically. The data set must be preallocated and cataloged.
- After unloading, the image copy data sets for all areas are dynamically allocated by IMS HP Image Copy. The data set name of each image copy is determined by IMS HP Image Copy as follows:

```
ICOUT.HFP.IC1.DEDBJN22.areaname
```

For more details, see *IMS High Performance Image Copy User's Guide*.

Tips:

- To take image copies of the areas that have pointer errors but to prevent the image copy data sets of those areas from being registered to DBRC by the NOTIFY.IC command, specify PTRCHK=BYPASS or PTRCHK=DETAIL, and ICNOTIFY=COND.
- To take image copies of the areas that have key sequence errors but to prevent the image copy data sets of those areas from being registered to DBRC by the NOTIFY.IC command, specify KEYSEQCHK=BYPASS or KEYSEQCHK=WARNING, and ICNOTIFY=COND.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//         DD DISP=SHR,DSN=HPS.SHPSLMD0
//         DD DISP=SHR,DSN=IMSVS.SDFSRESL
//         DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//RECON1  DD DISP=SHR,DSN=IMSVS.RECON1
//RECON2  DD DISP=SHR,DSN=IMSVS.RECON2
//RECON3  DD DISP=SHR,DSN=IMSVS.RECON3
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
UNLOAD
  DBD=DEDBJN22,
  IAREA=ALL,
  OAREA=ALL,
  ODSNMASK='HPFP.USRFILE.&AREA',
  ICOUT=YES
/*
//ICEIN   DD *
GLOBAL   ICHLQ=ICOUT.HFP,
         UNIT=SYSDA,
         SPACE=(CYL,100,100)
/*
//ICERPRT DD SYSOUT=*
```

Figure 29. Unloading all areas of a DEDB that are registered with DBRC and creating their image copies

Example 11: Unloading areas in user-specified format

The following figures show example JCL statements for unloading areas in user-specified format.

Two examples are provided in this topic. The first example is for creating an unloaded file that can be used as an input for the Reload function and the second example shows a case in which the created file is not enabled for the Reload function.

The unloaded file that is created by the following JCL can be used as an input for the RELOAD command to reload areas because this JCL satisfies all the conditions described under OAREAnnn/OARnnnnn DD in “Output DD statements” on page 62.

In this example:

- The SEGREC_PREFIX keyword defines the content of the segment prefix.
- Segment code is included in the segment prefix.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//          DD DISP=SHR,DSN=IMSVS.MDALIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//DB22AR01 DD DISP=SHR,DSN=HPFP.DB22AR0.ADS1
//OAREA001 DD DISP=OLD,DSN=HPFP.USRFIL0
//HFPPRINT DD SYSOUT=*
//HFPRPTS DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
UNLOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  OAREA=(DB22AR0),
  FORMAT=USER,
  SEGREC_PREFIX=(SEGMENT_CODE)
/*

```

Figure 30. Unloading areas in user-specified format (enabled for Reload)

The unloaded file that is created by the following JCL cannot be used with the Reload function because this JCL does not satisfy all the conditions described under OAREA n nnn/OAR n nnnnn DD in [“Output DD statements”](#) on page 62.

In this example:

- The USER_RECORD subcommand defines the position and the content of the header and trailer.
- BREAK=(DATABASE,BEFORE) defines the database-level header.
- FIELDS=(DBD_NAME) specifies to include the DBD name in the header.
- The SEGREC_PREFIX keyword defines the content of the segment prefix.
- Segment code is included in the segment prefix.
- The SEGREC_SUFFIX keyword defines the content of the segment suffix.
- Segment name is included in the segment suffix.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//          DD DISP=SHR,DSN=IMSVS.MDALIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//DB22AR01 DD DISP=SHR,DSN=HPFP.DB22AR0.ADS1
//OAREA001 DD DISP=OLD,DSN=HPFP.USRFIL0
//HFPPRINT DD SYSOUT=*
//HFPRPTS DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
UNLOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  OAREA=(DB22AR0),
  FORMAT=USER,
  SEGREC_PREFIX=(SEGMENT_CODE),
  SEGREC_SUFFIX=(SEGMENT_NAME)
USER_RECORD
  BREAK=(DATABASE,BEFORE),
  FIELDS=(DBD_NAME)
/*

```

Figure 31. Unloading areas in user-specified format (not enabled for Reload)

Example 12: Unloading all areas of a DEDB that are registered with DBRC and creating their image copies with the checkpoint restart option

The following figures show example JCL statements for unloading all areas of a DEDB that are registered with DBRC and creating their image copies with the checkpoint restart option.

In this example:

- The input ADSs that are to be unloaded for all areas are dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC.
- The output unloaded segment records data set for each area is HPFP.USRFILE.*areaname* that is specified by the ODSNMASK='HPFP.USRFILE.&AREA' parameter, and is allocated dynamically. The data set must be preallocated and cataloged.
- After unloading, the image copy data sets for all areas are created as stacked data sets.

```
//HFPEXE EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=HPS.SHPSLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPPRINT DD SYSOUT=*
//HFPRPTS DD SYSOUT=*
//ICEPRINT DD SYSOUT=*
//ICERPRT DD SYSOUT=*
//HFPCHKPT DD DISP=SHR,DSN=HPFP.HFPCHKPT
//STACK1 DD UNIT=TAPE,DSN=HFP001.ICDS,
//          DISP=OLD,VOL=SER=HFP001
//STACK2 DD UNIT=TAPE,DSN=HFP002.ICDS,
//          DISP=OLD,VOL=SER=HFP002
//STACK3 DD UNIT=TAPE,DSN=HFP003.ICDS,
//          DISP=OLD,VOL=SER=HFP003
//STACK4 DD UNIT=TAPE,DSN=HFP004.ICDS,
//          DISP=OLD,VOL=SER=HFP004
//HFPSYSIN DD *
GLOBAL DBRC=YES,
RESTART=YES
UNLOAD
DBD=DEDBJN22,
IAREA=ALL,
OAREA=ALL,
ODSNMASK='HPFP.USRFILE.&AREA',
KEYSEQCHK=ABEND,
ICOUT=YES
/*
//ICEIN DD *
GLOBAL ICHLQ=HFPICDS,UNIT=TAPE
IC DBD=DEDBJN22,AREA=DB22AR1,STACK=STACK1
IC DBD=DEDBJN22,AREA=DB22AR2,STACK=STACK1
IC DBD=DEDBJN22,AREA=DB22AR3,STACK=STACK2
IC DBD=DEDBJN22,AREA=DB22AR4,STACK=STACK3
IC DBD=DEDBJN22,AREA=DB22AR5,STACK=STACK4
IC DBD=DEDBJN22,AREA=DB22AR6,STACK=STACK4
IC DBD=DEDBJN22,AREA=DB22AR7,STACK=STACK2
IC DBD=DEDBJN22,AREA=DB22AR8,STACK=STACK1
IC DBD=DEDBJN22,AREA=DB22AR9,STACK=STACK1
/*
```

Figure 32. Unloading all areas of a DEDB that are registered with DBRC and creating their image copies with the checkpoint restart option

Example 13: Unloading multiple areas of a DEDB into one output unloaded segment record data set in SMS environment

The following figure shows example JCL for unloading multiple areas of a DEDB into one output unloaded segment record data set in an SMS environment.

In this example:

- The DBRC=NO option specifies not to check the area status in DBRC, so RECON data sets are not needed.
- The input ADSs to be unloaded for specified areas are dynamically allocated by using the IDSNMASK parameter. The following ADSs are allocated:

Area	Allocated ADS
DB22AR0	HPFP.DB22AR0.ADS1
DB22AR1	HPFP.DB22AR1.ADS1
DB22AR2	HPFP.DB22AR2.ADS1
DB22AR3	HPFP.DB22AR3.ADS1

- The output unloaded segment records data set for all areas is HPFP.USRFIL, which is specified by DSNAME='HPFP.USRFIL' in the FILECTL subcommand. This data set is allocated dynamically.
- Because the STORCLAS keyword is specified, the output data set is an SMS-managed data set.

Tip: When the output data set is an SMS-managed data set, the UNIT keyword and the VOLSER keyword, which are required for non-SMS managed data sets, are not required.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=NO
UNLOAD
  DBD=DEDBJN22,
  IAREA=ALL,
  IDSNMASK=' HPFP.&AREA.ADS1 ',
  SORT=YES
FILECTL
  OAREA=ALL,
  DSNAME=' HPFP.USRFIL ',
  DISP=(NEW,CATLG),
  SPACE=(CYL,5,1),
  STORCLAS=HFPSTOR
/*
```

Figure 33. Unloading multiple areas of a DEDB into one output unloaded segment record data set in an SMS environment

Example 14: Unloading an area without activating the DBRC process

The following figure shows example JCL for unloading an area without activating the DBRC process.

In this example:

- The DBRC=IGNORE option specifies not to check the area status in DBRC, so the RECON data sets are not needed. Even if DBRC=FORCE is set for IMS, DBRC is not activated when the DBRC=IGNORE option is specified.
- Area authorization is not obtained from DBRC, so the integrity of the area is not protected.
- The input ADS to be unloaded for the specified area is HPFP.DB22AR0.ADS1.
- The output unloaded segment records data set is specified on the OAREA001 DD statement.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//         DD DISP=SHR,DSN=IMSVS.SDFSRESL
//         DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//DB22AR0 DD DISP=SHR,DSN=HPFP.DB22AR0.ADS1
//OAREA001 DD DISP=OLD,DSN=HPFP.USRFILE.DB22AR0
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=IGNORE
UNLOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  OAREA=(DB22AR0)
/*

```

Figure 34. Unloading an area without activating the DBRC process

Example 15: Unloading an area and changing the DEDB area definition when IMS management of ACBs is enabled

The following figure shows example JCL for unloading an area and changing the DEDB area definition when ACBs are managed by IMS.

To run this example, the IMS catalog and the IMS management of ACBs must be enabled.

In this example, it is assumed that an area-specific randomizer is used so that a DEDB record is randomized to its original area. Also, in this example:

- The IMSACB DD statement is not specified. The ACB member is read from the IMS directory instead of the ACB library because IMSCATHLQ=IMSVS.CATALOG parameter is specified.
- To use IMS Tools Catalog Interface, the SGLXLOAD library of IMS Tools Base is included in the STEPLIB DD concatenation.
- The ACB member for the current DEDB definition is read from the IMS directory data set because IMSCATACB_INPUT=CURRENT is specified.
- The ACB member for new DEDB definition is read from the staging data set because IMSCATACB_OUTPUT=PENDING is specified.
- All database records are randomized by the randomizer that is loaded from the NEWRESLB DD data set.
- The output unloaded segment records data set is specified in the OAREA001 DD statement.
- The DBD information is written to the output unloaded segment records data set because ACBCHK=YES is specified.
- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS to be unloaded is dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//          DD DISP=SHR,DSN=ITB.SGLXLOAD
//NEWRESLB DD DISP=SHR,DSN=IMSVS.PGMLIB.NEW
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//OAREA001 DD DISP=OLD,DSN=HPFP.USRFILE
//HFPRPTS  DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES,
  IMSCATHLQ=IMSVS.CATALOG,
  IMSCATACB_INPUT=CURRENT
UNLOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  OAREA=(DB22AR0),
  IMSCATACB_OUTPUT=PENDING,
  ACBCHK=YES
/*

```

Figure 35. Unloading an area and changing the DEDB area definition when IMS management of ACBs is enabled

Chapter 5. Reloading a DEDB offline with FPA

You can reload a DEDB by using the FPA Reload function.

Topics:

- [“Functions of Reload” on page 89](#)
- [“Input and output for the Reload function” on page 90](#)
- [“Running the Reload function” on page 90](#)
- [“Region size requirements for the Reload function” on page 93](#)
- [“DD statements for the Reload function” on page 94](#)
- [“Utility control statements for the Reload function” on page 108](#)
- [“Examples for the Reload function” on page 115](#)

Functions of Reload

The Reload function of FPA efficiently reloads a single DEDB area. It can also concurrently reload multiple DEDB areas without impacting all areas of a database.

Specifically, the Reload function reloads multiple DEDB areas from the QSAM data sets. If necessary, a sort step and an analyze step can be included in the Reload process. Additionally, the Reload function has the following characteristics:

- It operates independently of the IMS control region. Hence, one or more areas of a multi-area database can be reloaded while the application continues to use the areas that are not included in the maintenance process. This feature can significantly increase application availability.

Note: This requires specific control of area selection and application code capable of handling 'FH' status codes.

- It can concurrently process multiple database areas with no database contention. This concurrent processing capability significantly decreases the time that is required for database maintenance, and it further increases application availability.
- An empty area unloaded by the Unload or Change process is clearly identified with an information message. Then, during the Reload process, an output data set for this empty area is initialized with no segment to prevent troubles when IMS DB/DC gets access to the output data set. The Audit report also shows the empty DEDB area.
- During the Reload process, dependent segment twin chains that are abnormally long can be controlled by specifying the load control information so that the impact on other database records in the same RAP CI is minimized. This feature is implemented by placing a user-specified limit on the number of segment occurrences that are to be placed closed to their root segments.
- The new area and RAP values for the database record are determined during unload processing. This allows all records for an area to be written to the same output data set, avoids an extra pass of the file, and simplifies the Reload process.
- There are no source code modifications made to any user-written or IMS program or control block. Because IMS program integrity is maintained, new exposures are not introduced.
- The Reload process can reload segment data in one area into multi-area data sets (MADS).
- The Reload process can optionally check the consistency between the DBD information in the library that is specified on the IMSACB DD statement and the DBD information in the USR files that were created during the Unload process.

Input and output for the Reload function

The Reload function is invoked by the RELOAD command in the HFPSYSIN DD control statement to reload multiple DEDB areas from the QSAM data sets.

The SORT program can be invoked internally to improve the performance of SORT by eliminating the SORTIN and SORTOUT file I/O operation. When the reload process is completed, IMS HP Image Copy can be invoked internally to create the image copies.

The following figure shows the input and the output for the Reload function.

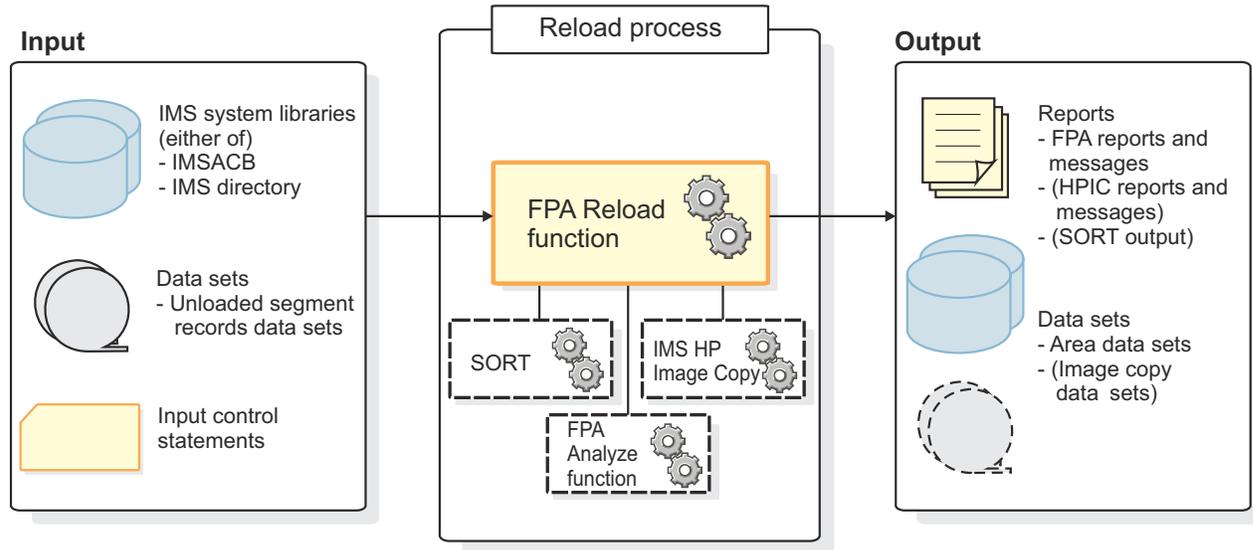


Figure 36. Input and output for the Reload function

The Reload function reloads multiple DEDB areas from the QSAM data sets and analyzes the DEDB areas by using the RELOAD command in the HFPSYSIN DD control statement. The Analyze function can be invoked internally after the Reload function. When the Analyze process is completed, IMS HP Image Copy can be invoked internally to create the image copies.

Running the Reload function

FPA runs the Reload function as a standard z/OS batch job. You must specify an EXEC statement and DD statements that define the input and output data sets in your JCL.

Procedure

- Optional: Estimate the region size that is required to run the job.
See [“Region size requirements for the Reload function”](#) on page 93.
- Specify the EXEC statement in the following format:

```
//RELOAD EXEC PGM=HFPMAIN0,REGION=rrrrM,
//      PARM='DBRCGRP=dbrcgrp,GSGNAME=gsgname,IMSPLEX=imsplex,
//      IMSID=imsid'
```

Parameter	Description
DBRCGRP=	DBRCGRP is an optional 1- to 3-character parameter that specifies the identifier assigned to a group of DBRC instances (DBRC sharing group) that access the same RECON data set in the IMSplex.
GSGNAME=	GSGNAME is an optional 1- to 8-character parameter that specifies the global service group.

Parameter	Description
IMSPLEX=	IMSPLEX is an optional 1- to 5-character parameter that specifies the names of the IMSplex groups that share databases or message queues.
IMSID=	<p>IMSID is an optional 1- to 8-character parameter that specifies the IMS system identification name.</p> <p>If you specify this parameter, FPA generates the buffers for temporary ESCD and SCD. When FPA issues a call to a randomizing module, register 11 contains the address of the temporary ESCD. FPA sets only the ESCDSCD field, which contains the address of SCD, in the temporary ESCD. Also, FPA sets only the SSCDIMID field, which contains the IMS system ID, in the temporary SCD. Other fields of temporary ESCD and temporary SCD cannot be referred to.</p>

3. Specify the DD statements to define input data sets and output data sets.

4. Optional step. If you want to create an RDE during the FPA process, follow these steps:

a) Activate Integrity Checker.

To create a new RDE by using the Integrity Checker utility of IMS Library Integrity Utilities, follow the instructions in "Activating Integrity Checker" in the *IMS Library Integrity Utilities User's Guide*.

b) Request that FPA creates an RDE by using the Integrity Checker utility of IMS Library Integrity Utilities.

i) If you are to use a specific LIU global option module, specify the name of the LIU global option module in the LIUGOPT parameter of the FPA control statement (HFPSYSIN). You also need to specify DBRC=YES in the FPA JCL because Integrity Checker requires that DBRC be active.

ii) In the JOBLIB/STEPLIB DD statement, concatenate the following load module libraries.

- The IMS Library Integrity Utilities load module library
- The LIU global option modules (LIU@INST, LIU@*imsi*, or both)

5. Run the JCL job.

Example

The following figure shows a JCL example for the RELOAD command.

```
//*****
//** RELOAD THE AREA(S) **
//*****
//RELOAD EXEC PGM=HFPMAIN0
//*
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSRESLB DD DISP=SHR,DSN=IMSVS.SDFSRESL
//HFPPRINT DD SYSOUT=A
//HFPRPTS DD SYSOUT=A
//HFPSYSIN DD *
GLOBAL DBRC=NO
RELOAD
DBD=IVPDB3
/*
//IAREA001 DD DISP=(OLD,DELETE),DSN=&OAREA001
//IAREA002 DD DISP=(OLD,DELETE),DSN=&OAREA002
//DFSIVD3A DD DISP=OLD,DSN=IMSVS.DFSIVD31.NEW
//DFSIVD3B DD DISP=OLD,DSN=IMSVS.DFSIVD33.NEW
```

Figure 37. JCL example for the RELOAD command

Related concepts

[Region size requirements for the Reload function](#)

The Reload function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Related reference

LIUGOPT keyword

Use the LIUGOPT keyword to specify the LIU global option module that is to be used by IMS Library Integrity Utilities. The Integrity Checker utility of IMS Library Integrity Utilities requires that DBRC be active.

Examples for the Reload function

Review examples and learn techniques for reloading DEDB areas.

Tips and best practices for the FPA Reload function

The FPA Reload function supports several keywords that improve the performance of FPA Reload jobs and that further simplify management of DEDBs.

Considerations on using the Integrity Checker utility

Some considerations apply when using the Integrity Checker utility of IMS Library Integrity Utilities.

- When you change a DBD and reorganize DEDB areas by using FPA, Integrity Checker creates new RDEs for the reorganized areas. If reorganization completes successfully, but Integrity Checker fails to create new RDEs, you must take one of the following actions:
 - Rerun the reorganization job to create new RDEs for the reorganized areas.
 - If you decide not to rerun the reorganization job and to use the original areas, you must delete the RDEs that were created during the reorganization by using the DELETE.DB command of the LICON utility before using the areas.
 - If you decide not to rerun the reorganization job and to use the reorganized areas, you must create the RDEs whose creation failed during the reorganization by using the INIT.DB command of the LICON utility before using the areas.
- When any of the following error conditions were detected in FPA or Integrity Checker processing, the RDEs are not created:
 - Either the return code is equal to or greater than 8 or the processing ends abnormally for one or more areas in the Reload, Change, or Reorg processing.
 - The return code is equal to or greater than 8 for the Analyze function even when Reload, Change, or Reorg function completes successfully.
 - Integrity Checker fails to create an RDE.
- When the randomizer routine, edit/compression routine, or both are defined in the DBD library, checksum values of the routines are stored in the RDE. You must make sure that the libraries that contain the correct routines are specified in one of the following DD statements:
 - Reload and Change functions
 - IMSRESLB
 - When IMSRESLB is not specified, JOBLIB/STEPLIB
 - Reorg function
 - JOBLIB/STEPLIB
 - If the library that contains the Edit/Compression routine is not concatenated to JOBLIB/STEPLIB, FABL0212W message is issued by Integrity Checker.

Region size requirements for the Reload function

The Reload function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Region size for the Reload function depends on the size of a UOW, the number of files being reloaded, the number of output areas, and the ITASKCTL specifications. Typically, database maintenance is a relatively important function. To increase the performance of the reload program, it would be advisable to run it in a region that has a relatively high dispatching priority (that is, at least equal to the dispatching priority of a message processing region).

Estimating the region size for reload

The region size required for the Reload function is a function of the size of a UOW, the number of areas being reloaded, the number of output files, and the ITASKCTL specifications.

In most cases, the specification on the EXEC statement for HFPMAIN0 of a region that is too small, is detected during the initialization processing of the main task. This causes processing to be terminated at that point. Under certain conditions, the region size shortfall does not become evident until the actual reload has begun. The job must be completely rerun. Thus, the region size calculations must always be performed prior to initiating reload processing.

If SORT=YES is specified, the sort program also requires region space. For how to estimate the required region size, see the *DFSORT Application Programming Guide*.

The following worksheet is for estimating the region size that is required for a typical reload job. The actual region size depends on the keywords that are specified for the job. Therefore, when you specify the region size in your JCL, use a higher value than the value calculated from this worksheet.

FPA programs, internal tables, control blocks	3,020,000	
Number of areas defined in DBD * 1,300	-----	
Internal tables, control blocks	21,000	
Maximum CISZ * 6	-----	
Internal I/O buffer	1,000,000	(If ICHECK=DEFER)
subtotal	-----	
Value specified on ITASKCTL * subtotal	-----	
Internal tables, control blocks	3,700	
Storage used by Sort program	-----	(If SORT=YES)
subtotal	-----	
Number of reloaded areas * subtotal	-----	
Internal tables, control blocks	4,300	
Internal I/O buffer	1,000,000	(If ICHECK=NODEFER)
subtotal	-----	
Number of input USR files * subtotal	-----	
Total of CISZ * UOW1 of each area * 2	-----	
Total bytes of reload storage	-----	

Figure 38. Worksheet for estimating the region size for the Reload function

If an analyze step is included in the reload process, estimate the region size for the analyze step and use the greater region size. For how to estimate the region size for an analyze step, see [“Region size requirements for the Analyze function”](#) on page 200.

If OCACHE=YES is specified, the Reload function preloads IOVF and SDEP parts into the data space. The data space size required for the Reload function is a total of the size of IOVF CIs used after reloading and the size of SDEP parts in each area.

The following figure shows a worksheet for estimating the data space size for the Reload function.

Total IOVF CIs used after reloading in each area	-----
Total size of SDEP parts in all areas	-----
Total bytes of a data space storage	-----

Figure 39. Worksheet for estimating data space size for the Reload function

DD statements for the Reload function

DD statements of the FPA Reload function determine the input and output data sets and specify how to run the Reload function.

Subsections:

- [“List of DD statements” on page 94](#)
- [“Input DD statements” on page 95](#)
- [“Output DD statements” on page 100](#)
- [“Input/Output DD statements” on page 106](#)

List of DD statements

The following table summarizes the DD statements for the Reload function.

Table 7. DD statements for the Reload function

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
JOBLIB or STEPLIB	Input	PDS	No	Required
IMSACB, IMSACBA, or IMSACBB	Input	PDS	No	Required when IMSCATHLQ=*NO
IMSDALIB	Input	PDS	No	Optional
MODSTAT	Input	PS	Yes	Optional
MODSTAT2	Input	PS	Yes	Optional
OLCSTAT	Input	PS	Yes	Optional
EXITLIB	Input	PDS	No	Optional
IMSRESLB	Input	PDS	No	Optional
<i>addnname</i> or <i>areaname</i>	Output	VSAM ESDS cluster	Yes	Required under certain conditions (for details, see <i>addnname/areaname</i> DD)
IAREAnnn or IARnnnnn	Input	PS	Yes	Required when IDSNMASK keyword is not specified
HFPSYSIN	Input	LRECL=80	No	Required
HFPAPARM	Input	LRECL=80	No	Optional
HFPAUOWC	Input	LRECL=80	No	Required when HFPAHST2 DD is specified
HFPCSITE	Input	PDS	Yes	Optional

Table 7. DD statements for the Reload function (continued)

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
HFPPRINT	Output	SYSOUT	Yes	Optional
HFPRPTS	Output	LRECL=133	Yes	Optional
HFPSRPT	Output	LRECL=133	Yes	Optional
HFPAHST	Output	DISP=MOD	No	Optional
HFPAHST2	Output	DISP=MOD	No	Required when HFPAUOWC DD is specified
HFPARERG	Output	LRECL=80	No	Optional
HFPASNAP	Output	LRECL=133	No	Required when CISNAP=YES
HFPTOTAL	Output	PS	No	Optional
HFPABEND	Output	LRECL=125	Yes	Optional
HFPCHKPT	Input and output	PS	No	Required when RESTART=YES
RECON n	Input and output	KSDS	Yes	Required when both of the following conditions are met: <ul style="list-style-type: none"> • DBRC=YES • DFSMDA members for RECONn data sets are not used
<i>ctttWKn</i>	Input and output		Yes	Optional
<i>itttWKn</i>	Input and output		Yes	Optional
<i>ctttSMMSG</i>	Output	LRECL=133	Yes	Optional
<i>itttSMMSG</i>	Output	LRECL=133	Yes	Optional
DFSPRINT	Output	LRECL=122	Yes	Optional
ICEIN	Input	LRECL=80	No	Optional
ICEPRINT	Output	LRECL=133	Yes	Optional
ICERPRT	Output	LRECL=133	No	Optional
<i>dataout1</i>	Output	PS	Yes	Optional
<i>dataout2</i>	Output	PS	Yes	Optional

Input DD statements

The following input DD statements are used for the Reload function.

JOBLIB/STEPLIB DD

This DD statement is a required DD statement. This DD statement defines the input data sets as follows:

- The FPA load module library (required).

- The library that contains DFSMDA dynamic allocation members for the RECON data set. When DBRC=YES is specified and the RECON n DD statement is omitted, this DD statement is required.
- The library that contains randomizer/edit compression members that were specified in the IMSACB data set, if IMSRESLB DD statement is not specified.
- The library that contains a user exit routine member that was specified in the EXITRTN= keyword, if EXITLIB DD statement is not specified.
- The library that contains a default site default member (HFPCSITE) to determine the data set name and the member of the site default table.
- The IMS load module library (required).
- The IMS Tools Base library (SHKTLOAD), if you specify the ITKBSRVR keyword.
- The IMS Tools Base library (SGLXLOAD), if you specify the IMSCATHLQ keyword.
- The IMS HP Image Copy library, if you specify the ICOUT=YES option.
- The IMS Library Integrity Utilities library, if you want to activate Integrity Checker.

FPA must be run as an APF-authorized program. All of the data sets that are listed within the JOBLIB or the STEPLIB concatenation must also be APF-authorized. If you use DFSMDA members and the DFSMDA members are in a non-APF authorized library, you can specify the DFSMDA library on the IMSDALIB DD statement.

IMSACB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. The IMSACB DD statements (IMSACB, IMSACBA, IMSACBB, MODSTAT, MODSTAT2, and OLCSTAT) can be used to identify the active ACB library. When both OLCSTAT or MODSTAT and MODSTAT2 are found, FPA checks the time stamp and determines the latest one.

The active IMSACB library is determined according to the following rules:

1. Check IMSACB DD statement in the JCL statement.
2. Check OLCSTAT DD statement in the JCL statement.
3. Check MODSTAT and MODSTAT2 DD statement in the JCL statement.
4. Allocate OLCSTAT dynamically and check the OLCSTAT member.
5. Allocate MODSTAT and MODSTAT2 dynamically and check the MODSTAT and MODSTAT2 members.

If you specify the IMSCATHLQ keyword, IMSACB DD statements are not used. The IMS directory is used instead of the ACB library.

IMSACBA DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSACBB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSDALIB DD

This DD statement is an optional DD statement that defines the IMSDALIB data set. This DD statement is useful when the data set for dynamic allocation members is not APF-authorized.

MODSTAT DD

This DD statement is an optional DD statement that defines the MODSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

MODSTAT2 DD

This DD statement is an optional DD statement that defines the MODSTAT2 data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT2 library by using a TYPE=DATABASE statement of the DFSMDA macro.

OLCSTAT DD

This DD statement is an optional DD statement that defines the OLCSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the OLCSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

EXITLIB DD

This DD statement is an optional DD statement. This statement defines the user exit routine library. This DD statement is used when the EXITRTN keyword is specified for the RELOAD command. When the EXITLIB DD statement is specified in the JCL statement, user exit routines are loaded directly only from the EXITLIB DD statement. When the EXITLIB DD statement is not specified in the JCL statement, the routines are loaded from the JOBLIB or the STEPLIB DD statement.

All of the data sets that are listed within the EXITLIB concatenation must be APF-authorized.

IMSRESLB DD

This DD statement is an optional DD statement that defines the libraries that contain the randomizer routines and compression routines that are specified in the ACB member in the IMSACB DD statements. If the IMSRESLB DD statement is specified in the JCL statement, only the randomizer and compression routines are loaded directly from the IMSRESLB DD statement. If the IMSRESLB DD statement is not specified in the JCL statement, they are loaded from the JOBLIB or the STEPLIB DD statement.

The IMS version of IMSRESLB DD statement must be the same as those of IMSACB DD.

All the data sets that are listed within the IMSRESLB concatenation must be APF-authorized.

IAREAnnn/IARnnnnn DD

This DD statement is a required DD statement when the IDSNMASK keyword is not specified. This statement defines the input data set that contains the unloaded (or created) segment data records, each of which is composed of a prefix and segment data, for one or more areas for the RELOAD command.

This data set can be allocated dynamically by the IDSNMASK keyword parameter of the RELOAD command, or it can be specified with one of following DD statements explicitly:

- //IAREAnnn DD JCL statement
- //IARnnnnn DD JCL statement

An IAREAnnn DD statement is for areas in the range of 1 - 999, and an IARnnnnn DD statement is for areas in the range of 1 - 2048. If the area number of the unloaded area is greater than 999, you must use the IARnnnnn DD statement. The value of *nnn* or *nnnnn* must be right-aligned, with leading zeros if necessary. The area number is the area number that is specified for the IAREA= parameter of the RELOAD command.

The area name that is specified for the IAREA= parameter with the RELOAD command affects only the DD name that is used for the input data set. The Reload function does not check whether the DD name of an unload segment records data set corresponds to the area to be reloaded.

Data set that is created by the EXTRACT command with EFORMAT=DBT or EFORMAT=TFMT, or by the UNLOAD command with FORMAT=DBT or FORMAT=TFMT can be used as an input for the RELOAD command to reload areas.

If the unloaded data set, which was unloaded in a user-specified format with the FORMAT=USER keyword, satisfies all of the following conditions, the data set can be used as an input for the Reload function.

- The USER_RECORD subcommand is not specified.
- SEGMENT_NAME or SEGMENT_CODE is specified on the SEGREC_PREFIX keyword.
- When COMPRTN is specified in the DBD definition, SEGMENT_ATTR is specified on the SEGREC_PREFIX keyword.
- When SEGMENT_CKEY is specified on the SEGREC_PREFIX keyword, SEGMENT_CKEY_LENGTH is specified before SEGMENT_CKEY.
- When KEY or SEGMENT_KEY is specified on the SEGREC_PREFIX keyword, SEGMENT_KEY_LENGTH is specified before KEY or SEGMENT_KEY.

For the format of the unloaded segment record that was created by the UNLOAD command with FORMAT=DBT, or for the format of the extracted file record that was created by the EXTRACT command with EFORMAT=DBT, see [“Unloaded segment records layout”](#) on page 1177.

For the format of the unloaded segment record that was created by the UNLOAD command with FORMAT=USER, see [“Extracted records layout”](#) on page 1186.

Note on dynamic allocation:

When this statement is not specified in the JCL statement, the data set is allocated dynamically by using the IDSNMASK keyword.

HFPSYSIN DD

This DD statement is a required DD statement that defines the input control statement that specifies the FPA functions.

The HFPSYSIN DD statement can be coded as a standard SYSIN file, a sequential data set, or a PDS member. LRECL=80 is required for the DCB of this data set. Only one command, besides the GLOBAL command, can be specified in HFPSYSIN. FPA analyzes the command syntax in columns 1 - 72. Columns 73 - 80 are regarded as comment.

For details about coding the HFPSYSIN DD statement, see [“Utility control statements for the Reload function”](#) on page 108.

HFPAPARM DD

This DD statement is an optional DD statement. This statement defines the control parameters for generating the analysis reports.

When SENSOR=YES is specified, the values of RBASEFS and RDOVFFS keywords are stored in the Sensor Data repository as the values of data elements DB_THRESHOLD_RBASEFS and DB_THRESHOLD_RDOVFFS. These values are used by the Online Space Management utility as the values of RBASEFS and RDOVFFS keywords for the OER utility process (online reorganization process).

The HFPAPARM DD statement can be coded as a sequential data set or as a PDS member. LRECL=80 is required for the DCB of this data set. The HFPAPARM keywords must be written on one line. It can be coded as follows:

```
//HFPAPARM DD *
   RDOVFFS=40 SYNTBL=20 UD0VFFS=35 RBASEFS=40
/*
```

You can specify the keywords that are listed in the following table.

Table 8. HFPAPARM keywords

Keyword	Description
ARDIGIT	This keyword specifies the digit of the area number field for the UOW group definition records that are defined in the HFP AUOWC file. The value is 3 or 5. The default is 3. If the HFP AUOWC file is not specified in the JCL statement, this keyword is ignored.

Table 8. HFPAPARM keywords (continued)

Keyword	Description
RBASEFS	This keyword is an optional keyword. If a RAP CI contains free space that is greater than the percentage specified, and the RAP CI also uses overflow CIs, a reorganization control statement is generated for the UOW, and the UOW is included in the UOW Exception report. The default is 100%, which means that reorganization control statements are never generated. The maximum value is 99, and the minimum value is 0.
RDOVFFS	This keyword is an optional keyword. If the DOVF part of a UOW contains free space that is greater than the percentage specified, and the UOW also uses IOVF CIs, a reorganization control statement is generated for the UOW and the UOW is included in the UOW Exception report. The default is 100%, which means that reorganization control statements are never generated. The maximum value is 99, and the minimum value is 0.
RGROUP	This keyword sets the group identifier for the statistics that are collected and stored in the History2 file or in a sensor data record set. For the History2 file, the keyword value is set in the Repository Group field, and for a sensor data record set, the value is set for the DB_SENSOR_DATA_GROUP_ID data element. This keyword has no default value. If this keyword is not specified, FPA sets the group identifier as follows: <ul style="list-style-type: none">• The Repository Group field in the History2 file contains 4 bytes of blank characters.• The DB_SENSOR_DATA_GROUP_ID data element is not included in the sensor data record set.
SYNTBL	This keyword is an optional keyword. It controls the number of reporting entries in the Synonym Chain Analysis table. The default value is 41 (that is, all synonym chains longer than 40 are grouped and reported together). The maximum value is 999, and the minimum value is 12.
UDOVFFS	This keyword is an optional keyword. If the DOVF UOW contains free space that is less than the percentage specified, the UOW is included in the UOW Exception report. The default is 0%, which means that the UOW is never included in the UOW Exception report. The maximum value is 99, and the minimum value is 0.

Note: The RBASEFS keyword and the RDOVFFS keyword that are used for the HFPAPARM data set are different from those used for the REORG command.

Requirement: Generating reorganization control statements for the High-Speed DEDB Direct Reorganization utility, you must specify the free space control percentages. You must also specify the HFPARERG DD statement in the JCL stream. For details, see the description of HFPARERG DD.

HFPAUOWC DD

This DD statement is a required DD statement when HFPAHST2 DD is specified. Each record in this data set specifies a range of UOW numbers of an area, which defines a UOW group of the area. The information about the UOW range is recorded in the HISTORY2 data set or in a sensor data record set in the order that is specified in the HFPAUOWC data set, even if the UOW ranges overlap or are in reverse order. You can specify the UOW ranges for each area in any order. However, you must specify the area numbers in ascending order.

The data set can reside in the JCL input stream. The data set can also be defined as a sequential data set or as a member of a partitioned data set. If a data set is specified, the data set must have the following attributes:

- DSORG=PS
- RECFM=FB

- LRECL=80
- BLKSIZE=*a multiple of 80*

If the data set is empty, all UOWs of the area are regarded as the target UOW range.

When you specify the HFPAHST2 DD statement, you must also specify the HFPAUOWC DD statement. If an HFPAUOWC DD statement is not specified, the HFPAHST2 DD statement is ignored.

When SENSOR_UOW=YES is specified, the data elements that will be stored differ by the HFPAUOWC DD specification as follows:

- If you specify the HFPAUOWC DD statement, data elements are collected at the UOW group level.
- If you do not specify the HFPAUOWC DD statement, data elements are collected at the UOW level.

For information about defining a UOW group, see [“UOW group definition record layout” on page 1191](#).

HFPCSITE DD

This DD statement is an optional DD statement. The HFPCSITE DD statement can be used to define the data set name of the site default table. When the SITE_DSNAME keyword is specified in the HFPSYSIN DD statement, the HFPCSITE DD statement is allocated dynamically. HFPCSITE DD in the JCL is mutually exclusive with the SITE_DSNAME keyword in the HFPSYSIN DD statement. The data set attribute of HFPCSITE DD is PDS.

ICEIN DD

This DD statement is an optional DD statement. This statement specifies the input data set that contains the control statements for IMS HP Image Copy. The data set can reside on a tape, a direct-access volume, or it can be routed through the input stream (DD * or DD DATA).

This data set is used only when you specify ICOUT=YES. If the DD statement is not coded and if ICOUT=YES is specified, the default control statements are used. For details about the ICEIN control statements, see *IMS High Performance Image Copy User's Guide*.

Output DD statements

The following output DD statements are used for the Reload function.

***addnname/areaname* DD**

This DD statement is a required DD statement. This statement defines an output area data set, which is the VSAM ESDS for an area to be reloaded. The ddname used must be the ddname for the area that is defined in the output ACB library (IMSACB DD). There can be multiple *addnname* DD statements for each area where data exists in the input file (IAREAnn or IARnnnnn DD) for the RELOAD command.

It is possible that no segments are reloaded. In this case, the output area data set is initialized (that is, an empty area).

This data set can be allocated dynamically by one of following methods:

- The ODSNMASK keyword parameter of the RELOAD command
- When the GLOBAL DBRC=YES option is specified, the ADS information in the DBRC RECON data sets
- The area name member in the IMS DFSMDA library

Or it can be specified with one of the following DD statements explicitly:

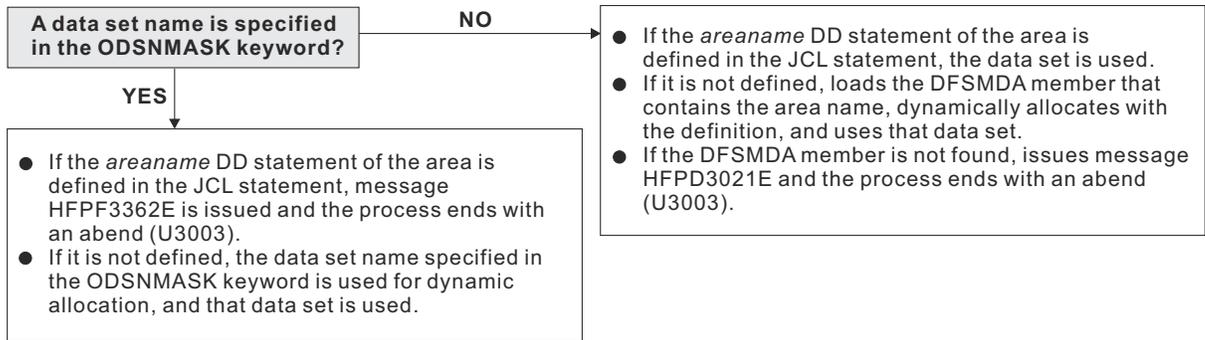
- *//addnname* DD JCL statement
- *//areaname* DD JCL statement

The area data set can be also defined by the ALLOCATE subcommand with the RELOAD command.

For the RELOAD command output, if the area is registered with DBRC, it must be in the RECOV NEEDED status; that is, all ADSs of the area must be in the UNAVAILABLE status in DBRC.

The following figure shows how FPA determines area data set names.

When DBRC=NO:



When DBRC=YES:

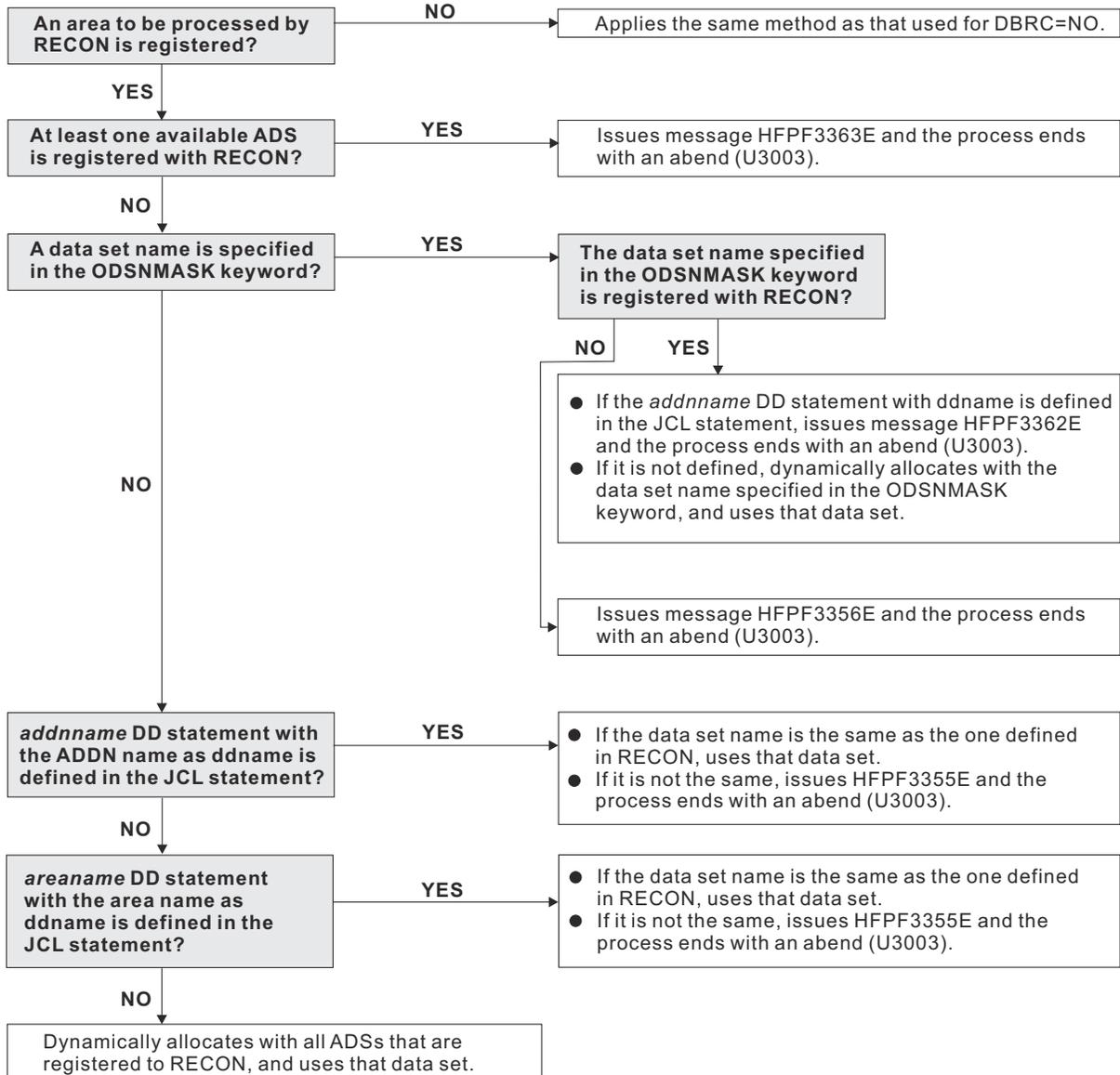


Figure 40. Output data set names for the RELOAD command

HFPPRINT DD

This DD statement is an optional DD statement that defines the processing log output data set. The data set contains the processing messages that are issued by FPA.

The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set. You cannot specify DUMMY.

The HFPPRINT data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPPRINT DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*.

The following figure shows an example of the processing messages that are issued for the RELOAD command.

```
IMS HPFP UTILITIES                                "Processing messages"                                PAGE: 1
5698-FPP V2R1                                     2019-11-22 13:03:37
HFPC0001I COMMAND ANALYZE PROCESSING STARTED.
HFPC0005I DBRC=NO IS SPECIFIED.
- EEQE DETECTION IS NOT PERFORMED.
HFPC0002I COMMAND ANALYZE PROCESSING COMPLETED.
HFPP0001I RELOAD PROCESSING STARTED FOR DATABASE: DEDBJN21
HFPK0011I RELOAD PROCESSING STARTED FOR UNLOADED SEGMENT RECORD FILE.
- DDNAME: IAREA001, DSNAME: HPFP.HFPUSR.DB21AR0
HFPK0001I RELOAD PROCESSING STARTED FOR AREA NO: 1, AREANAME: DB21AR0, USR FILE DDNAME: IAREA001
HFPK0002I RELOAD PROCESSING COMPLETED FOR AREA NO: 1, AREANAME: DB21AR0, USR FILE DDNAME: IAREA001
HFPK0012I RELOAD PROCESSING COMPLETED FOR UNLOADED SEGMENT RECORD FILE.
- DDNAME: IAREA001, DSNAME: HPFP.HFPUSR.DB21AR0
HFPP0002I RELOAD PROCESSING COMPLETED FOR DATABASE: DEDBJN21
HFPL0001I REPORT STARTED FOR COMMAND: RELOAD, DATABASE: DEDBJN21
HFPL0002I REPORT COMPLETED FOR COMMAND: RELOAD, DATABASE: DEDBJN21
```

Figure 41. Processing messages issued for the RELOAD command

HFPRPTS DD

This DD statement is an optional DD statement that defines the report output data set. The data set contains the reports that can be used to verify the process of the command. The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

The following reports are generated in this data set:

- Audit report
- Processing report
- DBD Definition report
- Reloaded report
- Input File report

When the REPORT subcommand and PTRCHKLVL=FULL are specified, the following reports are generated:

- DEDB Area Analysis reports
- Process Summary of Analysis report
- Resolve Pointers report
- UOW report
- UOW Exception report
- Largest Database Records report (When LARGEREC_REPORT=YES is specified)

- Consolidated Largest Database Records report (When LARGEREC_REPORT=YES is specified and the second parameter of the LARGEREC_COUNT keyword specifies a value that is equal to or larger than 1)
- Segment Length Distribution report
- DB Record Length Distribution report
- RBA of Used IOVF CI report

When the REPORT subcommand and PTRCHKLV=QUICK are specified, the following reports are generated:

- DEDB Area Analysis reports (FREESPACE ANALYSIS only)
- Process Summary of Analysis report
- UOW report (without DB record information)
- UOW Exception report (without DB record information)
- Pointer Checksum report
- RBA of Used IOVF CI report

You can generate reports related to the Analyze process in another DD data set that is specified by the REPORTDD= keyword parameter of the REPORT subcommand.

The HFPRPTS data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPRPTS DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

HFPSRT DD

This DD statement is an optional DD statement that defines the report output data set when SENSOR=YES is specified. This output data set contains the Sensor Data Statistics report that can be used to verify which sensor data is stored in the Sensor Data repository of IMS Tools KB. The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set. You can specify NULLFILE or DUMMY for this data set.

The HFPSRT data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPSRT DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

HFAHST DD

This DD statement is an optional DD statement. This statement defines an output data set that contains historical records. The analysis report generation process adds one record for each DEDB area that it processes. The record contains the key space utilization and performance information for that area. This statement is used only when you specify both the REPORT subcommand and the PTRCHKLV=FULL keyword.

If this feature is used, the HFAHST data set must be preallocated with the following attributes:

- DSORG=PS

- RECFM=FB
- LRECL=100
- BLKSIZE=*user-specified*

The DD statement in the JOB stream is specified as:

```
//HFPAHST DD DSN=HPFP.HISTORY,DISP=(MOD,KEEP,KEEP)
```

You can generate the records in another DD data set specified by the HISTORYDD= keyword parameter instead of HFPAHST DD.

HFPAHST2 DD

This DD statement is a required DD statement when HFPAUOWC DD is specified. This statement defines an output data set that contains historical records. The analysis report generation process adds one record for each UOW group that is specified by HFPAUOWC DD. The record contains the key space utilization and performance information for that area, which is based on the information obtained from the specified UOW range. This statement is used only when you specify both the REPORT subcommand and the PTRCHKLVL=FULL keyword. If HFPAUOWC DD statement is not specified, this statement is ignored.

If this feature is used, the HFPAHST2 data set must be preallocated with the following attributes:

- DSORG=PS
- RECFM=FB
- LRECL=200
- BLKSIZE=*user-specified*

The DD statement in the JOB stream is specified as:

```
//HFPAHST2 DD DSN=HPFP.HISTORY2,DISP=(MOD,KEEP,KEEP)
```

HFPARERG DD

This DD statement is an optional DD statement. This statement defines an output data set that contains the reorganization control statements for the High-Speed DEDB Direct Reorganization utility (DBFUHDR0). This statement is used only when you specify both the REPORT subcommand and the PTRCHKLVL=FULL or the PTRCHKLVL=QUICK keyword.

The following DCB attributes of the file are set by the Analyze process:

- DSORG=PS
- RECFM=FB
- LRECL=80
- BLKSIZE=800

Do not code the DCB information in your JCL statement. Do not specify DISP=MOD for this DD statement.

Requirement:

Generating reorganization control statements for the High-Speed DEDB Direct Reorganization utility, you must specify the HFPARERG DD statement in the JCL stream. You must also specify the free space control percentages. For details about specifying the free space control percentages, see the description of HFPAPARM DD.

HFPA SNAP DD

This DD statement is a required DD statement when CISNAP=YES. This statement defines the output data set for the CI Map/CI Dump report and the Pointer Chain Reconstruction report. This DD statement is effective when the REPORT subcommand, PTRCHKLVL=FULL or PTRCHKLVL=QUICK, and CISNAP=YES are specified.

The following DCB attributes of the file are set by the Analyze process:

- RECFM=FBA

- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement.

The HFPASNAP data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPASNAP DD SYSOUT=A
```

HFPTOTAL DD

This DD statement is an optional DD statement. This statement defines an output data set that contains the total number of reloaded segments in an area. When this DD statement is present, the Reload process writes the total number of reloaded segments of each segment type in an area.

The following DCB attributes of the file are set by the FPA main task:

- DSORG=PS
- LRECL=60
- BLKSIZE=6000

Do not code the DCB information in your JCL statement.

The format of this record is specified in the HFPZTOT macro, which is provided as a product-sensitive user interface. For details of the record format, see [“Total segments records layout” on page 1183](#).

HFPABEND DD

This DD statement is an optional DD statement. It defines a snap dump output data set that is used for diagnostic purposes. This data set might be used when the FPA job ends abnormally. Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

ctttSMMSG DD

This DD statement is an optional DD statement. However, this DD statement is required when DFSORT is used. This statement defines an output data set for storing DFSORT messages. For the condition for running DFSORT and the naming rules for the DD names, see the description of *ctttWKnn* DD.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

itttSMMSG DD

This DD statement is an optional DD statement. However, this DD statement is required when DFSORT is used. This statement defines an output data set for storing DFSORT messages. For the condition for running DFSORT and the naming rules for the DD names, see the description of *itttWKnn* DD.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

DFSPRINT DD

This DD statement is an optional DD statement. This statement specifies the output data set that contains the messages that are produced by IMS HP Image Copy. The data set can reside on a tape, a direct-access volume, or a printer, or it can be routed through the output stream (SYSOUT).

This data set is used only when you specify ICOUT=YES. If the DD statement is not coded and ICOUT=YES is specified, a SYSOUT stream is dynamically allocated for the DD.

ICEPRINT DD

This DD statement is an optional DD statement. This statement specifies the output data set that contains the reports that are produced by IMS HP Image Copy. The data set can reside on a tape, a direct-access volume, or a printer, or it can be routed through the output stream (SYSOUT).

This data set is used only when you specify ICOUT=YES. If the DD statement is not coded and ICOUT=YES is specified, a SYSOUT stream is dynamically allocated for the DD.

ICERPRT DD

This DD statement is an optional DD statement. This statement specifies the output data set that contains the Image Copy Dump Process report, which is produced by IMS HP Image Copy. It can reside on a tape, a direct-access volume, or a printer, or it can be routed through the output stream (SYSOUT).

This data set is used only when you specify ICOUT=YES.

dataout1 DD

This DD statement is an optional DD statement. This statement defines the primary image copy data set. One DD statement is required for each data set that is to be dumped. The DD name can be any 1- to 8-character string. The data set specified by the DD statement must be on either a direct-access device or a tape device.

This data set is used only when you specify ICOUT=YES. For details of the format and the characteristics of the image copy data set, see *IMS High Performance Image Copy User's Guide*. You can request IMS HP Image Copy to allocate the image copy data set dynamically by omitting this DD statement and specifying the allocation parameters that are associated with the data set in the ICEIN data set.

dataout2 DD

This DD statement is an optional DD statement. This statement defines the secondary image copy data set. This statement is required only if the associated ICEIN control statement requests the secondary image copy. The data set specified by the DD statement must be on either a direct-access device or a tape device.

This data set is used only when you specify ICOUT=YES. For details of the format and the characteristics of the image copy data set, see *IMS High Performance Image Copy User's Guide*.

Input/Output DD statements

The following input/output DD statements are used for the Reload function.

HFPCHKPT DD

This DD statement is a function-unique DD statement for the checkpoint restart option. The DD statement is required when RESTART=YES is specified. This statement defines the checkpoint file. It must be the same file used both for a particular checkpoint and for the associated attempted subsequent restarts.

If RESTART=YES is specified, the checkpoint file (HFPCHKPT) must be specified in the job step. It must not be specified as DUMMY or NULLFILE, otherwise, the checkpoint restart capability is disabled. The checkpoint file must be allocated with the following attributes, otherwise, the job ends with error.

- DISP=MOD is not specified
- RECFM=VB is specified
- LRECL is larger than or equal to 22000

For more information, see [“RESTART keyword” on page 514](#).

RECON1/RECON2/RECON3 DD

These statements define the DBRC RECON data sets. These DD statements are required DD statements when both of the following conditions are met:

- DBRC=YES is specified for the GLOBAL command
- DFSMDA members for RECON n data sets are not used

The RECON n data set must be the same data set that is used in the IMS control region. Because FPA issues a signon, a signoff, an authorization call, and an unauthorization call to DBRC, it updates RECON n data sets.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is necessary, the data set is allocated dynamically by using a DFSMDA member.

ctttWKnn DD

These DD statements are optional DD statements. These data sets are intermediate storage data sets that are used by DFSORT. For more information about coding the SORTWKnn DD statements, see *DFSORT Application Programming Guide*.

The adequate size of storage data sets is 1.5 to 2.5 times the input file size. In the Reload function, the input file is the unloaded segment record file. The size of the file is computed by multiplying the average length of the unloaded segment record by the number of segments. Allocating twice the space used by the SORTIN data set is usually adequate for each work data set. This data set is allocated dynamically by OS. When the SORTIN is too large for sort, a B37 abend occurs. In this case, this DD is a required DD.

Do not specify DISP=MOD for these DD statements.

The naming rules of the ddnames are as follows:

c:

The thousands digit of *n* for *n*-th input thread subtask.

U:

If *n* is 999 or less, U must be specified.

V:

If *n* is 1000 - 1999, V must be specified.

W:

If *n* is 2000 - 2048, W must be specified.

Note: When a subtask ends processing an area, it starts processing the next area. For this reason, the *n*-th input thread subtask does not always process the *n*-th input data set.

ttt:

The hundreds, tenths, and units digits of *n*.

nn:

Specifies the number of the SORT work data sets. This value must be 01 - 03.

Note on dynamic allocation:

If this statement is not specified in the JCL statement when these data sets are necessary, FPA allocates them dynamically.

itttWKnn DD

These are optional DD statements. These are intermediate storage data sets that are used by DFSORT. For more information about coding the SORTWKnn DD statements, see *DFSORT Application Programming Guide*.

Do not specify DISP=MOD for these DD statements.

The adequate size of storage data sets is 1.5 to 2.5 times the input file size. In the analyze step, the size of each input file is as follows:

PTRCHKLVL=FULL

30 x the number of pointers in the area

SDEPCHKLVL=FULL

30 x the number of SDEP segments in the area

CISNAP=YES

112 x the number of pointers in which pointer errors are detected

REPORT subcommand and PTRCHKLVL=FULL are specified

X x (the number of non-SDEP segments in the areas + the number of free space elements in the areas + the number of references to other CIs)

Where *X* is:

- When LARGEREC_REPORT=NO is specified, X is calculated as follows:

$$34 + (\text{the maximum segment level} \times 4)$$

However, if X is smaller than 48, 48 is used for X .

- When LARGEREC_REPORT=YES is specified, X is calculated as follows:

$$34 + (\text{the maximum segment level} \times 4) + \text{root key length}$$

However, if X is smaller than 48, 48 is used for X .

REPORT subcommand and PTRCHKLVL=QUICK are specified

$$23 \times (\text{the number of free space elements in the area} + \text{the number of references to other CIs})$$

Allocating twice the space used by the SORTIN data set is usually adequate for each work data set.

The naming rules of the ddnames are as follows:

i:

A, B, and C:

If PTRCHKLVL=FULL is specified.

D, E, and F:

If SDEPCHKLVL=FULL is specified.

G, H, and I:

If CISNAP=YES is specified.

J, K, and L:

If the REPORT subcommand is specified.

The thousands digit of the area number of the area to be analyzed, or the thousands digit of n for n -th input thread subtask.

A, D, G, and J:

If the number of areas to be analyzed or n is 999 or less.

B, E, H, and K:

If the number of areas to be analyzed or n is 1000 - 1999.

C, F, I, and L:

If the number of areas to be analyzed or n is 2000 - 2048.

Note: When a subtask ends processing an area, it starts processing the next area. For this reason, the n -th input thread subtask does not always process the n -th input data set.

ttt:

The hundreds, tenths, and units digits of the area number of the area to be analyzed, or those of n .

nn:

The number of the SORT work data sets. This value must be 01 - 03.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when these data sets are necessary, FPA allocates them dynamically.

Utility control statements for the Reload function

The utility control statements for the Reload function define the user description of the GLOBAL command, the RELOAD command, and the optional ALLOCATE, LOADCTL, REPORT, and THRESHOLD subcommands, all of which control the Reload process.

The utility control statements must be specified in the HFPSYSIN data set. This control statement data set usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte, fixed-length records. Block size, if coded, must be a multiple of 80.

The following figure shows a sample of how to code the HFPSYSIN data set.

```
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
  RELOAD
  DBD=dbdname
/*
```

Figure 42. RELOAD HFPSYSIN data set

Subsections:

- [“GLOBAL command keywords” on page 109](#)
- [“RELOAD command keywords” on page 110](#)
- [“ALLOCATE subcommand keywords” on page 112](#)
- [“LOADCTL subcommand keywords” on page 113](#)
- [“REPORT subcommand keywords” on page 113](#)
- [“THRESHOLD subcommand keywords” on page 114](#)

GLOBAL command keywords

For the GLOBAL command, you can specify the keywords that are listed in the following table. For the format of the GLOBAL command, see [“GLOBAL command” on page 394](#).

Table 9. GLOBAL command keywords

Keyword	Required or optional	Description	Topic
ADXCGRP	Optional	Specifies the name of the Autonomics Director XCF group.	“ADXCGRP keyword” on page 439
CISNAP	Optional	Requests the CI Map/CI Dump report and the Pointer Chain Reconstruction report.	“CISNAP keyword” on page 448
DBRC	Optional	Specifies to request or bypass the database recovery control (DBRC) facility process for the database that is specified by the DBD.	“DBRC keyword” on page 451
HISTORYDD	Optional	Specifies the DD statement for the output History file data set that is specified in the JCL.	“HISTORYDD keyword” on page 469
ICINCOMP	Optional	Specifies, when an image copy is used as an input data set, whether the image copy records are to be compressed by using the CSRCEserv MVS macro when they are stored into a data space storage.	“ICINCOMP keyword” on page 474
ICNOTIFY	Optional	Specifies whether to issue the NOTIFY.IC command to DBRC.	“ICNOTIFY keyword” on page 475
ICOUT	Optional	Specifies whether to generate one or more image copies.	“ICOUT keyword” on page 476
IMSCATACB_INPUT	Optional	Specifies whether to retrieve the currently active ACB definition or the pending ACB definition from the IMS directory.	“IMSCATACB_INPUT keyword” on page 480
IMSCATHLQ	Optional	Specifies the high-level qualifier of the bootstrap data set of the IMS directory.	“IMSCATHLQ keyword” on page 481

Table 9. GLOBAL command keywords (continued)

Keyword	Required or optional	Description	Topic
ITKBDATA	Optional	Specifies the type of additional data to store in the IMS Tools Base IMS Tools Knowledge Base repository.	“ITKBDATA keyword” on page 487
ITKBLOAD	Optional	Specifies the library name of IMS Tools KB.	“ITKBLOAD keyword” on page 487
ITKBSRVR	Optional	Specifies the name of the IMS Tools KB server XCF group.	“ITKBSRVR keyword” on page 488
LIUGOPT	Optional	Specifies the LIU global option module that is to be used by IMS Library Integrity Utilities.	“LIUGOPT keyword” on page 493
MSTIMESTAMP	Optional	Specifies whether to include a time stamp in the messages.	“MSTIMESTAMP keyword” on page 497
PTRCHKLVL	Optional	Specifies the method that is used to run integrity verification in the root addressable part and in the independent overflow part.	“PTRCHKLVL keyword” on page 506
RESTART	Optional	Enables the checkpoint restart option for reprocessing failed areas.	“RESTART keyword” on page 514
RESTART_COUNT	Optional	Specifies the maximum number of restart processing runs for the checkpoint restart option.	“RESTART_COUNT keyword” on page 515
SCAN	Optional	Verifies the syntax of command, keywords, and parameters in command sets without running FPA. The Scan function also allocates some (but not all) internal resources for running the process function.	“SCAN keyword” on page 523
SDEPCHKLVL	Optional	Specifies the method that is used to run integrity verification in the sequential dependent part.	“SDEPCHKLVL keyword” on page 526
SDEP_ORPHAN_MSGLVL	Optional	Specifies the error message severity level to be set when an orphan SDEP segment is detected.	SDEP_ORPHAN_MSGLVL keyword
SITE_DSNAME	Optional	Specifies the data set name of the site default table.	“SITE_DSNAME keyword” on page 539
SITE_MEMBER	Optional	Specifies the member name of the site default table.	“SITE_MEMBER keyword” on page 540

RELOAD command keywords

For the RELOAD command, you can specify the keywords that are listed in the following table. For the format of the RELOAD command, see [“RELOAD command” on page 405](#).

Table 10. RELOAD command keywords

Keyword	Required or optional	Description	Topic
ACBCHK	Optional	Specifies whether to check the DBD information in the library that is specified on the IMSACB DD statement with the DBD information in the USR files.	“ACBCHK keyword” on page 438
ATASKCTL	Optional	Specifies the maximum number of output areas to be analyzed concurrently.	“ATASKCTL keyword” on page 443
CISNAP	Optional	Requests the CI Map/CI Dump report and the Pointer Chain Reconstruction report when the reloaded area is analyzed concurrently.	“CISNAP keyword” on page 448
DBD	Required	Specifies the database description (DBD) that contains the areas to be processed.	“DBD keyword” on page 450
EXITRTN	Optional	Specifies the name of the user exit routine to be invoked prior to the write process of the Reload function.	“EXITRTN keyword” on page 461
FORMAT	Optional	Specifies the format of the unloaded file.	“FORMAT keyword” on page 465
HISTORYDD	Optional	Specifies the DD statement for the output History file data set that is specified in the JCL when the reloaded area is analyzed concurrently.	“HISTORYDD keyword” on page 469
IAREA	Optional	Specifies one or more input areas for the Reload process.	“IAREA keyword” on page 469
ICHECK	Optional	Controls the concurrent number of input devices to access by using the ITASKCTL keyword.	“ICHECK keyword” on page 473
ICNOTIFY	Optional	Specifies whether to issue the NOTIFY.IC command to DBRC.	“ICNOTIFY keyword” on page 475
ICOUT	Optional	Specifies whether to generate one or more image copies.	“ICOUT keyword” on page 476
IDSNMASK	Optional	Specifies the data set name or the mask for dynamically allocating cataloged pre-existing input data sets.	“IDSNMASK keyword” on page 476
ITASKCTL	Optional	Specifies the maximum number of input areas to be processed concurrently.	“ITASKCTL keyword” on page 486
KEYCHGCHK	Optional	Specifies whether to check if the key sequence field was not changed by a user exit routine.	“KEYCHGCHK keyword” on page 489
KEYSEQCHK	Optional	Specifies what to do if a key sequence error is detected.	“KEYSEQCHK keyword” on page 490
MAXERROR	Optional	Specifies the maximum number of pointer errors that might be encountered before processing ends with an error.	“MAXERROR keyword” on page 495
OAREA	Optional	Specifies one or more output areas to the process.	“OAREA keyword” on page 497
OCACHE	Optional	Specifies the type of control intervals that are held in a data space storage until all processing has completed.	“OCACHE keyword” on page 499
OCHECK	Optional	Controls the concurrent number of output devices to access.	“OCHECK keyword” on page 500
ODSNMASK	Optional	Specifies the data set name or mask for dynamically allocating cataloged pre-existing output data sets.	“ODSNMASK keyword” on page 500
PTRCHKLVL	Optional	Specifies the method that is used to run integrity verification in the root addressable part and in the independent overflow part.	“PTRCHKLVL keyword” on page 506
SDEPCHKLVL	Optional	Specifies the method that is used to run integrity verification in the sequential dependent part.	“SDEPCHKLVL keyword” on page 526
SDEP_ORPHAN_MSGLVL	Optional	Specifies the error message severity level to be set when the reloaded area is analyzed concurrently and an orphan SDEP segment is detected.	SDEP_ORPHAN_MSGLVL keyword

Table 10. RELOAD command keywords (continued)

Keyword	Required or optional	Description	Topic
SEGREC_PREFIX	Optional	Specifies the contents of the fields within the prefix portion of the segment records.	“SEGREC_PREFIX keyword” on page 531
SEGREC_SUFFIX	Optional	Specifies the contents of the fields within the suffix portion of the segment records.	“SEGREC_SUFFIX keyword” on page 533
SENSOR	Optional	Specifies whether to store the sensor data in the Sensor Data repository of IMS Tools KB.	“SENSOR keyword” on page 535
SENSOR_DBREC	Optional	Specifies whether to store the data elements that are related to database records and root segment distribution in the Sensor Data repository of IMS Tools KB.	“SENSOR_DBREC keyword” on page 536
SENSOR_PTRERR	Optional	Specifies to store the sensor data in the Sensor Data repository of IMS Tools KB even when pointer errors are detected in the area.	“SENSOR_PTRERR keyword” on page 537
SENSOR_UOW	Optional	Specifies whether to store the data elements that are related to UOW in the Sensor Data repository of IMS Tools KB.	“SENSOR_UOW keyword” on page 537
SENSOR_UOW_REPORT	Optional	Specifies whether to print the data elements that are related to UOW in the Sensor Data Statistics report.	SENSOR_UOW_REPORT keyword
SORT	Optional	Loads in sequence based on RAP (root anchor point).	“SORT keyword” on page 541
SORTOPT_FILSZ	Optional	Specifies that FPA adds the FILSZ parameter to the SORT control statement.	“SORTOPT_FILSZ keyword” on page 542
SSP	Optional	Specifies whether subset pointers should be retained or cleared.	“SSP keyword” on page 544

ALLOCATE subcommand keywords

The ALLOCATE subcommand dynamically defines a VSAM cluster that is used for the processing of the RELOAD command.

For the ALLOCATE subcommand, you can specify the keywords that are listed in the following table. For the format of the ALLOCATE subcommand, see [“ALLOCATE subcommand” on page 419](#).

Table 11. ALLOCATE subcommand keywords

Keyword	Required or optional	Description	Topic
AMSOPT	Optional	Passes optional parameters to the Access Method Services (AMS) to define VSAM clusters.	“AMSOPT keyword” on page 440
ATTRIBUTE	Optional	Requests optional processing related to allocation of a VSAM cluster.	“ATTRIBUTE keyword” on page 444
AVGREC	Optional	Identifies the space requirements of the output data set that is created by dynamic allocation.	“AVGREC keyword” on page 445
DATACLAS	Optional	Specifies the SMS data class of an output data set that is created by dynamic allocation.	“DATACLAS keyword” on page 449
KEYLABEL	Optional	Specifies the key label for encrypting a new VSAM cluster.	“KEYLABEL keyword” on page 489
MGMTCLAS	Optional	Specifies the SMS management class of an output data set that is created by dynamic allocation.	“MGMTCLAS keyword” on page 496
OAREA	Required	Specifies one or more output areas for the process.	“OAREA keyword” on page 497
SPACE	Optional	Specifies the space requirements of an output data set that is created by dynamic allocation.	“SPACE keyword” on page 543

Table 11. ALLOCATE subcommand keywords (continued)

Keyword	Required or optional	Description	Topic
STORCLAS	Optional	Specifies the SMS storage class of an output data set that is created by dynamic allocation.	“STORCLAS keyword” on page 546
VOLSER	Optional	Specifies one or more volume-serial identifiers for the data set that is accessed by dynamic allocation. If this keyword is omitted for an output data set, nonspecific (private) volumes are required.	“VOLSER keyword” on page 556

LOADCTL subcommand keywords

The LOADCTL subcommand specifies the load control information for a segment. You can specify an Insert Limit Count (ILC) and a place to insert dependent segments (either DOVF or IOVF). ILC is the maximum number of occurrences of segments that are inserted close to their root.

If the LOADCTL subcommand is specified multiple times for the same segment, they will be checked from the first to the last, and the first LOADCTL statement that matches the condition will be used for segment insertion.

For the LOADCTL subcommand, you can specify the keywords that are listed in the following table. For the format of the LOADCTL subcommand, see [“LOADCTL subcommand” on page 429](#).

Table 12. LOADCTL subcommand keywords

Keyword	Required or optional	Description	Topic
INSERT_LIMIT_COUNT	Optional	Specifies the maximum number of segments that are selected by the SEGMENT keyword, that can be inserted close to their root segment. The maximum value is 32767 and the minimum value is 0. The minimum value 0 specifies that all specified segments are stored far from their root segment.	INSERT_LIMIT_COUNT keyword
LOCATION	Optional	Specifies the location to insert dependent segments, either DOVF or IOVF. You can specify the location where segments that were selected by the SEGMENT keyword are loaded.	“LOCATION keyword” on page 494
SEGMENT	Required	Specifies the segment name to be inserted at the specified location. The segment name must be a name that is coded on the SEGM statement in your DBD.	“SEGMENT keyword” on page 530

REPORT subcommand keywords

The REPORT subcommand requests the analysis reports. If you do not specify PTRCHKLVL=FULL or PTRCHKLVL=QUICK, this subcommand is ignored.

For the REPORT subcommand, you can specify the keywords that are listed in the following table. For the format of the REPORT subcommand, see [“REPORT subcommand” on page 432](#).

Table 13. REPORT subcommand keywords

Keyword	Required or optional	Description	Topic
DBRLDIST	Optional	Specifies whether to generate the DB Record Length Distribution report.	“DBRLDIST keyword” on page 452
IOVFCI_RBA_REPORT	Optional	Specifies whether to generate the RBA of Used IOVF CI report.	“IOVFCI_RBA_REPORT keyword” on page 484
LARGEREC_COUNT	Optional	Specifies the number of reporting entries in the Largest Database Records report and the Consolidated Largest Database Records report. This keyword is effective only when PTRCHKLVL=FULL is specified.	“LARGEREC_COUNT keyword” on page 491

Table 13. REPORT subcommand keywords (continued)

Keyword	Required or optional	Description	Topic
LARGEREC_REPORT	Optional	Specifies whether to generate the Largest Database Records report. This keyword is effective only when PTRCHKLVL=FULL is specified.	“LARGEREC_REPORT keyword” on page 491
REPORTDD	Optional	Specifies the destination ddname for the analysis reports.	“REPORTDD keyword” on page 513
SEGLDIST	Optional	Specifies whether to generate the Segment Length Distribution report.	“SEGLDIST keyword” on page 529
SEGLSTAT	Optional	Specifies the format of the DB Record Profile Analysis report, which is one of the DEDB Area Analysis reports.	“SEGLSTAT keyword” on page 529
STARTUOW	Optional	Specifies the number that defines the first UOW to be reported in the UOW report and the UOW Exception report.	“STARTUOW keyword” on page 545
STOPUOW	Optional	Specifies the number that defines the last UOW to be reported in the UOW report and the UOW Exception report.	“STOPUOW keyword” on page 546
UOW_REPORT	Optional	Specifies whether to generate the UOW report.	“UOW_REPORT keyword” on page 555

THRESHOLD subcommand keywords

The THRESHOLD subcommand specifies the threshold parameter values, which cause warning messages to be generated when the actual values do not meet the specified criteria. If you do not specify the REPORT subcommand, this subcommand is ignored.

For the THRESHOLD subcommand, you can specify the keywords that are listed in the following table. For the format of the THRESHOLD subcommand, see [“THRESHOLD subcommand” on page 435](#).

Table 14. THRESHOLD subcommand keywords

Keyword	Required or optional	Description	Topic
DOVFFS_PCT	Optional	Specifies a threshold setting for the percentage of free space in DOVF. If the percentage of free space in DOVF is less than the value that you specified, a warning message is issued.	“DOVFFS_PCT keyword” on page 455
FS_DOVFIOVF	Optional	Specifies a threshold setting for the free space in DOVF and in IOVF. If the free space in DOVF is greater than the value that you specified, and if the free space in IOVF is less than the value that you specified, a warning message is issued.	“FS_DOVFIOVF keyword” on page 466
FS_RAADOVF	Optional	Specifies a threshold setting for the free space in RAA BASE and in DOVF. If the free space in RAA BASE is greater than the value that you specified, and if the free space in DOVF is less than the value that you specified, a warning message is issued.	“FS_RAADOVF keyword” on page 467
FS_RAADOVFIOVF	Optional	Specifies a threshold setting for the free space in RAA BASE, DOVF, and IOVF. If the free space in RAA BASE is greater than the value that you specified, and if the free space in either DOVF or IOVF is less than the value that you specified, a warning message is issued.	“FS_RAADOVFIOVF keyword” on page 467
FS_RAAIOVF	Optional	Specifies a threshold setting for the free space in RAA BASE and in IOVF. If the free space in RAA BASE is greater than the value that you specified, and if the free space in IOVF is less than the value that you specified, a warning message is issued.	“FS_RAAIOVF keyword” on page 468
IOVFFS_PCT	Optional	Specifies a threshold setting for the percentage of free space in IOVF. If the percentage is less than value that you specified, a warning message is issued.	“IOVFFS_PCT keyword” on page 485
IOVFUSED_PCT	Optional	Specifies a threshold setting for the percentage of IOVF CIs used. If the percentage is greater than the value that you specified, a warning message is issued.	“IOVFUSED_PCT keyword” on page 485

Table 14. THRESHOLD subcommand keywords (continued)

Keyword	Required or optional	Description	Topic
RAAFS_PCT	Optional	Specifies a threshold setting for the percentage of free space in RAA BASE. If the percentage is less than the value that you specified, a warning message is issued.	“RAAFS_PCT keyword” on page 507
RAPOVFL_PCT	Optional	Specifies a threshold setting for the percentage of RAA BASE CIs that use either DOVF or IOVF. If the percentage is greater than the value that you specified, a warning message is issued.	“RAPOVFL_PCT keyword” on page 508
RECIO_AVG	Optional	Specifies a threshold setting for the average number of I/Os that can be done to access a database record. If the number is greater than the value that you specified, a warning message is issued.	“RECIO_AVG keyword” on page 510
RECIO_MAX	Optional	Specifies a threshold setting for the maximum number of I/Os that can be done to access a database record. If the number is greater than the value that you specified, a warning message is issued.	“RECIO_MAX keyword” on page 511
RECIOVF_PCT	Optional	Specifies a threshold setting for the percentage of records using IOVF CIs. If the percentage is greater than the value that you specified, a warning message is issued.	“RECIOVF_PCT keyword” on page 510
ROOTIO_AVG	Optional	Specifies a threshold setting for the average number of I/Os to access a root segment. If the number is greater than the value that you specified, a warning message is issued.	“ROOTIO_AVG keyword” on page 522
ROOTIO_MAX	Optional	Specifies a threshold setting for the maximum number of I/Os that can be done to access a root segment. If the number is greater than the value that you specified, a warning message is issued.	“ROOTIO_MAX keyword” on page 523
SYN_LEN_AVG	Optional	Specifies a threshold setting for the average synonym chain length. If the length is greater than the value that you specified, a warning message is issued.	“SYN_LEN_AVG keyword” on page 547
SYN_MAX	Optional	Specifies a threshold setting for the maximum synonym chain length. If the length is greater than the value that you specified, a warning message is issued.	“SYN_MAX keyword” on page 548
SYN_PCT_LEN	Optional	Specifies a threshold setting for the percentage of RAPs that have a long synonym chain among all the RAPs that have synonyms. If the percentage of the RAPs that have a long synonym chain is greater than the threshold value, a warning message is issued.	“SYN_PCT_LEN keyword” on page 548
UOWDOVF_PCT	Optional	Specifies a threshold setting for the percentage of UOWs that use DOVF. If the percentage is greater than the value that you specified, a warning message is issued.	“UOWDOVF_PCT keyword” on page 553
UOWIOVF_AVG	Optional	Specifies a threshold setting for the average number of IOVF CIs that are used by a UOW. If the number is greater than the value that you specified, a warning message is issued.	“UOWIOVF_AVG keyword” on page 554
UOWIOVF_MAX	Optional	Specifies a threshold setting for the maximum number of IOVF CIs that are used by a UOW. If the number is greater than the value that you specified, a warning message is issued.	“UOWIOVF_MAX keyword” on page 554
UOWIOVF_PCT	Optional	Specifies a threshold setting for the percentage of UOWs that use IOVF. If the percentage is greater than the value that you specified, a warning message is issued.	“UOWIOVF_PCT keyword” on page 555

Examples for the Reload function

Review examples and learn techniques for reloading DEDB areas.

In these examples, the following DEDB is assumed:

```

DEDB name:  DEDBJN22 has four areas
  area name:  DB22AR0 (area number is 1 and has three ADSs)
    ADS DD name: DB22AR01,      DS name:  HPFP.DB22AR0.ADS1
    ADS DD name: DB22AR02,      DS name:  HPFP.DB22AR0.ADS2
    ADS DD name: DB22AR03,      DS name:  HPFP.DB22AR0.ADS3
  area name:  DB22AR1 (area number is 2 and has two ADSs)
    ADS DD name: DB22AR11,      DS name:  HPFP.DB22AR1.ADS1
    ADS DD name: DB22AR12,      DS name:  HPFP.DB22AR1.ADS2
  area name:  DB22AR2 (area number is 3 and has two ADSs)
    ADS DD name: DB22AR21,      DS name:  HPFP.DB22AR2.ADS1
    ADS DD name: DB22AR22,      DS name:  HPFP.DB22AR2.ADS2
  area name:  DB22AR3 (area number is 4 and has one ADS)
    ADS DD name: DB22AR31,      DS name:  HPFP.DB22AR3.ADS1

```

In DBRC, all ADSs are in the UNAVAILABLE status.

You can find additional JCL examples that help you to monitor and tune DEDB areas in [“FPA Reload JCL examples for monitoring and tuning DEDBs”](#) on page 1135.

Related tasks

Scenario: [Reloading all areas of a DEDB by using the checkpoint restart option](#)

To reload all areas of a DEDB by using the checkpoint restart option, modify your reload JCL accordingly. Then, run the job and check the reports. If processing fails for one or more areas, correct the error conditions and repeat the steps until all areas are processed.

Example 1: Reloading an area that is registered with DBRC

The following figure shows example JCL for reloading an area that is registered with DBRC.

This example shows the case of a Reload by using the unloaded segment records data set that is produced by [“Example 1: Unloading an area that is registered with DBRC”](#) on page 74.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input unloaded segment records data set for the area is HPFP.USRFILE.DB22AR0, which is specified by the IDSNMASK='HPFP.USRFILE.&AREA' parameter. This data set is allocated dynamically.
- The output ADSs to be reloaded are dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC. The output ADSs must be defined before running this example.
- After reloading, the RECOV NEEDED status of area DB22AR0 is changed to OFF, and all three ADSs of the area are made AVAILABLE with DBRC.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  IDSNMASK='HPFP.USRFILE.&AREA',
  OAREA=(DB22AR0)
/*

```

Figure 43. Reloading an area that is registered with DBRC

Example 2: Reloading all areas of a DEDB that are registered with DBRC

The following figure shows example JCL for reloading all areas of a DEDB that are registered with DBRC.

This example shows the case of a Reload by using the unloaded segment records data set that is produced by the following examples:

- “[Example 2: Unloading all areas of a DEDB that is registered with DBRC](#)” on page 74
- “[Example 6: Unloading all areas of a DEDB and changing some or all of the DEDB area definitions \(IAREA=ALL, OAREA=ALL\)](#)” on page 77
- “[Example 7: Unloading all areas of a DEDB and changing some or all of the DEDB area definitions \(IAREA=ALL, OAREA=specific\)](#)” on page 78

In this example:

- The input unloaded segment records data set for each area is HPFP.USRFILE.*areaname*, which is specified by the IDSNMASK='HPFP.USRFILE.&AREA' parameter. This data set is allocated dynamically.
- The output ADSs to be reloaded are dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC. The output ADSs must be defined before running this example.
- After reloading, the RECOV NEEDED status of the areas DB22AR0, DB22AR1, DB22AR2, and DB22AR3 are changed to OFF, and all ADSs of these areas are made AVAILABLE with DBRC.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//         DD DISP=SHR,DSN=IMSVS.SDFSRESL
//         DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//RECON1  DD DISP=SHR,DSN=IMSVS.RECON1
//RECON2  DD DISP=SHR,DSN=IMSVS.RECON2
//RECON3  DD DISP=SHR,DSN=IMSVS.RECON3
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
RELOAD
  DBD=DEDBJN22,
  IAREA=ALL,
  IDSNMASK='HPFP.USRFILE.&AREA',
  OAREA=ALL
/*
```

Figure 44. Reloading all areas of a DEDB that are registered with DBRC

Example 3: Reloading multiple areas of a DEDB from one input unloaded segment record data set

The following figure shows example JCL for reloading multiple areas of a DEDB from one input unloaded segment record data set.

This example shows a case of Reload by using the unloaded segment records data sets that are produced by “[Example 3: Unloading multiple areas of a DEDB into one output unloaded segment record data set](#)” on page 75.

In this example:

- The DBRC=NO option specifies not to check the area status in DBRC, so RECON data sets are not needed.
- The input unloaded segment records data set for area DB22AR0 and DB22AR1 is HPFP.USRFILE1 that is specified by the IAREA001 DD statement.
- The input unloaded segment records data set for area DB22AR2 and DB22AR3 is HPFP.USRFILE2 that is specified by the IAREA002 DD statement.

- Because the two input unloaded segment records data sets are already sorted during the unloading process, it is not necessary to specify the SORT=YES parameter. If these data sets were not sorted during the Unload process, SORT=YES must be specified.
- The output ADSs to be reloaded for specified areas are dynamically allocated by using the ODSNMASK parameter and the ALLOCATE subcommand.
- The output ADSs to be reloaded for specified areas are encrypted with a key label associated with the area data sets. The key label is specified on the KEYLABEL keyword of the ALLOCATE subcommand.
- The following ADSs are allocated:

Area	Allocated ADS
DB22AR0	HPFP.DB22AR0.ADS1
DB22AR1	HPFP.DB22AR1.ADS1
DB22AR2	HPFP.DB22AR2.ADS1
DB22AR3	HPFP.DB22AR3.ADS1

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//         DD DISP=SHR,DSN=IMSVS.SDFSRESL
//         DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IAREA001 DD DISP=SHR,DSN=HPFP.USRFIL1
//IAREA002 DD DISP=SHR,DSN=HPFP.USRFIL2
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=NO
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  OAREA=(DB22AR0,DB22AR1,DB22AR2,DB22AR3),
  ODSNMASK='HPFP.&AREA.ADS1'
ALLOCATE
  OAREA=ALL,
  SPACE=(CYL,5),
  VOLSER=HFPVOL,
  DATACLAS=ataclas,
  KEYLABEL=keylabel,
  ATTRIBUTE=(DELETE)
/*
```

Figure 45. Reloading multiple areas of a DEDB from one input unloaded segment record data set

Example 4: Reloading an area that is registered with DBRC as MADS area

The following figure shows example JCL to reload an area that is registered with DBRC as MADS area.

This example shows the case of a Reload by using the unloaded segment records data set that is produced by “[Example 4: Unloading an area from the specified ADS of MADS area](#)” on page 76.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSVS.MDALIB data set. Because IMSVS.MDALIB is an APF registered data set, it is concatenated in STEPLIB.
- The input unloaded segment records data sets for the area is dynamically allocated by the IDSNMASK='HPFP.USRFIL1' parameter. The data set must be cataloged.
- The output ADSs to be reloaded for the specified areas are dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC. The output ADSs must be defined before running this example.
- After reload is done, the RECOV NEEDED status of the area DB22AR0 is changed to OFF. ADS DB22AR01, DB22AR02, and DB22AR03 of the area are made AVAILABLE with DBRC.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//          DD DISP=SHR,DSN=IMSVS.MDALIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  IDSNMASK='HPFP.USRFILE',
  OAREA=(DB22AR0)
/*

```

Figure 46. Reloading an area that is registered with DBRC as MADS area

Example 5: Reloading an area of which some of the parameters for the DEDB area definition are changed

The following figure shows example JCL for reloading an area of which some of DEDB area definitions are changed.

This example is a case of Reload by using the unloaded segment records data set that is produced by “Example 5: Unloading an area and changing the DEDB area definition” on page 77.

In this example:

- The ACB library that has new DEDB definition and specified in NEWACB DD statement during unloading must be specified in IMSACB DD statement.
- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input unloaded segment records data set for the area is HPFP.USRFILE.DB22AR0, which is specified by the IDSNMASK='HPFP.USRFILE.&AREA' parameter. This data set is allocated dynamically.
- The output ADS DD statement names DB22AR01 and DB22AR02 that are to be reloaded for the area are specified in JCL DD statements.
- After reloading, the RECOV NEEDED status of area DB22AR0 is changed to OFF. Two ADSs of the area, DB22AR01 and DB22AR02, are made AVAILABLE, but ADS DB22AR03 is kept UNAVAILABLE with DBRC.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB.NEW
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//DB22AR01 DD DISP=SHR,DSN=HPFP.DB22AR0.ADS1
//DB22AR02 DD DISP=SHR,DSN=HPFP.DB22AR0.ADS2
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  IDSNMASK='HPFP.USRFILE.&AREA',
  OAREA=(DB22AR0)
/*

```

Figure 47. Reloading an area of which some of the parameters for the DEDB area definition are changed

Example 6: Reloading and analyzing an area that is registered with DBRC

The following figure shows example JCL for reloading and analyzing an area that is registered with DBRC.

This example shows the case of a Reload by using the unloaded segment records data set that is produced by [“Example 1: Unloading an area that is registered with DBRC”](#) on page 74.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input unloaded segment records data set for the area is HPPF.USRFILE.DB22AR0, which is specified by the IDSNMASK='HPPF.USRFILE.&AREA' parameter. This data set is allocated dynamically.
- The output ADSs to be unloaded are dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC. The output ADSs must be defined before running this example.
- After reloading, the RECOV NEEDED status of area DB22AR0 is changed to OFF, and all three ADSs of the area are made AVAILABLE with DBRC.
- After reloading, the reloaded area DB22AR0 is analyzed with the quick check option.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPPF.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//        DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  IDSNMASK='HPPF.USRFILE.&AREA',
  OAREA=(DB22AR0),
  PTRCHKLVL=QUICK
/*
```

Figure 48. Reloading and analyzing an area that is registered with DBRC

Example 7: Reloading an area that is registered with DBRC by using the LOADCTL subcommand

The following figure shows example JCL for reloading an area that is registered with DBRC and the LOADCTL subcommand.

This example shows the case of a Reload by using the unloaded segment records data set that is produced by [“Example 1: Unloading an area that is registered with DBRC”](#) on page 74.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input unloaded segment records data set for the area is HPPF.USRFILE.DB22AR0, which is specified by the IDSNMASK='HPPF.USRFILE.&AREA' parameter. This data set is allocated dynamically.
- The output ADSs to be unloaded are dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC. The output ADSs must be defined before running this example.
- After reloading, the RECOV NEEDED status of area DB22AR0 is changed to OFF, and all three ADSs of the area are made AVAILABLE with DBRC.
- By the specification of the first LOADCTL subcommand, up to 10 segments of segment name DD1 are inserted ordinarily, and segments after the 11th are inserted into IOVF.

- By the specification of the second LOADCTL subcommand, up to five segments of segment name *DD4* are inserted ordinarily, and segments over six, *DD4*, and its dependent segments are inserted into DOVF.
- By the specification of the third LOADCTL subcommand, all dependent segments of segment name *DD5* are inserted into IOVF.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//         DD DISP=SHR,DSN=IMSVS.SDFSRESL
//         DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  IDSNMASK='HPFP.USRFILE.&AREA',
  OAREA=(DB22AR0)
  LOADCTL SEGMENT=(DD1,ONLY),LOCATION=IOVF,ILC=10
  LOADCTL SEGMENT=(DD4,BOTH),LOCATION=DOVF,ILC=5
  LOADCTL SEGMENT=(DD5,DEPENDENTS),LOCATION=IOVF,ILC=0
/*
```

Figure 49. Reloading an area that is registered with DBRC by using the LOADCTL subcommand

Example 8: Reloading all areas by using the concatenated unloaded segment records data sets

The following figure shows example JCL that is used for reloading all areas of a DEDB that are registered with DBRC.

This example shows the case of a Reload that uses the unloaded segment records data set that is produced by “[Example 8: Unloading all areas of a DEDB and changing some or all of the DEDB area definitions \(IAREA=specific, OAREA=ALL\)](#)” on page 79.

In this example:

- The input unloaded segment records data set for area DB22AR0 is a concatenation of HPFP.USRFILE.DB22AR0.RUN1 and HPFP.USRFILE.DB22AR0.RUN2, which are specified by the IAREA001 DD statement.
- The input unloaded segment records data set for area DB22AR1 is a concatenation of HPFP.USRFILE.DB22AR1.RUN1 and HPFP.USRFILE.DB22AR1.RUN2, which are specified by the IAREA002 DD statement.
- The input unloaded segment records data set for area DB22AR2 is a concatenation of HPFP.USRFILE.DB22AR2.RUN1 and HPFP.USRFILE.DB22AR2.RUN2, which are specified by the IAREA003 DD statement.
- The input unloaded segment records data set for area DB22AR3 is a concatenation of HPFP.USRFILE.DB22AR3.RUN1 and HPFP.USRFILE.DB22AR3.RUN2, which are specified by the IAREA004 DD statement.
- The output ADSs to be reloaded are dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC. The output ADSs must be defined before running this example.
- After reloading, the RECOV NEEDED status of the areas DB22AR0, DB22AR1, DB22AR2, and DB22AR3 are changed to OFF, and all ADSs of these areas are made AVAILABLE with DBRC.

Notes:

1. The SORT=YES option is needed to process the concatenated input unloaded segment records data set.

2. If your system resource is not enough to process all areas in one job step, you can reload each area per job step.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//        DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//NEWRESLB DD DISP=SHR,DSN=IMSVS.PGMLIB.NEW
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//IAREA001 DD DISP=SHR,DSN=HPFP.USRFILE.DB22AR0.RUN1
//        DD DISP=SHR,DSN=HPFP.USRFILE.DB22AR0.RUN2
//IAREA002 DD DISP=SHR,DSN=HPFP.USRFILE.DB22AR1.RUN1
//        DD DISP=SHR,DSN=HPFP.USRFILE.DB22AR1.RUN2
//IAREA003 DD DISP=SHR,DSN=HPFP.USRFILE.DB22AR2.RUN1
//        DD DISP=SHR,DSN=HPFP.USRFILE.DB22AR2.RUN2
//IAREA004 DD DISP=SHR,DSN=HPFP.USRFILE.DB22AR3.RUN1
//        DD DISP=SHR,DSN=HPFP.USRFILE.DB22AR3.RUN2
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
RELOAD
  DBD=DEDBJN22,
  SORT=YES,
  IAREA=ALL,
  OAREA=ALL
/*
```

Figure 50. Reloading all areas by using the concatenated unloaded segment records data sets

Example 9: Reloading areas by using the concatenated unloaded segment records data sets

The following figure shows example JCL that is used for reloading specific areas of a DEDB that are registered with DBRC.

This example shows the case of a Reload that uses the unloaded segment records data set that is produced by [“Example 9: Unloading all areas of a DEDB and changing some or all of the DEDB area definitions \(IAREA=specific, OAREA=specific\)”](#) on page 80.

In this example:

- The input unloaded segment records data set for area *area-n* is a concatenation of HPFP.USRFILE.*area-n*.RUN1, HPFP.USRFILE.*area-n*.RUN2, HPFP.USRFILE.*area-n*.RUN3, and HPFP.USRFILE.*area-n*.RUN4, which are specified by the IAREAnnn DD statement. *nnn* is the area number of the *area-n*.
- The output ADSs that are to be reloaded are dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC. The output ADSs must be defined before running this example.
- After reloading, the RECOV NEEDED status of the area *area-n* is changed to OFF, and all ADSs of the area are made AVAILABLE with DBRC.
- You need to run the job step four times by changing *area-n* to DB22AR0, DB22AR1, DB22AR3, and DB22AR4.

Note: The SORT=YES option is needed to process the concatenated input unloaded segment records data set.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//         DD DISP=SHR,DSN=IMSVS.SDFSRESL
//         DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//NEWRESLB DD DISP=SHR,DSN=IMSVS.PGMLIB.NEW
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//IAREAnnn DD DISP=SHR,DSN=HPFP.USRFILE.area-n.RUN1
//         DD DISP=SHR,DSN=HPFP.USRFILE.area-n.RUN2
//         DD DISP=SHR,DSN=HPFP.USRFILE.area-n.RUN3
//         DD DISP=SHR,DSN=HPFP.USRFILE.area-n.RUN4
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
RELOAD
  DBD=DEDBJN22,
  SORT=YES,
  IAREA=area-n,
  OAREA=area-n
/*

```

Figure 51. Reloading areas by using the concatenated unloaded segment records data sets

Example 10: Reloading an area that is registered with DBRC and creating its image copy

The following figure shows example JCL that is used to reload an area that is registered with DBRC and to create its image copy.

This example shows the case of a Reload by using the unloaded segment records data set that is produced by [“Example 1: Unloading an area that is registered with DBRC” on page 74](#).

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input unloaded segment records data set for the area is HPFP.USRFILE.DB22AR0, which is specified by the IDSNMASK='HPFP.USRFILE.&AREA' parameter. This data set is allocated dynamically.
- The output ADSs to be reloaded are dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC. The output ADSs must be defined before running this example.
- After reloading, the RECOV NEEDED status of area DB22AR0 is changed to OFF, and all three ADSs of the area are made AVAILABLE with DBRC.
- After reloading, the image copy data set for the area is dynamically allocated by IMS HP Image Copy. The data set name of the image copy is determined by IMS HP Image Copy as follows:

```
ICOUT.HFP.IC1.DEDBJN22.DB22AR0
```

For more information, see the *IMS High Performance Image Copy User's Guide*.

- After creating image copy, a NOTIFY.IC command is issued to DBRC.

```

//HFP EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=HPS.SHPSLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  IDSNMASK='HPFP.USRFILE.&AREA',
  OAREA=(DB22AR0),
  ICOUT=YES,
  ICNOTIFY=YES
/*
//ICEIN DD *
GLOBAL ICHLQ=ICOUT.HFP,
        UNIT=SYSDA,
        SPACE=(CYL,100,100)
/*
//ICERPRT DD SYSOUT=*

```

Figure 52. Reloading an area that is registered with DBRC and creating its image copy

Example 11: Reloading an area from a user-specified format data set

The following figure shows example JCL for reloading an area from a user-specified format data set.

This example shows the case of reload by using the unloaded segment records data set that is created by the JCL shown in “Example 11: Unloading areas in user-specified format” on page 82.

In this example:

- The content of the segment prefix is defined by the SEGREC_PREFIX keyword.
- Segment code is included in the segment prefix.

```

//HFP EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//          DD DISP=SHR,DSN=IMSVS.MDALIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//DB22AR01 DD DISP=SHR,DSN=HPFP.DB22AR0.ADS1
//IAREA001 DD DISP=SHR,DSN=HPFP.USRFILE
//HFPPRINT DD SYSOUT=*
//HFPRPTS DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  OAREA=(DB22AR0),
  FORMAT=USER,
  SEGREC_PREFIX=(SEGMENT_CODE)
/*

```

Figure 53. Reloading an area from a user-specified format data set

Example 12: Reloading and analyzing an area and storing the sensor data in the IMS Tools KB repository

The following figure shows example JCL for reloading an area that is registered with DBRC, analyzing that area, and storing the sensor data in the Sensor Data repository of IMS Tools KB.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input unloaded segment records data set for the area is HPFP.USRFILE.DB22AR0, which is specified by the IDSNMASK='HPFP.USRFILE.&AREA' parameter. This data set is allocated dynamically.
- The output ADSs to be unloaded are dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC. The output ADSs must be defined before running this example.
- After the reloading process is complete, the RECOV NEEDED status of area DB22AR0 is changed to OFF, and all three ADSs of the area are made AVAILABLE with DBRC.
- After the reloading process is complete, the reloaded area DB22AR0 is analyzed with the quick check option, and sensor data is collected and stored in the Sensor Data repository of IMS Tools KB.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//        DD DISP=SHR,DSN=IMSVS.PGMLIB
//        DD DISP=SHR,DSN=ITB.SHKTLOAD
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSRPT DD SYSOUT=*
//HFPSYSIN DD *
  GLOBAL
    DBRC=YES,
    ITKBSRVR=FPQSRV01,
    ADXCFGRP=ADSRV01
  RELOAD
    DBD=DEDBJN22,
    IAREA=(DB22AR0),
    IDSNMASK='HPFP.USRFILE.&AREA',
    OAREA=(DB22AR0),
    PTRCHKLVL=QUICK,
    SENSOR=YES
/*
```

Figure 54. Reloading and analyzing an area and storing the sensor data in the IMS Tools KB repository

Example 13: Reloading multiple areas of a DEDB from one input unloaded segment record data set in SMS environment

The following figure shows example JCL for reloading multiple areas of a DEDB from one input unloaded segment record data set in an SMS environment.

This example shows a case of Reload by using unloaded segment records data sets that are produced by “[Example 13: Unloading multiple areas of a DEDB into one output unloaded segment record data set in SMS environment](#)” on page 84.

In this example:

- The DBRC=NO option specifies not to check the area status in DBRC, so RECON data sets are not needed.
- The input unloaded segment records data set for all areas is HPFP.USRFILE, which is specified by the IAREA001 DD statement.
- Because the input unloaded segment records data set is already sorted during the unloading process, the SORT=YES option is not needed. If the data set was not sorted during the Unload process, SORT=YES must be specified.
- The output ADSs to be reloaded for all areas are dynamically allocated by using the ODSNMASK parameter and the ALLOCATE subcommand. The following ADSs are allocated:

Area	Allocated ADS
DB22AR0	HPFP.DB22AR0.ADS1

Area	Allocated ADS
DB22AR1	HPFP.DB22AR1.ADS1
DB22AR2	HPFP.DB22AR2.ADS1
DB22AR3	HPFP.DB22AR3.ADS1

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IAREA001 DD DISP=SHR,DSN=HPFP.USRFIL
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=NO
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  ODSNMASK='HPFP.&AREA.ADS1'
ALLOCATE
  OAREA=ALL,
  SPACE=(CYL,5),
  STORCLAS=HFPSTOR
/*
```

Figure 55. Reloading multiple areas of a DEDB from one input unloaded segment record data set in an SMS environment

Example 14: Reloading an area without activating the DBRC process

The following figure shows example JCL for reloading an area without activating the DBRC process.

This example shows the case of a Reload by using the unloaded segment records data set that is produced by [“Example 14: Unloading an area without activating the DBRC process”](#) on page 85.

In this example:

- The DBRC=IGNORE option specifies not to check the area status in DBRC, so the RECON data sets are not needed. Even if DBRC=FORCE is set for IMS, DBRC is not activated when the DBRC=IGNORE option is specified.
- Area authorization is not obtained from DBRC, so the integrity of the area is not protected.
- The input unloaded segment records data set for the area is specified on the IAREA001 DD statement.
- The output ADS to be reloaded for the specified area is HPFP.DB22AR0.ADS1.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//DB22AR0 DD DISP=SHR,DSN=HPFP.DB22AR0.ADS1
//IAREA001 DD DISP=OLD,DSN=HPFP.USRFIL.DB22AR0
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=IGNORE
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  OAREA=(DB22AR0)
/*
```

Figure 56. Reloading an area without activating the DBRC process

Example 15: Reloading an area of which some of the parameters for the DEDB area definition are changed when IMS management of ACBs is enabled

The following figure shows example JCL for reloading an area of which some of DEDB area definitions are changed when ACBs are managed by IMS.

This example is a case of Reload by using the unloaded segment records data set that is produced by “[Example 15: Unloading an area and changing the DEDB area definition when IMS management of ACBs is enabled](#)” on page 86.

In this example:

- The IMSACB DD statement is not specified. The ACB member is read from the IMS directory instead of the ACB library because IMSCATHLQ=IMSVS.CATALOG parameter is specified.
- To use IMS Tools Catalog Interface, the SGLXLOAD library of IMS Tools Base is included in the STEPLIB DD concatenation.
- The ACB member that defines the reloaded DEDB is read from the staging data set because IMSCATACB_INPUT=PENDING is specified.
- The input unloaded segment records data set for the area is HFPF.USRFILE, which is specified by the IDSNMASK='HFPF.USRFILE' parameter. This data set is allocated dynamically.
- The consistency between the DBD information in the staging data set and the DBD information in the input unloaded segment records data set is checked because ACBCHK=YES is specified.
- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS to be reloaded is dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC.
- After reloading, the RECOV NEEDED status of area DB22AR0 is changed to OFF, and all three ADSs of the area are made AVAILABLE with DBRC.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HFPF.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//        DD DISP=SHR,DSN=IMSVS.PGMLIB
//        DD DISP=SHR,DSN=ITB.SGLXLOAD
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFP RPTS DD SYSOUT=*
//HFP PRINT DD SYSOUT=*
//HFP SYSIN DD *
GLOBAL
  DBRC=YES,
  IMSCATHLQ=IMSVS.CATALOG,
  IMSCATACB_INPUT=PENDING
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  IDSNMASK='HFPF.USRFILE',
  OAREA=(DB22AR0),
  ACBCHK=YES
/*
```

Figure 57. Reloading an area of which some of the parameters for the DEDB area definition are changed when IMS management of ACBs is enabled

Chapter 6. Changing a DEDB offline with FPA

You can change a DEDB by using the FPA Change function.

Topics:

- [“Functions of Change” on page 129](#)
- [“Input and output for the Change function” on page 130](#)
- [“Running the Change function” on page 131](#)
- [“Region size requirements for the Change function” on page 133](#)
- [“DD statements for the Change function” on page 134](#)
- [“Utility control statements for the Change function” on page 149](#)
- [“Examples for the Change function” on page 156](#)

Functions of Change

The Change function of FPA efficiently reorganizes or restructures a single DEDB area. It can also concurrently reorganize or restructure multiple DEDB areas without impacting all areas of a database.

Specifically, the Change function reorganizes or restructures multiple DEDB areas in one step. If necessary, an analyze step can be included in the Change process. Additionally, the Change function has the following characteristics:

- It operates independently of the IMS control region. Hence, one or more areas of a multi-area database can be reorganized or restructured while the application continues to use the areas that are not included in the maintenance process. This feature can significantly increase application availability.

Note: This requires specific control of area selection and application code capable of handling 'FH' status codes.

- It can concurrently process multiple database areas with no database contention. This concurrent processing capability significantly decreases the time that is required for database maintenance, and it further increases application availability.
- During the Change process, you can change any or all of the following database specifications:
 - Number of database areas
 - Randomizing module
 - Segment edit/compression routine
 - UOW parameter values
 - ROOT parameter values
 - CI size
 - Size of the VSAM data set
 - Pointer options
- During the Change process, you can move segments in the hierarchical structure, and define new segments.

Restrictions:

- The names of the existing segments cannot be changed.
 - The existing segments cannot be deleted.
 - The maximum length of the existing segments cannot be reduced.
- An empty area unloaded by the Change process is clearly identified with an information message. Then during the Change process, an output data set for this empty area is initialized with no segment to

prevent troubles when IMS DB/DC gets access to the output data set. The Audit report also shows the empty DEDB area.

- During the Change process, dependent segment twin chains that are abnormally long can be controlled by specifying the load control information so that the impact on other database records in the same RAP CI is minimized. This feature is implemented by placing a user-specified limit on the number of segment occurrences that are to be placed closed to their root segments.
- There are no source code modifications made to any user-written or IMS program or control block. Because IMS program integrity is maintained, new exposures are not introduced.
- The Change process can reload segment data in one area into multi-area data sets (MADS).

Input and output for the Change function

The Change function is invoked by the CHANGE command in the HFPSYSIN DD control statement to restructure or reorganize multiple DEDB areas.

By using the Change function, DEDB areas can be reorganized in one step by a single command, as opposed to two steps (running both the UNLOAD and RELOAD commands). To reorganize areas of a DEDB, only the IMSACB library is required. If you want to change the definition of a DEDB (for example, to extend an area by increasing any of the sizes of UOW, IOVF, and CI), you also need to specify the OLDACB library that has the DMCB member of the old DEDB definition. The SORT program can be invoked internally to achieve better SORT performance. Invoking the SORT program internally eliminates the SORTIN and the SORTOUT file I/O operation. When the change process is completed, IMS HP Image Copy can be invoked internally to create the image copies.

If you enable the IMS catalog and the IMS management of ACBs, ACB definitions are retrieved from the IMS directory instead of IMSACB and OLDACB libraries.

The output area data set must be different from the input area data set, except when using the IRENAME subcommand. If you want to specify the same name as the input area data set name in the CHANGE command for the output area data set name, as in the Reorganize function, specify the IRENAME subcommand.

The following figure shows the input and the output for the Change function.

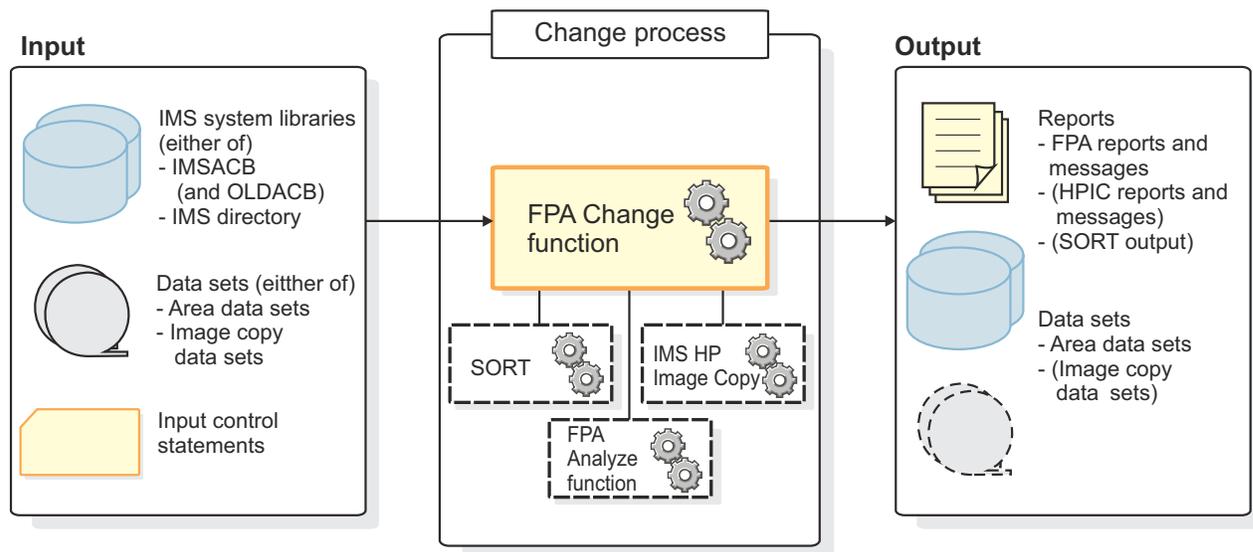


Figure 58. Input and output for the Change function

The Change function restructures or reorganizes multiple DEDB areas and analyzes the DEDB areas by using the CHANGE command in the HFPSYSIN DD control statement. The Analyze function can be invoked internally after the Change function. When the Analyze process is completed, IMS HP Image Copy can be invoked internally to create the image copies.

The change function can process the following image copies of a DEDB as input data sets to restructure or reorganize the DEDB:

- Standard image copies that are created by IMS Image Copy utility (DFSUDMP0) or IMS HP Image Copy
- Compressed image copies that are created by IMS HP Image Copy

When image copies are provided as input data sets for the change function, new image copies cannot be created after the completion of the Change process.

Running the Change function

FPA runs the Change function as a standard z/OS batch job. You must specify an EXEC statement and DD statements that define the input and output data sets in your JCL.

Procedure

1. Optional: Estimate the region size that is required to run the job.
See [“Region size requirements for the Change function”](#) on page 133.
2. Specify the EXEC statement in the following format:

```
//CHANGE EXEC PGM=HFPMAIN0,REGION=rrrrm,
//      PARM='DBRCGRP=dbrcgrp,GSGNAME=gsgname,IMSPLEX=imsplex,
//      IMSID=imsid'
```

Parameter	Description
DBRCGRP=	DBRCGRP is an optional 1- to 3-character parameter that specifies the identifier assigned to a group of DBRC instances (DBRC sharing group) that access the same RECON data set in the IMSplex.
GSGNAME=	GSGNAME is an optional 1- to 8-character parameter that specifies the global service group.
IMSPLEX=	IMSPLEX is an optional 1- to 5-character parameter that specifies the names of the IMSplex groups that share databases or message queues.
IMSID=	IMSID is an optional 1- to 8-character parameter that specifies the IMS system identification name. If you specify this parameter, FPA generates the buffers for temporary ESCD and SCD. When FPA issues a call to a randomizing module, register 11 contains the address of the temporary ESCD. FPA sets only the ESCDSCD field, which contains the address of SCD, in the temporary ESCD. Also, FPA sets only the SSCDIMID field, which contains the IMS system ID, in the temporary SCD. Other fields of temporary ESCD and temporary SCD cannot be referred to.

3. Specify the DD statements to define input data sets and output data sets.
4. Optional step. If you want to create an RDE during the FPA process, follow these steps:
 - a) Activate Integrity Checker.
To create a new RDE by using the Integrity Checker utility of IMS Library Integrity Utilities, follow the instructions in "Activating Integrity Checker" in the *IMS Library Integrity Utilities User's Guide*.
 - b) Request that FPA creates an RDE by using the Integrity Checker utility of IMS Library Integrity Utilities.
 - i) If you are to use a specific LIU global option module, specify the name of the LIU global option module in the LIUGOPT parameter of the FPA control statement (HFPSYSIN). You also need to specify DBRC=YES in the FPA JCL because Integrity Checker requires that DBRC be active.
 - ii) In the JOBLIB/STEPLIB DD statement, concatenate the following load module libraries.

- The IMS Library Integrity Utilities load module library
 - The LIU global option modules (LIU@INST, LIU@imsi, or both)
5. Run the JCL job.

Example

The following figure shows a JCL example for the CHANGE command.

```
//*****
//** CHANGE THE AREA(S)      **
//*****
//CHANGE EXEC PGM=HFPMAIN0
//*
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//        DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSRESLB DD DISP=SHR,DSN=IMSVS.SDFSRESL
//HFPPRINT DD SYSOUT=A
//HFPRPTS DD SYSOUT=A
//*
//HFPSYSIN DD *
//        GLOBAL DBRC=NO
//        CHANGE
//        DBD=IVPDB3
//*
//IAREA001 DD DISP=OLD,DSN=IMSVS.DFSIVD31
//IAREA002 DD DISP=OLD,DSN=IMSVS.DFSIVD33
//DFSIVD3A DD DISP=OLD,DSN=IMSVS.DFSIVD31.NEW
//DFSIVD3B DD DISP=OLD,DSN=IMSVS.DFSIVD33.NEW
//*
```

Figure 59. JCL example for the CHANGE command

Related concepts

[Region size requirements for the Change function](#)

The Change function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Related reference

[LIUGOPT keyword](#)

Use the LIUGOPT keyword to specify the LIU global option module that is to be used by IMS Library Integrity Utilities. The Integrity Checker utility of IMS Library Integrity Utilities requires that DBRC be active.

[Examples for the Change function](#)

Review examples and learn techniques for changing DEDB areas.

[Tips and best practices for the FPA Change function](#)

The FPA Change function supports several keywords that improve the performance of FPA Change jobs and that further simplify management of DEDBs.

Considerations on using the Integrity Checker utility

Some considerations apply when using the Integrity Checker utility of IMS Library Integrity Utilities.

- When you change a DBD and reorganize DEDB areas by using FPA, Integrity Checker creates new RDEs for the reorganized areas. If reorganization completes successfully, but Integrity Checker fails to create new RDEs, you must take one of the following actions:
 - Rerun the reorganization job to create new RDEs for the reorganized areas.
 - If you decide not to rerun the reorganization job and to use the original areas, you must delete the RDEs that were created during the reorganization by using the DELETE.DB command of the LICON utility before using the areas.

- If you decide not to rerun the reorganization job and to use the reorganized areas, you must create the RDEs whose creation failed during the reorganization by using the INIT.DB command of the LICON utility before using the areas.
- When any of the following error conditions were detected in FPA or Integrity Checker processing, the RDEs are not created:
 - Either the return code is equal to or greater than 8 or the processing ends abnormally for one or more areas in the Reload, Change, or Reorg processing.
 - The return code is equal to or greater than 8 for the Analyze function even when Reload, Change, or Reorg function completes successfully.
 - Integrity Checker fails to create an RDE.
- When the randomizer routine, edit/compression routine, or both are defined in the DBD library, checksum values of the routines are stored in the RDE. You must make sure that the libraries that contain the correct routines are specified in one of the following DD statements:
 - Reload and Change functions
 - IMSRESLB
 - When IMSRESLB is not specified, JOBLIB/STEPLIB
 - Reorg function
 - JOBLIB/STEPLIB
 - If the library that contains the Edit/Compression routine is not concatenated to JOBLIB/STEPLIB, FABLO212W message is issued by Integrity Checker.

Region size requirements for the Change function

The Change function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Region size for the Change function depends on the size of a UOW, the number of areas being changed, and the ITASKCTL and OTASKCTL specifications. Typically, database maintenance is a relatively important function. To increase the performance of the change program, it would be advisable to run it in a region that has a relatively high dispatching priority (that is, at least equal to the dispatching priority of a message processing region).

Estimating the region size for Change

The region size required for the Change function is a function of the size of a UOW, the number of areas being changed, and the ITASKCTL and OTASKCTL specifications.

In most cases, the specification on the EXEC statement for HFPMAIN0 of a region that is too small, is detected during the initialization processing of the main task. This causes processing to be terminated at that point. Under certain conditions, the region size shortfall does not become evident until the actual change has begun. The job must be completely rerun. Thus, the region size calculations must always be performed prior to initiating change processing.

In the restructure mode, sort program also requires region space. For how to estimate the required region size, see the *DFSORT Application Programming Guide*.

The following worksheet is for estimating the region size that is required for a typical change job. The actual region size depends on the keywords that are specified for the job. Therefore, when you specify the region size in your JCL, use a higher value than the value calculated from this worksheet.

FPA programs, internal tables, control blocks	3,020,000	
Number of areas defined in input DBD * 1,300	-----	
Number of areas defined in output DBD * 1,300	-----	(If restructure mode)
Internal tables, control blocks	21,000	
Maximum Segment length * 1,000	-----	
Maximum CISZ * 4	-----	
subtotal	-----	
Value specified on ITASKCTL * subtotal	-----	
Internal tables, control blocks	21,000	
Maximum CISZ * 6	-----	
Storage used by Sort program	-----	(If restructure mode)
subtotal	-----	
Value specified on OTASKCTL * subtotal	-----	
Total of input areas and output area * 3,700	-----	
Total of CISZ * UOW1 of each area * 4	-----	
Total bytes of change storage	-----	

Figure 60. Worksheet for estimating the region size for the Change function

If an analyze step is included in the change process, estimate the region size for the analyze step and use the greater region size. For how to estimate the region size for an analyze step, see [“Region size requirements for the Analyze function”](#) on page 200.

If ICACHE=YES is specified, the Change function preloads IOVF and SDEP parts into the data space. The data space size required for the Change function is a total of the size of IOVF and SDEP parts in each area. In addition, if the input data is an image copy data set, all image copy records are stored into the data space. In this case, the data space for the image copy data set must be added to the size that is calculated by the preceding worksheet.

If OCACHE=YES is specified, the Change function preloads IOVF and SDEP parts into the data space. The data space size required for the Change function is a total of the size of IOVF CIs that are used after reloading and the size of SDEP parts in each area.

The following figure shows a worksheet for estimating the data space size for the Change function.

Total size of IOVF parts in all areas	-----	(If ICACHE=YES)
Total size of SDEP parts in all areas	-----	(If ICACHE=YES and SDEP=LOGICAL or PHYSICAL)
Total of CISZ * UOW1 * (ROOT1 - ROOT2) in each area	-----	(If ICACHE=YES and input data is IC)
Total IOVF CIs used after reloading in each area	-----	(If OCACHE=YES)
Total size of SDEP parts in all areas	-----	(If OCACHE=YES and SDEP=LOGICAL or PHYSICAL)
Total bytes of a data space storage	-----	

Figure 61. Worksheet for estimating data space size for the Change function

DD statements for the Change function

DD statements of the FPA Change function determine the input and output data sets and specify how to run the Change function.

Subsections:

- [“List of DD statements”](#) on page 135

- “Input DD statements” on page 136
- “Output DD statements” on page 141
- “Input/Output DD statements” on page 147

List of DD statements

The following table summarizes the DD statements for the Change function.

Table 15. DD statements for the Change function

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
JOBLIB or STEPLIB	Input	PDS	No	Required
IMSACB, IMSACBA, or IMSACBB	Input	PDS	No	Required when IMSCATHLQ=*NO
IMSDALIB	Input	PDS	No	Optional
MODSTAT	Input	PS	Yes	Optional
MODSTAT2	Input	PS	Yes	Optional
OLCSTAT	Input	PS	Yes	Optional
EXITLIB	Input	PDS	No	Optional
IMSRESLB	Input	PDS	No	Optional
OLDACB	Input	PDS	Yes	Optional
OLDRESLB	Input	PDS	No	Optional
IAREAnn or IARnnnnn	Input	VSAM ESDS cluster	Yes	Required when IDSNMASK keyword is not specified
addnname or areaname	Output	VSAM ESDS cluster or PS (Note)	Yes	Required under certain conditions (for details, see addnname/areaname DD)
HFPSYSIN	Input	LRECL=80	No	Required
HFPAPARM	Input	LRECL=80	No	Optional
HFPAUOWC	Input	LRECL=80	No	Required when HFPAHST2 DD is specified
HFPCSITE	Input	PDS	Yes	Optional
HFPPRINT	Output	SYSOUT	Yes	Optional
HFPRPTS	Output	LRECL=133	Yes	Optional
HFSPRT	Output	LRECL=133	Yes	Optional
HFPAHST	Output	DISP=MOD	No	Optional
HFPAHST2	Output	DISP=MOD	No	Required when HFPAUOWC DD is specified
HFPARERG	Output	LRECL=80	No	Optional
HFPASNAP	Output	LRECL=133	No	Required when CISNAP=YES
HFPSEQER	Output	PS	No	Required when KEYSEQCHK=BYPASS or KEYSEQCHK=WARNING

Table 15. DD statements for the Change function (continued)

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
HFPTOTAL	Output	PS	No	Optional
HFPABEND	Output	LRECL=125	Yes	Optional
RECON n	Input and output	KSDS	Yes	Required when both of the following conditions are met: <ul style="list-style-type: none"> • DBRC=YES • DFSMDA members for RECONn data sets are not used
<i>ctttWKn</i>	Input and output		Yes	Optional
<i>itttWKn</i>	Input and output		Yes	Optional
<i>ctttSMMSG</i>	Output	LRECL=133	Yes	Optional
<i>itttSMMSG</i>	Output	LRECL=133	Yes	Optional
DFSPRINT	Output	LRECL=122	Yes	Optional
ICEIN	Input	LRECL=80	No	Optional
ICEPRINT	Output	LRECL=133	Yes	Optional
ICERPRT	Output	LRECL=133	No	Optional
<i>dataout1</i>	Output	PS	Yes	Optional
<i>dataout2</i>	Output	PS	Yes	Optional

Note: If image copy is used as an input data set.

Input DD statements

The following input DD statements are used for the Change function.

JOBLIB/STEPLIB DD

This DD statement is a required DD statement. This DD statement defines the input data sets as follows:

- The FPA load module library (required).
- The library that contains DFSMDA dynamic allocation members for the RECON data set. When DBRC=YES is specified and the RECON n DD statement is omitted, this DD statement is required.
- The library that contains a user exit routine member that was specified for the EXITRTN= keyword, if EXITLIB DD statement is not specified.
- The library that contains a default site default member (HFPCSITE) to determine the data set name and the member of the site default table.
- The IMS load module library (required).
- The IMS Tools Base library (SHKTLOAD), if you specify the ITKBSRVR keyword.
- The IMS Tools Base library (SGLXLOAD), if you specify the IMSCATHLQ keyword.
- The IMS HP Image Copy library, if you specify the ICOUT=YES option.
- The IMS Library Integrity Utilities library, if you want to activate Integrity Checker.

FPA must be run as an APF-authorized program. All of the data sets that are listed within the JOBLIB or the STEPLIB concatenation must also be APF-authorized. If you use DFSMDA members and the DFSMDA members are in a non-APF authorized library, you can specify the DFSMDA library on the IMSDALIB DD statement.

IMSACB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. The IMSACB DD statements (IMSACB, IMSACBA, IMSACBB, MODSTAT, MODSTAT2, and OLCSTAT) can be used to identify the active ACB library. When both OLCSTAT or MODSTAT and MODSTAT2 are found, FPA checks the time stamp and determines the latest one.

The active IMSACB library is determined according to the following rules:

1. Check IMSACB DD statement in the JCL statement.
2. Check OLCSTAT DD statement in the JCL statement.
3. Check MODSTAT and MODSTAT2 DD statement in the JCL statement.
4. Allocate OLCSTAT dynamically and check the OLCSTAT member.
5. Allocate MODSTAT and MODSTAT2 dynamically and check the MODSTAT and MODSTAT2 members.

If you specify the IMSCATHLQ keyword, IMSACB DD statements are not used. The IMS directory is used instead of the ACB library.

IMSACBA DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSACBB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSDALIB DD

This DD statement is an optional DD statement that defines the IMSDALIB data set. This DD statement is useful when the data set for dynamic allocation members is not APF-authorized.

MODSTAT DD

This DD statement is an optional DD statement that defines the MODSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

MODSTAT2 DD

This DD statement is an optional DD statement that defines the MODSTAT2 data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT2 library by using a TYPE=DATABASE statement of the DFSMDA macro.

OLCSTAT DD

This DD statement is an optional DD statement that defines the OLCSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the OLCSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

EXITLIB DD

This DD statement is an optional DD statement. This statement defines the user exit routine library. This DD statement is used when the IEXITRTN or OEXITRTN, or both keywords are specified for the CHANGE command. When the EXITLIB DD statement is specified in the JCL statement, user exit routines are loaded directly only from the EXITLIB DD statement. When the EXITLIB DD statement is not specified in the JCL statement, the routines are loaded from the JOBLIB or the STEPLIB DD statement.

All of the data sets that are listed within the EXITLIB concatenation must be APF-authorized.

IMSRESLB DD

This DD statement is an optional DD statement that defines the libraries that contain the randomizer routines and compression routines that are specified in the ACB member in the IMSACB DD statements. If the IMSRESLB DD statement is specified in the JCL statement, only the randomizer and compression routines are loaded directly from the IMSRESLB DD statement. If the IMSRESLB DD statement is not specified in the JCL statement, they are loaded from the JOBLIB or the STEPLIB DD statement.

The IMS version of IMSRESLB DD statement must be the same as those of IMSACB DD.

All the data sets that are listed within the IMSRESLB concatenation must be APF-authorized.

OLDACB DD

This DD statement is an optional DD statement. This statement defines the library that contains the original ACB library for the database. If this statement is specified, the CHANGE command is processed in the non-REORG mode (restructure). However, if the ACB member in the IMSACB DD library and the ACB member in the OLDACB DD library are the same, and the OLDRESLB DD statement is not specified in the JCL statement, the CHANGE command is processed in the REORG mode. If you want to reorganize the area data sets, do not specify this DD statement and do not allocate the data set dynamically.

If you specify the IMSCATHLQ keyword, OLDACB DD statement is not used. The IMS directory is used instead of the ACB library.

Note on dynamic allocation:

When this statement is not specified in the JCL statement, the data set is allocated dynamically by using a DFSMDA member. This DD library is allocated dynamically even if you do not specify in the JCL statement or a DFSMDA member. You can build the DFSMDA member for the OLDACB library by using a TYPE=DATABASE statement of the DFSMDA macro.

OLDRESLB DD

This DD statement is an optional DD statement. This statement is effective when the OLDACB DD statement is specified in JCL statement or the IMSCATACB_OUTPUT keyword is specified. This statement defines the libraries that contain the randomizer routines and compression routines that are specified in the ACB member pointed to from the OLDACB DD statements or the IMSCATACB_INPUT keyword.

If the OLDACB DD statement or the IMSCATACB_OUTPUT keyword is specified, the randomizer and compression routines are loaded from the OLDRESLB DD statement. If the OLDRESLB DD statement is not specified when the OLDACB DD statement or the IMSCATACB_OUTPUT keyword is specified, the randomizer and compression routines are loaded from the JOBLIB or the STEPLIB DD statement.

All of the data sets that are listed within the OLDRESLB concatenation must be APF-authorized.

IAREAnn/IARnnnn DD

This DD statement is a required DD statement when the IDSNMASK keyword is not specified. This statement defines the input area data set, which is the VSAM ESDS or an image copy data set for an area to be changed.

This data set can be allocated dynamically by using the IDSNMASK keyword parameter with the CHANGE command, or it can be specified with one of the following DD statements explicitly:

- //IAREAnnn DD JCL statement
- //IARnnnnn DD JCL statement

An IAREAnnn DD statement is for areas in the range of 1 - 999, and an IARnnnnn DD statement is for areas in the range of 1 - 2048. If the area number of the changed area is greater than 999, you must use the IARnnnnn DD statement. The value of nnn or nnnnn must be right-aligned, with leading zeros if necessary. The area number is the area number that is specified for the IAREA= parameter of the CHANGE command.

Note on dynamic allocation:

When this statement is not specified in the JCL statement, the data set is allocated dynamically by using the IDSNMASK keyword.

The following figure shows how FPA determines area data set names.

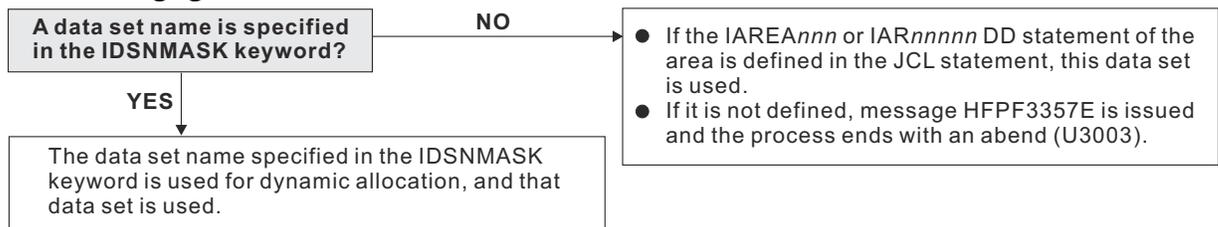


Figure 62. Input data set name for the CHANGE command

HFPSYSIN DD

This DD statement is a required DD statement that defines the input control statement that specifies the FPA functions.

The HFPSYSIN DD statement can be coded as a standard SYSIN file, a sequential data set, or a PDS member. LRECL=80 is required for the DCB of this data set. Only one command, besides the GLOBAL command, can be specified in HFPSYSIN. FPA analyzes the command syntax in columns 1 - 72. Columns 73 - 80 are regarded as comment.

For details about coding the HFPSYSIN DD statement, see [“Utility control statements for the Change function” on page 149.](#)

HFPAPARM DD

This DD statement is an optional DD statement. This statement defines the control parameters for generating the analysis reports.

When SENSOR=YES is specified, the values of RBASEFS and RDOVFFS keywords are stored in the Sensor Data repository as the values of data elements DB_THRESHOLD_RBASEFS and DB_THRESHOLD_RDOVFFS. These values are used by the Online Space Management utility as the values of RBASEFS and RDOVFFS keywords for the OER utility process (online reorganization process).

The HFPAPARM DD statement can be coded as a sequential data set or as a PDS member. LRECL=80 is required for the DCB of this data set. The HFPAPARM keywords must be written on one line. It can be coded as follows:

```
//HFPAPARM DD *
RDOVFFS=40 SYNTBL=20 UD0VFFS=35 RBASEFS=40
/*
```

You can specify the keywords that are listed in the following table.

Table 16. HFPAPARM keywords

Keyword	Description
ARDIGIT	This keyword specifies the digit of the area number field for the UOW group definition records that are defined in the HFPAUOWC file. The value is 3 or 5. The default is 3. If the HFPAUOWC file is not specified in the JCL statement, this keyword is ignored.
RBASEFS	This keyword is an optional keyword. If a RAP CI contains free space that is greater than the percentage specified, and the RAP CI also uses overflow CIs, a reorganization control statement is generated for the UOW, and the UOW is included in the UOW Exception report. The default is 100%, which means that reorganization control statements are never generated. The maximum value is 99, and the minimum value is 0.
RDOVFFS	This keyword is an optional keyword. If the DOVF part of a UOW contains free space that is greater than the percentage specified, and the UOW also uses IOVF CIs, a reorganization control statement is generated for the UOW and the UOW is included in the UOW Exception report. The default is 100%, which means that reorganization control statements are never generated. The maximum value is 99, and the minimum value is 0.
RGROUP	This keyword sets the group identifier for the statistics that are collected and stored in the History2 file or in a sensor data record set. For the History2 file, the keyword value is set in the Repository Group field, and for a sensor data record set, the value is set for the DB_SENSOR_DATA_GROUP_ID data element. This keyword has no default value. If this keyword is not specified, FPA sets the group identifier as follows: <ul style="list-style-type: none">• The Repository Group field in the History2 file contains 4 bytes of blank characters.• The DB_SENSOR_DATA_GROUP_ID data element is not included in the sensor data record set.
SYNTBL	This keyword is an optional keyword. It controls the number of reporting entries in the Synonym Chain Analysis table. The default value is 41 (that is, all synonym chains longer than 40 are grouped and reported together). The maximum value is 999, and the minimum value is 12.
UDOVFFS	This keyword is an optional keyword. If the DOVF UOW contains free space that is less than the percentage specified, the UOW is included in the UOW Exception report. The default is 0%, which means that the UOW is never included in the UOW Exception report. The maximum value is 99, and the minimum value is 0.

Note: The RBASEFS keyword and the RDOVFFS keyword that are used for the HFPAPARM data set are different from those used for the REORG command.

Requirement: Generating reorganization control statements for the High-Speed DEDB Direct Reorganization utility, you must specify the free space control percentages. You must also specify the HFPARERG DD statement in the JCL stream. For details, see the description of HFPARERG DD.

HFPAUOWC DD

This DD statement is a required DD statement when HFPAHST2 DD is specified. Each record in this data set specifies a range of UOW numbers of an area, which defines a UOW group of the area. The information about the UOW range is recorded in the HISTORY2 data set or in a sensor data record set in the order that is specified in the HFPAUOWC data set, even if the UOW ranges overlap or are in reverse order. You can specify the UOW ranges for each area in any order. However, you must specify the area numbers in ascending order.

The data set can reside in the JCL input stream. The data set can also be defined as a sequential data set or as a member of a partitioned data set. If a data set is specified, the data set must have the following attributes:

- DSORG=PS
- RECFM=FB
- LRECL=80
- BLKSIZE=*a multiple of 80*

If the data set is empty, all UOWs of the area are regarded as the target UOW range.

When you specify the HFPAHST2 DD statement, you must also specify the HFPAUOWC DD statement. If an HFPAUOWC DD statement is not specified, the HFPAHST2 DD statement is ignored.

When SENSOR_UOW=YES is specified, the data elements that will be stored differ by the HFPAUOWC DD specification as follows:

- If you specify the HFPAUOWC DD statement, data elements are collected at the UOW group level.
- If you do not specify the HFPAUOWC DD statement, data elements are collected at the UOW level.

For information about defining a UOW group, see [“UOW group definition record layout” on page 1191](#).

HFPCSITE DD

This DD statement is an optional DD statement. The HFPCSITE DD statement can be used to define the data set name of the site default table. When the SITE_DSNAME keyword is specified in the HFPSYSIN DD statement, the HFPCSITE DD statement is allocated dynamically. HFPCSITE DD in the JCL is mutually exclusive with the SITE_DSNAME keyword in the HFPSYSIN DD statement. The data set attribute of HFPCSITE DD is PDS.

ICEIN DD

This DD statement is an optional DD statement. This statement specifies the input data set that contains the control statements for IMS HP Image Copy. The data set can reside on a tape, a direct-access volume, or it can be routed through the input stream (DD * or DD DATA).

This data set is used only when you specify ICOUT=YES. If the DD statement is not coded and if ICOUT=YES is specified, the default control statements are used. For details about the ICEIN control statements, see *IMS High Performance Image Copy User's Guide*.

Output DD statements

The following output DD statements are used for the Change function.

***addnname/areaname* DD**

This DD statement is a required DD statement. This statement defines an output area data set, which is a VSAM ESDS that is generated by the Change process. The ddname used must be the ddname for the area that is defined in the output ACB library (IMSACB DD).

This data set can be allocated dynamically by one of following methods:

- The ODSNMASK keyword parameter of the CHANGE command
- When the GLOBAL DBRC=YES option is specified, the ADS information in the DBRC RECON data sets
- The area name member in the IMS DFSMDA library

Or it can be specified with one of the following DD statements explicitly:

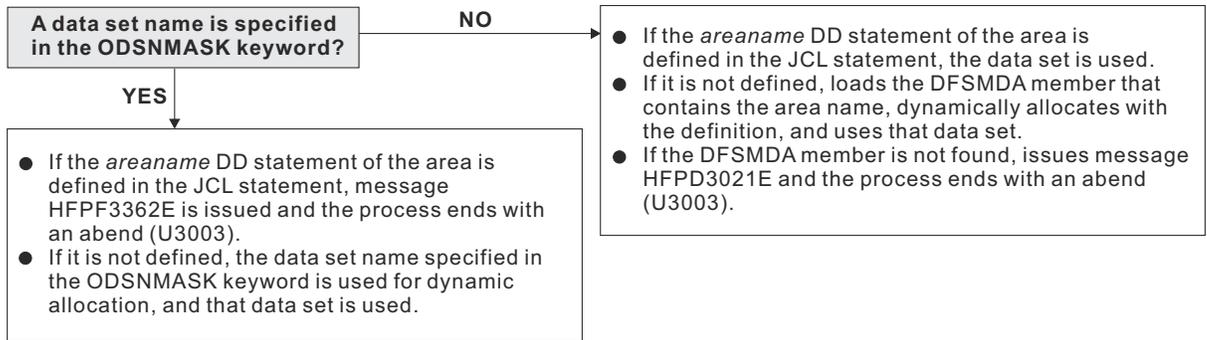
- *//addnname* DD JCL statement
- *//areaname* DD JCL statement

The area data set can be also defined by the ALLOCATE subcommand with the CHANGE command.

For the CHANGE command output, if the area is registered with DBRC, it must be in the RECOV NEEDED status; that is, all ADSs of the area must be in the UNAVAILABLE status in DBRC.

The following figure shows how FPA determines area data set names.

When DBRC=NO:



When DBRC=YES:

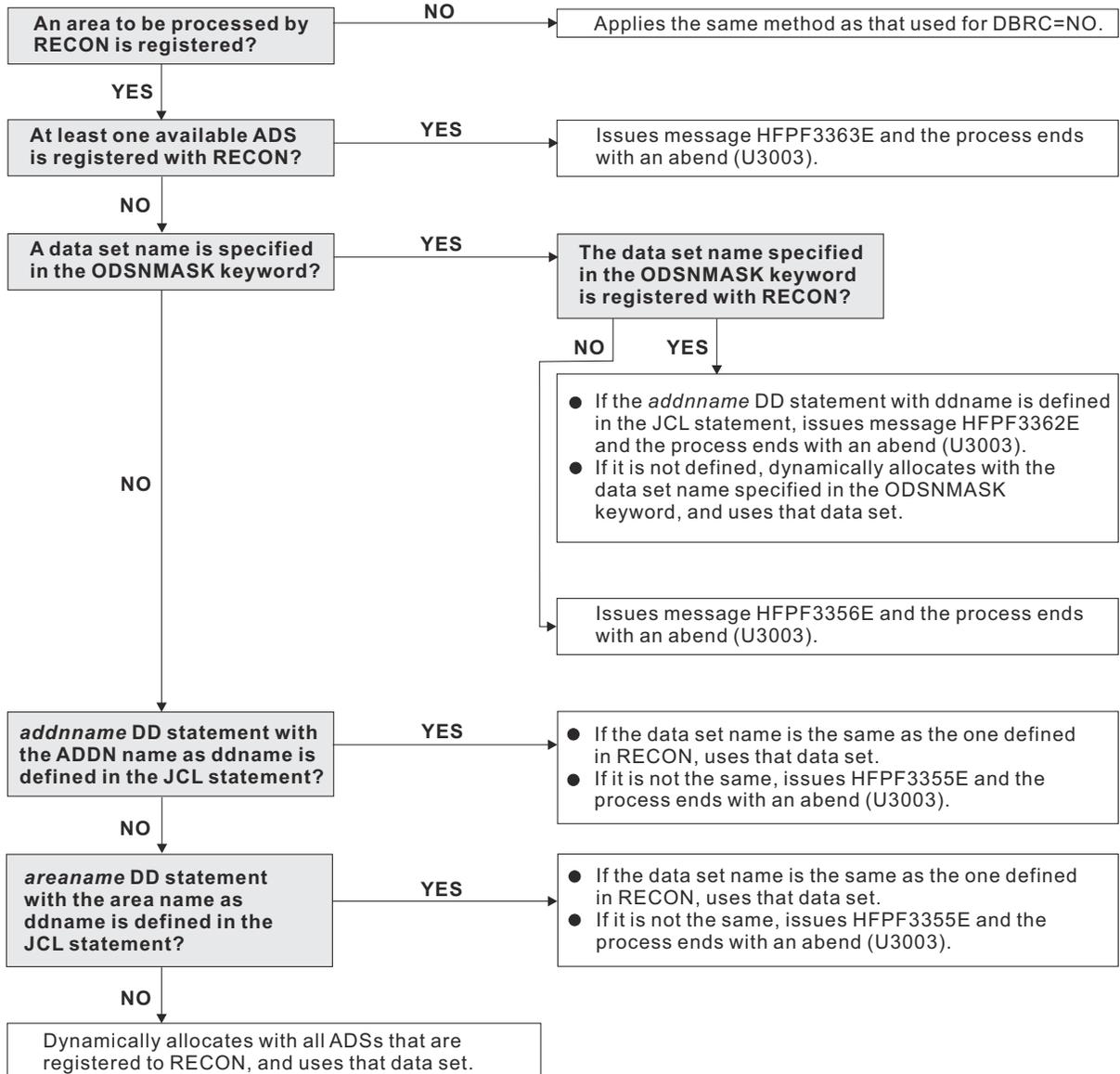


Figure 63. Output data set names for the CHANGE command

HFPPRINT DD

This DD statement is an optional DD statement that defines the processing log output data set. The data set contains the processing messages that are issued by FPA.

The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set. You cannot specify DUMMY.

The HFPPRINT data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPPRINT DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*.

The following figure shows an example of the processing messages that are issued for the CHANGE command.

```
IMS HPFP UTILITIES                                "Processing messages"                                PAGE: 1
5698-FPP V2R1                                     2019-11-22 13:03:37
HFPC0001I COMMAND ANALYZE PROCESSING STARTED.
HFPC0005I DBRC=NO IS SPECIFIED.
          - EEQE DETECTION IS NOT PERFORMED.
HFPD0001I OLDACB DYNAMICALLY ALLOCATED TO
          - DSNNAME: IMSVS.ACBLIB
HFPC0002I COMMAND ANALYZE PROCESSING COMPLETED.
HFPF0001I CHANGE PROCESSING STARTED FOR DATABASE: DEDBJN21
HFPJ0001I UNLOAD PROCESSING STARTED FOR AREA NO: 1, AREANAME: DB21AR0
HFPJ0002I UNLOAD PROCESSING COMPLETED FOR AREA NO: 1, AREANAME: DB21AR0
HFPK0001I RELOAD PROCESSING STARTED FOR AREA NO: 1, AREANAME: DB21AR0
HFPK0002I RELOAD PROCESSING COMPLETED FOR AREA NO: 1, AREANAME: DB21AR0
HFPF0002I CHANGE PROCESSING COMPLETED FOR DATABASE: DEDBJN21
HFPL0001I REPORT STARTED FOR COMMAND: CHANGE, DATABASE: DEDBJN21
HFPL0002I REPORT COMPLETED FOR COMMAND: CHANGE, DATABASE: DEDBJN21
```

Figure 64. Processing messages issued for the CHANGE command

HFPRPTS DD

This DD statement is an optional DD statement that defines the report output data set. The data set contains the reports that can be used to verify the process of the command. The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

The following reports are generated in this data set:

- Audit report
- Processing report
- DBD Definition report
- Unloaded report
- Reloaded report

When the REPORT subcommand and PTRCHKLVL=FULL are specified the following reports are generated:

- DEDB Area Analysis reports
- Process Summary of Analysis report
- Resolve Pointers report
- UOW report
- UOW Exception report
- Largest Database Records report (When LARGEREC_REPORT=YES is specified)

- Consolidated Largest Database Records report (When LARGEREC_REPORT=YES is specified and the second parameter of the LARGEREC_COUNT keyword specifies a value that is equal to or larger than 1)
- Segment Length Distribution report
- DB Record Length Distribution report
- RBA of Used IOVF CI report

When the REPORT subcommand and PTRCHKLVL=QUICK are specified, the following reports are generated:

- DEDB Area Analysis reports (FREESPACE ANALYSIS only)
- Process Summary of Analysis report
- UOW report (without DB record information)
- UOW Exception report (without DB record information)
- Pointer Checksum report
- RBA of Used IOVF CI report

You can generate reports related to the Analyze process in another DD data set that is specified by the REPORTDD=keyword parameter of the REPORT subcommand.

The HFPRPTS data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPRPTS DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

HFPSRT DD

This DD statement is an optional DD statement that defines the report output data set when SENSOR=YES is specified. This output data set contains the Sensor Data Statistics report that can be used to verify which sensor data is stored in the Sensor Data repository of IMS Tools KB. The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set. You can specify NULLFILE or DUMMY for this data set.

The HFPSRT data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPSRT DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

HFAHST DD

This DD statement is an optional DD statement. This statement defines an output data set that contains historical records. The analysis report generation process adds one record for each DEDB area that it processes. The record contains the key space utilization and performance information for that area. This statement is used only when you specify both the REPORT subcommand and the PTRCHKLVL=FULL keyword.

If this feature is used, the HFAHST data set must be preallocated with the following attributes:

- DSORG=PS

- RECFM=FB
- LRECL=100
- BLKSIZE=*user-specified*

The DD statement in the JOB stream is specified as:

```
//HFPAHST DD DSN=HPFP.HISTORY,DISP=(MOD,KEEP,KEEP)
```

You can generate the records in another DD data set specified by the HISTORYDD= keyword parameter instead of HFPAHST DD.

HFPAHST2 DD

This DD statement is a required DD statement when HFPAUOWC DD is specified. This statement defines an output data set that contains historical records. The analysis report generation process adds one record for each UOW group that is specified by HFPAUOWC DD. The record contains the key space utilization and performance information for that area, which is based on the information obtained from the specified UOW range. This statement is used only when you specify both the REPORT subcommand and the PTRCHKLVL=FULL keyword. If HFPAUOWC DD statement is not specified, this statement is ignored.

If this feature is used, the HFPAHST2 data set must be preallocated with the following attributes:

- DSORG=PS
- RECFM=FB
- LRECL=200
- BLKSIZE=*user-specified*

The DD statement in the JOB stream is specified as:

```
//HFPAHST2 DD DSN=HPFP.HISTORY2,DISP=(MOD,KEEP,KEEP)
```

HFPARERG DD

This DD statement is an optional DD statement. This statement defines an output data set that contains the reorganization control statements for the High-Speed DEDB Direct Reorganization utility (DBFUHDR0). This statement is used only when you specify both the REPORT subcommand and the PTRCHKLVL=FULL or the PTRCHKLVL=QUICK keyword.

The following DCB attributes of the file are set by the Analyze process:

- DSORG=PS
- RECFM=FB
- LRECL=80
- BLKSIZE=800

Do not code the DCB information in your JCL statement. Do not specify DISP=MOD for this DD statement.

Requirement:

Generating reorganization control statements for the High-Speed DEDB Direct Reorganization utility, you must specify the HFPARERG DD statement in the JCL stream. You must also specify the free space control percentages. For details about specifying the free space control percentages, see the description of HFPAPARM DD.

HFPA SNAP DD

This DD statement is a required DD statement when CISNAP=YES. This statement defines the output data set for the CI Map/CI Dump report and the Pointer Chain Reconstruction report. This DD statement is effective when the REPORT subcommand, PTRCHKLVL=FULL or PTRCHKLVL=QUICK, and CISNAP=YES are specified.

The following DCB attributes of the file are set by the Analyze process:

- RECFM=FBA

- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement.

The HFPASNAP data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPASNAP DD SYSOUT=A
```

HFPSEQER DD

This DD statement is a required DD statement when KEYSEQCHK=BYPASS or KEYSEQCHK=WARNING. This statement defines the output data set that contains the sequence error segment data records for one or more areas.

This statement defines an output data set for the sequence error segment data records that are produced for one or more of the areas defined in the "output" ACB library (NEWACB DD statement or IMSACB DD statement in Reorg mode).

The format of this record is specified in the HFPZSER macro, which is provided as a product-sensitive user interface. For details of the record format, see [“Sequence error segment records layout” on page 1183](#).

HFPTOTAL DD

This DD statement is an optional DD statement. This statement defines an output data set that contains the total number of reloaded segments in an area. When this DD statement is present, the Change process writes the total number of reloaded segments of each segment type in an area.

The following DCB attributes of the file are set by the FPA main task:

- DSORG=PS
- LRECL=60
- BLKSIZE=6000

Do not code the DCB information in your JCL statement.

The format of this record is specified in the HFPZTOT macro, which is provided as a product-sensitive user interface. For details of the record format, see [“Total segments records layout” on page 1183](#).

HFPABEND DD

This DD statement is an optional DD statement. It defines a snap dump output data set that is used for diagnostic purposes. This data set might be used when the FPA job ends abnormally. Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

ctttSMMSG DD

This DD statement is an optional DD statement. However, this DD statement is required when DFSORT is used. This statement defines an output data set for storing DFSORT messages. For the condition for running DFSORT and the naming rules for the DD names, see the description of *ctttWKnn* DD.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

itttSMMSG DD

This DD statement is an optional DD statement. However, this DD statement is required when DFSORT is used. This statement defines an output data set for storing DFSORT messages. For the condition for running DFSORT and the naming rules for the DD names, see the description of *itttWKnn* DD.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

DFSPRINT DD

This DD statement is an optional DD statement. This statement specifies the output data set that contains the messages that are produced by IMS HP Image Copy. The data set can reside on a tape, a direct-access volume, or a printer, or it can be routed through the output stream (SYSOUT).

This data set is used only when you specify ICOUT=YES. If the DD statement is not coded and ICOUT=YES is specified, a SYSOUT stream is dynamically allocated for the DD.

ICEPRINT DD

This DD statement is an optional DD statement. This statement specifies the output data set that contains the reports that are produced by IMS HP Image Copy. The data set can reside on a tape, a direct-access volume, or a printer, or it can be routed through the output stream (SYSOUT).

This data set is used only when you specify ICOUT=YES. If the DD statement is not coded and ICOUT=YES is specified, a SYSOUT stream is dynamically allocated for the DD.

ICERPRT DD

This DD statement is an optional DD statement. This statement specifies the output data set that contains the Image Copy Dump Process report, which is produced by IMS HP Image Copy. It can reside on a tape, a direct-access volume, or a printer, or it can be routed through the output stream (SYSOUT).

This data set is used only when you specify ICOUT=YES.

dataout1 DD

This DD statement is an optional DD statement. This statement defines the primary image copy data set. One DD statement is required for each data set that is to be dumped. The DD name can be any 1- to 8-character string. The data set specified by the DD statement must be on either a direct-access device or a tape device.

This data set is used only when you specify ICOUT=YES. For details of the format and the characteristics of the image copy data set, see *IMS High Performance Image Copy User's Guide*. You can request IMS HP Image Copy to allocate the image copy data set dynamically by omitting this DD statement and specifying the allocation parameters that are associated with the data set in the ICEIN data set.

dataout2 DD

This DD statement is an optional DD statement. This statement defines the secondary image copy data set. This statement is required only if the associated ICEIN control statement requests the secondary image copy. The data set specified by the DD statement must be on either a direct-access device or a tape device.

This data set is used only when you specify ICOUT=YES. For details of the format and the characteristics of the image copy data set, see *IMS High Performance Image Copy User's Guide*.

Input/Output DD statements

The following input/output DD statements are used for the Change function.

RECON1/RECON2/RECON3 DD

These statements define the DBRC RECON data sets. These DD statements are required DD statements when both of the following conditions are met:

- DBRC=YES is specified for the GLOBAL command
- DFSMDA members for RECON n data sets are not used

The RECON n data set must be the same data set that is used in the IMS control region. Because FPA issues a signon, a signoff, an authorization call, and an unauthorization call to DBRC, it updates RECON n data sets.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is necessary, the data set is allocated dynamically by using a DFSMDA member.

ctttWKnn DD

These DD statements are optional DD statements. These data sets are intermediate storage data sets that are used by DFSORT. For more information about coding the SORTWKnn DD statements, see *DFSORT Application Programming Guide*.

The adequate size of storage data sets is 1.5 to 2.5 times the input file size. In the Change function, the input file is the unloaded segment record file that is generated internally. The size of the file is computed by multiplying the average length of the unloaded segment record by the number of segments. Allocating twice the space used by the SORTIN data set is usually adequate for each work data set. This data set is allocated dynamically by OS. When the SORTIN is too large for sort, a B37 abend occurs. In this case, this DD is a required DD.

Do not specify DISP=MOD for these DD statements.

The naming rules of the ddnames are as follows:

c:

The thousands digit of *n* for *n*-th output thread subtask.

U:

If *n* is 999 or less, U must be specified.

V:

If *n* is 1000 - 1999, V must be specified.

W:

If *n* is 2000 - 2048, W must be specified.

Note: When a subtask ends processing an area, it starts processing the next area. For this reason, the *n*-th output thread subtask does not always process the *n*-th output data set.

ttt:

The hundreds, tenths, and units digits of *n*.

nn:

Specifies the number of the SORT work data sets. This value must be 01 - 03.

Note on dynamic allocation:

If this statement is not specified in the JCL statement when these data sets are necessary, FPA allocates them dynamically.

itttWKnn DD

These are optional DD statements. These are intermediate storage data sets that are used by DFSORT. For more information about coding the SORTWKnn DD statements, see *DFSORT Application Programming Guide*.

Do not specify DISP=MOD for these DD statements.

The adequate size of storage data sets is 1.5 to 2.5 times the input file size. In the analyze step, the size of each input file is as follows:

PTRCHKLVL=FULL

30 x the number of pointers in the area

SDEPCHKLVL=FULL

30 x the number of SDEP segments in the area

CISNAP=YES

112 x the number of pointers in which pointer errors are detected

REPORT subcommand and PTRCHKLVL=FULL are specified

X x (the number of non-SDEP segments in the areas + the number of free space elements in the areas + the number of references to other CIs)

Where *X* is:

- When LARGEREC_REPORT=NO is specified, *X* is calculated as follows:

34 + (the maximum segment level x 4)

However, if X is smaller than 48, 48 is used for X .

- When LARGEREC_REPORT=YES is specified, X is calculated as follows:

$$34 + (\text{the maximum segment level} \times 4) + \text{root key length}$$

However, if X is smaller than 48, 48 is used for X .

REPORT subcommand and PTRCHKLVL=QUICK are specified

$$23 \times (\text{the number of free space elements in the area} + \text{the number of references to other CIs})$$

Allocating twice the space used by the SORTIN data set is usually adequate for each work data set.

The naming rules of the ddnames are as follows:

i:

A, B, and C:

If PTRCHKLVL=FULL is specified.

D, E, and F:

If SDEPCHKLVL=FULL is specified.

G, H, and I:

If CISNAP=YES is specified.

J, K, and L:

If the REPORT subcommand is specified.

The thousands digit of the area number of the area to be analyzed, or the thousands digit of n for n -th input thread subtask.

A, D, G, and J:

If the number of areas to be analyzed or n is 999 or less.

B, E, H, and K:

If the number of areas to be analyzed or n is 1000 - 1999.

C, F, I, and L:

If the number of areas to be analyzed or n is 2000 - 2048.

Note: When a subtask ends processing an area, it starts processing the next area. For this reason, the n -th input thread subtask does not always process the n -th input data set.

ttt:

The hundreds, tenths, and units digits of the area number of the area to be analyzed, or those of n .

nn:

The number of the SORT work data sets. This value must be 01 - 03.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when these data sets are necessary, FPA allocates them dynamically.

Utility control statements for the Change function

The utility control statements for the Change function define the user description of the GLOBAL command, the CHANGE command, and the optional ALLOCATE, IRENAME, LOADCTL, REPORT, and THRESHOLD subcommands, all of which control the Change process.

The utility control statements must be specified in the HFPSYSIN data set. This control statement data set usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte, fixed-length records. Block size, if coded, must be a multiple of 80.

The following figure shows a sample of how to code the HFPSYSIN data set.

```
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
CHANGE
  DBD=dbdname
/*
```

Figure 65. CHANGE HFPSYSIN data set

Subsections:

- [“GLOBAL command keywords” on page 150](#)
- [“CHANGE command keywords” on page 151](#)
- [“ALLOCATE subcommand keywords” on page 153](#)
- [“IRENAME subcommand keywords” on page 153](#)
- [“LOADCTL subcommand keywords” on page 154](#)
- [“REPORT subcommand keywords” on page 154](#)
- [“THRESHOLD subcommand keywords” on page 155](#)

GLOBAL command keywords

For the GLOBAL command, you can specify the keywords that are listed in the following table. For the format of the GLOBAL command, see [“GLOBAL command” on page 394](#).

Table 17. GLOBAL command keywords

Keyword	Required or optional	Description	Topic
ADXCFCGRP	Optional	Specifies the name of the Autonomics Director XCF group.	“ADXCFCGRP keyword” on page 439
CISNAP	Optional	Requests the CI Map/CI Dump report and the Pointer Chain Reconstruction report.	“CISNAP keyword” on page 448
DBRC	Optional	Specifies to request or bypass the database recovery control (DBRC) facility process for the database that is specified by the DBD.	“DBRC keyword” on page 451
HISTORYDD	Optional	Specifies the DD statement for the output History file data set that is specified in the JCL.	“HISTORYDD keyword” on page 469
ICINCOMP	Optional	Specifies, when an image copy is used as an input data set, whether the image copy records are to be compressed by using the CSRCSRVS MVS macro when they are stored into a data space storage.	“ICINCOMP keyword” on page 474
ICNOTIFY	Optional	Specifies whether to issue NOTIFY.IC command to DBRC.	“ICNOTIFY keyword” on page 475
ICOUT	Optional	Specifies whether to generate one or more image copies.	“ICOUT keyword” on page 476
IMSCATACB_INPUT	Optional	Specifies whether to retrieve the currently active ACB definition or the pending ACB definition from the IMS directory.	“IMSCATACB_INPUT keyword” on page 480
IMSCATHLQ	Optional	Specifies the high-level qualifier of the bootstrap data set of the IMS directory.	“IMSCATHLQ keyword” on page 481
ITKBDATA	Optional	Specifies the type of additional data to store in the IMS Tools Base IMS Tools Knowledge Base repository.	“ITKBDATA keyword” on page 487
ITKBLOAD	Optional	Specifies the library name of IMS Tools KB.	“ITKBLOAD keyword” on page 487
ITKBSRVR	Optional	Specifies the name of the IMS Tools KB server XCF group.	“ITKBSRVR keyword” on page 488
LIUGOPT	Optional	Specifies the LIU global option module that is to be used by IMS Library Integrity Utilities.	“LIUGOPT keyword” on page 493

Table 17. GLOBAL command keywords (continued)

Keyword	Required or optional	Description	Topic
MSTIMESTAMP	Optional	Specifies whether to include a timestamp in the messages.	“MSTIMESTAMP keyword” on page 497
PTRCHKLVL	Optional	Specifies the method that is used to run integrity verification in the root addressable part and in the independent overflow part.	“PTRCHKLVL keyword” on page 506
SCAN	Optional	Specifies to verify the syntax of command, keywords, and parameters in command sets without running FPA. The Scan function also allocates some (but not all) internal resources for running the process function.	“SCAN keyword” on page 523
SDEPCHKLVL	Optional	Specifies the method that is used to run integrity verification in the sequential dependent part.	“SDEPCHKLVL keyword” on page 526
SDEP_ORPHAN_MSGLVL	Optional	Specifies the error message severity level to be set when an orphan SDEP segment is detected.	SDEP_ORPHAN_MSGLVL keyword
SITE_DSNAME	Optional	Specifies the data set name of the site default table.	“SITE_DSNAME keyword” on page 539
SITE_MEMBER	Optional	Specifies the member name of the site default table.	“SITE_MEMBER keyword” on page 540

CHANGE command keywords

For the CHANGE command, you can specify the keywords that are listed in the following table. For the format of the CHANGE command, see [“CHANGE command” on page 386](#).

Table 18. CHANGE command keywords

Keyword	Required or optional	Description	Topic
ATASKCTL	Optional	Specifies the maximum number of output areas to be analyzed concurrently.	“ATASKCTL keyword” on page 443
CISNAP	Optional	Specifies to request the CI Map/CI Dump report and the Pointer Chain Reconstruction report when the reloaded area is analyzed concurrently.	“CISNAP keyword” on page 448
DBD	Required	Specifies the database description (DBD) that contains the areas to be processed.	“DBD keyword” on page 450
DBREC_DISCARD_MSGLVL	Optional	Specifies the severity level of the message when a randomized database record is discarded due to a USR file not being specified.	DBREC_DISCARD_MSGLVL keyword
EDITCOMP_SKIP	Optional	Specifies whether to skip the segment edit/compression routine process.	“EDITCOMP_SKIP keyword” on page 459
HISTORYDD	Optional	Specifies the DD statement for the output History file data set that is specified in the JCL when the reloaded area is analyzed concurrently.	“HISTORYDD keyword” on page 469
IAREA	Optional	Specifies one or more input areas for the Unload process.	“IAREA keyword” on page 469
ICACHE	Optional	Specifies the type of control intervals that are to be preloaded into a data space storage for processing.	“ICACHE keyword” on page 472
ICHECK	Optional	Controls the concurrent number of input devices to access by using the ITASKCTL keyword.	“ICHECK keyword” on page 473
ICINCOMP	Optional	Specifies, when an image copy is used as an input data set, whether the image copy records are to be compressed by using the CSRCSRV MVS macro when they are stored into a data space storage.	“ICINCOMP keyword” on page 474
ICNOTIFY	Optional	Specifies whether to issue the NOTIFY.IC command to DBRC.	“ICNOTIFY keyword” on page 475

Table 18. CHANGE command keywords (continued)

Keyword	Required or optional	Description	Topic
ICOUT	Optional	Specifies whether to generate one or more image copies.	“ICOUT keyword” on page 476
IDSNMASK	Optional	Specifies the data set name or the mask for dynamically allocating cataloged pre-existing input data sets.	“IDSNMASK keyword” on page 476
IEXITRTN	Optional	Specifies the name of the user exit routine to be invoked after the Unload process of the area data set by the Change function.	“IEXITRTN keyword” on page 479
IMSCATACB_OUTPUT	Optional	Specifies whether to retrieve the currently active ACB definition or the pending ACB definition from the IMS directory. This keyword specifies the new ACB definition, which is used when an ACB definition change is requested.	“IMSCATACB_OUTPUT keyword” on page 481
ITASKCTL	Optional	Specifies the maximum number of input areas to be processed concurrently.	“ITASKCTL keyword” on page 486
KEYCHGCHK	Optional	Specifies whether to check if the key sequence field was not changed by a user exit routine.	“KEYCHGCHK keyword” on page 489
KEYSEQCHK	Optional	Specifies what to do if a key sequence error is detected.	“KEYSEQCHK keyword” on page 490
MAXERROR	Optional	Specifies the maximum number of pointer errors that might be encountered before processing ends with an error.	“MAXERROR keyword” on page 495
OAREA	Optional	Specifies one or more output areas for the process.	“OAREA keyword” on page 497
OCACHE	Optional	Specifies the types of control intervals that are held in a data space storage until all processing has completed.	“OCACHE keyword” on page 499
ODSNMASK	Optional	Specifies the data set name or mask for dynamically allocating cataloged pre-existing output data sets.	“ODSNMASK keyword” on page 500
OEXITRTN	Optional	Specifies the name of the user exit routine to be invoked prior to the write process of a Change function.	“OEXITRTN keyword” on page 502
OTASKCTL	Optional	Specifies the maximum number of output areas to be processed concurrently.	“OTASKCTL keyword” on page 503
PAD	Optional	Specifies the padding character to be used when the segment minimum length is increased due to DBD definition change. The PAD keyword is effective for fixed-length segments and variable-length segments.	“PAD keyword” on page 504
PTRCHK	Optional	Specifies what to do when a pointer error is detected.	“PTRCHK keyword” on page 505
PTRCHKLVL	Optional	Specifies the method that is used to run integrity verification in the root addressable part and in the independent overflow part.	“PTRCHKLVL keyword” on page 506
RELOCATE_SDEP	Optional	Specifies that SDEP information will be relocated.	“RELOCATE_SDEP keyword” on page 512
SDEP	Optional	Specifies how the SDEP segment data is to be processed.	“SDEP keyword” on page 524
SDEP_BYPASS_MSGLVL	Optional	Specifies the severity level of the message that FPA issues when the SDEP is defined but SDEP segment data is not unloaded.	“SDEP_BYPASS_MSGLVL keyword” on page 525
SDEPCHKLVL	Optional	Specifies the method that is used to run integrity verification in the sequential dependent part.	“SDEPCHKLVL keyword” on page 526
SDEP_ORPHAN_MSGLVL	Optional	Specifies the error message severity level to be set when the reloaded area is analyzed concurrently and an orphan SDEP segment is detected.	SDEP_ORPHAN_MSGLVL keyword
SENSOR	Optional	Specifies whether to store the sensor data in the Sensor Data repository of IMS Tools KB.	“SENSOR keyword” on page 535

Table 18. CHANGE command keywords (continued)

Keyword	Required or optional	Description	Topic
SENSOR_DBREC	Optional	Specifies whether to store the data elements that are related to database records and root segment distribution in the Sensor Data repository of IMS Tools KB.	"SENSOR_DBREC keyword" on page 536
SENSOR_PTRERR	Optional	Specifies to store the sensor data in the Sensor Data repository of IMS Tools KB even when pointer errors are detected in the area.	"SENSOR_PTRERR keyword" on page 537
SENSOR_UOW	Optional	Specifies whether to store the data elements that are related to UOW in the Sensor Data repository of IMS Tools KB.	"SENSOR_UOW keyword" on page 537
SENSOR_UOW_REPORT	Optional	Specifies whether to print the data elements that are related to UOW in the Sensor Data Statistics report.	SENSOR_UOW_REPORT keyword
SORTOPT_FILSZ	Optional	Specifies that FPA adds the FILSZ parameter to the SORT control statement.	"SORTOPT_FILSZ keyword" on page 542
SSP	Optional	Specifies whether subset pointers should be retained or cleared.	"SSP keyword" on page 544

ALLOCATE subcommand keywords

The ALLOCATE subcommand dynamically defines a VSAM cluster that is used for the processing of the CHANGE command.

For the ALLOCATE subcommand, you can specify the keywords that are listed in the following table. For the format of the ALLOCATE subcommand, see "ALLOCATE subcommand" on page 419.

Table 19. ALLOCATE subcommand keywords

Keyword	Required or optional	Description	Topic
AMSOPT	Optional	Passes optional parameters to the Access Method Services (AMS) to define VSAM clusters.	"AMSOPT keyword" on page 440
ATTRIBUTE	Optional	Requests optional processing related to allocation of a VSAM cluster.	"ATTRIBUTE keyword" on page 444
AVGREC	Optional	Identifies the space requirements of the output data set that is created by dynamic allocation.	"AVGREC keyword" on page 445
DATACLAS	Optional	Specifies the SMS data class of an output data set that is created by dynamic allocation.	"DATACLAS keyword" on page 449
KEYLABEL	Optional	Specifies the key label for encrypting a new VSAM cluster.	"KEYLABEL keyword" on page 489
MGMTCLAS	Optional	Specifies the SMS management class of an output data set that is created by dynamic allocation.	"MGMTCLAS keyword" on page 496
OAREA	Required	Specifies one or more output areas for the process.	"OAREA keyword" on page 497
SPACE	Optional	Specifies the space requirements of an output data set that is created by dynamic allocation.	"SPACE keyword" on page 543
STORCLAS	Optional	Specifies the SMS storage class of an output data set that is created by dynamic allocation.	"STORCLAS keyword" on page 546
VOLSER	Optional	Specifies one or more volume-serial identifiers for the data set that is accessed by dynamic allocation. If this keyword is omitted for an output data set, nonspecific (private) volumes are required.	"VOLSER keyword" on page 556

IRENAME subcommand keywords

The IRENAME subcommand renames the input VSAM cluster that is used for the processing of the CHANGE command. The same area data set name can be specified with the input and output area data sets by specifying the IRENAME subcommand.

For the IRENAME subcommand, you can specify the keywords that are listed in the following table. For the format of the IRENAME subcommand, see [“IRENAME subcommand”](#) on page 428.

Table 20. IRENAME subcommand keywords

Keyword	Required or optional	Description	Topic
IAREA	Optional	Specifies one or more input areas for the rename process before the Change process restructures or reorganizes a DEDB.	“IAREA keyword” on page 469
DSNAME	Optional	Specifies a mask for the data set name to rename the input area data set by the subcommand.	“DSNAME keyword” on page 456

LOADCTL subcommand keywords

The LOADCTL subcommand specifies the load control information for a segment. You can specify an Insert Limit Count (ILC) and a place to insert dependent segments (either DOVF or IOVF). ILC is the maximum number of occurrences of segments that are inserted close to their root.

If the LOADCTL subcommand is specified multiple times for the same segment, they will be checked from the first to the last, and the first LOADCTL statement that matches the condition will be used for segment insertion.

For the LOADCTL subcommand, you can specify the keywords that are listed in the following table. For the format of the LOADCTL subcommand, see [“LOADCTL subcommand”](#) on page 429.

Table 21. LOADCTL subcommand keywords

Keyword	Required or optional	Description	Topic
INSERT_LIMIT_COUNT	Optional	Specifies the maximum number of segments that are selected by the SEGMENT keyword, that can be inserted close to their root segment. The maximum value is 32767 and the minimum value is 0. The minimum value 0 specifies that all specified segments are stored far from their root segment.	INSERT_LIMIT_COUNT keyword
LOCATION	Optional	Specifies the location to insert dependent segments, either DOVF or IOVF. You can specify the location where segments that were selected by the SEGMENT keyword are loaded.	“LOCATION keyword” on page 494
SEGMENT	Required	Specifies the segment name to be inserted at the specified location. The segment name must be a name that is coded on the SEGM statement in your DBD.	“SEGMENT keyword” on page 530

REPORT subcommand keywords

The REPORT subcommand requests the analysis reports. If you do not specify PTRCHKLVL=FULL or PTRCHKLVL=QUICK, this subcommand is ignored.

For the REPORT subcommand, you can specify the keywords that are listed in the following table. For the format of the REPORT subcommand, see [“REPORT subcommand”](#) on page 432.

Table 22. REPORT subcommand keywords

Keyword	Required or optional	Description	Topic
DBRLDIST	Optional	Specifies whether to generate the DB Record Length Distribution report.	“DBRLDIST keyword” on page 452

Table 22. REPORT subcommand keywords (continued)

Keyword	Required or optional	Description	Topic
IOVFCI_RBA_REPORT	Optional	Specifies whether to generate the RBA of Used IOVF CI report.	“IOVFCI_RBA_REPORT keyword” on page 484
LARGEREC_COUNT	Optional	Specifies the number of reporting entries in the Largest Database Records report and the Consolidated Largest Database Records report. This keyword is effective only when PTRCHKLVL=FULL is specified.	“LARGEREC_COUNT keyword” on page 491
LARGEREC_REPORT	Optional	Specifies whether to generate the Largest Database Records report. This keyword is effective only when PTRCHKLVL=FULL is specified.	“LARGEREC_REPORT keyword” on page 491
REPORTDD	Optional	Specifies the destination ddname for the analysis reports.	“REPORTDD keyword” on page 513
SEGLDIST	Optional	Specifies whether to generate the Segment Length Distribution report.	“SEGLDIST keyword” on page 529
SEGLSTAT	Optional	Specifies the format of the DB Record Profile Analysis report, which is one of the DEDB Area Analysis reports.	“SEGLSTAT keyword” on page 529
STARTUOW	Optional	Specifies the number that defines the first UOW to be reported in the UOW report and the UOW Exception report.	“STARTUOW keyword” on page 545
STOPUOW	Optional	Specifies the number that defines the last UOW to be reported in the UOW report and the UOW Exception report.	“STOPUOW keyword” on page 546
UOW_REPORT	Optional	Specifies whether to generate the UOW report.	“UOW_REPORT keyword” on page 555

THRESHOLD subcommand keywords

The THRESHOLD subcommand specifies the threshold parameter values, which cause warning messages to be generated when the actual values do not meet the specified criteria. If you do not specify the REPORT subcommand, this subcommand is ignored.

For the THRESHOLD subcommand, you can specify the keywords that are listed in the following table. For the format of the THRESHOLD subcommand, see [“THRESHOLD subcommand” on page 435](#).

Table 23. THRESHOLD subcommand keywords

Keyword	Required or optional	Description	Topic
DOVFFS_PCT	Optional	Specifies a threshold setting for the percentage of free space in DOVF. If the percentage of free space in DOVF is less than the value that you specified, a warning message is issued.	“DOVFFS_PCT keyword” on page 455
FS_DOVFIOVF	Optional	Specifies a threshold setting for the free space in DOVF and in IOVF. If the free space in DOVF is greater than the value that you specified, and if the free space in IOVF is less than the value that you specified, a warning message is issued.	“FS_DOVFIOVF keyword” on page 466
FS_RAADOVF	Optional	Specifies a threshold setting for the free space in RAA BASE and in DOVF. If the free space in RAA BASE is greater than the value that you specified, and if the free space in DOVF is less than the value that you specified, a warning message is issued.	“FS_RAADOVF keyword” on page 467
FS_RAADOVFIOVF	Optional	Specifies a threshold setting for the free space in RAA BASE, DOVF, and IOVF. If the free space in RAA BASE is greater than the value that you specified, and if the free space in either DOVF or IOVF is less than the value that you specified, a warning message is issued.	“FS_RAADOVFIOVF keyword” on page 467
FS_RAAIOVF	Optional	Specifies a threshold setting for the free space in RAA BASE and in IOVF. If the free space in RAA BASE is greater than the value that you specified, and if the free space in IOVF is less than the value that you specified, a warning message is issued.	“FS_RAAIOVF keyword” on page 468

Table 23. THRESHOLD subcommand keywords (continued)

Keyword	Required or optional	Description	Topic
IOVFFS_PCT	Optional	Specifies a threshold setting for the percentage of free space in IOVF. If the percentage is less than value that you specified, a warning message is issued.	“IOVFFS_PCT keyword” on page 485
IOVFUSED_PCT	Optional	Specifies a threshold setting for the percentage of IOVF CIs used. If the percentage is greater than the value that you specified, a warning message is issued.	“IOVFUSED_PCT keyword” on page 485
RAAFS_PCT	Optional	Specifies a threshold setting for the percentage of free space in RAA BASE. If the percentage is less than the value that you specified, a warning message is issued.	“RAAFS_PCT keyword” on page 507
RAPOVFL_PCT	Optional	Specifies a threshold setting for the percentage of RAA BASE CIs that use either DOVF or IOVF. If the percentage is greater than the value that you specified, a warning message is issued.	“RAPOVFL_PCT keyword” on page 508
RECIO_AVG	Optional	Specifies a threshold setting for the average number of I/Os that can be done to access a database record. If the number is greater than the value that you specified, a warning message is issued.	“RECIO_AVG keyword” on page 510
RECIO_MAX	Optional	Specifies a threshold setting for the maximum number of I/Os that can be done to access a database record. If the number is greater than the value that you specified, a warning message is issued.	“RECIO_MAX keyword” on page 511
RECIOVF_PCT	Optional	Specifies a threshold setting for the percentage of records using IOVF CIs. If the percentage is greater than the value that you specified, a warning message is issued.	“RECIOVF_PCT keyword” on page 510
ROOTIO_AVG	Optional	Specifies a threshold setting for the average number of I/Os to access a root segment. If the number is greater than the value that you specified, a warning message is issued.	“ROOTIO_AVG keyword” on page 522
ROOTIO_MAX	Optional	Specifies a threshold setting for the maximum number of I/Os that can be done to access a root segment. If the number is greater than the value that you specified, a warning message is issued.	“ROOTIO_MAX keyword” on page 523
SYN_LEN_AVG	Optional	Specifies a threshold setting for the average synonym chain length. If the length is greater than the value that you specified, a warning message is issued.	“SYN_LEN_AVG keyword” on page 547
SYN_MAX	Optional	Specifies a threshold setting for the maximum synonym chain length. If the length is greater than the value that you specified, a warning message is issued.	“SYN_MAX keyword” on page 548
SYN_PCT_LEN	Optional	Specifies a threshold setting for the percentage of RAPs that have a long synonym chain among all the RAPs that have synonyms. If the percentage of the RAPs that have a long synonym chain is greater than the threshold value, a warning message is issued.	“SYN_PCT_LEN keyword” on page 548
UOWDOVF_PCT	Optional	Specifies a threshold setting for the percentage of UOWs that use DOVF. If the percentage is greater than the value that you specified, a warning message is issued.	“UOWDOVF_PCT keyword” on page 553
UOWIOVF_AVG	Optional	Specifies a threshold setting for the average number of IOVF CIs that are used by a UOW. If the number is greater than the value that you specified, a warning message is issued.	“UOWIOVF_AVG keyword” on page 554
UOWIOVF_MAX	Optional	Specifies a threshold setting for the maximum number of IOVF CIs that are used by a UOW. If the number is greater than the value that you specified, a warning message is issued.	“UOWIOVF_MAX keyword” on page 554
UOWIOVF_PCT	Optional	Specifies a threshold setting for the percentage of UOWs that use IOVF. If the percentage is greater than the value that you specified, a warning message is issued.	“UOWIOVF_PCT keyword” on page 555

Examples for the Change function

Review examples and learn techniques for changing DEDB areas.

In these examples, the following DEDB is assumed:

```

DEDB name:  DEDBJN22 has four areas
area name:  DB22AR0 (area number is 1 and has three ADSs)
ADS DD name: DB22AR01, DS name: HPFP.DB22AR0.ADS1
area name:  DB22AR1 (area number is 2 and has two ADSs)
ADS DD name: DB22AR11, DS name: HPFP.DB22AR1.ADS1
area name:  DB22AR2 (area number is 3 and has two ADSs)
ADS DD name: DB22AR21, DS name: HPFP.DB22AR2.ADS1
area name:  DB22AR3 (area number is 4 and has one ADS)
ADS DD name: DB22AR31, DS name: HPFP.DB22AR3.ADS1

```

In DBRC, all ADSs are in UNAVAILABLE status.

You can find additional JCL examples that help you to monitor and tune DEDB areas in [“FPA Change JCL examples for monitoring and tuning DEDBs”](#) on page 1141.

Example 1: Reorganizing an area that is registered with DBRC

The following figure shows example JCL for reorganizing an area that is registered with DBRC.

The following steps must be done before running this example:

- Rename the area data set name to 'HPFP.OLD.DB22AR0.ADS1'.
- Define the area data set with the original data set name 'HPFP.DB22AR0.ADS1'.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS to be unloaded is specified by IAREA001 DD in the JCL statement.
- The output ADSs to be reloaded are dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC.
- Area DB22AR0 is reorganized.
- After reorganizing, the RECOV NEEDED status of area DB22AR0 is changed to OFF, and the ADS is made AVAILABLE with DBRC

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//        DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//IAREA001 DD DISP=SHR,DSN=HPFP.OLD.DB22AR0.ADS1
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
CHANGE
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  OAREA=(DB22AR0)
/*

```

Figure 66. Reorganizing an area that is registered with DBRC

Example 2: Restructuring all areas that are registered with DBRC

The following figure shows example JCL for restructuring all areas that are registered with DBRC.

The following steps must be done before running this example:

- Rename the area data set name for DB22AR n to 'HPFP.OLD.DB22AR n .ADS1', where value n is 1, 2, 3, or 4.
- Define the area data set with the original data set name 'HPFP.DB22AR n .ADS1'.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADSs to be unloaded for all areas are dynamically defined and allocated by using the IDSNMASK parameter.
- The output ADSs to be reloaded are dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC.
- The ACBLIB member for the old DEDB definition is in the OLDACB DD data set and all areas that are restructured.
- All database records are randomized by the randomizer, which is loaded from the IMSRESLB DD data set.
- After restructuring, the RECOV NEEDED status of areas DB22AR0, DB22AR1, DB22AR2, and DB22AR3 are changed to OFF, and all ADSs of these areas are made AVAILABLE with DBRC.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//         DD DISP=SHR,DSN=IMSVS.SDFSRESL
//         DD DISP=SHR,DSN=IMSVS.PGMLIB
//OLDACB  DD DISP=SHR,DSN=IMSVS.ACBLIB.OLD
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
CHANGE
  DBD=DEDBJN22,
  IAREA=ALL,
  IDSNMASK='HPFP.OLD.&AREA.ADS1',
  OAREA=ALL
/*
```

Figure 67. Restructuring all areas that are registered with DBRC

Example 3: Reorganizing multiple areas that are registered with DBRC while renaming the input ADS

The following figure shows example JCL for reorganizing multiple areas that are registered with DBRC while renaming the input ADS.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADSs to be unloaded for the specified areas are dynamically defined and allocated by using the IDSNMASK parameter.
- The input ADSs are renamed by using a mask before processing the CHANGE command.
- The output ADSs are defined by the ALLOCATE subcommand before processing the CHANGE command.
- After reorganizing, the RECOV NEEDED status of areas DB22AR0, DB22AR1 are changed to OFF, and both ADSs of these areas are made AVAILABLE with DBRC.
- The reorganized areas DB22AR0 and DB22AR1 are analyzed with the FULL check option.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//         DD DISP=SHR,DSN=IMSVS.SDFSRESL
//         DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
CHANGE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  IDSNMASK='HPFP.&AREA.ADS1',
  OAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL
IRENAME
  IAREA=(DB22AR0,DB22AR1),
  DSNAME='HPFP.OLD.&AREA.ADS1'
ALLOCATE
  OAREA=(DB22AR0,DB22AR1),
  SPACE=(CYL,5),
  VOLSER=HFPVOL
/*

```

Figure 68. Reorganizing multiple areas that are registered with DBRC while renaming the input ADS

Example 4: Reorganizing an area that is registered with DBRC and creating its image copy

The following figure shows example JCL to reorganize an area that is registered with DBRC and to create its image copy.

The following steps must be done before running this example:

- Rename the area data set name for DB22AR0 to 'HPFP.OLD.DB22AR0.ADS1'.
- Define the area data set with the original data set name 'HPFP.DB22AR0.ADS1'.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS to be unloaded is specified by IAREA001 DD in the JCL statement.
- The output ADSs to be reloaded are dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC.
- Area DB22AR0 is reorganized.
- After reorganizing, the RECOV NEEDED status of area DB22AR0 is changed to OFF, and the ADS is made AVAILABLE with DBRC.
- After reorganizing, the image copy data set for the area is dynamically allocated by IMS HP Image Copy. The data set name of the image copy is determined by IMS HP Image Copy as follows:

```
ICOUT.HFP.IC1.DEDBJN22.DB22AR0
```

For more details, see *IMS High Performance Image Copy User's Guide*.

- After creating image copy, a NOTIFY.IC command is issued to DBRC.

```

//HFP EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=HPS.SHPSLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//IAREA001 DD DISP=SHR,DSN=HPFP.OLD.DB22AR0.ADS1
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
CHANGE
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  OAREA=(DB22AR0),
  ICOUT=YES,
  ICNOTIFY=YES
/*
//ICEIN DD *
GLOBAL ICHLQ=ICOUT.HFP,
        UNIT=SYSDA,
        SPACE=(CYL,100,100)
/*
//ICERPRT DD SYSOUT=*

```

Figure 69. Reorganizing an area that is registered with DBRC and creating its image copy

Example 5: Reorganizing an area, analyzing the reorganized area, and storing the sensor data in the IMS Tools KB repository

The following figure shows example JCL for reorganizing an area that is registered with DBRC, analyzing the reorganized area, and storing the sensor data in the Sensor Data repository of IMS Tools KB.

Before you run this example, complete the following tasks:

1. Rename the area data set name to 'HPFP.OLD.DB22AR0.ADS1'.
2. Define the area data set with the original data set name 'HPFP.DB22AR0.ADS1'.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS to be unloaded is specified by IAREA001 DD in the JCL statement.
- The output ADSs to be reloaded are dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC.
- Area DB22AR0 is reorganized.
- After the reorganization process is complete, the RECOV NEEDED status of area DB22AR0 is changed to OFF, and the ADS is made AVAILABLE with DBRC.
- After the reorganization process is complete, the reorganized area DB22AR0 is analyzed with the full check option, and sensor data is collected and stored in the Sensor Data repository of IMS Tools KB. In addition to the default data elements that are collected by the SENSOR keyword, the data elements that are related to database records and root segment distribution are stored (SENSOR_DBREC=YES specification).

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//          DD DISP=SHR,DSN=ITB.SHKTLOAD
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//IAREA001 DD DISP=SHR,DSN=HPFP.OLD.DB22AR0.ADS1
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSRPT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES,
  ITKBSRVR=FPQSRV01,
  ADXCFGRP=ADSRV01
CHANGE
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  OAREA=(DB22AR0),
  PTRCHKLVL=FULL,
  SENSOR=YES,
  SENSOR_DBREC=YES
/*

```

Figure 70. Reorganizing an area, analyzing the reorganized area, and storing the sensor data in the IMS Tools KB repository

Chapter 7. Reorganizing a DEDB offline with FPA

You can reorganize a DEDB by using the FPA Reorg function.

Topics:

- [“Functions of Reorg” on page 163](#)
- [“Input and output for the Reorg function” on page 166](#)
- [“Considerations for running the Reorg function” on page 166](#)
- [“Running the Reorg function” on page 167](#)
- [“Region size requirements for the Reorg function” on page 169](#)
- [“DD statements for the Reorg function” on page 170](#)
- [“Utility control statements for the Reorg function” on page 183](#)
- [“Examples for the Reorg function” on page 188](#)

Functions of Reorg

The Reorg function reorganizes a set of UOWs in a DEDB area on the basis of a ranked set of UOWs.

Reorganization of a ranked set of UOWs

The Reorg function goes through the phases to reorganize a ranked set of UOWs. These phases include the statistics gathering phase, the UOW Evaluation phase, the UOW Ranking phases, and the UOW Reorganization phase.

Statistics gathering phase

In the statistics gathering phase, the Reorg function runs Area Sensor to gather statistics for a set of UOWs in the DEDB area. The statistics gathered is used in the subsequent UOW Evaluation phase.

UOW Evaluation phase

In the UOW Evaluation phase, each UOW is evaluated by the RBASEFS and RDOVFFS criteria. These criteria measure the free space percentage in the CIs where the overflow data might be relocated by reorganization. These criteria are specified by the RBASEFS and RDOVFFS parameters respectively:

- The RBASEFS parameter specifies a percentage of free space in a RAP CI. If the UOW contains at least one RAP CI that contains free space that is greater than the percentage that is specified, and the RAP CI also uses a DOVF CI or an IOVF CI, the UOW can be selected as a candidate for the reorganization (See [Figure 71 on page 164](#)). RBASEFS=100 has a special meaning; if RBASEFS=100, this parameter is not used when selecting the candidate UOWs.
- The RDOVFFS parameter specifies a percentage of free space in the DOVF section in a UOW. If the free space percentage in the DOVF section in a UOW is greater than the percentage that is specified, and the UOW also uses IOVF CIs, the UOW can be selected as a candidate for the reorganization (See [Figure 72 on page 164](#)). RDOVFFS=100 has a special meaning; if RDOVFFS=100, this parameter is not used when selecting the candidate UOWs.

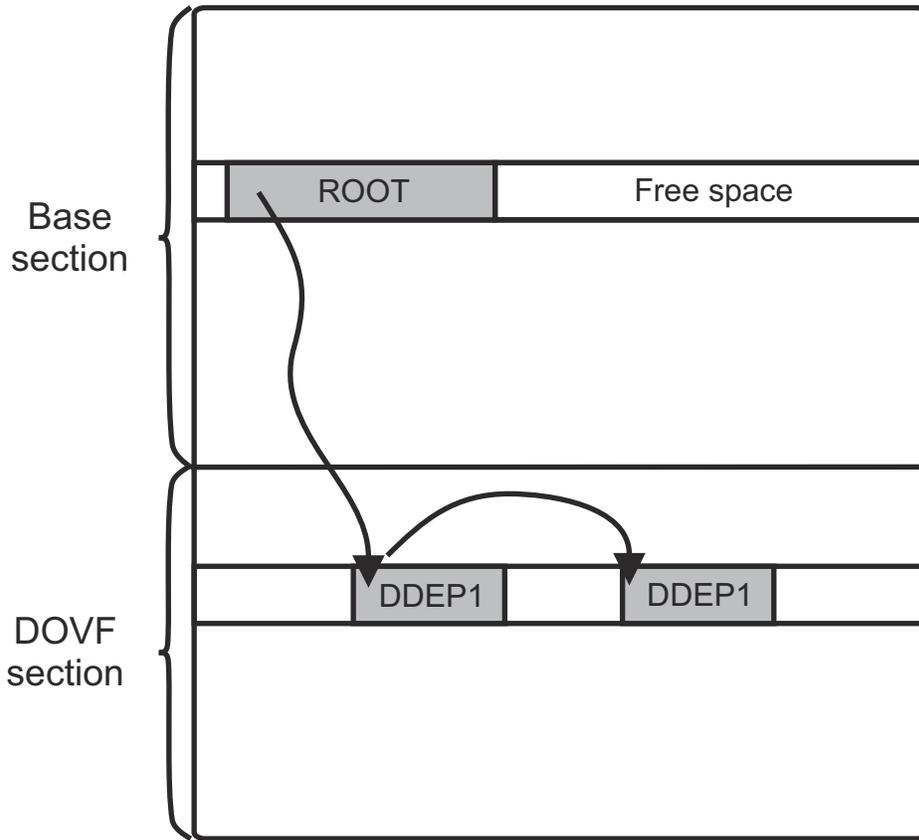


Figure 71. A UOW that satisfies the RBASEFS condition

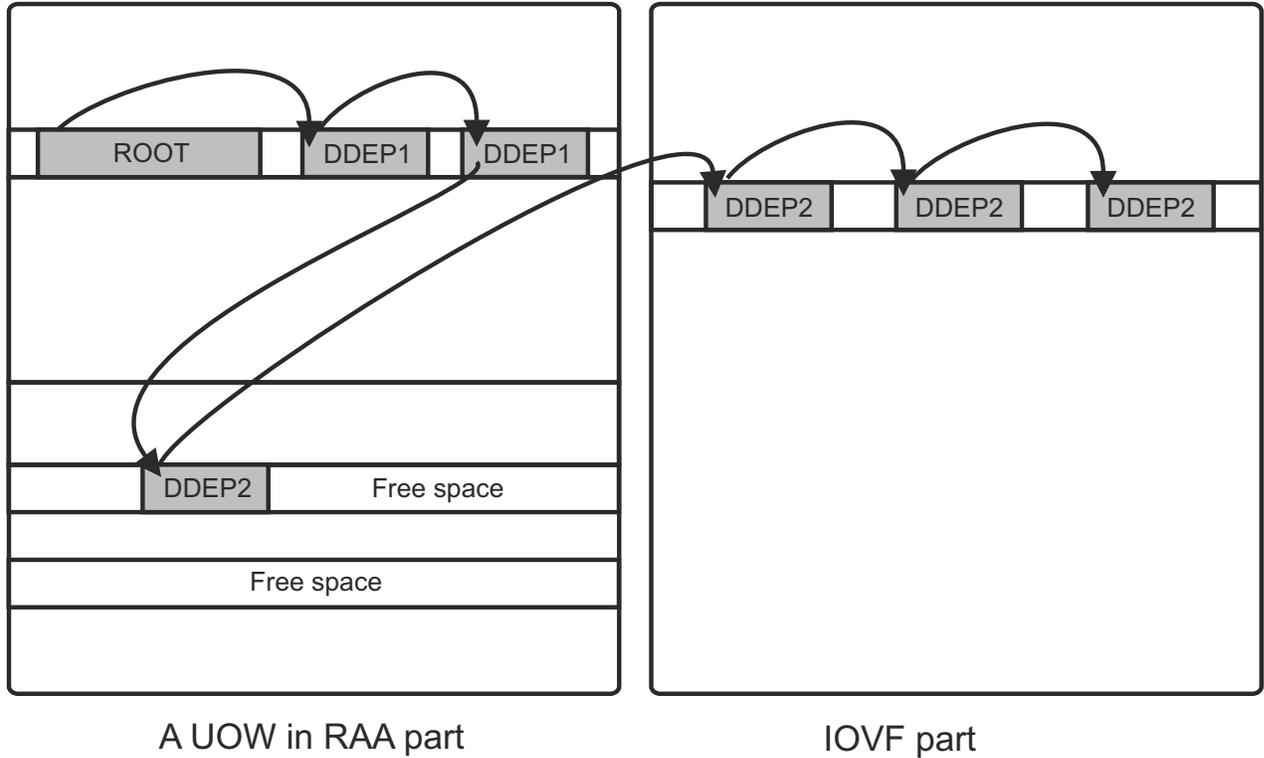


Figure 72. A UOW that satisfies the RDOVFFS condition

In the UOW Evaluation phase, an integer called Reorganization Free Space Score (RFS Score) is calculated for each of the candidate UOWs that are selected by the RBASEFS and RDOVFFS criteria. The RFS score is a combined score that is based on the RBASEFS and RDOVFFS criteria.

UOW Ranking phase

In the UOW Ranking phase, the Reorg function ranks, according to the RFS score, the set of candidate UOWs that is selected by the RBASEFS and RDOVFFS criteria.

UOW Reorganization phase

In the UOW Reorganization phase, the Reorg function reorganizes the ranked set of UOWs.

There are two phases within the reorganization of one UOW:

- Build phase, in which rebuilding of the UOW is done
- Copy phase, in which the rebuilt UOW is copied into the original location, and all independent overflow control intervals that are no longer used by the UOW are freed

During the build phase, the Reorg function copies, to the UOW buffer, the root segments of one UOW and their direct dependent segments in the logical order. If the UOW buffer cannot hold all the segments of the chains, the Reorg function allocates independent overflow space and fills it with the remaining segments. If the overflow is insufficient, the reorganization process for the UOW being reorganized is canceled. The Reorg function does not change the UOW that is being reorganized until it is completely rebuilt in the UOW buffer and copied to the reorganization UOW in the area, at which time the Reorg function sets the "copy phase reached" indicator.

During the copy phase, the Reorg function writes the UOW buffer into the corresponding location within the UOW that is being reorganized. The "copy phase reached" indicator is reset when the rebuilt UOW is successfully copied into the proper UOW location in the area. Any IOVF CIs that are no longer needed are then freed.

The user can specify an allowable maximum number of UOWs to be reorganized with the REORGLIM keyword. The number is called the reorganization limit. If the reorganization limit was specified, the highly ranked UOWs are reorganized up to the limit number of UOWs. The reorganization limit can be supplied as a number (quantity) of UOWs or as a percentage of the UOWs defined for the DEDB area.

Related reference

RBASEFS keyword

Use the RBASEFS keyword to specify a criterion to select UOWs to reorganize.

RDOVFFS keyword

Use the RDOVFFS keyword to specify a criterion to select UOWs to reorganize.

REORGLIM keyword

Use the REORGLIM keyword to specify the reorganization limit for the amount of UOWs that can be reorganized in the area.

The number of UOWs to be reorganized

The number of UOWs to be reorganized is determined by the combination of the REORG command parameters.

In this topic, it is assumed that YES is specified in the first parameter of the RFSSCAN keyword of the Reorg process. This enables the UOW Evaluation and UOW Ranking functions of the REORG command.

The number of UOWs to be reorganized is determined by the combination of the following parameters of the REORG command:

- RBASEFS
- RDOVFFS
- REORGLIM

Both the RBASEFS and the RDOVFFS parameters can be used to evaluate each UOW in the area. If at least one of RBASEFS and RDOVFFS is less than 100, the set of UOWs are classified into two subsets:

- The group of UOWs that satisfy the RBASEFS condition or the RDOVFFS condition (SUBSET-1)
- The group of UOWs that satisfy neither the RBASEFS condition nor the RDOVFFS condition

UOWs in SUBSET-1 are selected as initial candidates for the reorganization. If value 100 is specified for both the RBASEFS and the RDOVFFS parameters, all UOWs in the area are classified as SUBSET-1, that is, all UOWs are selected as candidates for the reorganization.

The REORGLIM parameter is used to limit the number of UOWs that are actually reorganized in the Reorg process. The parameter works on the ranked UOWs in SUBSET-1.

If the maximum number of UOWs determined by the REORGLIM parameter is larger than the number of UOWs in SUBSET-1, all UOWs in SUBSET-1 are reorganized, but no other UOWs are reorganized.

Input and output for the Reorg function

The Reorg function is invoked by the REORG command in the HFPSYSIN DD control statement to reorganize a set of UOWs in DEDB area on the basis of a ranked set of UOWs.

When the reorganization process is completed, IMS HP Image Copy can be invoked internally to create the image copies.

The following figure shows the input and the output for the Reorg function.

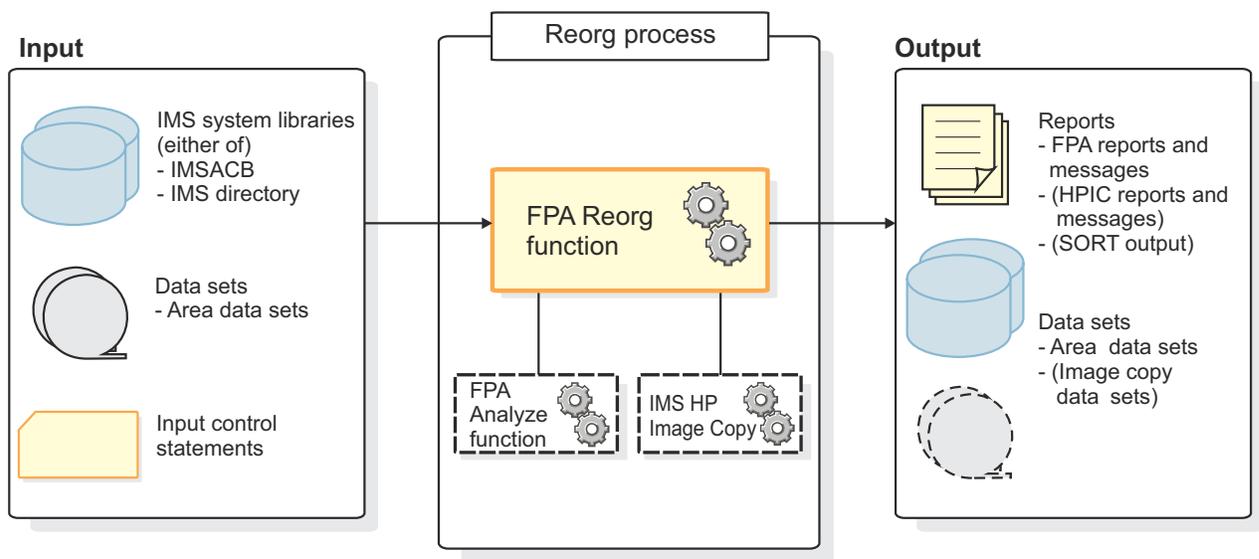


Figure 73. Input and output for the Reorg function

The Reorg function reorganizes a set of UOWs in a DEDB area on the basis of a ranked set of UOW and analyzes the DEDB areas by using the REORG command in the HFPSYSIN DD control statement. The Analyze function can be invoked internally after the Reorg function. When the Analyze process is completed, IMS HP Image Copy can be invoked internally to create the image copies.

Considerations for running the Reorg function

The following topics describe what you need to consider before running the Reorg function.

Because the Reorg function does not create the update logs during the process, the continuity of log records from the last copy will be broken after reorganizing. Therefore, the IC NEEDED flag is turned on for the database recovery.

It is highly recommended that you take a valid image copy of the output area after running the Reorg function. If you do not take the image copy of the output area, and if, by any chance, the area is broken, there is no way to recover the output area.

Recovery and restart

If the Reorg function ends abnormally, reorganization can be restarted by rerunning the job.

If the Reorg function ends abnormally during the build phase, reorganization can be restarted by rerunning the job. When restarted, the process starts from the beginning of the build phase on the UOW that was being processed at the time the process ended.

If the Reorg function ends abnormally during the copy phase, the copy phase can be completed by rerunning the job. When restarted, the process starts from the beginning of the copy phase on the UOW that was being processed at the time the process ended.

You can choose the method for recovering an area by the RECOVOPT keyword.

Related reference

[RECOVOPT keyword](#)

Use the RECOVOPT keyword to specify the method for recovering an area when the process ends abnormally.

MADS considerations

Multiple area data sets are not supported.

The Reorg function will select one of the available error-free ADSs that are registered with DBRC. All ADSs other than the selected ADS will be made unavailable. You can use the DEDB Area Data Set Create Utility (DBFUMRIO) to synchronize the unavailable ADSs after the completion of the Reorg function.

Running the Reorg function

FPA runs the Reorg function as a standard z/OS batch job. You must specify an EXEC statement and DD statements that define the input and output data sets in your JCL.

Procedure

1. Optional: Estimate the region size that is required to run the job.
See [“Region size requirements for the Reorg function”](#) on page 169.
2. Specify the EXEC statement in the following format:

```
//REORG EXEC PGM=HFPMAIN0,REGION=rrrrM,  
// PARM='DBRCGRP=dbrcgrp,GSGNAME=gsgname,IMSPLEX=imsplex'
```

For the parameters, see the following table.

Parameter	Description
DBRCGRP=	DBRCGRP is an optional 1- to 3-character parameter that specifies the identifier assigned to a group of DBRC instances (DBRC sharing group) that access the same RECON data set in the IMSplex.
GSGNAME=	GSGNAME is an optional 1- to 8-character parameter that specifies the global service group.
IMSPLEX=	IMSPLEX is an optional 1- to 5-character parameter that specifies the IMSplex group names to share either databases or message queues.

3. Specify the DD statements to define input data sets and output data sets.
4. Optional step. If you want to create an RDE during the FPA process, follow these steps:

a) Activate Integrity Checker.

To create a new RDE by using the Integrity Checker utility of IMS Library Integrity Utilities, follow the instructions in "Activating Integrity Checker" in the *IMS Library Integrity Utilities User's Guide*.

b) Request that FPA creates an RDE by using the Integrity Checker utility of IMS Library Integrity Utilities.

i) If you are to use a specific LIU global option module, specify the name of the LIU global option module in the LIUGOPT parameter of the FPA control statement (HFPSYSIN). You also need to specify DBRC=YES in the FPA JCL because Integrity Checker requires that DBRC be active.

ii) In the JOBLIB/STEPLIB DD statement, concatenate the following load module libraries.

- The IMS Library Integrity Utilities load module library
- The LIU global option modules (LIU@INST, LIU@*imsi*, or both)

5. Run the JCL job.

Example

The following figure shows example JCL for the REORG command.

```
//*****  
//** REORG THE AREA(S) **  
//*****  
//REORG EXEC PGM=HFPMAIN0  
//*  
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0  
// DD DISP=SHR,DSN=IMSVS.SDFSRESL  
// DD DISP=SHR,DSN=IMSVS.PGMLIB  
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB  
//IMSRESLB DD DISP=SHR,DSN=IMSVS.SDFSRESL  
//HFPPRINT DD SYSOUT=A  
//HFPRPTS DD SYSOUT=A  
//HFPSYSIN DD *  
GLOBAL DBRC=NO  
REORG  
DBD=IVPDB3  
/*  
//DFSIVD3A DD DISP=OLD,DSN=IMSVS.DFSIVD31  
//DFSIVD3B DD DISP=OLD,DSN=IMSVS.DFSIVD33  
//*
```

Figure 74. Example JCL for the REORG command

Related concepts

[Region size requirements for the Reorg function](#)

The Reorg function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Related reference

[LIUGOPT keyword](#)

Use the LIUGOPT keyword to specify the LIU global option module that is to be used by IMS Library Integrity Utilities. The Integrity Checker utility of IMS Library Integrity Utilities requires that DBRC be active.

[Examples for the Reorg function](#)

Review examples and learn techniques for reorganizing DEDB areas.

[Tips and best practices for the FPA Reorg function](#)

The FPA Reorg function supports several keywords that improve the performance of FPA Reorg jobs and that further simplify management of DEDBs.

Considerations on using the Integrity Checker utility

Some considerations apply when using the Integrity Checker utility of IMS Library Integrity Utilities.

- When you change a DBD and reorganize DEDB areas by using FPA, Integrity Checker creates new RDEs for the reorganized areas. If reorganization completes successfully, but Integrity Checker fails to create new RDEs, you must take one of the following actions:
 - Rerun the reorganization job to create new RDEs for the reorganized areas.
 - If you decide not to rerun the reorganization job and to use the original areas, you must delete the RDEs that were created during the reorganization by using the DELETE.DB command of the LICON utility before using the areas.
 - If you decide not to rerun the reorganization job and to use the reorganized areas, you must create the RDEs whose creation failed during the reorganization by using the INIT.DB command of the LICON utility before using the areas.
- When any of the following error conditions were detected in FPA or Integrity Checker processing, the RDEs are not created:
 - Either the return code is equal to or greater than 8 or the processing ends abnormally for one or more areas in the Reload, Change, or Reorg processing.
 - The return code is equal to or greater than 8 for the Analyze function even when Reload, Change, or Reorg function completes successfully.
 - Integrity Checker fails to create an RDE.
- When the randomizer routine, edit/compression routine, or both are defined in the DBD library, checksum values of the routines are stored in the RDE. You must make sure that the libraries that contain the correct routines are specified in one of the following DD statements:
 - Reload and Change functions
 - IMSRESLB
 - When IMSRESLB is not specified, JOBLIB/STEPLIB
 - Reorg function
 - JOBLIB/STEPLIB
 - If the library that contains the Edit/Compression routine is not concatenated to JOBLIB/STEPLIB, FABLO212W message is issued by Integrity Checker.

Region size requirements for the Reorg function

The Reorg function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Region size for the Reorg function depends on the size of a UOW, the number of areas being reorganized and of the ITASKCTL specifications used. Typically, database maintenance is a relatively important function. To increase the performance of the Reorg program, it would be advisable to run it in a region that has a relatively high dispatching priority (that is, at least equal to the dispatching priority of a message processing region).

Estimating the region size for Reorg

The region size required for the Reorg function is a function of the size of a UOW, the number of areas being reorganized, and the ITASKCTL specifications.

In most cases, the specification on the EXEC statement for HFPMAIN0 of a region that is too small, is detected during the initialization processing of the main task. This causes processing to be terminated at

that point. Under certain conditions, the region size shortfall does not become evident until the actual reorganization has begun. The job must be completely rerun. Thus, the region size calculations must always be performed prior to initiating reorganization processing.

The following worksheet is for estimating the region size that is required for a typical reorganization job. The actual region size depends on the keywords that are specified for the job. Therefore, when you specify the region size in your JCL, use a higher value than the value calculated from this worksheet.

FPA programs, internal tables, control blocks	3,020,000	
Number of areas defined in DBD * 1,300	-----	
Internal tables, control blocks	37,500	
Maximum Segment length * 1,000	-----	
Maximum CISZ * 17	-----	
subtotal	-----	
Value specified on ITASKCTL * subtotal	-----	
Number of input areas * 7,500	-----	
Total of CISZ * UOW1 of each area * 4	-----	
Number of UOWs * 130	-----	(If RFSSCAN=(YES,YES))
Number of UOWs * 70 RFSSCAN=(NO,YES))	-----	(If RFSSCAN=(YES,NO) or
Total bytes of reorganization storage	-----	

Figure 75. Worksheet for estimating the region size for the Reorg function

If an analyze step is included in the reorganization process, estimate the region size for the analyze step and use the greater region size. For how to estimate the region size for an analyze step, see [“Region size requirements for the Analyze function”](#) on page 200.

If ICACHE=YES is specified, the Reorg function preloads the IOVF part into the data space. The data space size required for the Reorg function is a total of the size of the IOVF part in each area.

The following figure shows a worksheet for estimating the data space size for the Reorg function.

Total size of IOVF parts in all areas	-----
Total bytes of a data space storage	-----

Figure 76. Worksheet for estimating data space size for the Reorg function

DD statements for the Reorg function

DD statements of the FPA Reorg function determine the input and output data sets and specify how to run the Reorg function.

Subsections:

- [“List of DD statements”](#) on page 170
- [“Input DD statements”](#) on page 172
- [“Output DD statements”](#) on page 175
- [“Input/Output DD statements”](#) on page 179

List of DD statements

The following table summarizes the DD statements for the Reorg function.

Table 24. DD statements for the Reorg function

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
JOBLIB or STEPLIB	Input	PDS	No	Required
IMSACB, IMSACBA, or IMSACBB	Input	PDS	No	Required when IMSCATHLQ=*NO
IMSDALIB	Input	PDS	No	Optional
MODSTAT	Input	PS	Yes	Optional
MODSTAT2	Input	PS	Yes	Optional
OLCSTAT	Input	PS	Yes	Optional
<i>addnname</i> or <i>areaname</i>	Input and output	VSAM ESDS	Yes	Required under certain conditions (for details, see <i>addnname/areaname</i> DD)
HFPSYSIN	Input	LRECL=80	No	Required
HFPAPARM	Input	LRECL=80	No	Optional
HFPAUOWC	Input	LRECL=80	No	Required when HFPAHST2 DD is specified
HFPCSITE	Input	PDS	Yes	Optional
HFPPRINT	Output	SYSOUT	Yes	Optional
HFPRPTS	Output	LRECL=133	Yes	Optional
HFPSPT	Output	LRECL=133	Yes	Optional
HFPAHST	Output	PS	No	Optional
HFPAHST2	Output	DISP=MOD	No	Required when HFPAUOWC DD is specified
HFPARERG	Output	LRECL=80	No	Optional
HFPASNAP	Output	LRECL=133	No	Required when CISNAP=YES
HFPABEND	Output	LRECL=125	Yes	Optional
RECON n	Input and output	KSDS	Yes	Required when both of the following conditions are met: <ul style="list-style-type: none"> • DBRC=YES • DFSMDA members for RECONn data sets are not used
<i>itttWKnn</i>	Input and output		Yes	Optional
<i>itttSMMSG</i>	Output	LRECL=133	Yes	Optional
DFSPRINT	Output	LRECL=122	Yes	Optional
ICEIN	Input	LRECL=80	No	Optional
ICEPRINT	Output	LRECL=133	Yes	Optional
ICERPRT	Output	LRECL=133	No	Optional
<i>dataout1</i>	Output	PS	Yes	Optional

Table 24. DD statements for the Reorg function (continued)

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
<i>dataout2</i>	Output	PS	Yes	Optional

Input DD statements

The following input DD statements are used for the Reorg function.

JOBLIB/STEPLIB DD

This DD statement is a required DD statement that defines the following input data sets:

- The FPA load module library (required).
- The library that contains DFSMDA dynamic allocation members for the RECON data set. When DBRC=YES is specified and the RECON n DD statement is omitted, this DD statement is required.
- The library that contains a default site default member (HFPCSITE) to determine the data set name and the member of the site default table.
- The IMS load module library (required).
- The IMS Tools Base library (SHKTLOAD), if you specify the ITKBSRVR keyword.
- The IMS Tools Base library (SGLXLOAD), if you specify the IMSCATHLQ keyword.
- The IMS HP Image Copy library, if you specify the ICOUT=YES option.
- The IMS Library Integrity Utilities library, if you want to activate Integrity Checker.

FPA must be run as an APF-authorized program. All of the data sets that are listed within the JOBLIB or the STEPLIB concatenation must also be APF-authorized. If you use DFSMDA members and the DFSMDA members are in a non-APF authorized library, you can specify the DFSMDA library on the IMSDALIB DD statement.

IMSACB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. The IMSACB DD statements (IMSACB, IMSACBA, IMSACBB, MODSTAT, MODSTAT2, and OLCSTAT) can be used to identify the active ACB library. When both OLCSTAT or MODSTAT and MODSTAT2 are found, FPA checks the time stamp and determines the latest one.

The active IMSACB library is determined according to the following rules:

1. Check IMSACB DD statement in the JCL statement.
2. Check OLCSTAT DD statement in the JCL statement.
3. Check MODSTAT and MODSTAT2 DD statement in the JCL statement.
4. Allocate OLCSTAT dynamically and check the OLCSTAT member.
5. Allocate MODSTAT and MODSTAT2 dynamically and check the MODSTAT and MODSTAT2 members.

If you specify the IMSCATHLQ keyword, IMSACB DD statements are not used. The IMS directory is used instead of the ACB library.

IMSACBA DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSACBB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSDALIB DD

This DD statement is an optional DD statement that defines the IMSDALIB data set. This DD statement is useful when the data set for dynamic allocation members is not APF-authorized.

MODSTAT DD

This DD statement is an optional DD statement that defines the MODSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

MODSTAT2 DD

This DD statement is an optional DD statement that defines the MODSTAT2 data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT2 library by using a TYPE=DATABASE statement of the DFSMDA macro.

OLCSTAT DD

This DD statement is an optional DD statement that defines the OLCSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the OLCSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

HFPSYSIN DD

This DD statement is a required DD statement that defines the input control statement that specifies the FPA functions.

The HFPSYSIN DD statement can be coded as a standard SYSIN file, a sequential data set, or a PDS member. LRECL=80 is required for the DCB of this data set. Only one command, besides the GLOBAL command, can be specified in HFPSYSIN. FPA analyzes the command syntax in columns 1 - 72. Columns 73 - 80 are regarded as comment.

For details about coding the HFPSYSIN DD statement, see [“Utility control statements for the Reorg function”](#) on page 183.

HFPAPARM DD

This DD statement is an optional DD statement. This statement defines the control parameters for generating the analysis reports.

When SENSOR=YES is specified, the values of RBASEFS and RDOVFFS keywords are stored in the Sensor Data repository as the values of data elements DB_THRESHOLD_RBASEFS and DB_THRESHOLD_RDOVFFS. These values are used by the Online Space Management utility as the values of RBASEFS and RDOVFFS keywords for the OER utility process (online reorganization process).

The HFPAPARM DD statement can be coded as a sequential data set or as a PDS member. LRECL=80 is required for the DCB of this data set. The HFPAPARM keywords must be written on one line. It can be coded as follows:

```
//HFPAPARM DD *  
RDOVFFS=40 SYNTBL=20 UD0VFFS=35 RBASEFS=40  
/*
```

You can specify the keywords that are listed in the following table.

Table 25. HFPAPARM keywords

Keyword	Description
ARDIGIT	This keyword specifies the digit of the area number field for the UOW group definition records that are defined in the HFPAUOWC file. The value is 3 or 5. The default is 3. If the HFPAUOWC file is not specified in the JCL statement, this keyword is ignored.
RBASEFS	This keyword is an optional keyword. If a RAP CI contains free space that is greater than the percentage specified, and the RAP CI also uses overflow CIs, a reorganization control statement is generated for the UOW, and the UOW is included in the UOW Exception report. The default is 100%, which means that reorganization control statements are never generated. The maximum value is 99, and the minimum value is 0.
RDOVFFS	This keyword is an optional keyword. If the DOVF part of a UOW contains free space that is greater than the percentage specified, and the UOW also uses IOVF CIs, a reorganization control statement is generated for the UOW and the UOW is included in the UOW Exception report. The default is 100%, which means that reorganization control statements are never generated. The maximum value is 99, and the minimum value is 0.
RGROUP	This keyword sets the group identifier for the statistics that are collected and stored in the History2 file or in a sensor data record set. For the History2 file, the keyword value is set in the Repository Group field, and for a sensor data record set, the value is set for the DB_SENSOR_DATA_GROUP_ID data element. This keyword has no default value. If this keyword is not specified, FPA sets the group identifier as follows: <ul style="list-style-type: none">• The Repository Group field in the History2 file contains 4 bytes of blank characters.• The DB_SENSOR_DATA_GROUP_ID data element is not included in the sensor data record set.
SYNTBL	This keyword is an optional keyword. It controls the number of reporting entries in the Synonym Chain Analysis table. The default value is 41 (that is, all synonym chains longer than 40 are grouped and reported together). The maximum value is 999, and the minimum value is 12.
UDOVFFS	This keyword is an optional keyword. If the DOVF UOW contains free space that is less than the percentage specified, the UOW is included in the UOW Exception report. The default is 0%, which means that the UOW is never included in the UOW Exception report. The maximum value is 99, and the minimum value is 0.

Note: The RBASEFS keyword and the RDOVFFS keyword that are used for the HFPAPARM data set are different from those used for the REORG command.

Requirement: Generating reorganization control statements for the High-Speed DEDB Direct Reorganization utility, you must specify the free space control percentages. You must also specify the HFPARERG DD statement in the JCL stream. For details, see the description of HFPARERG DD.

HFPAUOWC DD

This DD statement is a required DD statement when HFPAHST2 DD is specified. Each record in this data set specifies a range of UOW numbers of an area, which defines a UOW group of the area. The information about the UOW range is recorded in the HISTORY2 data set or in a sensor data record set in the order that is specified in the HFPAUOWC data set, even if the UOW ranges overlap or are in reverse order. You can specify the UOW ranges for each area in any order. However, you must specify the area numbers in ascending order.

The data set can reside in the JCL input stream. The data set can also be defined as a sequential data set or as a member of a partitioned data set. If a data set is specified, the data set must have the following attributes:

- DSORG=PS
- RECFM=FB
- LRECL=80
- BLKSIZE=*a multiple of 80*

If the data set is empty, all UOWs of the area are regarded as the target UOW range.

When you specify the HFPAHST2 DD statement, you must also specify the HFPAUOWC DD statement. If an HFPAUOWC DD statement is not specified, the HFPAHST2 DD statement is ignored.

When SENSOR_UOW=YES is specified, the data elements that will be stored differ by the HFPAUOWC DD specification as follows:

- If you specify the HFPAUOWC DD statement, data elements are collected at the UOW group level.
- If you do not specify the HFPAUOWC DD statement, data elements are collected at the UOW level.

For information about defining a UOW group, see [“UOW group definition record layout” on page 1191](#).

HFPCSITE DD

This DD statement is an optional DD statement. The HFPCSITE DD statement can be used to define the data set name of the site default table. When the SITE_DSNAME keyword is specified in the HFPSYSIN DD statement, the HFPCSITE DD statement is allocated dynamically. HFPCSITE DD in the JCL is mutually exclusive with the SITE_DSNAME keyword in the HFPSYSIN DD statement. The data set attribute of HFPCSITE DD is PDS.

ICEIN DD

This DD statement is an optional DD statement. This statement specifies the input data set that contains the control statements for IMS HP Image Copy. The data set can reside on a tape, a direct-access volume, or it can be routed through the input stream (DD * or DD DATA).

This data set is used only when you specify ICOUT=YES. If the DD statement is not coded and if ICOUT=YES is specified, the default control statements are used. For details about the ICEIN control statements, see *IMS High Performance Image Copy User's Guide*.

Output DD statements

The following output DD statements are used for the Reorg function.

HFPPRINT DD

This DD statement is an optional DD statement that defines the processing log output data set. The data set contains the processing messages that are issued by FPA.

The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set. You cannot specify DUMMY.

The HFPPRINT data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPPRINT DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

The following figure shows an example of the processing messages that are issued for the REORG command.

```
IMS HFPF UTILITIES                                "Processing messages"                                PAGE: 1
5698-FPP V2R1                                     2019-11-22 13:47:01
HFPC0001I COMMAND ANALYZE PROCESSING STARTED.
HFPC0005I DBRC=NO IS SPECIFIED.
- EEQE DETECTION IS NOT PERFORMED.
HFPC0002I COMMAND ANALYZE PROCESSING COMPLETED.
HFPF0001I REORG PROCESSING STARTED FOR DATABASE: DEDBJN23
HFPJ0030I REORG PROCESSING STARTED FOR AREA NO: 1, AREANAME: DB23AR0
HFPJ0031I REORG PROCESSING COMPLETED FOR AREA NO: 1, AREANAME: DB23AR0
HFPF0002I REORG PROCESSING COMPLETED FOR DATABASE: DEDBJN23
HFPL0001I REPORT STARTED FOR COMMAND: REORG, DATABASE: DEDBJN23
HFPL0002I REPORT COMPLETED FOR COMMAND: REORG, DATABASE: DEDBJN23
```

Figure 77. Processing messages issued for the REORG command

HFPRPTS DD

This DD statement is an optional DD statement that defines the report output data set. The data set contains the reports that can be used to verify the process of the command. The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

The following reports are generated in this data set:

- Audit report
- Processing report
- DBD Definition report
- Reorganization Processing Summary report
- UOW Space Statistics report

When the REPORT subcommand and PTRCHKLVL=FULL are specified, the following reports are generated:

- DEDB Area Analysis reports
- Process Summary of Analysis report
- Resolve Pointers report
- UOW report
- UOW Exception report
- Largest Database Records report (When LARGEREC_REPORT=YES is specified)
- Consolidated Largest Database Records report (When LARGEREC_REPORT=YES is specified and the second parameter of the LARGEREC_COUNT keyword specifies a value that is equal to or larger than 1)
- Segment Length Distribution report
- DB Record Length Distribution report
- RBA of Used IOVF CI report

When the REPORT subcommand and PTRCHKLVL=QUICK are specified, the following reports are generated:

- DEDB Area Analysis reports (FREESPACE ANALYSIS only)
- Process Summary of Analysis report
- UOW report (without DB record information)
- UOW Exception report (without DB record information)

- Pointer Checksum report
- RBA of Used IOVF CI report

You can generate reports related to the Analyze process in another DD data set that is specified by the REPORTDD=keyword parameter of the REPORT subcommand.

The HFPRPTS data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPRPTS DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*.

HFPSRT DD

This DD statement is an optional DD statement that defines the report output data set when SENSOR=YES is specified. This output data set contains the Sensor Data Statistics report that can be used to verify which sensor data is stored in the Sensor Data repository of IMS Tools KB. The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set. You can specify NULLFILE or DUMMY for this data set.

The HFPSRT data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPSRT DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*.

HFAHST DD

This DD statement is an optional DD statement. This statement defines an output data set that contains historical records. The analysis report generation process adds one record for each DEDB area that it processes. The record contains the key space utilization and performance information for that area. This statement is used only when you specify both the REPORT subcommand and the PTRCHKLVL=FULL keyword.

If this feature is used, the HFAHST data set must be preallocated with the following attributes:

- DSORG=PS
- RECFM=FB
- LRECL=100
- BLKSIZE=*user-specified*

The DD statement in the JOB stream is specified as:

```
//HFAHST DD DSN=HPFP.HISTORY,DISP=(MOD,KEEP,KEEP)
```

You can generate the records in another DD data set specified by the HISTORYDD= keyword parameter instead of HFAHST DD.

HFAHST2 DD

This DD statement is a required DD statement when HFPAUOWC DD is specified. This statement defines an output data set that contains historical records. The analysis report generation process adds one record for each UOW group that is specified by HFPAUOWC DD. The record contains the key space utilization and performance information for that area, which is based on the information

obtained from the specified UOW range. This statement is used only when you specify both the REPORT subcommand and the PTRCHKLVL=FULL keyword. If HFPAUOWC DD statement is not specified, this statement is ignored.

If this feature is used, the HFPAHST2 data set must be preallocated with the following attributes:

- DSORG=PS
- RECFM=FB
- LRECL=200
- BLKSIZE=*user-specified*

The DD statement in the JOB stream is specified as:

```
//HFPAHST2 DD DSN=HPFP.HISTORY2,DISP=(MOD,KEEP,KEEP)
```

HFPARERG DD

This DD statement is an optional DD statement. This statement defines an output data set that contains the reorganization control statements for the High-Speed DEDB Direct Reorganization utility (DBFUHDR0). This statement is used only when you specify both the REPORT subcommand and the PTRCHKLVL=FULL or the PTRCHKLVL=QUICK keyword.

The following DCB attributes of the file are set by the Analyze process:

- DSORG=PS
- RECFM=FB
- LRECL=80
- BLKSIZE=800

Do not code the DCB information in your JCL statement. Do not specify DISP=MOD for this DD statement.

Requirement:

Generating reorganization control statements for the High-Speed DEDB Direct Reorganization utility, you must specify the HFPARERG DD statement in the JCL stream. You must also specify the free space control percentages. For details about specifying the free space control percentages, see the description of HFPAARM DD.

HFASNAP DD

This DD statement is a required DD statement when CISNAP=YES. This statement defines the output data set for the CI Map/CI Dump report and the Pointer Chain Reconstruction report. This DD statement is effective when the REPORT subcommand, PTRCHKLVL=FULL or PTRCHKLVL=QUICK, and CISNAP=YES are specified.

The following DCB attributes of the file are set by the Analyze process:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement.

The HFASNAP data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFASNAP DD SYSOUT=A
```

HFABEND DD

This DD statement is an optional DD statement. It defines a snap dump output data set that is used for diagnostic purposes. This data set might be used when the FPA job ends abnormally. Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

itttSMMSG DD

This DD statement is an optional DD statement. However, this DD statement is required when DFSORT is used. This statement defines an output data set for storing DFSORT messages. For the condition for running DFSORT and the naming rules for the DD names, see the description of *itttWKnn* DD.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

DFSPRINT DD

This DD statement is an optional DD statement. This statement specifies the output data set that contains the messages that are produced by IMS HP Image Copy. The data set can reside on a tape, a direct-access volume, or a printer, or it can be routed through the output stream (SYSOUT).

This data set is used only when you specify ICOUT=YES. If the DD statement is not coded and ICOUT=YES is specified, a SYSOUT stream is dynamically allocated for the DD.

ICEPRINT DD

This DD statement is an optional DD statement. This statement specifies the output data set that contains the reports that are produced by IMS HP Image Copy. The data set can reside on a tape, a direct-access volume, or a printer, or it can be routed through the output stream (SYSOUT).

This data set is used only when you specify ICOUT=YES. If the DD statement is not coded and ICOUT=YES is specified, a SYSOUT stream is dynamically allocated for the DD.

ICERPRT DD

This DD statement is an optional DD statement. This statement specifies the output data set that contains the Image Copy Dump Process report, which is produced by IMS HP Image Copy. It can reside on a tape, a direct-access volume, or a printer, or it can be routed through the output stream (SYSOUT).

This data set is used only when you specify ICOUT=YES.

dataout1 DD

This DD statement is an optional DD statement. This statement defines the primary image copy data set. One DD statement is required for each data set that is to be dumped. The DD name can be any 1- to 8-character string. The data set specified by the DD statement must be on either a direct-access device or a tape device.

This data set is used only when you specify ICOUT=YES. For details of the format and the characteristics of the image copy data set, see *IMS High Performance Image Copy User's Guide*. You can request IMS HP Image Copy to allocate the image copy data set dynamically by omitting this DD statement and specifying the allocation parameters that are associated with the data set in the ICEIN data set.

dataout2 DD

This DD statement is an optional DD statement. This statement defines the secondary image copy data set. This statement is required only if the associated ICEIN control statement requests the secondary image copy. The data set specified by the DD statement must be on either a direct-access device or a tape device.

This data set is used only when you specify ICOUT=YES. For details of the format and the characteristics of the image copy data set, see *IMS High Performance Image Copy User's Guide*.

Input/Output DD statements

The following input/output DD statements are used for the Reorg function.

addname/areaname DD

This DD statement is a required DD statement. This statement defines an input-and-output area data set, which is the VSAM ESDS for an area to be reorganized. The area data set contains the database records in an area. The ddname used must be the ddname for the area that is defined in the output ACB library (IMSACB DD). There can be multiple *addname* DD statements for each area included in the reorganization request.

The input-and-output area data can be allocated dynamically in one of following ways:

- The IDSNMASK keyword parameter of the REORG command
- When the GLOBAL DBRC=YES option is specified, the ADS information in the DBRC RECON data sets
- The area name member in the IMS DFSMDA library

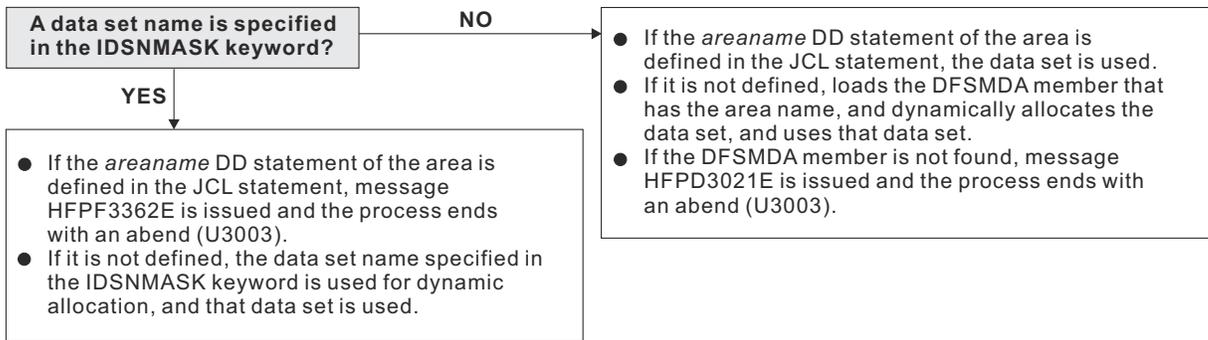
Or it can be specified with one of the following DD statements explicitly:

- *//addname* DD JCL statement
- *//areaname* DD JCL statement

If used for the REORG command input and output, it must be error free (that is, it has no EQEs). And if the area is registered with DBRC, the data set must have no EEQEs.

The following figure shows how FPA determines input-and-output data set name.

When DBRC=NO:



When DBRC=YES:

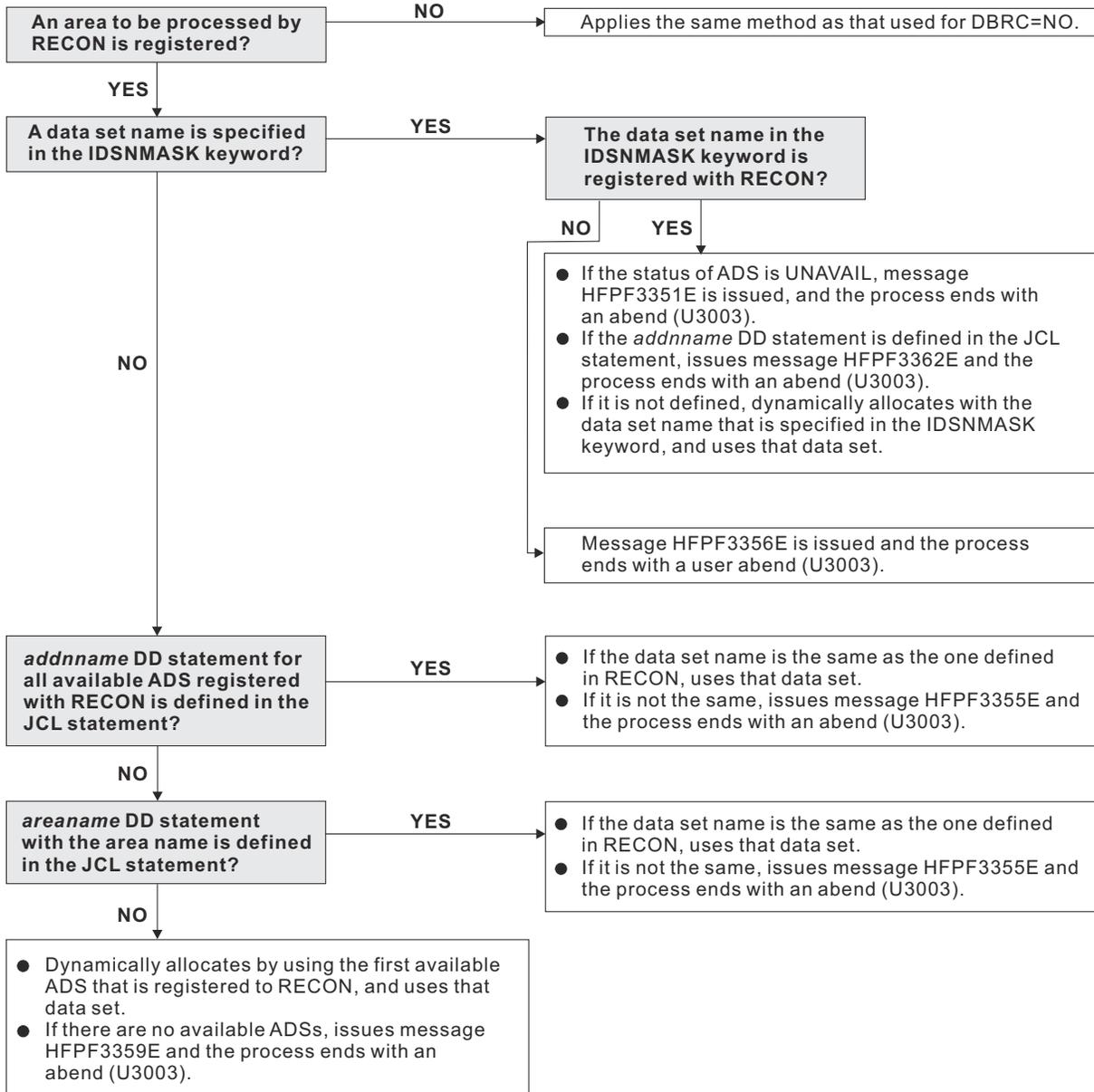


Figure 78. Input-and-output data set name for the REORG command

RECON1/RECON2/RECON3 DD

These statements define the DBRC RECON data sets. These DD statements are required DD statements when both of the following conditions are met:

- DBRC=YES is specified for the GLOBAL command
- DFSMDA members for RECON n data sets are not used

The RECON n data set must be the same data set that is used in the IMS control region. Because FPA issues a signon, a signoff, an authorization call, and an unauthorization call to DBRC, it updates RECON n data sets.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is necessary, the data set is allocated dynamically by using a DFSMDA member.

itttWKnn DD

These are optional DD statements. These are intermediate storage data sets that are used by DFSORT. For more information about coding the SORTWK nn DD statements, see *DFSORT Application Programming Guide*.

Do not specify DISP=MOD for these DD statements.

The adequate size of storage data sets is 1.5 to 2.5 times the input file size. In the analyze step, the size of each input file is as follows:

PTRCHKLVL=FULL

30 x the number of pointers in the area

SDEPCHKLVL=FULL

30 x the number of SDEP segments in the area

CISNAP=YES

112 x the number of pointers in which pointer errors are detected

REPORT subcommand and PTRCHKLVL=FULL are specified

X x (the number of non-SDEP segments in the areas + the number of free space elements in the areas + the number of references to other CIs)

Where X is:

- When LARGEREC_REPORT=NO is specified, X is calculated as follows:

34 + (the maximum segment level x 4)

However, if X is smaller than 48, 48 is used for X.

- When LARGEREC_REPORT=YES is specified, X is calculated as follows:

34 + (the maximum segment level x 4) + root key length

However, if X is smaller than 48, 48 is used for X.

REPORT subcommand and PTRCHKLVL=QUICK are specified

23 x (the number of free space elements in the area + the number of references to other CIs)

Allocating twice the space used by the SORTIN data set is usually adequate for each work data set.

The naming rules of the ddnames are as follows:

i:

A, B, and C:

If PTRCHKLVL=FULL is specified.

D, E, and F:

If SDEPCHKLVL=FULL is specified.

G, H, and I:

If CISNAP=YES is specified.

J, K, and L:

If the REPORT subcommand is specified.

The thousands digit of the area number of the area to be analyzed, or the thousands digit of n for n -th input thread subtask.

A, D, G, and J:

If the number of areas to be analyzed or n is 999 or less.

B, E, H, and K:

If the number of areas to be analyzed or n is 1000 - 1999.

C, F, I, and L:

If the number of areas to be analyzed or n is 2000 - 2048.

Note: When a subtask ends processing an area, it starts processing the next area. For this reason, the n -th input thread subtask does not always process the n -th input data set.

ttt:

The hundreds, tenths, and units digits of the area number of the area to be analyzed, or those of n .

nn:

The number of the SORT work data sets. This value must be 01 - 03.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when these data sets are necessary, FPA allocates them dynamically.

Utility control statements for the Reorg function

The utility control statements for the Reorg function define the user description of the GLOBAL command, the REORG command, and the optional LOADCTL, REPORT, and THRESHOLD subcommands, all of which control the Reorg process.

The utility control statements must be specified in the HFPSYSIN data set. This control statement data set usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte, fixed-length records. Block size, if coded, must be a multiple of 80.

The following figure shows a sample of how to code the HFPSYSIN data set.

```
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
  REORG
  DBD=dbdname
/*
```

Figure 79. REORG HFPSYSIN data set

Subsections:

- [“GLOBAL command keywords” on page 183](#)
- [“REORG command keywords” on page 184](#)
- [“LOADCTL subcommand keywords” on page 186](#)
- [“REPORT subcommand keywords” on page 186](#)
- [“THRESHOLD subcommand keywords” on page 187](#)

GLOBAL command keywords

For the GLOBAL command, you can specify the keywords that are listed in the following table. For the format of the GLOBAL command, see [“GLOBAL command” on page 394](#).

Table 26. GLOBAL command keywords

Keyword	Required or optional	Description	Topic
ADXCGRP	Optional	Specifies the name of the Autonomics Director XCF group.	“ADXCGRP keyword” on page 439

Table 26. GLOBAL command keywords (continued)

Keyword	Required or optional	Description	Topic
CISNAP	Optional	Requests the CI Map/CI Dump report and the Pointer Chain Reconstruction report.	“CISNAP keyword” on page 448
DBRC	Optional	Specifies to request or bypass the database recovery control (DBRC) facility process for the database that is specified by the DBD.	“DBRC keyword” on page 451
HISTORYDD	Optional	Specifies the DD statement for the output History file data set that is specified in the JCL.	“HISTORYDD keyword” on page 469
ICINCOMP	Optional	Specifies, when an image copy is used as an input data set, whether the image copy records are to be compressed by using the CSRCESTRV MVS macro when they are stored into a data space storage.	“ICINCOMP keyword” on page 474
ICNOTIFY	Optional	Specifies whether to issue NOTIFY.IC command to DBRC.	“ICNOTIFY keyword” on page 475
ICOUT	Optional	Specifies whether to generate one or more image copies.	“ICOUT keyword” on page 476
IMSCATACB_INPUT	Optional	Specifies whether to retrieve the currently active ACB definition or the pending ACB definition from the IMS directory.	“IMSCATACB_INPUT keyword” on page 480
IMSCATHLQ	Optional	Specifies the high-level qualifier of the bootstrap data set of the IMS directory.	“IMSCATHLQ keyword” on page 481
ITKBDATA	Optional	Specifies the type of additional data to store in the IMS Tools Base IMS Tools Knowledge Base repository.	“ITKBDATA keyword” on page 487
ITKBLOAD	Optional	Specifies the library name of IMS Tools KB.	“ITKBLOAD keyword” on page 487
ITKBSRVR	Optional	Specifies the name of the IMS Tools KB server XCF group.	“ITKBSRVR keyword” on page 488
LIUGOPT	Optional	Specifies the LIU global option module that is to be used by IMS Library Integrity Utilities.	“LIUGOPT keyword” on page 493
MSTIMESTAMP	Optional	Specifies whether to include a timestamp in the messages.	“MSTIMESTAMP keyword” on page 497
PTRCHKLVL	Optional	Specifies the method that is used to run integrity verification in the root addressable part and in the independent overflow part.	“PTRCHKLVL keyword” on page 506
SCAN	Optional	Specifies to verify the syntax of command, keywords, and parameters in command sets without running FPA. The Scan function also allocates some (but not all) internal resources for running the process function.	“SCAN keyword” on page 523
SDEPCHKLVL	Optional	Specifies the method that is used to run integrity verification in the sequential dependent part.	“SDEPCHKLVL keyword” on page 526
SDEP_ORPHAN_MSGLVL	Optional	Specifies the error message severity level to be set when an orphan SDEP segment is detected.	SDEP_ORPHAN_MSGLVL keyword
SITE_DSNAME	Optional	Specifies the data set name of the site default table.	“SITE_DSNAME keyword” on page 539
SITE_MEMBER	Optional	Specifies the member name of the site default table.	“SITE_MEMBER keyword” on page 540

REORG command keywords

For the REORG command, you can specify the keywords that are listed in the following table. For the format of the REORG command, see [“REORG command” on page 409](#).

Table 27. REORG command keywords

Keyword	Required or optional	Description	Topic
ATASKCTL	Optional	Specifies the maximum number of output areas to be analyzed concurrently.	“ATASKCTL keyword” on page 443
CISNAP	Optional	Requests the CI Map/CI Dump report and the Pointer Chain Reconstruction report when the reloaded area is analyzed concurrently.	“CISNAP keyword” on page 448
DBD	Required	Specifies the database description (DBD) that contains the areas to be processed.	“DBD keyword” on page 450
HISTORYDD	Optional	Specifies the DD statement for the output History file data set that is specified in the JCL when the reloaded area is analyzed concurrently.	“HISTORYDD keyword” on page 469
IAREA	Optional	Specifies one or more input areas for the Reorg process.	“IAREA keyword” on page 469
ICACHE	Optional	Specifies the type of control intervals that are to be preloaded into a data space storage for processing.	“ICACHE keyword” on page 472
ICNOTIFY	Optional	Specifies whether to issue the NOTIFY.IC command to DBRC.	“ICNOTIFY keyword” on page 475
ICOUT	Optional	Specifies whether to generate one or more image copies.	“ICOUT keyword” on page 476
IDSNMASK	Optional	Specifies the data set name or the mask to use when dynamically allocating cataloged pre-existing input data sets.	“IDSNMASK keyword” on page 476
ITASKCTL	Optional	Specifies the maximum number of input areas to be processed concurrently.	“ITASKCTL keyword” on page 486
MAXFAILS	Optional	Specifies the maximum allowable number of UOWs that fail to be reorganized in an area, before the utility ends abnormally.	“MAXFAILS keyword” on page 496
PTRCHKLVL	Optional	Specifies the method that is used to run integrity verification in the root addressable part and in the independent overflow part.	“PTRCHKLVL keyword” on page 506
RBASEFS	Optional	Specifies the criterion to select UOWs to reorganize.	“RBASEFS keyword” on page 509
RDOVFFS	Optional	Specifies the criterion to select UOWs to reorganize.	“RDOVFFS keyword” on page 509
RECOVOPT	Optional	Specifies the method for recovering an area when the process ends abnormally.	“RECOVOPT keyword” on page 511
REORGLIM	Optional	Specifies the reorganization limit for the amount of UOWs that can be reorganized in the area.	“REORGLIM keyword” on page 513
RFSSCAN	Optional	Specifies whether to gather the statistics before and after reorganization.	“RFSSCAN keyword” on page 519
RFSSTAT	Optional	Specifies whether to generate the UOW Space Statistics report before and after reorganization.	“RFSSTAT keyword” on page 520
SDEPCHKLVL	Optional	Specifies the method that is used to run integrity verification in the sequential dependent part.	“SDEPCHKLVL keyword” on page 526
SDEP_ORPHAN_MSGLVL	Optional	Specifies the error message severity level to be set when the reloaded area is analyzed concurrently and an orphan SDEP segment is detected.	SDEP_ORPHAN_MSGLVL keyword
SDEP_PTRIOVF_MSGLVL	Optional	Specifies the severity level of the message that is issued when detecting an SDEP pointer that points to the IOVF portion.	SDEP_PTRIOVF_MSGLVL keyword
SENSOR	Optional	Specifies whether to store the sensor data in the Sensor Data repository of IMS Tools KB.	“SENSOR keyword” on page 535
SENSOR_DBREC	Optional	Specifies whether to store the data elements that are related to database records and root segment distribution in the Sensor Data repository of IMS Tools KB.	“SENSOR_DBREC keyword” on page 536

Table 27. REORG command keywords (continued)

Keyword	Required or optional	Description	Topic
SENSOR_PTRERR	Optional	Specifies to store the sensor data in the Sensor Data repository of IMS Tools KB even when pointer errors are detected in the area.	“SENSOR_PTRERR keyword” on page 537
SENSOR_UOW	Optional	Specifies whether to store the data elements that are related to UOW in the Sensor Data repository of IMS Tools KB.	“SENSOR_UOW keyword” on page 537
SENSOR_UOW_REPORT	Optional	Specifies whether to print the data elements that are related to UOW in the Sensor Data Statistics report.	SENSOR_UOW_REPORT keyword
SORTOPT_FILSZ	Optional	Specifies that FPA adds the FILSZ parameter to the SORT control statement.	“SORTOPT_FILSZ keyword” on page 542

LOADCTL subcommand keywords

The LOADCTL subcommand specifies the load control information for a segment. You can specify an Insert Limit Count (ILC) and a place to insert dependent segments (either DOVF or IOVF). ILC is the maximum number of occurrences of segments that are inserted close to their root.

If the LOADCTL subcommand is specified multiple times for the same segment, they will be checked from the first to the last, and the first LOADCTL statement that matches the condition will be used for segment insertion.

For the LOADCTL subcommand, you can specify the keywords that are listed in the following table. For the format of the LOADCTL subcommand, see [“LOADCTL subcommand” on page 429](#).

Table 28. LOADCTL subcommand keywords

Keyword	Required or optional	Description	Topic
INSERT_LIMIT_COUNT	Optional	Specifies the maximum number of segments that are selected by the SEGMENT keyword, that can be inserted close to their root segment. The maximum value is 32767 and the minimum value is 0. The minimum value 0 specifies that all specified segments are stored far from their root segment.	INSERT_LIMIT_COUNT keyword
LOCATION	Optional	Specifies the location to insert dependent segments, either DOVF or IOVF. You can specify the location where segments that were selected by the SEGMENT keyword are loaded.	“LOCATION keyword” on page 494
SEGMENT	Required	Specifies the segment name to be inserted at the specified location. The segment name must be a name that is coded on the SEGM statement in your DBD.	“SEGMENT keyword” on page 530

REPORT subcommand keywords

The REPORT subcommand requests the analysis reports. If you do not specify PTRCHKLVL=FULL or PTRCHKLVL=QUICK, this subcommand is ignored.

For the REPORT subcommand, you can specify the keywords that are listed in the following table. For the format of the REPORT subcommand, see [“REPORT subcommand” on page 432](#).

Table 29. REPORT subcommand keywords

Keyword	Required or optional	Description	Topic
DBRLDIST	Optional	Specifies whether to generate the DB Record Length Distribution report.	“DBRLDIST keyword” on page 452
IOVFCI_RBA_REPORT	Optional	Specifies whether to generate the RBA of Used IOVF CI report.	“IOVFCI_RBA_REPORT keyword” on page 484

Table 29. REPORT subcommand keywords (continued)

Keyword	Required or optional	Description	Topic
LARGEREC_COUNT	Optional	Specifies the number of reporting entries in the Largest Database Records report and the Consolidated Largest Database Records report. This keyword is effective only when PTRCHKLVL=FULL is specified.	“LARGEREC_COUNT keyword” on page 491
LARGEREC_REPORT	Optional	Specifies whether to generate the Largest Database Records report. This keyword is effective only when PTRCHKLVL=FULL is specified.	“LARGEREC_REPORT keyword” on page 491
REPORTDD	Optional	Specifies the destination ddname for the analysis reports.	“REPORTDD keyword” on page 513
SEGLDIST	Optional	Specifies whether to generate the Segment Length Distribution report.	“SEGLDIST keyword” on page 529
SEGLSTAT	Optional	Specifies the format of the DB Record Profile Analysis report, which is one of the DEDB Area Analysis reports.	“SEGLSTAT keyword” on page 529
STARTUOW	Optional	Specifies the number that defines the first UOW to be reported in the UOW report and the UOW Exception report.	“STARTUOW keyword” on page 545
STOPUOW	Optional	Specifies the number that defines the last UOW to be reported in the UOW report and the UOW Exception report.	“STOPUOW keyword” on page 546
UOW_REPORT	Optional	Specifies whether to generate the UOW report.	“UOW_REPORT keyword” on page 555

THRESHOLD subcommand keywords

The THRESHOLD subcommand specifies the threshold parameter values, which cause warning messages to be generated when the actual values do not meet the specified criteria. If you do not specify the REPORT subcommand, this subcommand is ignored.

For the THRESHOLD subcommand, you can specify the keywords that are listed in the following table. For the format of the THRESHOLD subcommand, see [“THRESHOLD subcommand” on page 435](#).

Table 30. THRESHOLD subcommand keywords

Keyword	Required or optional	Description	Topic
DOVFFS_PCT	Optional	Specifies a threshold setting for the percentage of free space in DOVF. If the percentage of free space in DOVF is less than the value that you specified, a warning message is issued.	“DOVFFS_PCT keyword” on page 455
FS_DOVFIOVF	Optional	Specifies a threshold setting for the free space in DOVF and in IOVF. If the free space in DOVF is greater than the value that you specified, and if the free space in IOVF is less than the value that you specified, a warning message is issued.	“FS_DOVFIOVF keyword” on page 466
FS_RAADOVF	Optional	Specifies a threshold setting for the free space in RAA BASE and in DOVF. If the free space in RAA BASE is greater than the value that you specified, and if the free space in DOVF is less than the value that you specified, a warning message is issued.	“FS_RAADOVF keyword” on page 467
FS_RAADOVFIOVF	Optional	Specifies a threshold setting for the free space in RAA BASE, DOVF, and IOVF. If the free space in RAA BASE is greater than the value that you specified, and if the free space in either DOVF or IOVF is less than the value that you specified, a warning message is issued.	“FS_RAADOVFIOVF keyword” on page 467
FS_RAAIOVF	Optional	Specifies a threshold setting for the free space in RAA BASE and in IOVF. If the free space in RAA BASE is greater than the value that you specified, and if the free space in IOVF is less than the value that you specified, a warning message is issued.	“FS_RAAIOVF keyword” on page 468

Table 30. THRESHOLD subcommand keywords (continued)

Keyword	Required or optional	Description	Topic
IOVFFS_PCT	Optional	Specifies a threshold setting for the percentage of free space in IOVF. If the percentage is less than value that you specified, a warning message is issued.	“IOVFFS_PCT keyword” on page 485
IOVFUSED_PCT	Optional	Specifies a threshold setting for the percentage of IOVF CIs used. If the percentage is greater than the value that you specified, a warning message is issued.	“IOVFUSED_PCT keyword” on page 485
RAAFS_PCT	Optional	Specifies a threshold setting for the percentage of free space in RAA BASE. If the percentage is less than the value that you specified, a warning message is issued.	“RAAFS_PCT keyword” on page 507
RAPOVFL_PCT	Optional	Specifies a threshold setting for the percentage of RAA BASE CIs that use either DOVF or IOVF. If the percentage is greater than the value that you specified, a warning message is issued.	“RAPOVFL_PCT keyword” on page 508
RECIO_AVG	Optional	Specifies a threshold setting for the average number of I/Os that can be done to access a database record. If the number is greater than the value that you specified, a warning message is issued.	“RECIO_AVG keyword” on page 510
RECIO_MAX	Optional	Specifies a threshold setting for the maximum number of I/Os that can be done to access a database record. If the number is greater than the value that you specified, a warning message is issued.	“RECIO_MAX keyword” on page 511
RECIOVF_PCT	Optional	Specifies a threshold setting for the percentage of records using IOVF CIs. If the percentage is greater than the value that you specified, a warning message is issued.	“RECIOVF_PCT keyword” on page 510
ROOTIO_AVG	Optional	Specifies a threshold setting for the average number of I/Os to access a root segment. If the number is greater than the value that you specified, a warning message is issued.	“ROOTIO_AVG keyword” on page 522
ROOTIO_MAX	Optional	Specifies a threshold setting for the maximum number of I/Os that can be done to access a root segment. If the number is greater than the value that you specified, a warning message is issued.	“ROOTIO_MAX keyword” on page 523
SYN_LEN_AVG	Optional	Specifies a threshold setting for the average synonym chain length. If the length is greater than the value that you specified, a warning message is issued.	“SYN_LEN_AVG keyword” on page 547
SYN_MAX	Optional	Specifies a threshold setting for the maximum synonym chain length. If the length is greater than the value that you specified, a warning message is issued.	“SYN_MAX keyword” on page 548
SYN_PCT_LEN	Optional	Specifies a threshold setting for the percentage of RAPs that have a long synonym chain among all the RAPs that have synonyms. If the percentage of the RAPs that have a long synonym chain is greater than the threshold value, a warning message is issued.	“SYN_PCT_LEN keyword” on page 548
UOWDOVF_PCT	Optional	Specifies a threshold setting for the percentage of UOWs that use DOVF. If the percentage is greater than the value that you specified, a warning message is issued.	“UOWDOVF_PCT keyword” on page 553
UOWIOVF_AVG	Optional	Specifies a threshold setting for the average number of IOVF CIs that are used by a UOW. If the number is greater than the value that you specified, a warning message is issued.	“UOWIOVF_AVG keyword” on page 554
UOWIOVF_MAX	Optional	Specifies a threshold setting for the maximum number of IOVF CIs that are used by a UOW. If the number is greater than the value that you specified, a warning message is issued.	“UOWIOVF_MAX keyword” on page 554
UOWIOVF_PCT	Optional	Specifies a threshold setting for the percentage of UOWs that use IOVF. If the percentage is greater than the value that you specified, a warning message is issued.	“UOWIOVF_PCT keyword” on page 555

Examples for the Reorg function

Review examples and learn techniques for reorganizing DEDB areas.

In these examples, the following DEDB is assumed:

```

DEDB name:  DEDBJN22 has four areas
  area name:  DB22AR0 (area number is 1 and has three ADSs)
    ADS DD name:  DB22AR01,      DS name:  HPFP.DB22AR0.ADS1
  area name:  DB22AR1 (area number is 2 and has two ADSs)
    ADS DD name:  DB22AR11,     DS name:  HPFP.DB22AR1.ADS1
  area name:  DB22AR2 (area number is 3 and has two ADSs)
    ADS DD name:  DB22AR21,     DS name:  HPFP.DB22AR2.ADS1
  area name:  DB22AR3 (area number is 4 and has one ADS)
    ADS DD name:  DB22AR31,     DS name:  HPFP.DB22AR3.ADS1

```

In DBRC, all ADSs are in the AVAILABLE status.

You can find additional JCL examples that help you to monitor and tune DEDB areas in [“FPA Reorg JCL examples for monitoring and tuning DEDBs”](#) on page 1148.

Example 1: Reorganizing all UOWs in an area

The following figure shows example JCL for reorganizing all UOWs in an area.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input-and-output ADS that is to be reorganized is dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC.
- Because RFSSCAN=(NO,NO) is specified, all UOWs are reorganized unconditionally.
- After reorganizing, the IC NEEDED status of area DB22AR0 is changed to ON.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//         DD DISP=SHR,DSN=IMSVS.SDFSRESL
//         DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
REORG
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  RFSSCAN=(NO,NO)
/*

```

Figure 80. Reorganizing all UOWs in an area

Example 2: Reorganizing UOWs that use IOVF CIs and generating an image copy

The following figure shows example JCL for reorganizing UOWs that use IOVF CIs and generating an image copy.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input-and-output ADS that is to be reorganized is dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC.
- Because RDOVFFS=0 is specified, the UOWs that use IOVF CIs are reorganized.
- After reorganizing, the image copy data set for the area is dynamically allocated by IMS HP Image Copy. The data set name of the image copy is determined by IMS HP Image Copy as follows:

```
ICOUT.HFP.IC1.DEDBJN22.DB22AR0
```

For more details, see *IMS High Performance Image Copy User's Guide*.

- After creating the image copy, a NOTIFY.IC command is issued to DBRC.

```
//HFP EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=HPS.SHPSLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
  REORG
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  RDOVFFS=0,
  ICOUT=YES,
  ICNOTIFY=YES
/*
//ICEIN DD *
GLOBAL ICHLQ=ICOUT.HFP,
  UNIT=SYSDA,
  SPACE=(CYL,100,100)
/*
//ICERPRT DD SYSOUT=*
```

Figure 81. Reorganizing UOWs that use IOVF CIs and generating an image copy

Example 3: Reorganizing some UOWs that satisfy one of the RBASEFS and the RDOVFFS conditions

The following figure shows example JCL for reorganizing some UOWs that satisfy one of the RBASEFS and the RDOVFFS conditions.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input-and-output ADS that is to be reorganized is dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC.
- The UOWs that are to be reorganized are limited to the subset of UOWs that satisfy one of the RBASEFS and RDOVFFS conditions. Then, UOWs in the subset are ranked on the basis of statistics gathered by the Area Sensor.
- Because REORGLIM=20 is specified, higher ranked 20 UOWs are selected from the ranked subset of UOWs, and those selected UOWs are reorganized.
- After reorganizing, the IC NEEDED status of area DB22AR0 is changed to ON.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//         DD DISP=SHR,DSN=IMSVS.SDFSRESL
//         DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
REORG
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  RBASEFS=20,
  RDOVFFS=20,
  REORGLIM=20
/*

```

Figure 82. Reorganizing some UOWs that satisfy one of RBASEFS or RDOVFFS

Example 4: Reorganizing an area, analyzing the reorganized area, and storing the sensor data in the IMS Tools KB repository

The following figure shows example JCL for reorganizing several UOWs that satisfy one of the RBASEFS and the RDOVFFS conditions. When the UOWs are reorganized, the job analyzes the reorganized area and stores the sensor data in the Sensor Data repository of IMS Tools KB.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input-and-output ADS that will be reorganized is dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC.
- The UOWs that will be reorganized are limited to the subset of UOWs that satisfy one of the RBASEFS and the RDOVFFS conditions. Then, UOWs in the subset are ranked on the basis of statistics gathered by the Area Sensor.
- After the reorganization process is complete, the IC NEEDED status of area DB22AR0 is changed to ON.
- After the reorganization process is complete, the reorganized area DB22AR0 is analyzed with the full check option, and sensor data is collected and stored in the Sensor Data repository of IMS Tools KB. In addition to the default data elements that are collected by the SENSOR keyword, the data elements that are related to database records and root segment distribution are stored (SENSOR_DBREC=YES specification).

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//          DD DISP=SHR,DSN=ITB.SHKTLOAD
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSRPT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES,
  ITKBSRVR=FPQSRV01,
  ADXCFGRP=ADSRV01
REORG
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  RFSSCAN=(NO,NO),
  RBASEFS=20,
  RDOVFFS=20,
  PTRCHKLVL=FULL,
  SENSOR=YES,
  SENSOR_DBREC=YES
/*

```

Figure 83. Reorganizing an area, analyzing the reorganized area, and storing the sensor data in the IMS Tools KB repository

Chapter 8. Analyzing a DEDB offline with FPA

You can analyze a DEDB by using the FPA Analyze function.

The three Pointer Checkers of IMS HP Fast Path Utilities

There are three distinct pointer checker products. Therefore, before getting started, it is important to read this topic in order to avoid confusion later on.

This topic describes the relationship among the following three Pointer Checkers:

- Analyze function of FPA
- DEDB Pointer Checker of FPB
- Online Pointer Checker of FPO

The Analyze function of FPA and DEDB Pointer Checker of FPB run in the offline environment and Online Pointer Checker of FPO runs in the online environment.

The Analyze function of FPA is part of the IMS HP Fast Path Utilities package. It has the function similar to that of DEDB Pointer Checker, which runs while the database area is offline. FPA verifies the integrity of all IMS pointer values and verifies the integrity of the pointer segments in secondary index databases. It also produces complete analysis reports in one step, whereas DEDB Pointer Checker runs the same functions in multiple steps.

DEDB Pointer Checker of FPB is also part of the IMS HP Fast Path Utilities package. It uses program FABADA1 to collect pointer verification data. This program must be run while the database area is offline. The collected data can then be passed to the pointer integrity checking programs and various report generating programs in this package.

Online Pointer Checker of FPO is also part of the IMS HP Fast Path Utilities package. It runs as a Fast Path utility to verify the integrity of all IMS pointer values and produce complete analysis reports in one step while the database area is online. However, you can use OPC to run a fast scan of database integrity and collect pointer verification data from an online DEDB area, then pass it to DEDB Pointer Checker of FPB to check cross-reference of pointers and produce complete analysis reports. This type reduces the online run time but must be run in multiple steps.

Topics:

- [“Functions of Analyze” on page 194](#)
- [“Input and output for the Analyze function” on page 196](#)
- [“Running the Analyze function” on page 197](#)
- [“Region size requirements for the Analyze function” on page 200](#)
- [“DD statements for the Analyze function” on page 203](#)
- [“Utility control statements for the Analyze function” on page 218](#)
- [“Examples for the Analyze function” on page 223](#)

Related information

[Analyzing a DEDB online with FPO](#)

You can analyze an online DEDB by using the Online Pointer Checker utility (also referred to as OPC) of FPO.

Functions of Analyze

The Analyze function provides a fast, efficient mechanism for verifying the integrity of all IMS pointer values, free space element chains, VSAM control fields, and space utilization in IMS DEDBs.

The Analyze function provides a capability for verifying the integrity of pointer segments in all secondary index databases.

The Analyze function provides the comprehensive analysis information required to effectively manage the space utilization, performance characteristics, and physical attributes of IMS DEDBs. The Analyze function uses a VSAM ESDS or an image copy as input. The Analyze function can be an aid for database administrators and technical support personnel in database management and maintenance.

DEDB integrity verification

The Analyze function provides complete DEDB integrity verification.

Specifically, the following areas of integrity are addressed:

- All of the following pointer values are verified to ensure that they refer to valid segments:
 - Root Anchor Point (RAP)
 - Physical Twin Forward (PTF)
 - Physical Child First (PCF)
 - Physical Child Last (PCL)
 - Subset (SSP)
 - Sequential Dependent (SDEP)

All orphan segments, incorrect pointer values, and broken pointer chains are detected and reported. PCF/PCL/SSP pointer interdependency errors are also detected and reported.

- All free space element (FSE) chains are checked for validity.
- Space utilization is verified by totalizing the lengths of all segments, FSEs, and scraps within a CI, and ensuring that the calculated total equals the usable CI size (CI size minus VSAM and IMS overhead). This process will detect FSEs within segments, or segments within FSEs.
- The VSAM CIDF and RDF fields are also verified to ensure that their values are correct. The Analyze function will report the RBA of any incorrect CIs that are encountered and continue processing.
- The Analyze function provides the following reports that can be used to determine the best way to repair incorrect CIs:
 - CI Map/CI Dump report for incorrect CIs
 - Pointer Chain Reconstruction report

The Analyze function runs the integrity verification in two different operating modes. These modes are determined according to the specification of the PTRCHKLVL keyword parameter.

QUICK mode

The Analyze function runs the integrity verification in the QUICK mode when PTRCHKLVL=QUICK is specified. The QUICK mode provides a fast scan of database integrity. The Analyze function can detect and report the exact RBA of the following integrity exposures:

- Incorrect block type (DBLKBTID)
- Incorrect segment codes
- Incorrect FSE in the FSE chain

- Number or length of FSEs found during serial deblocking does not equal that tallied by chasing the FSE chain
- CI space utilization discrepancies (that is, the total length of all segments, FSEs, and scraps plus IMS and VSAM overhead does not equal CI size)
- The following pointer interdependency problems:
 - PCF pointer nonzero and PCL pointer zero
 - PCF pointer zero and PCL or Subset pointer nonzero
- SDEP pointers that contain a pointer value that is outside the sequential dependent part of the area
- Logical end (LE) in the DMAC is less than the logical beginning (LB)
- Incorrect VSAM control fields (CIDF, RDF) when the input data set is the area ESDS

The Analyze function also does a checksum verification of the following pointers versus the appropriate segment RBAs:

- PCF/PTF versus appropriate segment RBAs
- PCL versus appropriate segment RBAs
- SDEP pointer values and SDEP segments (between LB and LE)

The checksum verification process can detect the presence of incorrect pointers but cannot report the exact RBA of the problem. Checksum verification cannot be done for subset pointers.

FULL mode

The Analyze function runs the integrity verification in the FULL mode when PTRCHKLVL=FULL is specified. The FULL mode provides in-depth pointer verification. Besides the QUICK mode integrity verification functions, the Analyze function generates pointer and segment information entities for each pointer and segment to carry out in-depth pointer analysis through a matching process. This process reports the exact RBAs of the following pointer problems:

- Orphan segments (that is, segments that are not pointed to by any correct pointer)
- Pointers that reference nonexistent segments
- Pointers to wrong segment types (that is, the Analyze function verifies that the pointer points to the correct segment type)
- Segments pointed to by multiple parents
- Segments that cannot be resolved back to a correct RAP RBA

Subset pointer validation can be done only in this verification.

DEDB analysis and report

The complete analysis reports are provided to effectively manage the space utilization, performance characteristics, and physical attributes of IMS DEDBs. It provides integrity verification reports, and summary and detailed analysis reports for database administrators and technical support personnel in database management, maintenance, and performance tuning.

The statistical information includes:

- Free space analysis, including statistics on the number and size of the free space elements and inter-database section references.
- Database record and segment occurrence profiles.
- Database record and segment placement analysis.
- Complete overflow usage analysis of both dependent overflow (DOVF) and independent overflow (IOVF).
- Physical I/O statistics for each segment type and for the typical database record.
- Root distribution and synonym chain analysis.

- Optional automated threshold reporting, which generates warning messages when conditions within the database do not meet user-specified criteria. This feature can be used to show when reorganization or expansion is required, or if database performance falls below acceptable standards.
- Optional generation of:
 - A report that describes the largest database records.
 - A report that identifies the units of work (UOWs) that exceed a user-specified reorganization criteria.
 - The control statements that are required to run the IMS High-Speed DEDB Direct Reorganization utility (DBFUMDR0).

Reorganizing specific UOWs can significantly reduce the time and resources that are required for database maintenance.

Secondary index database integrity verification

The Analyze function provides simple secondary index database integrity verification by comparing the data in DEDB areas. It verifies DEDB areas and their secondary index databases at the same time.

Note: To verify the integrity of the secondary index databases, FULL mode (PTRCHKLVL=FULL keyword) must be specified.

The following pointer segment data is verified:

- Search field
- Subsequence field
- Duplicate data field
- Concatenated key field

For information about these fields, see *IMS Database Definition*.

The Analyze function detects the following errors and issues error messages:

- When there is no index pointer segment that points to the corresponding target segment.
- When there is no index target segment that is pointed by the corresponding index pointer segment.

Input and output for the Analyze function

You can invoke the Analyze function by several methods to analyze multiple DEDB areas and multiple secondary index databases.

To verify the integrity of all IMS pointer values, free space element chains, VSAM control fields, and space utilization in IMS DEDBs, the Analyze function is invoked by one of the following methods:

- The ANALYZE command
- The PTRCHKLVL=FULL keyword with the CHANGE, RELOAD, or REORG command in the HFPSYSIN DD control statement
- The PTRCHKLVL=QUICK keyword with the CHANGE, RELOAD, or REORG command in the HFPSYSIN DD control statement

To verify the integrity of pointer segments in secondary index databases, the names of the secondary index databases must be specified on the INDEXDBD keyword and the PTRCHKLVL=FULL keyword must be specified for the Analyze function.

The SORT program is invoked internally to achieve better SORT performance by eliminating the SORTIN and the SORTOUT file I/O operation. When the Analyze process is completed, IMS HP Image Copy can be invoked internally to create the image copies.

The following figure shows the input and the output for the Analyze function.

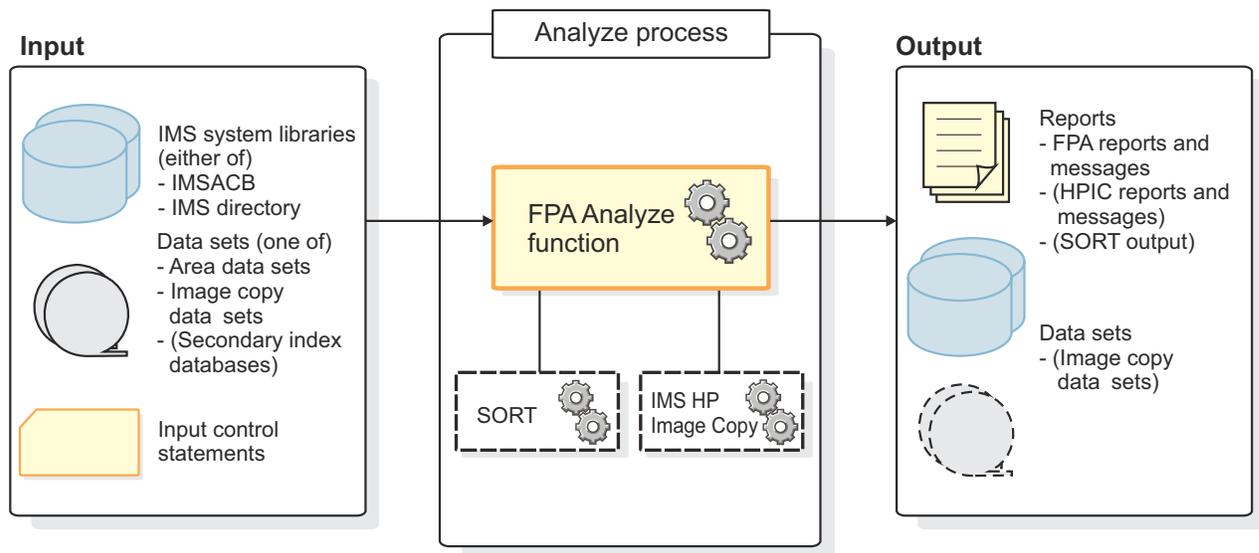


Figure 84. Input and output for the Analyze function

The Analyze function can read and analyze the following image copies of a DEDB:

- Standard image copies that are created by IMS Image Copy utility (DFSUDMP0) or IMS HP Image Copy
- Compressed image copies that are created by IMS HP Image Copy

When image copies are provided as input data sets for the Analyze function, new image copies cannot be created after the completion of the Analyze process.

Running the Analyze function

You can run the Analyze function to perform complete DEDB integrity verification and generate the complete analysis reports. You can also use this function to correct integrity problems that were detected by a DEDB integrity verification process. The reports generated by the Analyze function can be used to effectively manage IMS DEDBs.

About this task

There are four main steps in running the Analyze process.

1. Allocate and initialize the permanent data set required by FPA. This is a preprocess to run the Analyze process of FPA, and is done only once.
2. Run the Analyze process.
3. Correct integrity problems, if there are any.
4. If necessary, analyze the analysis reports and tune your environment.

Preprocess for the Analyze process: Initializing permanent data set

If you specify the REPORT subcommand and the PTRCHKLVL=FULL keyword for an Analyze process, you can maintain the history of your Analyze process results by defining a Statistics History data set or a Statistics History2 data set. If you want to maintain the history, you must allocate and initialize a Statistics History data set or a Statistics History2 data set.

About this task

Use of the Statistics History data set and the Statistics History2 data set is optional. If you do not intend to maintain a history of your Analyze process results, you do not need to allocate and initialize a Statistics History data set or a Statistics History2 data set; you can skip this preprocess.

Procedure

The Statistics History data set is used only when both the REPORT subcommand and the PTRCHKLVL=FULL keyword are specified, and either of the following specifications is present:

- The HISTORYDD keyword with a DD data set name as its parameter
- The HFFAHST DD statement

Unless you specify a DD data set name by using the HISTORYDD keyword, the Statistics History data set is automatically defined by the HFFAHST DD statement. When a Statistics History data set is defined for an Analyze process by either of these specifications, the Statistics History data set is updated every time the process is run. Each Analyze process adds one record for each DEDB area that it processes, which contains the key space utilization and performance information for that area.

The Statistics History2 data set is also used only when both the REPORT subcommand and the PTRCHKLVL=FULL keyword are specified and the HFFAHST2 DD statement is present. When a Statistics History2 data set is defined for an Analyze process, the Statistics History2 data set is updated every time the process is run. Each Analyze process adds one record for each UOW group in area that it processes, which contains the key space utilization and performance information for that UOW group. The UOW groups are specified by HFFAUOWC DD.

For the Statistics History data set and the Statistics History2 data set, you must use DISP=(MOD,KEEP,KEEP).

Example

The following figure shows the JCL that you can use to allocate and initialize a Statistics History data set and a Statistics History2 data set.

```
//*****  
//** ALLOCATE AND INITIALIZE HISTORY DATA SET  
//*****  
//IDCAMS EXEC PGM=IDCAMS  
//SYSPRINT DD SYSOUT=A  
//SYSIN DD *  
    DELETE ( HFFP.STAT.HIST ) NONVSAM  
    DELETE ( HFFP.STAT.HIST2 ) NONVSAM  
    SET MAXCC = 0  
/*  
//*****  
//** ALLOCATE STATISTICS HISTORY DATA SET  
//** ALLOCATE STATISTICS HISTORY2 DATA SET  
//*****  
//IEFBR14 EXEC PGM=IEFBR14  
//DASDHIO DD DSN=HFFP.STAT.HIST,DISP=(NEW,CATLG,CATLG),  
//          DCB=(DSORG=PS,RECFM=FB,LRECL=100,BLKSIZE=800),  
//          UNIT=SYSDA,SPACE=(CYL,(2,2),RLSE)  
//DASDHIO2 DD DSN=HFFP.STAT.HIST2,DISP=(NEW,CATLG,CATLG),  
//          DCB=(DSORG=PS,RECFM=FB,LRECL=200,BLKSIZE=1600),  
//          UNIT=SYSDA,SPACE=(CYL,(4,4),RLSE)
```

Figure 85. Sample JCL to initialize permanent data set (History data sets)

Running the Analyze process

You must specify an EXEC statement and DD statements that define the input and output data sets in your JCL.

Procedure

1. Optional: Estimate the region size that is required to run the job.

See [“Region size requirements for the Analyze function”](#) on page 200.

2. Specify the EXEC statement in the following format:

```
//ANALYZE EXEC PGM=HFPMAIN0,REGION=rrrrM,
//          PARM='DBRCGRP=dbrcgrp,GSGNAME=gsgname,IMSPLEX=impsplex,
//          IMSID=imsid'
```

Parameter	Description
DBRCGRP=	DBRCGRP is an optional 1- to 3-character parameter that specifies the identifier assigned to a group of DBRC instances (DBRC sharing group) that access the same RECON data set in the IMSplex.
GSGNAME=	GSGNAME is an optional 1- to 8-character parameter that specifies the global service group.
IMSPLEX=	IMSPLEX is an optional 1- to 5-character parameter that specifies the names of the IMSplex groups that share databases or message queues.
IMSID=	IMSID is an optional 1- to 8-character parameter that specifies the IMS system identification name. If you specify this parameter, FPA generates the buffers for temporary ESCD and SCD. When FPA issues a call to a randomizing module, register 11 contains the address of the temporary ESCD. FPA sets only the ESCDSCD field, which contains the address of SCD, in the temporary ESCD. Also, FPA sets only the SSCDIMID field, which contains the IMS system ID, in the temporary SCD. Other fields of temporary ESCD and temporary SCD cannot be referred to.

3. Specify the DD statements to define input data sets and output data sets.

4. Run the JCL job.

Example

The following figure shows a JCL example for the ANALYZE command.

```
//*****
//** ANALYZE THE AREA(S) **
//*****
//ANALYZE EXEC PGM=HFPMAIN0
//*
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//HFPPRINT DD SYSOUT=A
//HFPRPTS DD SYSOUT=A
//*
//HFPSYSIN DD *
//          GLOBAL DBRC=NO
//          ANALYZE
//          DBD=IVPDB3
//          *
//DFSIVD3A DD DISP=SHR,DSN=IMSVS.DFSIVD31
//DFSIVD3B DD DISP=SHR,DSN=IMSVS.DFSIVD33
//          *
```

Figure 86. JCL example the ANALYZE command

Related concepts

[Region size requirements for the Analyze function](#)

The Analyze function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Related reference

[Examples for the Analyze function](#)

Review examples and learn techniques for analyzing DEDB areas.

Tips and best practices for the FPA Analyze function

The FPA Analyze function supports several keywords that improve the performance of FPA jobs and that further simplify management of DEDBs.

Correcting integrity problems in DEDB areas

The integrity problem messages always provide the exact RBAs (addresses) of the pointer or integrity discrepancies.

Procedure

To correct integrity problems, follow these steps:

1. Run the Analyze function with PTRCHKLVL=FULL and CISNAP=YES to obtain a CI SNAP Dump.
2. Analyze the results from the reports. Verify the CI content and determine the appropriate corrections.
3. Repair the DEDB area. IMS Database Repair Facility provides a convenient tool for printing and changing the contents of a CI without modifying the Format 1 DSCB in the VTOC.
4. Rerun the Analyze function to ensure that the problem is fixed. Optionally, you can use the image copy option to make a backup of the repaired DEDB area.
5. If desired, run the IDCAMS program to print the area for additional information.

Correcting integrity problems in secondary index databases

The Secondary Index Analysis report provides information about the pointer segments and the source segments of the processed secondary index databases. When integrity problems were found, this report also identifies the databases where the problems were found.

Procedure

To correct integrity problems, follow these steps:

1. From the Secondary Index Analysis report, identify the databases where the problems were found.
2. Run the Build Index function to rebuild the databases that were identified in step “1” on [page 200](#).
3. Rerun the Analyze function to ensure that the problem is fixed.

Analyzing the complete analysis reports

The complete analysis reports can be used to effectively manage the space utilization, performance characteristics, and physical attributes of IMS DEDBs.

Related information

DEDB monitoring and tuning guide

You can use the utilities of IMS Fast Path Solution Pack to collect statistics data from DEDB areas, be notified of exceptional states of areas, and generate statistical reports to tune the areas.

Region size requirements for the Analyze function

The Analyze function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Region size for the Analyze function depends on the size of a UOW, the number of areas being analyzed, and the ATASKCTL specifications. Typically, database maintenance is a relatively important function. To increase the performance of the analyze program, it would be advisable to run it in a region that has a relatively high dispatching priority (that is, at least equal to the dispatching priority of a message processing region).

Estimating the region size for Analyze

The region size required for the Analyze function is a function of the size of a UOW, the number of areas being analyzed, and the ATASKCTL specifications.

In most cases, the specification on the EXEC statement for HFPMAIN0 of a region that is too small, is detected during the initialization processing of the main task. This causes processing to be terminated at that point. Under certain conditions, the region size shortfall does not become evident until the actual analyze has begun. The job must be completely rerun. Thus, the region size calculations must always be performed prior to initiating analyze processing.

If the following specifications are made, the sort program also requires storage region. For how to estimate the region size, see the *DFSORT Application Programming Guide*.

- PTRCHKLVL=FULL
- SDEPCHKLVL=FULL
- CISNAP=YES
- REPORT subcommand is specified

The following worksheet is for estimating the region size that is required for a typical analyze job. The actual region size depends on the keywords that are specified for the job. Therefore, when you specify the region size in your JCL, use a higher value than the value calculated from this worksheet.

FPA programs, internal tables, control blocks	3,020,000	
Number of areas defined in DBD * 1,300	-----	
Maximum Segment length * 1,100	-----	
Internal tables, control blocks	93,000	
Storage used by Sort program	-----	(If PTRCHKLVL=FULL)
(Maximum segment code + 1) * 350	-----	(If PTRCHKLVL=FULL)
Storage used by Sort program	-----	(If SDEPCHKLVL=FULL)
subtotal	-----	
Value specified on ATASKCTL * subtotal	-----	
Number of analyzed areas * 4,000	-----	
Total of CISZ * UOW1 of each area * 2	-----	
Total of CI size of each area	-----	
Total of X * (126 + MAX Segment Level * 4)	-----	(If PTRCHKLVL=FULL)
X : Total of maximum number of (RAP/PTF/PCF)		
per one UOW in each area		
Internal tables, control blocks	50,000	(If CISNAP=YES)
Storage used by Sort program	-----	(If CISNAP=YES)
CI size of the area where pointer errors are detected	-----	(If CISNAP=YES)
subtotal	-----	(If CISNAP=YES)
Number of areas where pointer errors are detected * subtotal	-----	(If CISNAP=YES)
Storage used by Sort program	-----	(If REPORT subcommand is specified)
Maximum number of UOW in all areas * 100	-----	(If REPORT subcommand is specified)
Storage used to generate RBA of Used IOVF CI report	-----	(If IOVF_CI_RBA_REPORT=YES is specified)
IOVF CI# = ROOT2 * UOW1	(A)-----	
IOVF SMP CI# = A / 120	(B)-----	
QHDR size=(ROOT1 - ROOT2) * 8	(C)-----	
SR buffer Pool=A * (8 +64)	(D)-----	
B * 4 * 119 + C + D	-----	
Total bytes of analyze storage	-----	

Figure 87. Worksheet for estimating the region size for the Analyze function

If ICACHE=YES is specified and if an image copy data set is used as the input data, the Analyze function preloads all image copy records into the data space. The data space size required for the Analyze function is a total of the size of the original area data set.

If ATASKCTL is specified, the data space size is smaller than that estimated by the following worksheet because a task reuses a data space that the task used for processing prior areas.

The following figure shows a worksheet for estimating the data space size for the Analyze function.

Total size of IOVF parts in all areas	-----	(If input data is IC)
Total size of SDEP parts in all areas (QUICK)	-----	(If input data is IC and SDEPCHKLVL=FULL or QUICK)
Total of CISZ * UOW1 * (ROOT1 - ROOT2) in each area	-----	(If input data is IC)
Total bytes of a data space storage	-----	

Figure 88. Worksheet for estimating data space size for the Analyze function

DD statements for the Analyze function

DD statements of the FPA Analyze function determine the input and output data sets and specify how to run the Analyze function.

Subsections:

- “List of DD statements” on page 203
- “Input DD statements” on page 204
- “Output DD statements” on page 211
- “Input/Output DD statements” on page 216

List of DD statements

The following table summarizes the DD statements for the Analyze function.

Table 31. DD statements for the Analyze function

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
JOBLIB or STEPLIB	Input	PDS	No	Required
IMSACB, IMSACBA, or IMSACBB	Input	PDS	No	Required when IMSCATHLQ=*NO
IMSDALIB	Input	PDS	No	Optional
MODSTAT	Input	PS	Yes	Optional
MODSTAT2	Input	PS	Yes	Optional
OLCSTAT	Input	PS	Yes	Optional
<i>addnname</i> or <i>areaname</i>	Input	VSAM ESDS cluster or PS (Note)	Yes	Required under certain conditions (for details, see <i>addnname/areaname</i> DD)
<i>index</i>	Input	KSDS	Yes	Required when both of the following conditions are met: <ul style="list-style-type: none"> • INDEXDBD keyword is specified • DFSMDA members for index databases are not used
<i>indexovrflw</i>	Input	ESDS	Yes	Required when both of the following conditions are met: <ul style="list-style-type: none"> • INDEXDBD keyword is specified • DFSMDA members for index databases are not used

Table 31. DD statements for the Analyze function (continued)

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
HFPSYSIN	Input	LRECL=80	No	Required
HFPAPARM	Input	LRECL=80	No	Optional
HFPAUOWC	Input	LRECL=80	No	Required when HFFAHST2 DD is specified
HFPCSITE	Input	PDS	Yes	Optional
HFPPRINT	Output	SYSOUT	Yes	Optional
HFPRPTS	Output	LRECL=133	Yes	Optional
HFPSPRT	Output	LRECL=133	Yes	Optional
HFFAHST	Output	DISP=MOD	No	Optional
HFFAHST2	Output	DISP=MOD	No	Required when HFPAUOWC DD is specified
HFFARERG	Output	LRECL=80	No	Optional
HFFASNAP	Output	LRECL=133	No	Required when CISNAP=YES
HFFABEND	Output	LRECL=125	Yes	Optional
RECON n	Input and output	KSDS	Yes	Required when both of the following conditions are met: <ul style="list-style-type: none"> • DBRC=YES • DFSMDA members for RECONn data sets are not used
<i>itttWKnn</i>	Input and output		Yes	Optional
<i>itttSMMSG</i>	Output	LRECL=133	Yes	Optional
HFFS $nnnn$	Input and output		Yes	Optional
HFFI $nnnn$	Input and output		Yes	Optional
DFSPRINT	Output	LRECL=122	Yes	Optional
ICEIN	Input	LRECL=80	No	Optional
ICEPRINT	Output	LRECL=133	Yes	Optional
ICERPRT	Output	LRECL=133	No	Optional
<i>dataout1</i>	Output	PS	Yes	Optional
<i>dataout2</i>	Output	PS	Yes	Optional

Note: If image copy is used as an input data set.

Input DD statements

The following input DD statements are used for the Analyze function.

JOBLIB/STEPLIB DD

This DD statement is a required DD statement that defines the following input data sets:

- The FPA load module library (required).
- The library that contains DFSMDA dynamic allocation members for the RECON data set. When DBRC=YES is specified and the RECONn DD statement is omitted, this DD statement is required.
- The library that contains the randomizer routines and compression routines that are specified in the ACB member (specified by the IMSACB DD statements).
- The library that contains a default site default member (HFPCSITE) to determine the data set name and the member of the site default table.
- The IMS load module library (required).
- The library that contains the user-defined data entry database partition selection exit routine that was specified on the XDFLD PSELRTN= parameter during DBD generation for the DEDB.
- The library that contains the secondary index database maintenance exit routine that was specified on the XDFLD EXTRTN= parameter during DBD generation for the DEDB.
- The IMS Tools Base library (SHKTLOAD), if you specify the ITKBSRVR keyword.
- The IMS Tools Base library (SGLXLOAD), if you specify the IMSCATHLQ keyword.
- The IMS HP Image Copy library, if you specify the ICOUT=YES option.

FPA must be run as an APF-authorized program. All of the data sets that are listed within the JOBLIB or the STEPLIB concatenation must also be APF-authorized. If you use DFSMDA members and the DFSMDA members are in a non-APF authorized library, you can specify the DFSMDA library on the IMSDALIB DD statement.

IMSACB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. The IMSACB DD statements (IMSACB, IMSACBA, IMSACBB, MODSTAT, MODSTAT2, and OLCSTAT) can be used to identify the active ACB library. When both OLCSTAT or MODSTAT and MODSTAT2 are found, FPA checks the time stamp and determines the latest one.

The active IMSACB library is determined according to the following rules:

1. Check IMSACB DD statement in the JCL statement.
2. Check OLCSTAT DD statement in the JCL statement.
3. Check MODSTAT and MODSTAT2 DD statement in the JCL statement.
4. Allocate OLCSTAT dynamically and check the OLCSTAT member.
5. Allocate MODSTAT and MODSTAT2 dynamically and check the MODSTAT and MODSTAT2 members.

If you specify the IMSCATHLQ keyword, IMSACB DD statements are not used. The IMS directory is used instead of the ACB library.

IMSACBA DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSACBB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSDALIB DD

This DD statement is an optional DD statement that defines the IMSDALIB data set. This DD statement is useful when the data set for dynamic allocation members is not APF-authorized.

MODSTAT DD

This DD statement is an optional DD statement that defines the MODSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

MODSTAT2 DD

This DD statement is an optional DD statement that defines the MODSTAT2 data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT2 library by using a TYPE=DATABASE statement of the DFSMDA macro.

OLCSTAT DD

This DD statement is an optional DD statement that defines the OLCSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the OLCSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

***addnname/areaname* DD**

This DD statement is a required DD statement. This statement defines an input data set, which is the VSAM ESDS or image copy for the area to be analyzed. The ddname used must be the ddname for the area that is defined in the input ACB library (IMSACB DD). There can be multiple *addnname* DD statements for each area included in the analyze request.

The input data set can be allocated dynamically in one of the following ways:

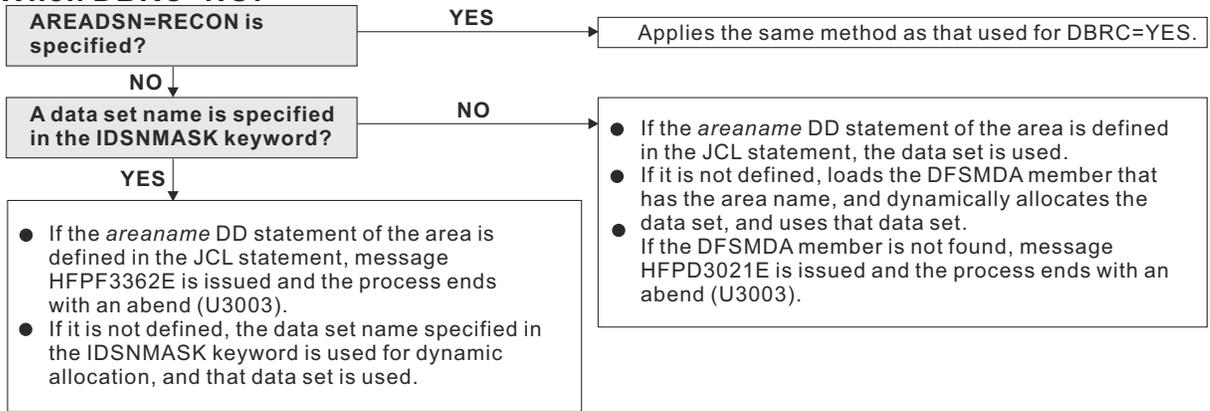
- The IDSNMASK keyword parameter of the ANALYZE command
- When the GLOBAL DBRC=YES option is specified, the ADS information in the DBRC RECON data sets (only for VSAM ESDS data sets)
- The area name member in the IMS DFSMDA library (only for VSAM ESDS data sets)

Or it can be specified with one of the following DD statements explicitly:

- *//addnname* DD JCL statement
- *//areaname* DD JCL statement

The following figure shows how FPA determines input area data set names for the ANALYZE command.

When DBRC=NO:



When DBRC=YES:

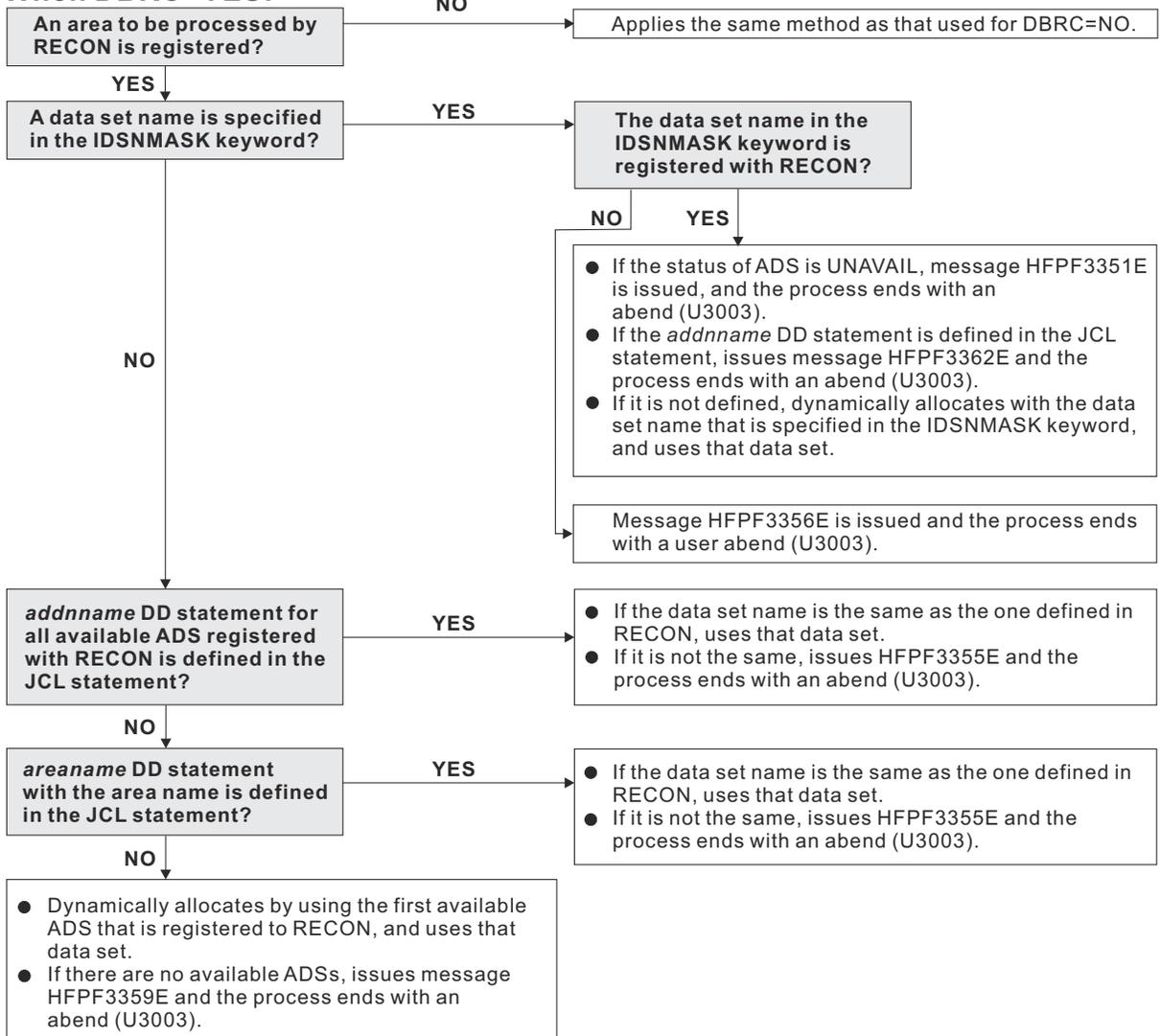


Figure 89. Input data set names for the ANALYZE command

The following figure shows how FPA determines area data set names for the Analyze process of the RELOAD and CHANGE commands.

When DBRC=NO:

The output data set name generated in the Reload or Change process is used.

When DBRC=YES:

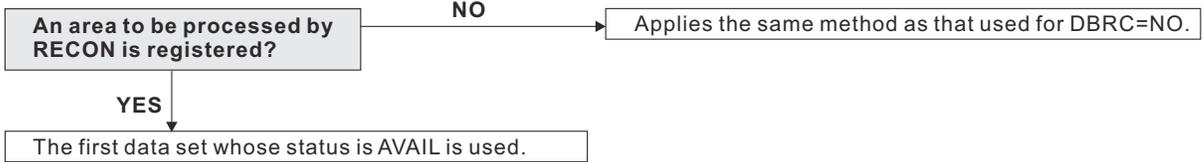


Figure 90. Input data set names for the Analyze process of the RELOAD and CHANGE commands

index DD

This DD statement is an optional input DD statement that defines the VSAM KSDS data set to be scanned. The DD name must be the DD name for the secondary index database that is defined in the input ACB library (IMSACB DD). Multiple *index DD* statements can be specified for each secondary index database that is processed by the Analyze function.

The input data set that is used by FPA is determined by the following elements:

- The name of the secondary index database member in the IMS DFSMDA library
- The *index DD* JCL statement

indexovrflw DD

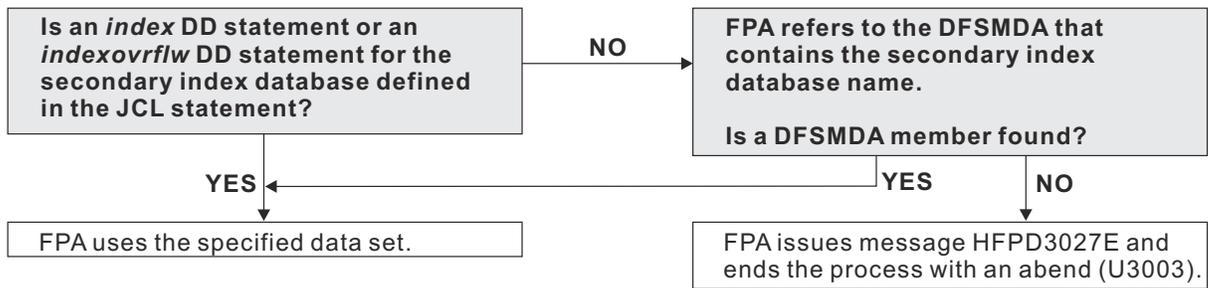
This DD statement is an optional input DD statement that defines the VSAM ESDS data set to be scanned. The DD name must be the DD name for the secondary index database that is defined in the input ACB library (IMSACB DD). Multiple *indexovrflw DD* statements can be specified for each secondary index database that is processed by the Analyze function.

The input data set that is used by FPA is determined by the following elements:

- The name of the secondary index database member in the IMS DFSMDA library
- The *indexovrflw DD* JCL statement

The following figure shows how FPA determines data set names for secondary index databases.

When DBRC=NO:



When DBRC=YES:

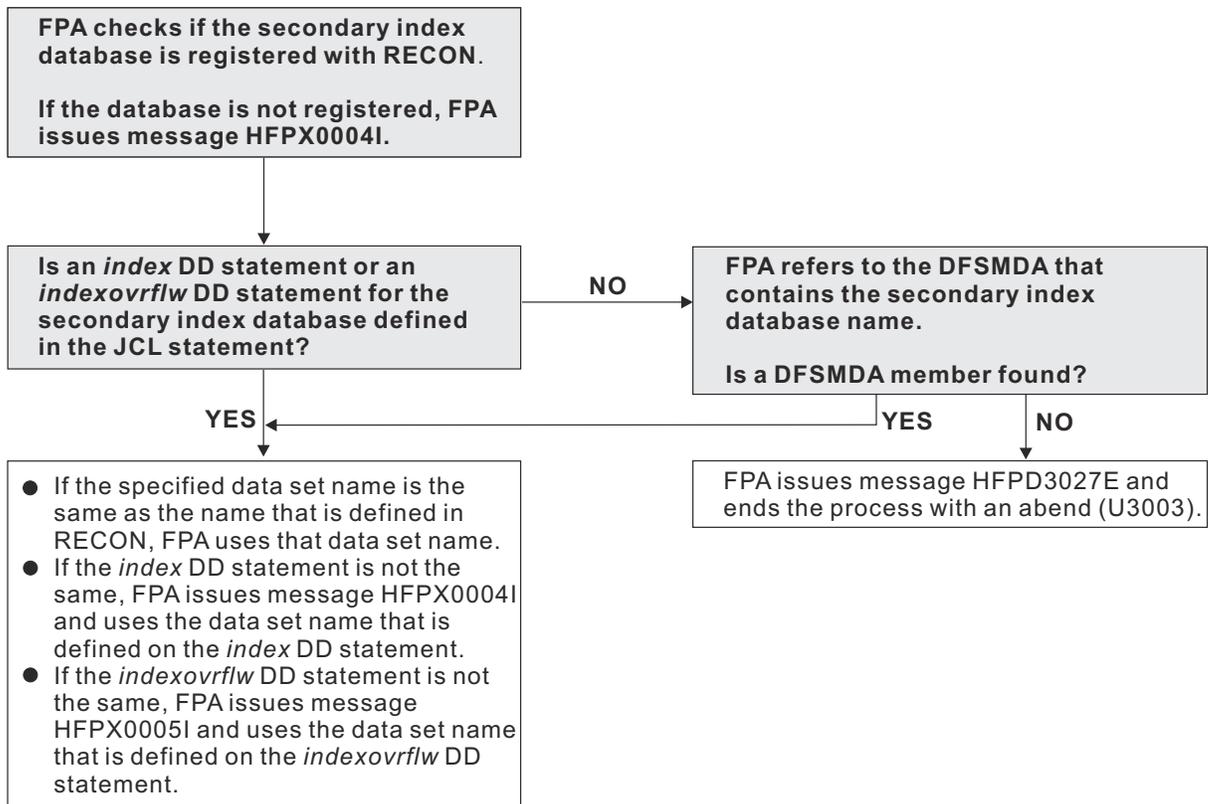


Figure 91. Input data set names for the secondary index database for the ANALYZE command

HFPSYSIN DD

This DD statement is a required DD statement that defines the input control statement that specifies the FPA functions.

The HFPSYSIN DD statement can be coded as a standard SYSIN file, a sequential data set, or a PDS member. LRECL=80 is required for the DCB of this data set. Only one command, besides the GLOBAL command, can be specified in HFPSYSIN. FPA analyzes the command syntax in columns 1 - 72. Columns 73 - 80 are regarded as comment.

For details about coding the HFPSYSIN DD statement, see [“Utility control statements for the Analyze function”](#) on page 218.

HFAPARM DD

This DD statement is an optional DD statement. This statement defines the control parameters for generating the analysis reports.

When SENSOR=YES is specified, the values of RBASEFS and RDOVFFS keywords are stored in the Sensor Data repository as the values of data elements DB_THRESHOLD_RBASEFS and DB_THRESHOLD_RDOVFFS. These values are used by the Online Space Management utility as the

values of RBASEFS and RDOVFFS keywords for the OER utility process (online reorganization process).

The HFPAPARM DD statement can be coded as a sequential data set or as a PDS member. LRECL=80 is required for the DCB of this data set. The HFPAPARM keywords must be written on one line. It can be coded as follows:

```
//HFPAPARM DD *
    RDOVFFS=40 SYNTBL=20 UDOVFFS=35 RBASEFS=40
/*
```

You can specify the keywords that are listed in the following table.

Table 32. HFPAPARM keywords

Keyword	Description
ARDIGIT	This keyword specifies the digit of the area number field for the UOW group definition records that are defined in the HFPAUOWC file. The value is 3 or 5. The default is 3. If the HFPAUOWC file is not specified in the JCL statement, this keyword is ignored.
RBASEFS	This keyword is an optional keyword. If a RAP CI contains free space that is greater than or equal to the percentage specified, and the RAP CI also uses overflow CIs, a reorganization control statement is generated for the UOW, and the UOW is included in the UOW Exception report. The default is 100%. The maximum value is 99, and the minimum value is 0.
RDOVFFS	This keyword is an optional keyword. If the DOVF part of a UOW contains free space that is greater than or equal to the percentage specified, and the UOW also uses IOVF CIs, a reorganization control statement is generated for the UOW, and the UOW is included in the UOW Exception report. The default is 100%. The maximum value is 99, and the minimum value is 0.
RGROUP	This keyword sets the group identifier for the statistics that are collected and stored in the History2 file or in a sensor data record set. For the History2 file, the keyword value is set in the Repository Group field, and for a sensor data record set, the value is set for the DB_SENSOR_DATA_GROUP_ID data element. This keyword has no default value. If this keyword is not specified, FPA sets the group identifier as follows: <ul style="list-style-type: none"> • The Repository Group field in the History2 file contains 4 bytes of blank characters. • The DB_SENSOR_DATA_GROUP_ID data element is not included in the sensor data record set.
SYNTBL	This keyword is an optional keyword. It controls the number of reporting entries in the Synonym Chain Analysis table. The default value is 41 (that is, all synonym chains longer than 40 are grouped and reported together). The maximum value is 999, and the minimum value is 12. This keyword is ignored if you specify the REPORT subcommand with the PTRCHKLVL=QUICK keyword.
UDOVFFS	This keyword is an optional keyword. If the DOVF UOW contains free space that is less than or equal to the percentage specified, the UOW is included in the UOW Exception report. The default is 0%. The maximum value is 99, and the minimum value is 0.

Note: The RBASEFS keyword and the RDOVFFS keyword that are used for the HFPAPARM data set are different from those used for the REORG command.

Requirement: Generating reorganization control statements for the High-Speed DEDB Direct Reorganization utility, you must specify the free space control percentages. You must also specify the HFPARERG DD statement in the JCL stream. For details, see the description of HFPARERG DD.

HFAUOWC DD

This DD statement is a required DD statement when HFAHST2 DD is specified. Each record in this data set specifies a range of UOW numbers of an area, which defines a UOW group of the area. The information about the UOW range is recorded in the HISTORY2 data set or in a sensor data record set in the order that is specified in the HFAUOWC data set, even if the UOW ranges overlap or are in reverse order. You can specify the UOW ranges for each area in any order. However, you must specify the area numbers in ascending order.

The data set can reside in the JCL input stream. The data set can also be defined as a sequential data set or as a member of a partitioned data set. If a data set is specified, the data set must have the following attributes:

- DSORG=PS
- RECFM=FB
- LRECL=80
- BLKSIZE=*a multiple of 80*

If the data set is empty, all UOWs of the area are regarded as the target UOW range.

When you specify the HFAHST2 DD statement, you must also specify the HFAUOWC DD statement. If an HFAUOWC DD statement is not specified, the HFAHST2 DD statement is ignored.

When SENSOR_UOW=YES is specified, the data elements that will be stored differ by the HFAUOWC DD specification as follows:

- If you specify the HFAUOWC DD statement, data elements are collected at the UOW group level.
- If you do not specify the HFAUOWC DD statement, data elements are collected at the UOW level.

For information about defining a UOW group, see [“UOW group definition record layout” on page 1191](#).

HFPCSITE DD

This DD statement is an optional DD statement. The HFPCSITE DD statement can be used to define the data set name of the site default table. When the SITE_DSNAME keyword is specified in the HFPSYSIN DD statement, the HFPCSITE DD statement is allocated dynamically. HFPCSITE DD in the JCL is mutually exclusive with the SITE_DSNAME keyword in the HFPSYSIN DD statement. The data set attribute of HFPCSITE DD is PDS.

ICEIN DD

This DD statement is an optional DD statement. This statement specifies the input data set that contains the control statements for IMS HP Image Copy. The data set can reside on a tape, a direct-access volume, or it can be routed through the input stream (DD * or DD DATA).

This data set is used only when you specify ICOUT=YES. If the DD statement is not coded and if ICOUT=YES is specified, the default control statements are used. For details about the ICEIN control statements, see *IMS High Performance Image Copy User's Guide*.

Output DD statements

The following output DD statements are used for the Analyze function.

HFPPRINT DD

This DD statement is an optional DD statement that defines the processing log output data set. The data set contains the processing messages that are issued by FPA.

The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set. You cannot specify DUMMY.

The HFPPRINT data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPPRINT DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*.

The following figure shows an example of the processing messages that are issued for the ANALYZE command.

```
IMS HPFP UTILITIES                                "Processing messages"                                PAGE: 1
5698-FPP V2R1                                     2019-11-22 13:03:37
HFPC0001I COMMAND ANALYZE PROCESSING STARTED.
HFPC0005I DBRC=NO IS SPECIFIED.
- EEQE DETECTION IS NOT PERFORMED.
HFPC0002I COMMAND ANALYZE PROCESSING COMPLETED.
HFPA0001I ANALYZE PROCESSING STARTED FOR DATABASE: DEDBJN21
HFPA0002I ANALYZER PROCESSING STARTED FOR DATABASE: DEDBJN21 AREA NO: 1 AREANAME: DB21AR0
HFPA0002I NO POINTER ERRORS WERE DETECTED IN DATABASE: DEDBJN21 AREA NO: 1 AREANAME: DB21AR0
HFPA0004I ANALYSER PROCESSING COMPLETED FOR DATABASE: DEDBJN21 AREA NO: 1 AREANAME: DB21AR0
HFPP0002I ANALYZE PROCESSING COMPLETED FOR DATABASE: DEDBJN21
HFPL0001I REPORT STARTED FOR COMMAND: ANALYZE, DATABASE: DEDBJN21
HFPL0002I REPORT COMPLETED FOR COMMAND: ANALYZE, DATABASE: DEDBJN21
```

Figure 92. Processing messages issued for the ANALYZE command

HFPRPTS DD

This DD statement is an optional DD statement that defines the report output data set. The data set contains the reports that can be used to verify the process of the command. The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

The following reports are generated in this data set.

- Audit report
- Processing report
- DBD Definition report

When the REPORT subcommand and PTRCHKLVL=FULL are specified, the following reports are generated:

- DEDB Area Analysis reports
- Process Summary of Analysis report
- UOW report
- UOW Exception report
- Largest Database Records report (When LARGEREC_REPORT=YES is specified)
- Consolidated Largest Database Records report (When LARGEREC_REPORT=YES is specified and the second parameter of the LARGEREC_COUNT keyword specifies a value that is equal to or larger than 1)
- Resolve Pointers report
- Segment Length Distribution report
- DB Record Length Distribution report
- RBA of Used IOVF CI report

When the REPORT subcommand and PTRCHKLV=QUICK are specified, the following reports are generated:

- DEDB Area Analysis reports (FREESPACE ANALYSIS only)
- Process Summary of Analysis report
- UOW report (without DB record information)
- UOW Exception report (without DB record information)
- Pointer Checksum report
- RBA of Used IOVF CI report

The following reports are also generated:

- The Secondary Index Definition report when secondary index databases are defined
- The Secondary Index Analysis report when the INDEXDBD keyword is specified
- The Pointer Segment Dump report when the IDXDUMP=YES option is specified and if incorrect pointer segments are detected

You can generate the following reports in another DD data set that is specified by the REPORTDD keyword parameter of the REPORT subcommand:

- DEDB Area Analysis reports
- UOW report
- UOW Exception report
- Largest Database Records report
- Consolidated Largest Database Records report
- Segment Length Distribution report
- DB Record Length Distribution report
- RBA of Used IOVF CI report

The HFPRPTS data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPRPTS DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

HFPSRT DD

This DD statement is an optional DD statement that defines the report output data set when SENSOR=YES is specified. This output data set contains the Sensor Data Statistics report that can be used to verify which sensor data is stored in the Sensor Data repository of IMS Tools KB. The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set. You can specify NULLFILE or DUMMY for this data set.

The HFPSRT data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPSRT DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

HFPAHST DD

This DD statement is an optional DD statement. This statement defines an output data set that contains historical records. The analysis report generation process adds one record for each DEDB area that it processes. The record contains the key space utilization and performance information for that area. This statement is used only when you specify both the REPORT subcommand and the PTRCHKLVL=FULL keyword.

If this feature is used, the HFPAHST data set must be preallocated with the following attributes:

- DSORG=PS
- RECFM=FB
- LRECL=100
- BLKSIZE=*user-specified*

The DD statement in the JOB stream is specified as:

```
//HFPAHST DD DSN=HPFP.HISTORY,DISP=(MOD,KEEP,KEEP)
```

You can generate the records in another DD data set specified by the HISTORYDD= keyword parameter instead of HFPAHST DD.

HFPAHST2 DD

This DD statement is a required DD statement when HFPAUOWC DD is specified. This statement defines an output data set that contains historical records. The analysis report generation process adds one record for each UOW group that is specified by HFPAUOWC DD. The record contains the key space utilization and performance information for that area, which is based on the information obtained from the specified UOW range. This statement is used only when you specify both the REPORT subcommand and the PTRCHKLVL=FULL keyword. If HFPAUOWC DD statement is not specified, this statement is ignored.

If this feature is used, the HFPAHST2 data set must be preallocated with the following attributes:

- DSORG=PS
- RECFM=FB
- LRECL=200
- BLKSIZE=*user-specified*

The DD statement in the JOB stream is specified as:

```
//HFPAHST2 DD DSN=HPFP.HISTORY2,DISP=(MOD,KEEP,KEEP)
```

HFPARERG DD

This DD statement is an optional DD statement. This statement defines an output data set that contains the reorganization control statements for the High-Speed DEDB Direct Reorganization utility (DBFUHDR0). This statement is used only when you specify both the REPORT subcommand and the PTRCHKLVL=FULL or the PTRCHKLVL=QUICK keyword.

The following DCB attributes of the file are set by the Analyze process:

- DSORG=PS
- RECFM=FB
- LRECL=80
- BLKSIZE=800

Do not code the DCB information in your JCL statement. Do not specify DISP=MOD for this DD statement.

Requirement:

Generating reorganization control statements for the High-Speed DEDB Direct Reorganization utility, you must specify the HFPARERG DD statement in the JCL stream. You must also specify the free space control percentages. For details about specifying the free space control percentages, see the description of HFPAPARM DD.

HFPASNAP DD

This DD statement is a required DD statement when CISNAP=YES. This statement defines the output data set for the CI Map/CI Dump report and the Pointer Chain Reconstruction report. This statement is used only when you specify the CISNAP=YES keyword.

The following DCB attributes of the file are set by the Analyze process:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement.

The HFPASNAP data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPASNAP DD SYSOUT=A
```

HFPABEND DD

This DD statement is an optional DD statement. It defines a snap dump output data set that is used for diagnostic purposes. This data set might be used when the FPA job ends abnormally. Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

itttSMMSG DD

These optional DD statements point to the data sets for storing DFSORT messages. The thread task number is used for the first four characters (*ittt*). For variable *i* and conditions for running DFSORT, see the description of *itttWKnn* DD.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

DFSPRINT DD

This DD statement is an optional DD statement. This statement specifies the output data set that contains the messages that are produced by IMS HP Image Copy. The data set can reside on a tape, a direct-access volume, or a printer, or it can be routed through the output stream (SYSOUT).

This data set is used only when you specify ICOUT=YES. If the DD statement is not coded and ICOUT=YES is specified, a SYSOUT stream is dynamically allocated for the DD.

ICEPRINT DD

This DD statement is an optional DD statement. This statement specifies the output data set that contains the reports that are produced by IMS HP Image Copy. The data set can reside on a tape, a direct-access volume, or a printer, or it can be routed through the output stream (SYSOUT).

This data set is used only when you specify ICOUT=YES. If the DD statement is not coded and ICOUT=YES is specified, a SYSOUT stream is dynamically allocated for the DD.

ICERPRT DD

This DD statement is an optional DD statement. This statement specifies the output data set that contains the Image Copy Dump Process report, which is produced by IMS HP Image Copy. It can reside on a tape, a direct-access volume, or a printer, or it can be routed through the output stream (SYSOUT).

This data set is used only when you specify ICOUT=YES.

dataout1 DD

This DD statement is an optional DD statement. This statement defines the primary image copy data set. One DD statement is required for each data set that is to be dumped. The DD name can be any 1- to 8-character string. The data set specified by the DD statement must be on either a direct-access device or a tape device.

This data set is used only when you specify ICOUT=YES. For details of the format and the characteristics of the image copy data set, see *IMS High Performance Image Copy User's Guide*. You can request IMS HP Image Copy to allocate the image copy data set dynamically by omitting this DD statement and specifying the allocation parameters that are associated with the data set in the ICEIN data set.

dataout2 DD

This DD statement is an optional DD statement. This statement defines the secondary image copy data set. This statement is required only if the associated ICEIN control statement requests the secondary image copy. The data set specified by the DD statement must be on either a direct-access device or a tape device.

This data set is used only when you specify ICOUT=YES. For details of the format and the characteristics of the image copy data set, see *IMS High Performance Image Copy User's Guide*.

Input/Output DD statements

The following input/output DD statements are used for the Analyze function.

RECON1/RECON2/RECON3 DD

These statements define the DBRC RECON data sets. These DD statements are required DD statements when both of the following conditions are met:

- DBRC=YES is specified for the GLOBAL command
- DFSMDA members for RECON n data sets are not used

The RECON n data set must be the same data set that is used in the IMS control region. Because FPA issues a signon, a signoff, an authorization call, and an unauthorization call to DBRC, it updates RECON n data sets.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is necessary, the data set is allocated dynamically by using a DFSMDA member.

itttWKnn DD

These DD statements are optional DD statements. These data sets are intermediate storage data sets that are used by DFSORT. For more information about coding the SORTWK nn DD statements, see *DFSORT Application Programming Guide*.

The adequate size of storage data sets is 1.5 to 2.5 times the input file size. Allocating twice the space used by the SORTIN data set is usually adequate for each work data set.

Do not specify DISP=MOD for these DD statements.

The naming rules of the ddnames and the size of each input file are as follows:

i:

A, B, and C: If PTRCHKLVL=FULL is specified

30 x the number of pointers in the area

D, E, and F: If SDEPCHKLVL=FULL is specified

30 x the number of SDEP segments in the area

G, H, and I: If CISNAP=YES is specified

112 x the number of pointers in which pointer errors are detected

J, K, and L: If the REPORT subcommand is specified

PTRCHKLVL=FULL is specified

$X \times$ (the number of non-SDEP segments in the areas + the number of free space elements in the areas + the number of references to other CIs)

Where X is:

- When LARGEREC_REPORT=NO is specified, X is calculated as follows:

$34 +$ (the maximum segment level \times 4)

However, if X is smaller than 48, 48 is used for X .

- When LARGEREC_REPORT=YES is specified, X is calculated as follows:

$34 +$ (the maximum segment level \times 4) + root key length

However, if X is smaller than 48, 48 is used for X .

PTRCHKLVL=QUICK is specified

$23 \times$ (the number of free space elements in the area + the number of references to other CIs)

M, N, and O: If PTRCHKLVL=FULL is specified

$X \times$ the number of segments in the IOVF part

Where X is:

- When LARGEREC_REPORT=NO is specified, X is 34.
- When LARGEREC_REPORT=YES is specified, X is calculated as follows:

$34 +$ root key length

The thousands digit of the area number of the area to be analyzed, or the thousands digit of n for n -th input thread subtask.

A, D, G, J, and M:

If the number of areas to be analyzed or n is 999 or less.

B, E, H, K, and N:

If the number of areas to be analyzed or n is 1000 - 1999.

C, F, I, L, and O:

If the number of areas to be analyzed or n is 2000 - 2048.

Note: When a subtask ends processing an area, it starts processing the next area. For this reason, the n -th input thread subtask does not always process the n -th input data set.

ttt:

The hundreds, tenths, and units digits of the area number of the area to be analyzed, or those of n .

nn:

The number of the SORT work data sets. This value must be 01 - 03.

Note on dynamic allocation:

If this statement is not specified in the JCL statement when these data sets are necessary, FPA allocates them dynamically.

FPA uses the thread task number for the first four characters (*ittt*).

HFPSnnnn DD

HFPInnnn DD

These statements define the intermediate work data sets. HFPSnnnn data sets contain the segment information records for all segments that reside in the root addressable part of the database. HFPInnnn data sets contain the segment information records for all segments that reside in the IOVF part of the database. These data sets are used only when you specify PTRCHKLVL=FULL.

An HFPSnnnn DD statement or HFPInnnn DD statement is for an area or thread task. The value of nnnn must be right-aligned, with leading zeros if necessary. Do not specify DISP=MOD for these DD statements.

For example, if you specify IAREA=(11-20) and ATASKCTL=2 in the HFPSYSIN data set and HFPS0001 and HFPS0020 DD statements in the JCL, HFPS $nnnn$ data sets are used as follows:

- HFPS0001 data set is used by thread task number 1.
- HFPS0020 data set is used when area number 20 is processed by a thread task.
- If areas 11 to 19 are processed by thread task number 2, FPA allocates the HFPS0002 data set dynamically.

Note on dynamic allocation:

If this statement is not specified in the JCL statement when these data sets are necessary, FPA allocates them dynamically.

Utility control statements for the Analyze function

The utility control statements for the Analyze function define the user description of the GLOBAL command, the ANALYZE command, and two optional subcommands, REPORT and THRESHOLD, all of which control the Analyze process.

The utility control statements must be specified in the HFPSYSIN data set. This control statement data set usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte, fixed-length records. Block size, if coded, must be a multiple of 80.

The following figure is a sample showing how to code the HFPSYSIN data set.

```
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
ANALYZE
  DBD=dbdname
```

Figure 93. ANALYZE HFPSYSIN data set

Subsections:

- [“GLOBAL command keywords” on page 218](#)
- [“ANALYZE command keywords” on page 219](#)
- [“REPORT subcommand keywords” on page 221](#)
- [“THRESHOLD subcommand keywords” on page 221](#)

GLOBAL command keywords

For the GLOBAL command, you can specify the keywords that are listed in the following table. For the format of the GLOBAL command, see [“GLOBAL command” on page 394](#).

Table 33. GLOBAL command keywords

Keyword	Required or optional	Description	Topic
ADXCGRP	Optional	Specifies the name of the Autonomics Director XCF group.	“ADXCGRP keyword” on page 439
AREADSN	Optional	Specifies how to obtain area data set names.	“AREADSN keyword” on page 441
CISNAP	Optional	Specifies to request the CI Map/CI Dump report and the Pointer Chain Reconstruction report.	“CISNAP keyword” on page 448
DBRC	Optional	Specifies to request or bypass the processing of database recovery control (DBRC) facility for the database that is specified by the DBD.	“DBRC keyword” on page 451
HISTORYDD	Optional	Specifies the DD statement for the output History file data set that is specified in the JCL.	“HISTORYDD keyword” on page 469

Table 33. GLOBAL command keywords (continued)

Keyword	Required or optional	Description	Topic
ICINCOMP	Optional	Specifies, when an image copy is used as an input data set, whether the image copy records are to be compressed by using the CSRCSRV MVS macro when they are stored into a data space storage.	“ICINCOMP keyword” on page 474
ICNOTIFY	Optional	Specifies whether to issue the NOTIFY.IC command to DBRC.	“ICNOTIFY keyword” on page 475
ICOUT	Optional	Specifies whether to generate one or more image copies.	“ICOUT keyword” on page 476
IMSCATACB_INPUT	Optional	Specifies whether to retrieve the currently active ACB definition or the pending ACB definition from the IMS directory.	“IMSCATACB_INPUT keyword” on page 480
IMSCATHLQ	Optional	Specifies the high-level qualifier of the bootstrap data set of the IMS directory.	“IMSCATHLQ keyword” on page 481
ITKBLOAD	Optional	Specifies the library name of IMS Tools KB.	“ITKBLOAD keyword” on page 487
ITKBSRVR	Optional	Specifies the name of the IMS Tools KB server XCF group.	“ITKBSRVR keyword” on page 488
MSTIMESTAMP	Optional	Specifies whether to include a time stamp in the messages.	“MSTIMESTAMP keyword” on page 497
PTRCHKLVL	Optional	Specifies the method that is used to run integrity verification in the root addressable part and in the independent overflow part.	“PTRCHKLVL keyword” on page 506
SCAN	Optional	Specifies to verify the syntax of command, keywords, and parameters in command sets without running FPA. The Scan function also allocates some (but not all) internal resources for running the process function.	“SCAN keyword” on page 523
SDEPCHLVL	Optional	Specifies the method that is used to run integrity verification in the sequential dependent part.	“SDEPCHLVL keyword” on page 526
SDEP_ORPHAN_MSGLVL	Optional	Specifies the error message severity level to be set when an orphan SDEP segment is detected.	SDEP_ORPHAN_MSGLVL keyword
SITE_DSNAME	Optional	Specifies the data set name for the site default table.	“SITE_DSNAME keyword” on page 539
SITE_MEMBER	Optional	Specifies the member name for the site default table.	“SITE_MEMBER keyword” on page 540

ANALYZE command keywords

For the ANALYZE command, you can specify the keywords that are listed in the following table. For the format of the ANALYZE command, see [“ANALYZE command” on page 382](#).

Table 34. ANALYZE command keywords

Keyword	Required or optional	Description	Topic
ATASKCTL	Optional	Specifies the maximum number of input areas to be analyzed concurrently.	“ATASKCTL keyword” on page 443
CISNAP	Optional	Requests the CI Map/CI Dump report and the Pointer Chain Reconstruction report.	“CISNAP keyword” on page 448
DBD	Required	Specifies the database description (DBD) that contains the areas to be processed.	“DBD keyword” on page 450
DOVFNULPTR_MSGLVL	Optional	Specifies that FPA issues a warning message if the first DOVF CI in any UOW contains a null pointer for the current overflow CI.	DOVFNULPTR_MSGLVL keyword

Table 34. ANALYZE command keywords (continued)

Keyword	Required or optional	Description	Topic
HISTORYDD	Optional	Specifies the DD statement for the output History file data set that is specified in the JCL.	“HISTORYDD keyword” on page 469
IAREA	Optional	Specifies one or more input areas for the Analyze process.	“IAREA keyword” on page 469
ICACHE	Optional	Specifies the type of control intervals that are to be preloaded into a data space storage for processing.	“ICACHE keyword” on page 472
ICHECK	Optional	Controls the concurrent number of input devices to access by using the ITASKCTL keyword.	“ICHECK keyword” on page 473
ICINCOMP	Optional	Specifies, when an image copy is used as an input data set, whether the image copy records are to be compressed by using the CSRCEserv MVS macro when they are stored into a data space storage.	“ICINCOMP keyword” on page 474
ICNOTIFY	Optional	Specifies whether to issue the NOTIFY.IC command to DBRC.	“ICNOTIFY keyword” on page 475
ICOUT	Optional	Specifies whether to generate one or more image copies.	“ICOUT keyword” on page 476
IDSNMASK	Optional	Specifies the data set name or the mask for dynamically allocating cataloged pre-existing input data sets.	“IDSNMASK keyword” on page 476
IDXDUMP	Optional	Specifies whether to generate the Pointer Segment Dump report when incorrect pointers are detected.	“IDXDUMP keyword” on page 477
IDXMAXERROR	Optional	Specifies the maximum allowable number of pointer segment errors for the index database. When the maximum number is reached, the ANALYZE process ends abnormally.	“IDXMAXERROR keyword” on page 478
IDXTASKCTL	Optional	Specifies the maximum number of input secondary index databases that are to be processed concurrently.	“IDXTASKCTL keyword” on page 479
INDEXDBD	Optional	Specifies the database description (DBD) for building or processing secondary index databases.	“INDEXDBD keyword” on page 482
KEYSEQCHK	Optional	Specifies what to do if a key sequence error is detected.	“KEYSEQCHK keyword” on page 490
MAXERROR	Optional	Specifies the maximum number of pointer errors that might be encountered before processing ends with an error.	“MAXERROR keyword” on page 495
PTRCHKLVL	Optional	Specifies the method that is used to run integrity verification in RAA BASE, DOVF, and IOVF.	“PTRCHKLVL keyword” on page 506
SDEPCHKLVL	Optional	Specifies the method that is used to run integrity verification in the sequential dependent part	“SDEPCHKLVL keyword” on page 526
SDEP_ORPHAN_MSGLVL	Optional	Specifies the error message severity level to be set when an orphan SDEP segment is detected.	SDEP_ORPHAN_MSGLVL keyword
SDEP_PTRIOVF_MSGLVL	Optional	Specifies the severity level of the message that is issued when detecting an SDEP pointer that points to the IOVF portion.	SDEP_PTRIOVF_MSGLVL keyword
SENSOR	Optional	Specifies whether to store the sensor data in the Sensor Data repository of IMS Tools KB.	“SENSOR keyword” on page 535
SENSOR_DBREC	Optional	Specifies whether to store the data elements that are related to database records and root segment distribution in the Sensor Data repository of IMS Tools KB.	“SENSOR_DBREC keyword” on page 536
SENSOR_PTRERR	Optional	Specifies to store the sensor data in the Sensor Data repository of IMS Tools KB even when pointer errors are detected in the area.	“SENSOR_PTRERR keyword” on page 537
SENSOR_UOW	Optional	Specifies whether to store the data elements that are related to UOW in the Sensor Data repository of IMS Tools KB.	“SENSOR_UOW keyword” on page 537
SENSOR_UOW_REPORT	Optional	Specifies whether to print the data elements that are related to UOW in the Sensor Data Statistics report.	SENSOR_UOW_REPORT keyword

Table 34. ANALYZE command keywords (continued)

Keyword	Required or optional	Description	Topic
SORTOPT_FILSZ	Optional	Specifies that FPA adds the FILSZ parameter to the SORT control statement.	“SORTOPT_FILSZ keyword” on page 542

REPORT subcommand keywords

The REPORT subcommand requests the analysis reports. If you do not specify PTRCHKLVL=FULL or PTRCHKLVL=QUICK, this subcommand is ignored.

For the REPORT subcommand, you can specify the keywords that are listed in the following table. For the format of the REPORT subcommand, see [“REPORT subcommand” on page 432](#).

Table 35. REPORT subcommand keywords

Keyword	Required or optional	Description	Topic
DBRLDIST	Optional	Specifies whether to generate the DB Record Length Distribution report.	“DBRLDIST keyword” on page 452
IOVFCI_RBA_REPORT	Optional	Specifies whether to generate the RBA of Used IOVF CI report.	“IOVFCI_RBA_REPORT keyword” on page 484
LARGEREC_COUNT	Optional	Specifies the number of reporting entries in the Largest Database Records report and the Consolidated Largest Database Records report. This keyword is effective only when PTRCHKLVL=FULL is specified.	“LARGEREC_COUNT keyword” on page 491
LARGEREC_REPORT	Optional	Specifies whether to generate the Largest Database Records report. This keyword is effective only when PTRCHKLVL=FULL is specified.	“LARGEREC_REPORT keyword” on page 491
REPORTDD	Optional	Specifies the destination ddname for the analysis reports.	“REPORTDD keyword” on page 513
SEGLDIST	Optional	Specifies whether to generate the Segment Length Distribution report.	“SEGLDIST keyword” on page 529
SEGLSTAT	Optional	Specifies the format of the DB Record Profile Analysis report, which is one of the DEDB Area Analysis reports.	“SEGLSTAT keyword” on page 529
STARTUOW	Optional	Specifies the number that defines the first UOW to be reported in the UOW report and the UOW Exception report.	“STARTUOW keyword” on page 545
STOPUOW	Optional	Specifies the number that defines the last UOW to be reported in the UOW report and the UOW Exception report.	“STOPUOW keyword” on page 546
UOW_REPORT	Optional	Specifies whether to generate the UOW report.	“UOW_REPORT keyword” on page 555

THRESHOLD subcommand keywords

The THRESHOLD subcommand specifies the threshold parameter values, which cause warning messages to be generated when the actual values do not meet the specified criteria. If you do not specify the REPORT subcommand, this subcommand is ignored.

For the THRESHOLD subcommand, you can specify the keywords that are listed in the following table. For the format of the THRESHOLD subcommand, see [“THRESHOLD subcommand” on page 435](#).

Table 36. THRESHOLD subcommand keywords

Keyword	Required or optional	Description	Topic
DOVFFS_PCT	Optional	Specifies a threshold setting for the percentage of free space in DOVF. If the percentage of free space in DOVF is less than the value that you specified, a warning message is issued.	“DOVFFS_PCT keyword” on page 455
FS_DOVFIOVF	Optional	Specifies a threshold setting for the free space in DOVF and in IOVF. If the free space in DOVF is greater than the value that you specified, and if the free space in IOVF is less than the value that you specified, a warning message is issued.	“FS_DOVFIOVF keyword” on page 466
FS_RAADOVF	Optional	Specifies a threshold setting for the free space in RAA BASE and in DOVF. If the free space in RAA BASE is greater than the value that you specified, and if the free space in DOVF is less than the value that you specified, a warning message is issued.	“FS_RAADOVF keyword” on page 467
FS_RAADOVFIOVF	Optional	Specifies a threshold setting for the free space in RAA BASE, DOVF, and IOVF. If the free space in RAA BASE is greater than the value that you specified, and if the free space in either DOVF or IOVF is less than the value that you specified, a warning message is issued.	“FS_RAADOVFIOVF keyword” on page 467
FS_RAATIOVF	Optional	Specifies a threshold setting for the free space in RAA BASE and in IOVF. If the free space in RAA BASE is greater than the value that you specified, and if the free space in IOVF is less than the value that you specified, a warning message is issued.	“FS_RAATIOVF keyword” on page 468
IOVFFS_PCT	Optional	Specifies a threshold setting for the percentage of free space in IOVF. If the percentage is less than value that you specified, a warning message is issued.	“IOVFFS_PCT keyword” on page 485
IOVFUSED_PCT	Optional	Specifies a threshold setting for the percentage of IOVF CIs used. If the percentage is greater than the value that you specified, a warning message is issued.	“IOVFUSED_PCT keyword” on page 485
RAAFS_PCT	Optional	Specifies a threshold setting for the percentage of free space in RAA BASE. If the percentage is less than the value that you specified, a warning message is issued.	“RAAFS_PCT keyword” on page 507
RAPOVFL_PCT	Optional	Specifies a threshold setting for the percentage of RAA BASE CIs that use either DOVF or IOVF. If the percentage is greater than the value that you specified, a warning message is issued.	“RAPOVFL_PCT keyword” on page 508
RECIO_AVG	Optional	Specifies a threshold setting for the average number of I/Os that can be done to access a database record. If the number is greater than the value that you specified, a warning message is issued.	“RECIO_AVG keyword” on page 510
RECIO_MAX	Optional	Specifies a threshold setting for the maximum number of I/Os that can be done to access a database record. If the number is greater than the value that you specified, a warning message is issued.	“RECIO_MAX keyword” on page 511
RECIOVF_PCT	Optional	Specifies a threshold setting for the percentage of records using IOVF CIs. If the percentage is greater than the value that you specified, a warning message is issued.	“RECIOVF_PCT keyword” on page 510
ROOTIO_AVG	Optional	Specifies a threshold setting for the average number of I/Os to access a root segment. If the number is greater than the value that you specified, a warning message is issued.	“ROOTIO_AVG keyword” on page 522
ROOTIO_MAX	Optional	Specifies a threshold setting for the maximum number of I/Os that can be done to access a root segment. If the number is greater than the value that you specified, a warning message is issued.	“ROOTIO_MAX keyword” on page 523
SYN_LEN_AVG	Optional	Specifies a threshold setting for the average synonym chain length. If the length is greater than the value that you specified, a warning message is issued.	“SYN_LEN_AVG keyword” on page 547
SYN_MAX	Optional	Specifies a threshold setting for the maximum synonym chain length. If the length is greater than the value that you specified, a warning message is issued.	“SYN_MAX keyword” on page 548

Table 36. THRESHOLD subcommand keywords (continued)

Keyword	Required or optional	Description	Topic
SYN_PCT_LEN	Optional	Specifies a threshold setting for the percentage of RAPs that have a long synonym chain among all the RAPs that have synonyms. If the percentage of the RAPS that have a long synonym chain is greater than the threshold value, a warning message is issued.	“SYN_PCT_LEN keyword” on page 548
UOWDOVF_PCT	Optional	Specifies a threshold setting for the percentage of UOWs that use DOVF. If the percentage is greater than the value that you specified, a warning message is issued.	“UOWDOVF_PCT keyword” on page 553
UOWIOVF_AVG	Optional	Specifies a threshold setting for the average number of IOVF CIs that are used by a UOW. If the number is greater than the value that you specified, a warning message is issued.	“UOWIOVF_AVG keyword” on page 554
UOWIOVF_MAX	Optional	Specifies a threshold setting for the maximum number of IOVF CIs that are used by a UOW. If the number is greater than the value that you specified, a warning message is issued.	“UOWIOVF_MAX keyword” on page 554
UOWIOVF_PCT	Optional	Specifies a threshold setting for the percentage of UOWs that use IOVF. If the percentage is greater than the value that you specified, a warning message is issued.	“UOWIOVF_PCT keyword” on page 555

Examples for the Analyze function

Review examples and learn techniques for analyzing DEDB areas.

In these examples, the following DEDB is assumed:

```

DEDB name:  DEDBJN22 has four areas
  area name:  DB22AR0 (area number is 1 and has three ADSs)
    ADS DD name:  DB22AR01,      DS name:  HPFP.DB22AR0.ADS1
    ADS DD name:  DB22AR02,      DS name:  HPFP.DB22AR0.ADS2
    ADS DD name:  DB22AR03,      DS name:  HPFP.DB22AR0.ADS3
  area name:  DB22AR1 (area number is 2 and has two ADSs)
    ADS DD name:  DB22AR11,      DS name:  HPFP.DB22AR1.ADS1
    ADS DD name:  DB22AR12,      DS name:  HPFP.DB22AR1.ADS2
  area name:  DB22AR2 (area number is 3 and has two ADSs)
    ADS DD name:  DB22AR21,      DS name:  HPFP.DB22AR2.ADS1
    ADS DD name:  DB22AR22,      DS name:  HPFP.DB22AR2.ADS2
  area name:  DB22AR3 (area number is 4 and has one ADS)
    ADS DD name:  DB22AR31,      DS name:  HPFP.DB22AR3.ADS1

```

In DBRC, all ADSs are in the AVAILABLE status.

In the examples that use secondary index databases, the following secondary index databases are assumed:

```

Secondary index database name: INDEX1
  Primary DBDS DD name: INDEXK1, DS name: HPFP.INDEXK1
Secondary index database name: INDEX2
  Primary DBDS DD name: INDEXK2, DS name: HPFP.INDEXK2
  Overflow DBDS DD name: INDEXE2, DS name: HPFP.INDEXE2

```

You can find additional JCL examples that help you to monitor and tune DEDB areas in [“FPA Analyze JCL examples for monitoring and tuning DEDBs” on page 1129](#).

Example 1: Running checksum validation of pointers in an area

The following figure shows example JCL for the checksum validation of pointers in an area that is registered in DBRC.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.

- The input ADS to be validated is dynamically allocated. The allocation information (DD name and DS name of the ADS) is obtained from DBRC.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
ANALYZE
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  PTRCHKLVL=QUICK
/*
```

Figure 94. Running checksum validation of pointers in an area

Example 2: Running cross-reference validation of pointers in an area

The following figure shows example JCL for the cross-reference validation of pointers in an area that is registered in DBRC.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS to be validated is dynamically allocated. The allocation information (DD name and DS name of the ADS) is obtained from DBRC.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
ANALYZE
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  PTRCHKLVL=FULL,
  SDEPCHKLVL=FULL
/*
```

Figure 95. Running cross-reference validation of pointers in an area

Example 3: Generating the analysis report of an area in the HPFRPTS data set

The following figure shows example JCL for generating an analysis report of an area that is registered in DBRC.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS to be validated is dynamically allocated. The allocation information (DD name and DS name of the ADS) is obtained from DBRC.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
ANALYZE
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  PTRCHKLVL=FULL,
  SDEPCHKLVL=FULL
REPORT
/*

```

Figure 96. Generating the analysis report of an area in the HPFRPTS data set

Example 4: Running checksum validation of pointers in an image copy

The following figure shows example JCL for the checksum validation of pointers in an image copy.

In this example, the input image copy to be validated is ICOUT.HFP.IC1.DEDBJN22.DB22AR0, which is specified by the IDSNMASK='ICOUT.HFP.IC1.DEDBJN22.&AREA' parameter. This data set is allocated dynamically.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=NO
ANALYZE
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  IDSNMASK=('ICOUT.HFP.IC1.DEDBJN22.&AREA'),
  PTRCHKLVL=FULL,
  SDEPCHKLVL=FULL
/*

```

Figure 97. Running checksum validation of pointers in an image copy

Example 5: Running cross-reference validation of pointers in all areas and validation of pointer segments in all secondary index databases

The following figure shows example JCL for running the cross-reference validation of pointers in all areas and validating pointer segments in all secondary index databases.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS to be validated is dynamically allocated. The allocation information (DD name and DS name of the ADS) is obtained from DBRC.
- The input secondary index database data sets to be validated are dynamically allocated. The allocation information (DD name and DS name of the secondary index database data set) is obtained from DFSMDA members in the IMSDALIB DD data set.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL    DBRC=YES
ANALYZE   DBD=DEDBJN22,
          IAREA=ALL,
          PTRCHKLVL=FULL,
          INDEXDBD=ALL
/*

```

Figure 98. Running cross-reference validation of pointers in all areas and validation of pointer segments in all secondary index databases

Example 6: Running cross-reference validation of pointers in an area and validation of pointer segments in all secondary index databases

The following figure shows example JCL statement for running the cross-reference validation of pointers in an area and validating pointer segments in all secondary index databases.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS to be validated is dynamically allocated. The allocation information (DD name and DS name of the ADS) is obtained from DBRC.
- The input secondary index database data sets to be validated is dynamically allocated. The allocation information (DD name and DS name of the secondary index database data set) is obtained from DFSMDA members in the IMSDALIB DD data set.
- Pointer segments that point the target segment in area that is other than the input area (DB22AR0) are not validated.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL    DBRC=YES
ANALYZE   DBD=DEDBJN22,
          IAREA=DB22AR0,
          PTRCHKLVL=FULL,
          INDEXDBD=ALL
/*

```

Figure 99. Running cross-reference validation of pointers in an area and validation of pointer segments in all secondary index databases

Example 7: Running cross-reference validation of pointers in all areas and validation of pointer segments in a secondary index database

The following figure shows example JCL statement for running the cross-reference validation of pointers in all areas and validating pointer segments in a secondary index database.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.

- The input ADS to be validated is dynamically allocated. The allocation information (DD name and DS name of the ADS) is obtained from DBRC.
- The input secondary index database data sets to be validated is dynamically allocated. The allocation information (DD name and DS name of the secondary index database data set) is obtained from DFSMDA members in the IMSDALIB DD data set.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL    DBRC=YES
ANALYZE   DBD=DEDBJN22,
          IAREA=ALL,
          PTRCHKLVL=FULL,
          INDEXDBD=INDEX1
/*
```

Figure 100. Running cross-reference validation of pointers in all areas and validation of pointer segments in a secondary index database

Example 8: Generating the analysis reports of an area and storing the sensor data in the IMS Tools KB repository

The following figure shows example JCL for generating the analysis reports of an area in the HFPRPTS data set and storing the sensor data in the Sensor Data repository of IMS Tools KB.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS to be validated is dynamically allocated. The allocation information (DD name and DS name of the ADS) is obtained from DBRC.
- Sensor data is collected and stored in the Sensor Data repository of IMS Tools KB. In addition to the default data elements that are collected by the SENSOR keyword, the data elements that are related to database records and root segment distribution are stored (SENSOR_DBREC=YES specification).

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//          DD DISP=SHR,DSN=ITB.SHKTLOAD
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSRPT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL    DBRC=YES,
          ITKBSRVR=FPQSRV01,
          ADXCFGRP=ADSRV01
ANALYZE   DBD=DEDBJN22,
          IAREA=(DB22AR0),
          PTRCHKLVL=FULL,
          SDEPCHLVL=FULL,
          SENSOR=YES,
          SENSOR_DBREC=YES
REPORT
/*
```

Figure 101. Generating the analysis reports and storing the sensor data in the IMS Tools KB repository

Chapter 9. Tuning a DEDB offline with FPA

You can tune a DEDB by using the FPA Tune function.

Topics:

- [“Functions of Tune” on page 229](#)
- [“Input and output for the Tune function” on page 231](#)
- [“Running the Tune function” on page 233](#)
- [“Region size requirements for the Tune function” on page 236](#)
- [“DD statements for the Tune function” on page 238](#)
- [“Utility control statements for the Tune function” on page 247](#)
- [“Examples for the Tune function” on page 253](#)

Functions of Tune

The Tune function provides a comprehensive tuning facility for managing databases and tuning performance. The function simulates changes in the DBD parameters and randomizers without loading the segments. Use this function to select the physical database attributes that satisfy your performance and space utilization requirements.

Any or all of the following database specifications can be changed and evaluated with the Tune function:

- Randomizing module
- UOW parameter values
- ROOT parameter values
- Number of database areas
- CI sizes

The Tune function can help you accomplish the following tasks:

- Select database parameters and attributes with greater precision.
- Identify and select physical database attributes that meet performance and space utilization requirements (optimal values for the ROOT and UOW parameters).
- Simplify the evaluation and selection of suitable randomizing routines.
- Reduce maintenance frequency and costs.
- Eliminate the level of database maintenance required when a database, after an unload and reload, does not meet performance or space utilization expectations.
- Evaluate whether tuning helps or hinders online performance.

A tuning summary report, which shows effectiveness of the changed attributes for each area, helps you determine the optimal database attributes. This report contains the following information:

- Threshold exceptions detected in the current database and in the simulated database
- Characteristics of the areas and the number of database records in the current database and in the simulated database

A set of complete analysis reports is also provided to effectively manage the space utilization, performance characteristics, and physical attributes of the databases. Each set of the reports is provided for the current database and the simulated database. The statistical information includes:

- Free space analysis, including statistics on the number and size of the free space elements and inter-database section references.
- Database record and segment occurrence profiles.

- Database record and segment placement analysis.
- Complete overflow usage analysis of both dependent overflow (DOVF) and independent overflow (IOVF).
- Physical I/O statistics for each segment type and for the typical database record.
- Root distribution and synonym chain analysis.
- Optional automated threshold reporting, which generates warning messages when conditions within the database do not meet user-specified criteria. This feature can be used to show when reorganization or expansion is required, or if database performance falls below acceptable standards.
- Largest database record analysis.
- Units of work (UOWs) that exceed user-specified reorganization criteria.

Generally, you run the Tune function multiple times with different keyword parameters to determine the optimal database attributes. The Tune function scans the areas, generates analysis records, and simulates the new database attributes by using the analysis records, but because analysis records are kept after the run, for the second and subsequent runs, you can omit the scan process and make the job run faster.

The Tune function, when used alone, simulates the database while the database is offline. However, when used in combination with the Online Pointer Checker utility of FPO, the Online Pointer Checker utility can scan the database while the database is online and generate the analysis records. The Tune function can then simulate the new database attributes based on the analysis records.

Simulating modes

The Tune function supports two modes for simulating databases: reorganization mode and restructure mode. The mode is automatically selected based on the conditions you provide for the simulation.

Reorganization mode

Reorganization mode simulates a reorganization of the specified areas to evaluate the effect of database reorganization.

Reorganization mode does not invoke the randomizer. Any change to the database structure, the DBD, or the randomizer is not supported.

The function runs in reorganization mode when database parameter changes are not requested. That is, when the following conditions are met:

- Resources for simulating the database are not provided.
- A new DMB is not provided.
- The randomizer is not changed.

Restructure mode

Restructure mode simulates structure changes in the specified areas. Restructure mode invokes a new randomizer to help you determine the new RAP numbers for each database record. Use this mode to evaluate the effect of changing the database structure, the DBD, and the randomizer.

The function runs in restructure mode when database parameter changes are requested. That is, when one or more of the following conditions are met:

- A new DMB is provided.
- A randomizer change is requested.
- Changes to the area definition are requested.

The mode is selected based on the JCL DD statements and some keywords. For details about how the mode is selected, see [“Running the Tune function” on page 233](#).

Input and output for the Tune function

The Tune function is invoked by the TUNE command in the HFPSYSIN DD control statement.

The SORT program can be invoked internally to achieve better performance by eliminating the SORTIN and the SORTOUT file I/O operations.

The following figure shows the input and the output for the Tune function.

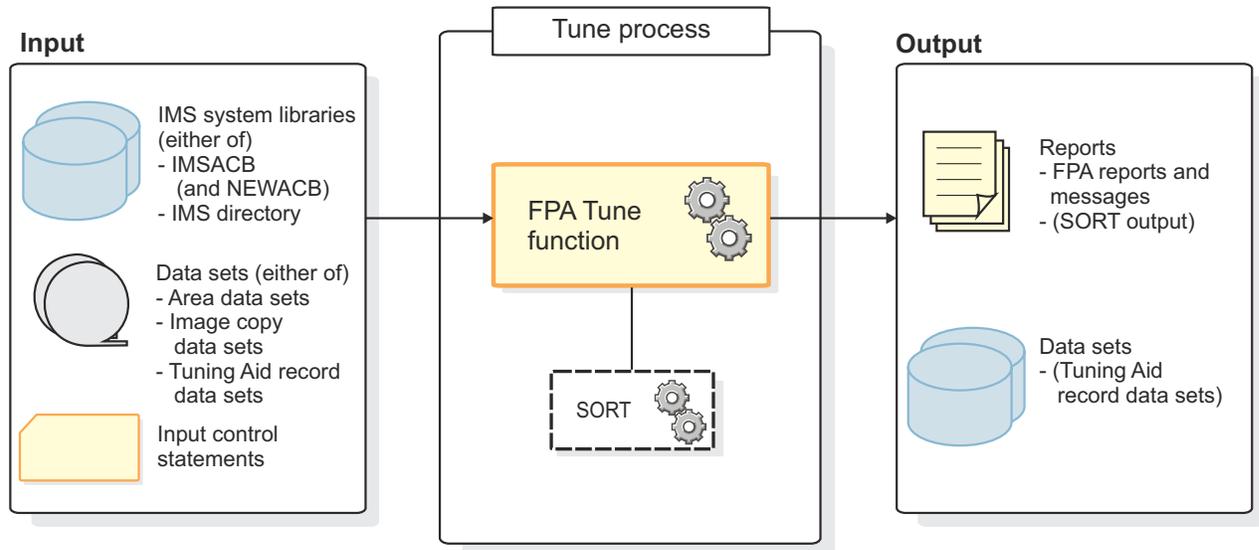


Figure 102. Input and output for the Tune function

The flow of the Tune process is as follows:

1. Scans the database and generates analysis records.
2. Using the analysis records and the DMB, calls the randomizer to simulate the new areas and the RAP number for each database record.
3. The function internally establishes an area model, inserts the segments into the model, and generates analysis records of the simulated areas.
4. Based on the two sets of analysis records, generates a tuning summary report and a set of analysis reports.

Tuning aid records

Tuning aid records are the sorted and resolved data analysis records. A tuning aid record data set contains the analysis records for an area. The Tune function uses the tuning aid records to simulate new database attributes.

Subsections:

- [“How tuning aid records are created and used” on page 231](#)
- [“Space requirement for tuning aid record data sets” on page 233](#)

How tuning aid records are created and used

Tuning aid records can be generated by the following three methods:

- FPA TUNE command with the AREASCAN=YES option
- FPO OPC utility with the TARECGEN=YES option
- FPO OER utility with the TARECGEN=YES option

Tuning aid records are stored in tuning aid record data sets. A tuning aid record data set cannot contain records of multiple areas. Therefore, you must prepare one tuning aid record data set for each area to simulate.

Tuning aid record data sets do not require DCB information to be specified in JCL.

TUNE command with the AREASCAN=YES option

When you specify AREASCAN=YES to scan the areas in the Tune job, the Tune function generates tuning aid records during the job and simulates the database by using the records.

The following table summarizes the methods for specifying tuning aid record data sets and the behavior of the Tune function.

Tuning aid record data sets specified with	Description
TARnnnnn DD statements	Generates the records in the data sets that are specified by the TARnnnnn DD statements. You must allocate tuning aid record data sets before you run the job.
TARDSNMASK keyword	Generates the records in the data sets that are specified by the TARDSNMASK keyword. You must allocate tuning aid record data sets before you run the job.
FILECTL subcommand with the DISP=NEW keyword	Dynamically generates tuning aid record data sets and generates the records in the data sets. You do not need to allocate tuning aid record data sets before you run the job.
FILECTL subcommand with the DISP=OLD or DISP=SHR keyword	Generates the records in the data sets that are specified by the FILECTL subcommand. You must allocate tuning aid record data sets before you run the job.
No DD statement or keyword specifies the data sets	Generates tuning aid record data sets as temporary data sets and generates the records in the data sets. The temporary data sets are deleted at the end of the job. You do not need to allocate tuning aid record data sets before you run the job.

TUNE command with the AREASCAN=NO option

When you specify AREASCAN=NO to skip the scanning of the areas in the Tune job, the Tune function obtains existing tuning aid records from the tuning aid record data sets and simulates the database by using the records. Therefore, tuning aid record data sets must exist and contain records before the job is run.

The following table summarizes the methods for specifying the tuning aid record data sets and the behavior of the Tune function.

Tuning aid record data sets specified with	Description
TARnnnnn DD statements	Obtains the records from the data sets that are specified by the TARnnnnn DD statements.

Tuning aid record data sets specified with	Description
TARDSNMASK keyword	Obtains the records from the data sets that are specified by the TARDSNMASK keyword.

When AREASCAN=NO is specified, the Tune function obtains tuning aid records from the existing tuning aid record data sets. You must specify all the tuning aid record data sets for the areas that you specify with the IAREA keyword.

FPO ANALYZE or PTRCHK statement with the TARECGEN=YES option

When you specify TARECGEN=YES on the ANALYZE or the PTRCHK statement of FPO, the job scans the online areas and generates tuning aid records. Areas are not simulated in the job, but you can use the generated tuning aid records as input for the Tune function.

The following table summarizes the methods for specifying the tuning aid record data sets.

Tuning aid record data sets specified with	Description
TARnnnnn DD statements	Generates the records in the data sets that are specified by the TARnnnnn DD statements. You must allocate tuning aid record data sets before you run the job.
TARDSNMASK keyword	Generates the records in the data sets that are specified by the TARDSNMASK keyword. You must allocate tuning aid record data sets before you run the job.

Space requirement for tuning aid record data sets

The space requirement for tuning aid record data sets varies depending on the size of the area, the number of free space elements, and the number of segments in the areas.

To estimate the space required for a tuning aid record data set, use the following formula. The unit is in bytes.

$$(\text{Number of CIs in the area} \times 30) + (\text{Number of segments in the area} \times (80 + \text{root key length}))$$

Running the Tune function

By running the Tune function, you can simulate changes in database parameters and generate tuning summary and analysis reports to determine the optimal parameters for the database.

About this task

The tuning aid records that are generated in a Tune job with AREASCAN=YES can be reused in subsequent Tune jobs. If tuning aid records exist, you can iteratively run the Tune function without scanning the database each time. By reusing the tuning aid records, you can run the Tune function faster and simulate various parameters with iterative runs with less amount of time.

Tip: If you want to simulate changes in an online database, you can use the FPO OPC utility or the OER utility with the TARECGEN=YES option to generate the tuning aid records. Then, use the tuning aid records as input to the Tune function.

Use the tuning summary and analysis reports generated by these runs and determine the optimal parameters for the database.

Procedure

1. Optional: Estimate the region size that is required to run the job.

See [“Region size requirements for the Tune function”](#) on page 236.

2. Specify the EXEC statement in the following format:

```
//TUNE EXEC PGM=HFPMAIN0,REGION=rrrrM,  
// PARM='DBRCGRP=dbrcgrp,GSGNAME=gsgname,IMSPLEX=imsplex,  
IMSID=imsid'
```

Parameter	Description
DBRCGRP=	DBRCGRP is an optional 1- to 3-character parameter that specifies the identifier assigned to a group of DBRC instances (DBRC sharing group) that access the same RECON data set in the IMSplex.
GSGNAME=	GSGNAME is an optional 1- to 8-character parameter that specifies the global service group.
IMSPLEX=	IMSPLEX is an optional 1- to 5-character parameter that specifies the names of the IMSplex groups that share databases or message queues.
IMSID=	IMSID is an optional 1- to 8-character parameter that specifies the IMS system identification name. If you specify this parameter, FPA generates the buffers for temporary ESCD and SCD. When FPA issues a call to a randomizing module, register 11 contains the address of the temporary ESCD. FPA sets only the ESCDSCD field, which contains the address of SCD, in the temporary ESCD. Also, FPA sets only the SSCDIMID field, which contains the IMS system ID, in the temporary SCD. Other fields of temporary ESCD and temporary SCD cannot be referred to.

3. Specify the DD statements to define input data sets and output data sets.

For descriptions of the DD statements, see [“DD statements for the Tune function”](#) on page 238.

To generate tuning aid records, tuning aid record data sets must be specified. Ensure that you specify a `TARnnnnn` DD statement for each area to simulate. `TARnnnnn` DD statements define the tuning aid record data sets for storing the tuning aid records.

4. Specify the utility control statements.

See [“Utility control statements for the Tune function”](#) on page 247 and specify the TUNE command, subcommands, and keywords.

- Ensure that `AREASCAN=NO` is not specified for the TUNE command unless tuning aid records are already generated. `AREASCAN=YES` (default) generates the tuning aid records in the corresponding tuning aid records data sets.
- To evaluate the effect of database reorganization with no database parameter changes (reorganization mode), do not specify the SIMULATE subcommand with the following keywords:
 - `AREADEF` subcommand keywords
 - `ACBLIB` keyword
 - `RMODLIB` keyword
 - `RMOD` keyword

If you request database parameter changes with these keywords, the job runs in restructure mode and evaluates the effect of database parameter changes.

- If you want to evaluate the areas by using thresholds, specify the `THRESHOLD` subcommand keywords to define the threshold values. When threshold values are defined, statistics values for the

evaluated factors, obtained from both the current areas and the simulated areas, and the difference between the values are printed in the Area Tuning Summary report.

5. Run the JCL job.

Results

The following reports are generated to help you tune the database:

- An Area Tuning Summary report that summarizes the effectiveness of the new database parameters.
- A set of analysis reports for the current database. These reports are the same as the reports that are generated by the Analyze process.
- A set of analysis reports for the simulated database. The format is the same as the analysis reports that are generated for the current database. You can compare these reports with the reports for the current database and evaluate whether the new database parameters are optimal.

Example

The following figure shows a JCL example for the TUNE command.

```
//*****  
//**      TUNE THE AREA(S)      **  
//*****  
//TUNE      EXEC PGM=HFPMAIN0  
//*  
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0  
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL  
//          DD DISP=SHR,DSN=IMSVS.PGMLIB  
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB  
//TAR00001 DD DISP=OLD,DSN=HPFP.TAR.DFSIVD3A  
//TAR00002 DD DISP=OLD,DSN=HPFP.TAR.DFSIVD3B  
//HFPPRINT DD SYSOUT=A  
//HFPRPTS DD SYSOUT=A  
//*  
//HFPSYSIN DD *  
GLOBAL  
DBRC=NO  
TUNE  
DBD=IVPDB3  
SIMULATE  
RMOD=RMOD5  
/*  
//DFSIVD3A DD DISP=SHR,DSN=IMSVS.DFSIVD31  
//DFSIVD3B DD DISP=SHR,DSN=IMSVS.DFSIVD33  
//*
```

Figure 103. JCL example for the TUNE command

What to do next

If you want to simulate the database with different database parameters, rerun the job with new keyword parameters. Because the tuning aid records are already created, you can run the job without the scan process. In the second and subsequent runs of the job, you can specify AREASCAN=NO to omit the scan process and shorten the elapsed time.

Related concepts

[Region size requirements for the Tune function](#)

The Tune function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Related reference

[Examples for the Tune function](#)

Review examples and learn techniques for tuning DEDB areas.

[Tips and best practices for the FPA Tune function](#)

The FPA Tune function supports several keywords that improve the performance of FPA Tune jobs and that further simplify management of DEDBs.

Related information

[DEDB monitoring and tuning guide](#)

You can use the utilities of IMS Fast Path Solution Pack to collect statistics data from DEDB areas, be notified of exceptional states of areas, and generate statistical reports to tune the areas.

Region size requirements for the Tune function

The Tune function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Region size for the Tune function depends on the size of a UOW, the number of areas to simulate, and the ITASKCTL and OTASKCTL specifications. Typically, database maintenance is a relatively important function. To increase the performance of the tune program, it would be advisable to run it in a region that has a relatively high dispatching priority (that is, at least equal to the dispatching priority of a message processing region).

Estimating the region size for Tune

The region size required for the Tune function is a function of the size of a UOW, the number of areas being analyzed, and the ITASKCTL and OTASKCTL specifications.

In most cases, the specification on the EXEC statement for HFPMAIN0 of a region that is too small, is detected during the initialization processing of the main task. This causes processing to be terminated at that point. Under certain conditions, the region size shortfall does not become evident until the actual tune process has begun. The job must be completely rerun. Thus, the region size calculations must always be performed prior to initiating tune processing.

The sort program also requires storage region. For how to estimate the region size, see the *DFSORT Application Programming Guide*.

The actual region size depends on the keywords that are specified for the job. Therefore, when you specify the region size in your JCL, use a higher value than the value calculated from the following worksheets.

The following worksheet is for estimating the region size that is required for a typical tune job with the AREASCAN=YES option.

FPA programs, internal tables, control blocks	3,020,000	
Number of areas defined in DBD * 1,300	-----	
Maximum Segment length * 1,100	-----	
Internal tables, control blocks	93,000	
Storage used by Sort program	-----	
(Maximum segment code + 1) * 350	-----	
subtotal	-----	
Value specified on ITASKCTL * subtotal	-----	
Number of input areas * 4,000	-----	
Total of CISZ * UOW1 of each area * 2	-----	
Total of CI size of each area	-----	
Total of X * (126 + MAX Segment Level * 4)	-----	
X : Total of maximum number of (RAP/PTF/PCF)		
per one UOW in each area		
Internal tables, control blocks	50,000	(If CISNAP=YES)
Storage used by Sort program	-----	(If CISNAP=YES)
CI size of the area where pointer errors are detected	-----	(If CISNAP=YES)
subtotal	-----	(If CISNAP=YES)
Number of areas where pointer errors are detected * subtotal	-----	(If CISNAP=YES)
Storage used by Sort program	-----	
Maximum number of UOW in all areas * 100	-----	
Total bytes of tuning storage	-----	

Figure 104. Worksheet for estimating the region size for the Tune function with AREASCAN=YES

The following worksheet is for estimating the region size that is required for a typical tune job with AREASCAN=NO.

- “Input DD statements” on page 239
- “Output DD statements” on page 243
- “Input/Output DD statements” on page 245

List of DD statements

The following table summarizes the DD statements for the Tune function.

Table 37. DD statements for the Tune function

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
JOBLIB or STEPLIB	Input	PDS	No	Required
IMSACB, IMSACBA, or IMSACBB	Input	PDS	No	Required when IMSCATHLQ=*NO
IMSDALIB	Input	PDS	No	Optional
MODSTAT	Input	PS	Yes	Optional
MODSTAT2	Input	PS	Yes	Optional
OLCSTAT	Input	PS	Yes	Optional
<i>addnname</i> or <i>areaname</i>	Input	VSAM ESDS cluster or PS (see Note)	Yes	Required under certain conditions (for details, see addnname/areaname DD)
TARnnnnn	Input and output	PS	Yes	Required under certain conditions (for details, see TARnnnnn DD)
HFPSYSIN	Input	LRECL=80	No	Required
HFPAPARM	Input	LRECL=80	No	Optional
HFPCSITE	Input	PDS	Yes	Optional
HFPPRINT	Output	SYOUT	Yes	Optional
HFPRPTS	Output	LRECL=133	Yes	Optional
HFPASNAP	Output	LRECL=133	No	Required when CISNAP=YES
HFPABEND	Output	LRECL=125	Yes	Optional
RECONn	Input and output	KSDS	Yes	Required when both of the following conditions are met: <ul style="list-style-type: none"> • DBRC=YES • DFSMDA members for RECONn data sets are not used
<i>itttWKnn</i>	Input and output		Yes	Optional
<i>itttMSG</i>	Output	LRECL=133	Yes	Optional

Note: If image copy is used as an input data set.

Input DD statements

The following input DD statements are used for the Tune function.

JOBLIB/STEPLIB DD

This DD statement is a required DD statement that defines the following input data sets:

- The FPA load module library (required).
- The library that contains DFSMDA dynamic allocation members for the RECON data set. When DBRC=YES is specified and the RECONn DD statement is omitted, this DD statement is required.
- The library that contains the randomizer routines and compression routines that are specified in the ACB member (specified by the IMSACB DD statements).
- The library that contains a default site default member (HFPCSITE) to determine the data set name and the member of the site default table.
- The IMS load module library (required).
- The IMS Tools Base library (SGLXLOAD), if you specify the IMSCATHLQ keyword.

FPA must be run as an APF-authorized program. All of the data sets that are listed within the JOBLIB or the STEPLIB concatenation must also be APF-authorized. If you use DFSMDA members and the DFSMDA members are in a non-APF authorized library, you can specify the DFSMDA library on the IMSDALIB DD statement.

To use a new randomizing routine that is in a non-APF authorized library, instead of specifying the library to the JOBLIB or STEPLIB concatenation, specify the library name on the RMODLIB keyword.

IMSACB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database to be analyzed.

If AREASCAN=NO is specified, this DD statement defines the library that contains the DMB for the database that was analyzed in a previous job with AREASCAN=YES or an FPO utility job with TARECGEN=YES.

The IMSACB DD statements (IMSACB, IMSACBA, IMSACBB, MODSTAT, MODSTAT2, and OLCSTAT) can be used to identify the active ACB library. When both OLCSTAT or MODSTAT and MODSTAT2 are found, FPA checks the time stamp and determines the latest one.

The active IMSACB library is determined according to the following rules:

1. Check IMSACB DD statement in the JCL statement.
2. Check OLCSTAT DD statement in the JCL statement.
3. Check MODSTAT and MODSTAT2 DD statement in the JCL statement.
4. Allocate OLCSTAT dynamically and check the OLCSTAT member.
5. Allocate MODSTAT and MODSTAT2 dynamically and check the MODSTAT and MODSTAT2 members.

If you specify the IMSCATHLQ keyword, IMSACB DD statements are not used. The IMS directory is used instead of the ACB library.

IMSACBA DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSACBB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSDALIB DD

This DD statement is an optional DD statement that defines the IMSDALIB data set. This DD statement is useful when the data set for dynamic allocation members is not APF-authorized.

MODSTAT DD

This DD statement is an optional DD statement that defines the MODSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

MODSTAT2 DD

This DD statement is an optional DD statement that defines the MODSTAT2 data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT2 library by using a TYPE=DATABASE statement of the DFSMDA macro.

OLCSTAT DD

This DD statement is an optional DD statement that defines the OLCSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the OLCSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

***addnname/areaname* DD**

This DD statement is a required DD statement when AREASCAN=YES is specified.

This statement defines an input data set, which is the VSAM ESDS or image copy for the area to be analyzed. The ddname used must be the ddname for the area that is defined in the input ACB library (IMSACB DD). There can be multiple *addnname* DD statements for each area included in the analyze request.

The input data set can be allocated dynamically in one of the following ways:

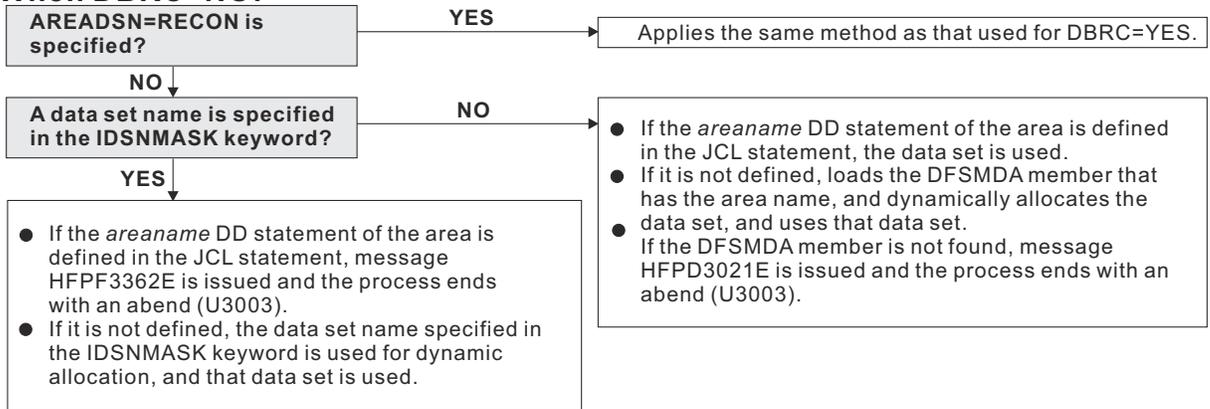
- The IDSNMASK keyword parameter of the TUNE command
- The ADS information in the DBRC RECON data sets (only for VSAM ESDS data sets)
- The area name member in the IMS DFSMDA library (only for VSAM ESDS data sets)

Or it can be specified with one of the following DD statements explicitly:

- *//addnname* DD JCL statement
- *//areaname* DD JCL statement

The following figure shows how FPA determines input area data set names for the TUNE command.

When DBRC=NO:



When DBRC=YES:

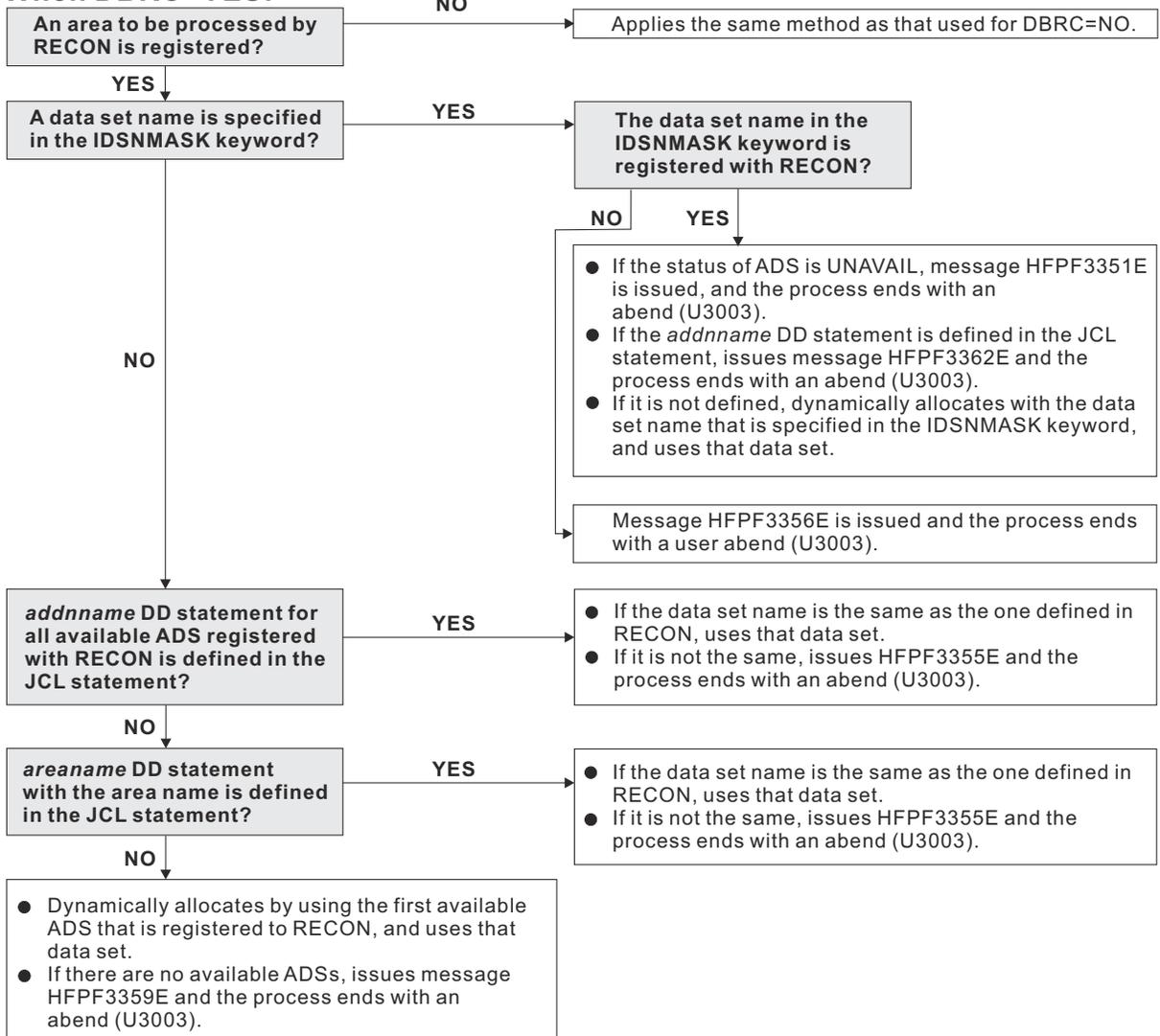


Figure 107. Input data set names for the TUNE command

HFPYSIN DD

This DD statement is a required DD statement that defines the input control statement that specifies the FPA functions.

The HFPYSIN DD statement can be coded as a standard SYSIN file, a sequential data set, or a PDS member. LRECL=80 is required for the DCB of this data set. Only one command, besides the GLOBAL

command, can be specified in HFPSYSIN. FPA analyzes the command syntax in columns 1 - 72. Columns 73 - 80 are regarded as comment.

For details about coding the HFPSYSIN DD statement, see [“Utility control statements for the Tune function”](#) on page 247.

HFAPARM DD

This DD statement is an optional DD statement. This statement defines the control parameters for generating the analysis reports. This statement is used only when you specify the REPORT subcommand.

The HFAPARM DD statement can be coded as a sequential data set or as a PDS member. LRECL=80 is required for the DCB of this data set. The HFAPARM keywords must be written on one line. It can be coded as follows:

```
//HFAPARM DD *  
  RDOVFFS=40 SYNTBL=20 UDOVFFS=35 RBASEFS=40  
/*
```

You can specify the keywords that are listed in the following table. All the keywords are optional.

Table 38. HFAPARM keywords

Keyword	Description
RBASEFS	If a RAP CI contains free space that is greater than or equal to the percentage specified, and the RAP CI also uses overflow CIs, the UOW is included in the UOW Exception report. The default is 100%. The maximum value is 99, and the minimum value is 0.
RDOVFFS	If the DOVF part of a UOW contains free space that is greater than or equal to the percentage specified, and the UOW also uses IOVF CIs, the UOW is included in the UOW Exception report. The default is 100%. The maximum value is 99, and the minimum value is 0.
SYNTBL	It controls the number of reporting entries in the Synonym Chain Analysis table. The default value is 41 (that is, all synonym chains longer than 40 are grouped and reported together). The maximum value is 999, and the minimum value is 12.
UDOVFFS	If the DOVF UOW contains free space that is less than or equal to the percentage specified, the UOW is included in the UOW Exception report. The default is 0%. The maximum value is 99, and the minimum value is 0.

Note: The RBASEFS keyword and the RDOVFFS keyword that are used for the HFAPARM data set are different from those used for the REORG command.

HFPCSITE DD

This DD statement is an optional DD statement. The HFPCSITE DD statement can be used to define the data set name of the site default table. When the SITE_DSNAME keyword is specified in the HFPSYSIN DD statement, the HFPCSITE DD statement is allocated dynamically. HFPCSITE DD in the JCL is mutually exclusive with the SITE_DSNAME keyword in the HFPSYSIN DD statement. The data set attribute of HFPCSITE DD is PDS.

Output DD statements

The following output DD statements are used for the Tune function.

HFPPRINT DD

This DD statement is an optional DD statement that defines the processing log output data set. The data set contains the processing messages that are issued by FPA.

The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set. You cannot specify DUMMY.

The HFPPRINT data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPPRINT DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

The following figure shows an example of the processing messages that are issued for the TUNE command.

```
HFPA0028I TUNE SCAN PROCESSING STARTED FOR DATABASE: DEDBJN22, AREA NO: 1, AREANAME: DB22AR0
HFPA0001I J000SMSG DYNAMICALLY ALLOCATED TO
- DSNAME: SYSOUT=*
HFPA0001I A000SMSG DYNAMICALLY ALLOCATED TO
- DSNAME: SYSOUT=*
HFPA0001I M001SMSG DYNAMICALLY ALLOCATED TO
- DSNAME: SYSOUT=*
HFPA0002I NO POINTER ERRORS WERE DETECTED IN DATABASE: DEDBJN22, AREA NO: 1, AREANAME: DB22AR0
HFPA0029I TUNE SCAN PROCESSING COMPLETED FOR DATABASE: DEDBJN22, AREA NO: 1, AREANAME: DB22AR0
HFPA0028I TUNE READ PROCESSING STARTED FOR DATABASE: DEDBJN22, AREA NO: 1, AREANAME: DB22AR0
HFPA0029I TUNE READ PROCESSING COMPLETED FOR DATABASE: DEDBJN22, AREA NO: 1, AREANAME: DB22AR0
HFPA0028I TUNE SIMULATE PROCESSING STARTED FOR DATABASE: DEDBJN22, AREA NO: 1, AREANAME: DB22AR0
HFPA0001I D001SMSG DYNAMICALLY ALLOCATED TO
- DSNAME: SYSOUT=*
HFPA0029I TUNE SIMULATE PROCESSING COMPLETED FOR DATABASE: DEDBJN22, AREA NO: 1, AREANAME: DB22AR0
HFPA0002I TUNE PROCESSING COMPLETED FOR DATABASE: DEDBJN22
HFPL0001I REPORT STARTED FOR COMMAND: TUNE, DATABASE: DEDBJN22
HFPL0002I REPORT COMPLETED FOR COMMAND: TUNE, DATABASE: DEDBJN22
```

Figure 108. Processing messages issued for the TUNE command

HFPRPTS DD

This DD statement is an optional DD statement that defines the report output data set. The data set contains the reports that can be used to verify the process of the command. The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

The following reports are generated in this data set.

- Audit report
- Processing report
- DBD Definition report
- Resolve Pointers report

When the REPORT subcommand is specified, the following reports are generated:

- DEDB Area Analysis reports
- Process Summary of Analysis report
- UOW report
- UOW Exception report

- Largest Database Records report (When LARGEREC_REPORT=YES is specified)

You can generate the following reports in another DD data set that is specified by the REPORTDD keyword parameter of the REPORT subcommand:

- DEDB Area Analysis reports
- UOW report
- UOW Exception report
- Largest Database Records report

The HFPRPTS data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPRPTS DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*.

HFPASNAP DD

This DD statement is a required DD statement when CISNAP=YES. This statement defines the output data set for the CI Map/CI Dump report and the Pointer Chain Reconstruction report. This statement is used only when you specify the CISNAP=YES keyword.

The following DCB attributes of the file are set by the Analyze process:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement.

The HFPASNAP data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPASNAP DD SYSOUT=A
```

HFPABEND DD

This DD statement is an optional DD statement. It defines a snap dump output data set that is used for diagnostic purposes. This data set might be used when the FPA job ends abnormally. Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*.

itttSMMSG DD

This DD statement is an optional DD statement. However, this DD statement is required when DFSORT is used. This statement defines an output data set for storing DFSORT messages. For the condition for running DFSORT and the naming rules for the DD names, see the description of *itttWKnn* DD.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*.

Input/Output DD statements

The following input/output DD statements are used for the Tune function.

TARnnnnn DD

These DD statements are optional DD statements. It points to the tuning aid record data set. A *TARnnnnn* DD statement specifies the tuning aid record data set for one area.

When AREASCAN=YES is specified, this statement defines the output data set for generating the tuning aid record. If this DD statement is not specified for an area, the function dynamically allocates a temporary data set and generates the record of that area in that data set.

When AREASCAN=NO is specified, this statement defines the input data set that contains the tuning aid record. You must specify one TARnnnnn DD statement per area for all the areas that are specified by the IAREA keyword.

A TARnnnnn DD statement is for areas in the range of 1 - 2048. The value of nnnnn must be right-aligned, with leading zeros if necessary.

This data set can be allocated dynamically by one of the following methods:

- The TARDSNMASK keyword parameter of the TUNE command
- The DSNNAME keyword parameter of the FILECTL subcommand

Or it can be specified with one of the following DD statements explicitly:

- //TARnnnnn DD JCL statement
- //ddname DD (The ddname is specified by the DDNAME keyword parameter of the FILECTL subcommand.)

This data set can also be defined by the FILECTL subcommand with the TUNE command.

For a description and space requirement for tuning aid records, see [“Tuning aid records” on page 231](#).

RECON1/RECON2/RECON3 DD

These statements define the DBRC RECON data sets. These DD statements are required DD statements when both of the following conditions are met:

- DBRC=YES is specified for the GLOBAL command
- DFSMDA members for RECONn data sets are not used

The RECONn data set must be the same data set that is used in the IMS control region. Because FPA issues a signon, a signoff, an authorization call, and an unauthorization call to DBRC, it updates RECONn data sets.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is necessary, the data set is allocated dynamically by using a DFSMDA member.

itttWKnn DD

These DD statements are optional DD statements. These data sets are intermediate storage data sets that are used by DFSORT. For more information about coding the SORTWKnn DD statements, see *DFSORT Application Programming Guide*.

The adequate size of storage data sets is 1.5 to 2.5 times the input file size. Allocating twice the space used by the SORTIN data set is usually adequate for each work data set.

Do not specify DISP=MOD for these DD statements.

The naming rules of the ddnames and the size of each input file are as follows:

i:

A, B, and C

30 x the number of pointers in the area

D, E, and F

Greater than one of the following values:

- {28 + (the maximum segment level x 4) + root key length} x the number of non-SDEP segments in the area
- 48 + (20 x TOTAL CIs of BASE, DOVF, and IOVF in the area) + (23 x the number of references to other CIs in the area)

G, H, and I: If CISNAP=YES is specified

$112 \times$ the number of pointers in which pointer errors are detected

J, K, and L: If the REPORT subcommand is specified

$X \times$ (the number of non-SDEP segments in the areas + the number of free space elements in the areas + the number of references to other CIs)

Where X is:

- When LARGEREC_REPORT=NO is specified, X is calculated as follows:

$$34 + (\text{the maximum segment level} \times 4)$$

However, if X is smaller than 48, 48 is used for X .

- When LARGEREC_REPORT=YES is specified, X is calculated as follows:

$$34 + (\text{the maximum segment level} \times 4) + \text{root key length}$$

However, if X is smaller than 48, 48 is used for X .

M, N, and O

$X \times$ the number of segments in the IOVF part

Where X is:

- When LARGEREC_REPORT=NO is specified, X is 34.
- When LARGEREC_REPORT=YES is specified, X is calculated as follows:

$$34 + \text{root key length}$$

The thousands digit of the area number of the area to be analyzed, or the thousands digit of n for n -th input thread subtask.

A, D, G, J, and M:

If the number of areas to be analyzed or n is 999 or less.

B, E, H, K, and N:

If the number of areas to be analyzed or n is 1000 - 1999.

C, F, I, L, and O:

If the number of areas to be analyzed or n is 2000 - 2048.

Note: When a subtask ends processing an area, it starts processing the next area. For this reason, the n -th input thread subtask does not always process the n -th input data set.

ttt:

The hundreds, tenths, and units digits of the area number of the area to be analyzed, or those of n .

nn:

The number of the SORT work data sets. This value must be 01 - 03.

Note on dynamic allocation:

If this statement is not specified in the JCL statement when these data sets are necessary, FPA allocates them dynamically.

Utility control statements for the Tune function

The utility control statements for the Tune function define the user description of the GLOBAL command, the TUNE command, and five optional subcommands, SIMULATE, AREADEF, FILECTL, REPORT, and THRESHOLD, all of which control the Tune process.

The utility control statements must be specified in the HFPSYSIN data set. This control statement data set usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte, fixed-length records. Block size, if coded, must be a multiple of 80.

The following figure is a sample showing how to code the HFPSYSIN data set.

```
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
  TUNE
  DBD=dbdname
SIMULATE
  ACBLIB=IMSVS.TEST.ACBLIB,
  RMODLIB=USER.RMODLIB
/*
```

Figure 109. TUNE HFPSYSIN data set

Subsections:

- [“GLOBAL command keywords” on page 248](#)
- [“TUNE command keywords” on page 249](#)
- [“SIMULATE subcommand keywords” on page 249](#)
- [“AREADEF subcommand keywords” on page 250](#)
- [“FILECTL subcommand keywords” on page 250](#)
- [“REPORT subcommand keywords” on page 251](#)
- [“THRESHOLD subcommand keywords” on page 252](#)

GLOBAL command keywords

For the GLOBAL command, you can specify the keywords that are listed in the following table. For the format of the GLOBAL command, see [“GLOBAL command” on page 394](#).

Table 39. GLOBAL command keywords

Keyword	Required or optional	Description	Topic
AREADSN	Optional	Specifies how to obtain area data set names.	“AREADSN keyword” on page 441
CISNAP	Optional	Specifies to request the CI Map/CI Dump report and the Pointer Chain Reconstruction report.	“CISNAP keyword” on page 448
DBRC	Optional	Specifies to request or bypass the processing of database recovery control (DBRC) facility for the database that is specified by the DBD.	“DBRC keyword” on page 451
IMSCATACB_INPUT	Optional	Specifies whether to retrieve the currently active ACB definition or the pending ACB definition from the IMS directory.	“IMSCATACB_INPUT keyword” on page 480
IMSCATHLQ	Optional	Specifies the high-level qualifier of the bootstrap data set of the IMS directory.	“IMSCATHLQ keyword” on page 481
MSTIMESTAMP	Optional	Specifies whether to include a time stamp in the messages.	“MSTIMESTAMP keyword” on page 497
SCAN	Optional	Specifies to verify the syntax of command, keywords, and parameters in command sets without running FPA. The Scan function also allocates some (but not all) internal resources for running the process function.	“SCAN keyword” on page 523
SITE_DSNAME	Optional	Specifies the data set name for the site default table.	“SITE_DSNAME keyword” on page 539
SITE_MEMBER	Optional	Specifies the member name for the site default table.	“SITE_MEMBER keyword” on page 540

TUNE command keywords

For the TUNE command, you can specify the keywords that are listed in the following table. For the format of the TUNE command, see [“TUNE command” on page 413](#).

Table 40. TUNE command keywords

Keyword	Required or optional	Description	Topic
AREASCAN	Optional	Requests to scan the areas and generate tuning aid record before simulating the database.	“AREASCAN keyword” on page 442
CISNAP	Optional	Specifies to request the CI Map/CI Dump report and the Pointer Chain Reconstruction report.	“CISNAP keyword” on page 448
DBD	Required	Specifies the database description (DBD) that contains the areas to be processed.	“DBD keyword” on page 450
IAREA	Optional	Specifies one or more input areas for the Tune process.	“IAREA keyword” on page 469
ICACHE	Optional	Specifies the type of control intervals that are to be preloaded into a data space storage for processing.	“ICACHE keyword” on page 472
ICHECK	Optional	Controls the concurrent number of input devices to access by using the ITASKCTL keyword.	“ICHECK keyword” on page 473
ICINCOMP	Optional	Specifies, when an image copy is used as an input data set, whether the image copy records are to be compressed by using the CSRCESTRV MVS macro when they are stored into a data space storage.	“ICINCOMP keyword” on page 474
IDSNMASK	Optional	Specifies the data set name or the mask for dynamically allocating cataloged pre-existing input data sets.	“IDSNMASK keyword” on page 476
ITASKCTL	Optional	Specifies the maximum number of input areas to be analyzed concurrently.	“ITASKCTL keyword” on page 486
MAXERROR	Optional	Specifies the maximum number of pointer errors that might be encountered before processing ends with an error.	“MAXERROR keyword” on page 495
OTASKCTL	Optional	Specifies the maximum number of input areas to be simulated concurrently.	“OTASKCTL keyword” on page 503
SORTOPT_FILSZ	Optional	Specifies that FPA adds the FILSZ parameter to the SORT control statement.	“SORTOPT_FILSZ keyword” on page 542
TARDSNMASK	Optional	Specifies the data set name or the mask for dynamically allocating cataloged pre-existing tuning aid record data sets.	“TARDSNMASK keyword” on page 549

SIMULATE subcommand keywords

The SIMULATE subcommand provides resources for simulating databases.

For the SIMULATE subcommand, you can specify the keywords that are listed in the following table. For the format of the SIMULATE subcommand, see [“SIMULATE subcommand” on page 434](#).

Table 41. SIMULATE subcommand keywords

Keyword	Required or optional	Description	Topic
ACBLIB	Optional	Specifies the name of the data set that contains the DMB that describes the database configuration to simulate.	“ACBLIB keyword” on page 439
IMSCATACB_OUTPUT	Optional	Specifies whether to retrieve the currently active ACB definition or the pending ACB definition from the IMS directory. This keyword specifies the new ACB definition, which is used when simulating changes in ACB definitions.	“IMSCATACB_OUTPUT keyword” on page 481
OAREA	Optional	Specifies one or more areas to be simulated.	“OAREA keyword” on page 497

Table 41. SIMULATE subcommand keywords (continued)

Keyword	Required or optional	Description	Topic
REPORTDD	Optional	Specifies the destination ddname for generating simulated analysis reports.	“REPORTDD keyword” on page 513
RMOD	Optional	Specifies the new randomizer name to replace.	“RMOD keyword” on page 521
RMODLIB	Optional	Specifies the name of the data set that contains the randomizer module to simulate.	“RMODLIB keyword” on page 521

AREADEF subcommand keywords

The AREADEF subcommand defines the new area definitions to simulate.

For the AREADEF subcommand, you can specify the keywords that are listed in the following table. For the format of the AREADEF subcommand, see [“AREADEF subcommand” on page 421](#).

Table 42. AREADEF subcommand keywords

Keyword	Required or optional	Description	Topic
CISIZE	Optional	Specifies to simulate the areas with a new CI size.	“CISIZE keyword” on page 448
OAREA	Optional	Specifies one or more areas to apply the definitions specified by this AREADEF subcommand.	“OAREA keyword” on page 497
ROOT	Optional	Specifies to simulate the areas with a new ROOT parameter. The ROOT parameter affects the total number of UOWs and the number of UOWs in the overflow area.	“ROOT keyword” on page 522
UOW	Optional	Specifies to simulate the areas with a new number of CIs for each unit of work (UOW).	“UOW keyword” on page 553

FILECTL subcommand keywords

The FILECTL subcommand defines the output files to be created during the Tune process. One or more tuning aid record data sets are produced by each FILECTL subcommand that is specified.

For the FILECTL subcommand, you can specify the keywords that are listed in the following table. For the format of the FILECTL subcommand, see [“FILECTL subcommand” on page 425](#).

Table 43. FILECTL subcommand keywords

Keyword	Required or optional	Description	Topic
AVGREC	Optional	Identifies the space requirements of the output data set that is created by dynamic allocation.	“AVGREC keyword” on page 445
DATACLAS	Optional	Specifies the SMS data class of an output data set that is created by dynamic allocation.	“DATACLAS keyword” on page 449
DDNAME	Optional	Specifies the ddnames for the data sets to be processed by the subcommand.	“DDNAME keyword” on page 453
DISP	Optional	Specifies the status and the disposition of the data set that is accessed by dynamic allocation.	“DISP keyword” on page 453
DSNAME	Optional	Specifies the mask for the data set name that is used for dynamic allocation of the data set to be processed by the subcommand.	“DSNAME keyword” on page 456
EXPDT	Optional	Specifies the expiration date of an output data set that is created by dynamic allocation. Do not specify both the EXPDT keyword and the RETPD keyword because they are mutually exclusive.	“EXPDT keyword” on page 462

Table 43. FILECTL subcommand keywords (continued)

Keyword	Required or optional	Description	Topic
IAREA	Optional	Specifies the areas for which you want a tuning aid record data set to be created.	“IAREA keyword” on page 469
LIKE	Optional	Specifies the SMS allocation attributes of an output data set that is created by dynamic allocation. The model data set specified must be an existing SMS data set.	“LIKE keyword” on page 492
MGMTCLAS	Optional	Specifies the SMS management class of an output data set that is created by dynamic allocation.	“MGMTCLAS keyword” on page 496
RETPD	Optional	Specifies the retention period of an output data set that is created by dynamic allocation. Do not specify both the EXPDT keyword and the RETPD keyword because they are mutually exclusive.	“RETPD keyword” on page 519
SPACE	Optional	Specifies the space requirements of the output data set that is created by dynamic allocation.	“SPACE keyword” on page 543
STORCLAS	Optional	Specifies the SMS storage class of the output data set that is created by dynamic allocation.	“STORCLAS keyword” on page 546
UNIT	Optional	Specifies the physical device type to be used for a data set that is accessed by dynamic allocation. Use this keyword also to specify the number of physical devices to be allocated.	“UNIT keyword” on page 552
VOLCNT	Optional	Specifies the maximum number of volumes that will be required by an output data set that is created by dynamic allocation that uses nonspecific volumes.	“VOLCNT keyword” on page 556
VOLSER	Optional	Specifies one or more volume-serial identifiers for the data set that is accessed by dynamic allocation. If this keyword is omitted for an output data set, nonspecific (private) volumes are required.	“VOLSER keyword” on page 556

REPORT subcommand keywords

The REPORT subcommand requests the analysis reports.

For the REPORT subcommand, you can specify the keywords that are listed in the following table. For the format of the REPORT subcommand, see [“REPORT subcommand” on page 432](#).

Table 44. REPORT subcommand keywords

Keyword	Required or optional	Description	Topic
DBRLDIST	Optional	Specifies whether to generate the DB Record Length Distribution report.	“DBRLDIST keyword” on page 452
LARGEREC_COUNT	Optional	Specifies the number of reporting entries in the Largest Database Records report and the Consolidated Largest Database Records report. This keyword is effective only when PTRCHKLVL=FULL is specified.	“LARGEREC_COUNT keyword” on page 491
LARGEREC_REPORT	Optional	Specifies whether to generate the Largest Database Records report. This keyword is effective only when PTRCHKLVL=FULL is specified.	“LARGEREC_REPORT keyword” on page 491
REPORTDD	Optional	Specifies the destination ddname for the analysis reports.	“REPORTDD keyword” on page 513
SEGLDIST	Optional	Specifies whether to generate the Segment Length Distribution report.	“SEGLDIST keyword” on page 529
SEGLSTAT	Optional	Specifies the format of the DB Record Profile Analysis report, which is one of the DEDB Area Analysis reports.	“SEGLSTAT keyword” on page 529
STARTUOW	Optional	Specifies the number that defines the first UOW to be reported in the UOW report and the UOW Exception report.	“STARTUOW keyword” on page 545

Table 44. REPORT subcommand keywords (continued)

Keyword	Required or optional	Description	Topic
STOPUOW	Optional	Specifies the number that defines the last UOW to be reported in the UOW report and the UOW Exception report.	“STOPUOW keyword” on page 546
UOW_REPORT	Optional	Specifies whether to generate the UOW report.	“UOW_REPORT keyword” on page 555

THRESHOLD subcommand keywords

The THRESHOLD subcommand specifies the threshold parameter values, which cause warning messages to be generated when the actual values do not meet the specified criteria.

For the THRESHOLD subcommand, you can specify the keywords that are listed in the following table. For the format of the THRESHOLD subcommand, see [“THRESHOLD subcommand” on page 435](#).

Table 45. THRESHOLD subcommand keywords

Keyword	Required or optional	Description	Topic
DOVFFS_PCT	Optional	Specifies a threshold setting for the percentage of free space in DOVF. If the percentage of free space in DOVF is less than the value that you specified, a warning message is issued.	“DOVFFS_PCT keyword” on page 455
FS_DOVFIOVF	Optional	Specifies a threshold setting for the free space in DOVF and in IOVF. If the free space in DOVF is greater than the value that you specified, and if the free space in IOVF is less than the value that you specified, a warning message is issued.	“FS_DOVFIOVF keyword” on page 466
FS_RAADOVF	Optional	Specifies a threshold setting for the free space in RAA BASE and in DOVF. If the free space in RAA BASE is greater than the value that you specified, and if the free space in DOVF is less than the value that you specified, a warning message is issued.	“FS_RAADOVF keyword” on page 467
FS_RAADOVFIOVF	Optional	Specifies a threshold setting for the free space in RAA BASE, DOVF, and IOVF. If the free space in RAA BASE is greater than the value that you specified, and if the free space in either DOVF or IOVF is less than the value that you specified, a warning message is issued.	“FS_RAADOVFIOVF keyword” on page 467
FS_RAATIOVF	Optional	Specifies a threshold setting for the free space in RAA BASE and in IOVF. If the free space in RAA BASE is greater than the value that you specified, and if the free space in IOVF is less than the value that you specified, a warning message is issued.	“FS_RAATIOVF keyword” on page 468
IOVFFS_PCT	Optional	Specifies a threshold setting for the percentage of free space in IOVF. If the percentage is less than value that you specified, a warning message is issued.	“IOVFFS_PCT keyword” on page 485
IOVFUSED_PCT	Optional	Specifies a threshold setting for the percentage of IOVF CIs used. If the percentage is greater than the value that you specified, a warning message is issued.	“IOVFUSED_PCT keyword” on page 485
RAAFS_PCT	Optional	Specifies a threshold setting for the percentage of free space in RAA BASE. If the percentage is less than the value that you specified, a warning message is issued.	“RAAFS_PCT keyword” on page 507
RAP_LONGSYN_PCT	Optional	Specifies a threshold setting for the percentage of RAPs that have a long synonym chain among all the RAPs that are used. If the percentage of the RAPs that have a long synonym chain is greater than the threshold value, a warning message is issued.	“RAP_LONGSYN_PCT keyword” on page 507
RAPOVFL_PCT	Optional	Specifies a threshold setting for the percentage of RAA BASE CIs that use either DOVF or IOVF. If the percentage is greater than the value that you specified, a warning message is issued.	“RAPOVFL_PCT keyword” on page 508
RECIO_AVG	Optional	Specifies a threshold setting for the average number of I/Os that can be done to access a database record. If the number is greater than the value that you specified, a warning message is issued.	“RECIO_AVG keyword” on page 510

Table 45. THRESHOLD subcommand keywords (continued)

Keyword	Required or optional	Description	Topic
RECIO_MAX	Optional	Specifies a threshold setting for the maximum number of I/Os that can be done to access a database record. If the number is greater than the value that you specified, a warning message is issued.	“RECIO_MAX keyword” on page 511
RECIOVF_PCT	Optional	Specifies a threshold setting for the percentage of records using IOVF CIs. If the percentage is greater than the value that you specified, a warning message is issued.	“RECIOVF_PCT keyword” on page 510
ROOTIO_AVG	Optional	Specifies a threshold setting for the average number of I/Os to access a root segment. If the number is greater than the value that you specified, a warning message is issued.	“ROOTIO_AVG keyword” on page 522
ROOTIO_MAX	Optional	Specifies a threshold setting for the maximum number of I/Os that can be done to access a root segment. If the number is greater than the value that you specified, a warning message is issued.	“ROOTIO_MAX keyword” on page 523
SYN_LEN_AVG	Optional	Specifies a threshold setting for the average synonym chain length. If the length is greater than the value that you specified, a warning message is issued.	“SYN_LEN_AVG keyword” on page 547
SYN_MAX	Optional	Specifies a threshold setting for the maximum synonym chain length. If the length is greater than the value that you specified, a warning message is issued.	“SYN_MAX keyword” on page 548
SYN_PCT_LEN	Optional	Specifies a threshold setting for the percentage of RAPs that have a long synonym chain among all the RAPs that have synonyms. If the percentage of the RAPs that have a long synonym chain is greater than the threshold value, a warning message is issued.	“SYN_PCT_LEN keyword” on page 548
UOWDOVF_PCT	Optional	Specifies a threshold setting for the percentage of UOWs that use DOVF. If the percentage is greater than the value that you specified, a warning message is issued.	“UOWDOVF_PCT keyword” on page 553
UOWIOVF_AVG	Optional	Specifies a threshold setting for the average number of IOVF CIs that are used by a UOW. If the number is greater than the value that you specified, a warning message is issued.	“UOWIOVF_AVG keyword” on page 554
UOWIOVF_MAX	Optional	Specifies a threshold setting for the maximum number of IOVF CIs that are used by a UOW. If the number is greater than the value that you specified, a warning message is issued.	“UOWIOVF_MAX keyword” on page 554
UOWIOVF_PCT	Optional	Specifies a threshold setting for the percentage of UOWs that use IOVF. If the percentage is greater than the value that you specified, a warning message is issued.	“UOWIOVF_PCT keyword” on page 555

Examples for the Tune function

Review examples and learn techniques for tuning DEDB areas.

In these examples, the following DEDB is assumed:

```

DEDB name:  DEDBJN22 has four areas
  area name:  DB22AR0 (area number is 1 and has three ADSs)
    ADS DD name:  DB22AR01,      DS name:  HPFP.DB22AR0.ADS1
    ADS DD name:  DB22AR02,      DS name:  HPFP.DB22AR0.ADS2
    ADS DD name:  DB22AR03,      DS name:  HPFP.DB22AR0.ADS3
  area name:  DB22AR1 (area number is 2 and has two ADSs)
    ADS DD name:  DB22AR11,      DS name:  HPFP.DB22AR1.ADS1
    ADS DD name:  DB22AR12,      DS name:  HPFP.DB22AR1.ADS2
  area name:  DB22AR2 (area number is 3 and has two ADSs)
    ADS DD name:  DB22AR21,      DS name:  HPFP.DB22AR2.ADS1
    ADS DD name:  DB22AR22,      DS name:  HPFP.DB22AR2.ADS2
  area name:  DB22AR3 (area number is 4 and has one ADS)
    ADS DD name:  DB22AR31,      DS name:  HPFP.DB22AR3.ADS1
    
```

In DBRC, all ADSs are in the AVAILABLE status.

Example 1: Evaluating the effect of the reorganization of a DEDB

The following figure shows a JCL example for simulating a reorganization of an area that is registered in DBRC.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS is dynamically allocated. The allocation information (DD name and DS name of the ADS) is obtained from DBRC.
- Because changes to the database parameters are not requested, the function simulates the effect of database reorganization without database parameter changes.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//        DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
TUNE
  DBD=DEDBJN22,
  IAREA=(DB22AR0)
REPORT
/*
```

Figure 110. Simulating a reorganization of an area

Example 2: Simulating changes in area definitions

The following figure shows a JCL example for simulating changes in area definitions of two areas that are registered in DBRC.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS is dynamically allocated. The allocation information (DD name and DS name of the ADS) is obtained from DBRC.
- The tuning aid records for area DB22AR0 and DB22AR1 are generated in the HPFP.TAR.DB22AR0 and HPFP.TAR.DB22AR1 data sets. The TAR00001 and TAR00002 DD statements specify these tuning aid record data sets.
- For area DB22AR0, the following conditions are simulated:
 - The UOW parameter is changed to (24,4).
 - The ROOT parameter is changed to (600,200).
- For area DB22AR1, the area is simulated with CI size of 8192 bytes.
- The threshold values of DOVFFS_PCT and IOVFFS_PCT are evaluated twice: for the current areas and for the simulated areas. The results are written in the Area Tuning Summary report.
- Two sets of analysis reports are generated in the data set pointed to by the HFPRPTS DD statement: one for the current areas and another for the simulated areas.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB   DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//TAR00001 DD DISP=SHR,DSN=HPFP.TAR.DB22AR0
//TAR00002 DD DISP=SHR,DSN=HPFP.TAR.DB22AR1
//HFPRPTS  DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
TUNE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1)
SIMULATE
  OAREA=(DB22AR0,DB22AR1),
  REPORTDD=HFPRPTS
AREADEF
  OAREA=(DB22AR0),
  UOW=(24,4),
  ROOT=(600,200)
AREADEF
  OAREA=(DB22AR1),
  CISIZE=8192
THRESHOLD
  DOVFFS_PCT=50,
  IOVFFS_PCT=80
REPORT
  REPORTDD=HFPRPTS
/*

```

Figure 111. Simulating changes in area definitions

Example 3: Evaluating a randomizer

The following figure shows a JCL example for evaluating a randomizer.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS is dynamically allocated. The allocation information (DD name and DS name of the ADS) is obtained from DBRC.
- The RMODLIB and the RMOD keywords specify the new randomizer, TSTRAND1 in USER.RMODLIB. FPA simulates placement of the segments with the new randomizer.
- The threshold values of SYN_LEN_AVG and SYN_MAX are evaluated twice: for the current areas and for the simulated areas. The results are written in the Area Tuning Summary report.
- The FILECTL subcommand specifies DSNAME='HPFP.TAR.&AREA', which causes a tuning aid record data set allocated dynamically for each area with the name of HPFP.TAR.*areaname*.
- Two sets of analysis reports are generated in the data set pointed to by the HFPRPTS DD statement: one for the current areas and another for the simulated areas.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
TUNE
  DBD=DEDBJN22,
  IAREA=ALL
SIMULATE
  OAREA=ALL,
  RMODLIB='USER.RMODLIB',
  RMOD=TSTRAND1,
  REPORTDD=HFPRPTS
THRESHOLD
  SYN_LEN_AVG=2,
  SYN_MAX=8
FILECTL
  DSNNAME='HPFP.TAR.&AREA',
  STORCLAS=STORCLAS,
  DISP=(NEW,CATLG),
  SPACE=(CYL,100,100)
REPORT
  REPORTDD=HFPRPTS
/*

```

Figure 112. Evaluating a randomizer

Example 4: Rerunning the simulation with different parameters

The following figure shows a JCL example for iterating the simulation.

This example assumes that the Tune job in [“Example 2: Simulating changes in area definitions”](#) on page 254 is already performed, and you want to change some parameters and run the Tune job again.

In this example:

- The areas are not allocated or scanned because AREASCAN=NO is specified.
- The input area information (tuning aid records) is obtained from the tuning aid record data sets that are specified by TARDSNMASK='HPFP.TAR.&AREA'. The data sets are allocated dynamically.
- For area DB22AR0, the following conditions are simulated:
 - The UOW parameter is changed to (30,10).
 - The ROOT parameter is changed to (500,100).
- For area DB22AR1, the area is simulated with CI size of 12288 bytes.
- The threshold values of DOVFFS_PCT and IOVFFS_PCT are evaluated twice: for the current areas and for the simulated areas. The results are written in the Area Tuning Summary report.
- Two sets of analysis reports are generated in the data set pointed to by the HFPRPTS DD statement: one for the current areas and another for the simulated areas.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//         DD DISP=SHR,DSN=IMSVS.SDFSRESL
//         DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
TUNE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  AREASCAN=NO,
  TARDSNMASK='HPFP.TAR.&AREA'
SIMULATE
  OAREA=(DB22AR0,DB22AR1),
  REPORTDD=HFPRPTS
AREADEF
  OAREA=(DB22AR0),
  UOW=(30,10),
  ROOT=(500,100)
AREADEF
  OAREA=(DB22AR1),
  CISIZE=12288
THRESHOLD
  DOVFFS_PCT=50,
  IOVFFS_PCT=80
REPORT
  REPORTDD=HFPRPTS
/*

```

Figure 113. Rerunning the simulation with different parameters

Example 5: Simulating changes in area definitions of online areas

The following figures show a JCL example for simulating the areas of an online DEDB. This JCL example consists of two steps. The first step is the FPO OPC step, and the second step is the FPA Tune step.

In the FPO OPC step:

- TARECGEN=YES causes the OPC utility to scan and analyze the online areas and generate a tuning aid record for each area.
- TARDSNMASK='HPFP.TAR.&AREA' specifies the tuning aid record data sets. The tuning aid records are generated in the tuning aid record data sets named HPFP.TAR.*areaname*. These data sets are allocated dynamically. Before the job is run, the data sets must be preallocated and cataloged.

In the FPA Tune step:

- The areas are not allocated or scanned because AREASCAN=NO is specified.
- The input area information (tuning aid records) is obtained from the tuning aid record data sets that are specified by TARDSNMASK='HPFP.TAR.&AREA'. The data sets are allocated dynamically.
- The database parameters used for this simulation are contained in the IMSVS.TEST.ACBLIB data set, which is specified by ACBLIB='IMSVS.TEST.ACBLIB'.
- Threshold values of RECIO_AVG, RECIO_MAX, and IOVFUSED_PCT are evaluated twice: for the current areas and for the simulated areas. The results are written to the Area Tuning Summary report.
- Two sets of analysis reports are generated in the data set pointed to by the HFPRPTS DD statement: one for the current areas and another for the simulated areas.

```

//ANALYZE EXEC FPUTIL, DBD=DEDBJN22, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//        DD DISP=SHR, DSN=IMSVS.SDFSRESL
//FPXIN   DD *
ANALYZE AREA=ALL,
        FULLSTEP=YES,
        TYPRUN=RPT,
        TARECGEN=YES,
        TARDSNMASK='HPFP.TAR.&AREA'
REPORT  REPORTDD=FPXRPTS
/*
//FPXRPTS DD SYSOUT=*
//SYSOUT  DD SYSOUT=*

```

Figure 114. Simulating changes in area definitions of online areas (FPO OPC step)

```

//HFP     EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//        DD DISP=SHR, DSN=IMSVS.SDFSRESL
//        DD DISP=SHR, DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR, DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR, DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
TUNE
  DBD=DEDBJN22,
  IAREA=ALL,
  AREASCAN=NO,
  TARDSNMASK='HPFP.TAR.&AREA'
SIMULATE
  OAREA=ALL,
  ACBLIB='IMSVS.TEST.ACBLIB',
  REPORTDD=HFPRPTS
THRESHOLD
  RECIO_AVG=2,
  RECIO_MAX=10,
  IOVFUSED_PCT=10
REPORT
  REPORTDD=HFPRPTS
/*

```

Figure 115. Simulating changes in area definitions of online areas (FPA Tune step)

Chapter 10. Extracting a DEDB offline with FPA

You can extract a DEDB by using the FPA Extract function.

Topics:

- [“Functions of Extract” on page 259](#)
- [“Input and output for the Extract function” on page 262](#)
- [“Running the Extract function” on page 262](#)
- [“Region size requirements for the Extract function” on page 263](#)
- [“DD statements for the Extract function” on page 265](#)
- [“Utility control statements for the Extract function” on page 272](#)
- [“Examples for the Extract function” on page 276](#)

Functions of Extract

The Extract function provides an easy-to-use, flexible method to extract segment data from one or more offline DEDB areas and writes the output to QSAM data sets.

The QSAM data set can be one of the following formats:

- Standard format
- FPB Unload/Reload format
- FPB Unload/Reload TFMT format
- TRIMAR Unload/Reload format
- HD Unload format
- User format

The Extract function can be used as an alternative to writing your own application to extract DEDB data. Other potential uses of this function include:

- Record reporting
- Extraction of records to populate a test database

Segment selection

Segment selection is done by using the SELECT subcommand.

The selection criteria can be based on:

- Key value
- Segment data content (On the basis of a positional offset, operator, and data type)

You can specify multiple comparisons per the SELECT subcommand by using the WHERE keyword. Each comparison supports:

- EQ, NE, GT, GE, LT, LE, ==, <>, >, >=, =>, <, <=, and =< operations
- Hexadecimal, character, fullword, and packed decimal data type

If you specify any comparison with the SELECT subcommand for a compressed segment, you must expand the segment with the COMPRESS=NO keyword.

The same segment name can appear in more than one SELECT subcommand. Each SELECT subcommand specification is evaluated independently for a match, which allows for Boolean OR logic between SELECT subcommands. You can do the following:

- Skip a number of segment occurrences

- Stop selection after a number of segment occurrences
- Select every *N*th occurrence of a segment

If the SELECT subcommand selection criteria are met, the selected segments, their dependents, and the parent hierarchy chain above the segments are forwarded for output processing.

If no SELECT subcommand is specified, all root segments and their dependent segments are selected, including SDEP segments between logical begin and logical end in the DMAC.

Related reference

WHERE keyword

Use the WHERE keyword to specify the criteria that is used to select segments for extraction.

Output file format

An OUTPUT subcommand controls the output selection and which data that matched the segment is written to the output file.

You can specify multiple comparisons per OUTPUT subcommand with the WHERE keyword to control output selection. Each comparison supports:

- EQ, NE, GT, GE, LT, LE, ==, <>, >, >=, =>, <, <=, and =< operations
- Hexadecimal, character, fullword, and packed decimal data type

You can specify the OUTPUT subcommand with the FIELDS keyword to change the style of the segment records that are written to the output file.

If no OUTPUT subcommands is specified, one record is written for each selected segment. When one or more OUTPUT subcommands are specified, segment records are written only for segments that are specified. No output record is written for segment types that are not specified on any OUTPUT subcommand. However, when the following format is used, a segment record is written for each selected segment, regardless of any OUTPUT subcommands that are specified.

- FPB Unload/Reload format
- FPB Unload/Reload TFMT format
- TRIMAR Unload/Reload format
- HD Unload format

Related reference

WHERE keyword

Use the WHERE keyword to specify the criteria that is used to select segments for extraction.

Samples of SELECT and OUTPUT commands

This topic provides samples for the SELECT and OUTPUT commands.

The following figure shows a sample of the segment hierarchy.

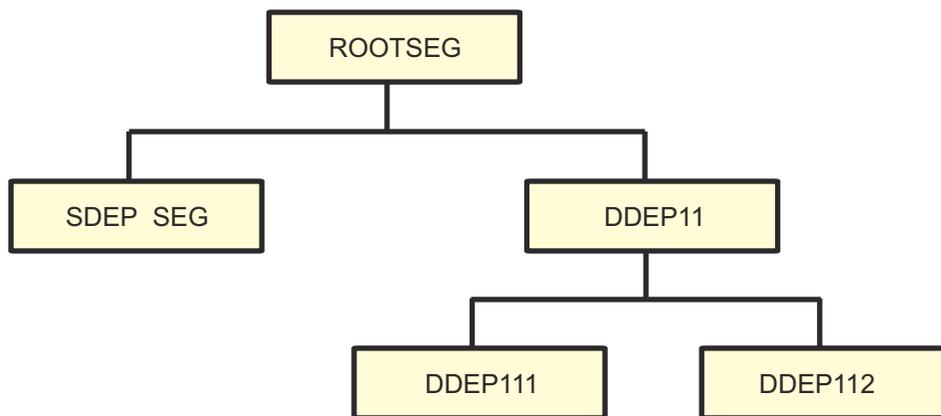


Figure 116. Sample segment hierarchy

The following table shows selected segments and segment output when EFORMAT=STD or EFORMAT=USER is specified.

Table 46. SELECT and OUTPUT subcommands with EFORMAT=STD or EFORMAT=USER

Command	Selected segments	Output segments
EXTRACT IAREA=ALL (No SELECT or OUTPUT subcommand is specified)	All roots and their dependent segments.	A segment record is written for each selected segment.
SELECT SEGMENT=ROOTSEG, WHERE=(criteria)	All ROOTSEG segments that meet the criteria and their dependents.	A segment record is written for each selected segment.
OUTPUT SEGMENT=DDEP112	All roots and their dependent segments.	A segment record is written for each DDEP112 that is selected.
SELECT SEGMENT=DDEP11, WHERE=(criteria) OUTPUT SEGMENT=DDEP111, WHERE=(criteria)	All roots, DDEP11 segments that meet the selection criteria, and their dependent segments (DDEP111 and DDEP112).	A segment record is written for each selected DDEP111 segment that meets the criteria on the OUTPUT subcommand.

The following table shows selected segments and segment output when EFORMAT=DBT, EFORMAT=TFMT, EFORMAT=UR or EFORMAT=HDUNLOAD is specified.

Table 47. SELECT and OUTPUT subcommands with EFORMAT=DBT, EFORMAT=TFMT, EFORMAT=UR or EFORMAT=HDUNLOAD

Command	Selected segments	Output segments
EXTRACT IAREA=ALL (No SELECT or OUTPUT subcommand is specified)	All roots and their dependent segments.	A segment record is written for each selected segment.
SELECT SEGMENT=ROOTSEG, WHERE=(criteria)	All ROOTSEG segments that meet the criteria and their dependents.	A segment record is written for each selected segment.
OUTPUT SEGMENT=DDEP112	All roots and their dependent segments.	A segment record is written for each selected segment.
SELECT SEGMENT=DDEP11, WHERE=(criteria) OUTPUT SEGMENT=DDEP111, WHERE=(criteria)	All roots, all SDEPSEG segments, DDEP11 segments that meet the selection criteria, and their dependent segments (DDEP111 and DDEP112).	All ROOTSEG and SDEPSEG segments are written. All DDEP11 (DDEP111 and DDEP112 children) segments that meet the selection criteria are written.

Input and output for the Extract function

The Extract function is invoked by the EXTRACT command in the HFPSYSIN DD control statement to extract multiple DEDB areas to QSAM data sets.

The SORT program can be invoked internally to achieve better performance by eliminating the SORTIN and the SORTOUT file I/O operation.

The extract function can process the following image copies of a DEDB as input data sets:

- Standard image copies that are created by IMS Image Copy utility (DFSUDUMP0) or IMS HP Image Copy
- Compressed image copies that are created by IMS HP Image Copy

The following figure shows the input and the output for the Extract function.

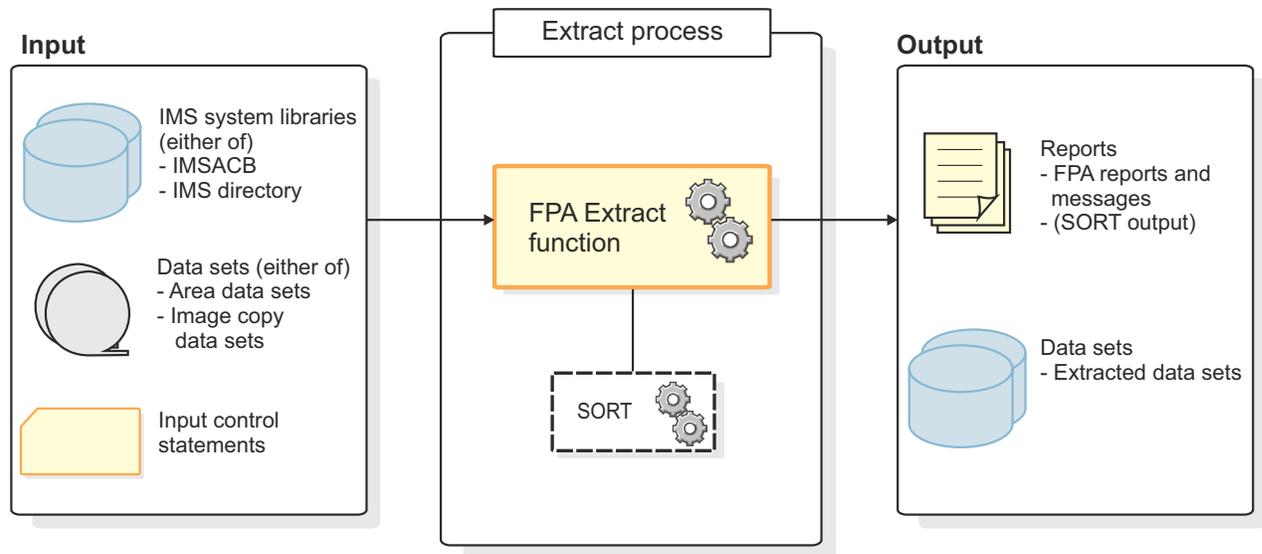


Figure 117. Input and output for the Extract function

Running the Extract function

FPA runs the Extract function as a standard z/OS batch job. You must specify an EXEC statement and DD statements that define the input and output data sets in your JCL.

Procedure

1. Optional: Estimate the region size that is required to run the job.
See [“Region size requirements for the Extract function”](#) on page 263.
2. Specify the EXEC statement in the following format:

```
//EXTRACT EXEC PGM=HFPMAIN0,REGION=rrrrM,  
// PARM='DBRCGRP=dbrcgrp,GSGNAME=gsgname,IMSPLEX=imsplex'
```

For the parameters, see the following table.

Parameter	Description
DBRCGRP=	DBRCGRP is an optional 1- to 3-character parameter that specifies the identifier assigned to a group of DBRC instances (DBRC sharing group) that access the same RECON data set in the IMSplex.
GSGNAME=	GSGNAME is an optional 1- to 8-character parameter that specifies the global service group.

Parameter	Description
IMSPLEX=	IMSPLEX is an optional 1- to 5-character parameter that specifies the IMSplex group names to share either databases or message queues.

3. Specify the DD statements to define input data sets and output data sets.
4. Run the JCL job.

Example

The following figure shows a JCL example for the EXTRACT command.

```

//*****
//**  EXTRACT THE AREA(S)  **
//*****
//EXTRACT EXEC PGM=HFPMAIN0
//*
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//         DD DISP=SHR,DSN=IMSVS.SDFSRESL
//         DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//HFPPRINT DD SYSOUT=A
//HFPRPTS DD SYSOUT=A
//*
//HFPSYSIN DD *
GLOBAL DBRC=NO
EXTRACT
    DBD=IVPDB3,
    EFORMAT=STD
SELECT
    SEGMENT=ROOTSEG,
    WHERE=(3:2 GT X'0123')
/*
//DFSIVD3A DD DISP=SHR,DSN=IMSVS.DFSIVD31
//DFSIVD3B DD DISP=SHR,DSN=IMSVS.DFSIVD33
//OAREA001 DD DISP=SHR,DSN=HPFP.OUTPUT1,SPACE=(CYL,5,1),
//          UNIT=SYSDA,VOL=HFPVOL
//OAREA002 DD DISP=SHR,DSN=HPFP.OUTPUT2,SPACE=(CYL,5,1),
//          UNIT=SYSDA,VOL=HFPVOL
//*

```

Figure 118. JCL example for the EXTRACT command

Related concepts

Region size requirements for the Extract function

The Extract function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Related reference

Examples for the Extract function

Review examples and learn techniques for extracting DEDB areas.

Tips and best practices for the FPA Extract function

The FPA Extract function supports several keywords that improve the performance of FPA Extract jobs and that further simplify management of DEDBs.

Region size requirements for the Extract function

The Extract function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Region size for the Extract function depends on the size of a UOW, the number of areas being extracted, the number of output files, and the ITASKCTL specifications. Typically, database maintenance is a relatively important function. To increase the performance of the extract program, it would be advisable to run it in a region that has a relatively high dispatching priority (that is, at least equal to the dispatching priority of a message processing region).

Estimating the region size for Extract

The region size required for the Extract function is a function of the size of a UOW, the number of areas being extracted, the number of output files, and the ITASKCTL specifications.

In most cases, the specification on the EXEC statement for HFPMAIN0 of a region that is too small, is detected during the initialization processing of the main task. This causes processing to be terminated at that point. Under certain conditions, the region size shortfall does not become evident until the actual extract has begun. The job must be completely rerun. Thus, the region size calculations must always be performed prior to initiating extract processing.

If SORT=YES is specified, the sort program also requires region space. For how to estimate the required region size, see the *DFSORT Application Programming Guide*.

The following worksheet is for estimating the region size that is required for a typical extract job. The actual region size depends on the keywords that are specified for the job. Therefore, when you specify the region size in your JCL, use a higher value than the value calculated from this worksheet.

FPA programs, internal tables, control blocks	3,020,000	
Number of areas defined in input DBD * 1,300	-----	
Internal tables, control blocks	18,200	
Maximum Segment length * 1,000	-----	
Maximum CISZ * 4	-----	
Internal I/O buffer	1,000,000	(If OCHECK=DEFER)
subtotal	-----	
Value specified on ITASKCTL * subtotal	-----	
Internal tables, control blocks	21,000	
Internal I/O buffer	1,000,000	(If OCHECK=NODEFER)
Storage used by Sort program	-----	(If SORT=YES)
subtotal	-----	
Number of output USR files * subtotal	-----	
Number of unloaded areas * 3,700	-----	
Total of CISZ * UOW1 of each area * 2	-----	
Total bytes of unload storage	-----	

Figure 119. Worksheet for estimating the region size for the Extract function

If ICACHE=YES is specified, the Extract function preloads IOVF and SDEP parts into the data space. The data space size required for the Extract function is a total of the size of IOVF and SDEP parts in each area. In addition, if the input data is an image copy data set, all image copy records are stored into the data space. In this case, the data space for the image copy data set must be added to the size that is calculated by the preceding worksheet.

The following figure shows a worksheet for estimating the data space size for the Extract function.

Total size of IOVF parts in all areas	-----	
Total size of SDEP parts in all areas	-----	
Total of CISZ * UOW1 * (R00T1 - R00T2) in each area	-----	(If input data is IC)
Total bytes of a data space storage	-----	

Figure 120. Worksheet for estimating data space size for the Extract function

DD statements for the Extract function

DD statements of the FPA Extract function determine the input and output data sets and specify how to run the Extract function.

Subsections:

- [“List of DD statements” on page 265](#)
- [“Input DD statements” on page 266](#)
- [“Output DD statements” on page 269](#)
- [“Input/Output DD statements” on page 271](#)

List of DD statements

The following table summarizes the DD statements for the Extract function.

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
JOBLIB or STEPLIB	Input	PDS	No	Required
IMSACB, IMSACBA, or IMSACBB	Input	PDS	No	Required when IMSCATHLQ=*NO
IMSDALIB	Input	PDS	No	Optional
MODSTAT	Input	PDS	Yes	Optional
MODSTAT2	Input	PDS	Yes	Optional
OLCSTAT	Input	PDS	Yes	Optional
<i>addnname</i> or <i>areaname</i>	Input	VSAM ESDS cluster or PS (Note)	Yes	Required under certain conditions (for details, see addnname/areaname DD)
OAREAnnn or OARnnnnn	Output	PS	Yes	Required when neither the ODSNMASK keyword nor the FILECTL subcommand is specified
HFPSYSIN	Input	LRECL=80	No	Required
HFPCSITE	Input	PDS	Yes	Optional
HFPPRINT	Output	SYSOUT	Yes	Optional
HFPRPTS	Output	LRECL=133	Yes	Optional
HFPTOTAL	Output	PS	No	Optional
HFPABEND	Output	LRECL=125	Yes	Optional
RECONn	Input and output	KSDS	Yes	Required when both of the following conditions are met: <ul style="list-style-type: none"> • DBRC=YES • DFSMDA members for RECONn data sets are not used
<i>cttWKnn</i>	Input and output	-	Yes	Optional

Table 48. DD statements for the Extract function (continued)

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
ctttSMMSG	Output	LRECL=133	Yes	Optional

Note: If image copy is used as an input data set.

Input DD statements

The following input DD statements are used for the Extract function.

JOBLIB/STEPLIB DD

This DD statement is a required DD statement. This DD statement defines the input data sets as follows:

- The FPA load module library (required).
- The library that contains DFSMDA dynamic allocation members for the RECON data set. When DBRC=YES is specified and the RECON n DD statement is omitted, this DD statement is required.
- The library that contains edit compression members that were specified in the IMSACB data set.
- The library that contains a default site default member (HFPCSITE) to determine the data set name and the member of the site default table.
- The IMS load module library (required).
- The IMS Tools Base library (SHKTLOAD), if you specify the ITKBSRVR keyword.
- The IMS Tools Base library (SGLXLOAD), if you specify the IMSCATHLQ keyword.

FPA must be run as an APF-authorized program. All of the data sets that are listed within the JOBLIB or the STEPLIB concatenation must also be APF-authorized. If you use DFSMDA members and the DFSMDA members are in a non-APF authorized library, you can specify the DFSMDA library on the IMSDALIB DD statement.

IMSACB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. The IMSACB DD statements (IMSACB, IMSACBA, IMSACBB, MODSTAT, MODSTAT2, and OLCSTAT) can be used to identify the active ACB library. When both OLCSTAT or MODSTAT and MODSTAT2 are found, FPA checks the time stamp and determines the latest one.

The active IMSACB library is determined according to the following rules:

1. Check IMSACB DD statement in the JCL statement.
2. Check OLCSTAT DD statement in the JCL statement.
3. Check MODSTAT and MODSTAT2 DD statement in the JCL statement.
4. Allocate OLCSTAT dynamically and check the OLCSTAT member.
5. Allocate MODSTAT and MODSTAT2 dynamically and check the MODSTAT and MODSTAT2 members.

If you specify the IMSCATHLQ keyword, IMSACB DD statements are not used. The IMS directory is used instead of the ACB library.

IMSACBA DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSACBB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSDALIB DD

This DD statement is an optional DD statement that defines the IMSDALIB data set. This DD statement is useful when the data set for dynamic allocation members is not APF-authorized.

MODSTAT DD

This DD statement is an optional DD statement that defines the MODSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

MODSTAT2 DD

This DD statement is an optional DD statement that defines the MODSTAT2 data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT2 library by using a TYPE=DATABASE statement of the DFSMDA macro.

OLCSTAT DD

This DD statement is an optional DD statement that defines the OLCSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the OLCSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

***addnname/areaname* DD**

This DD statement is a required DD statement. This statement defines an input data set, which is the VSAM ESDS or image copy for an area to be extracted. The *ddname* used must be the *ddname* for the area that is defined in the input ACB library (IMSACB DD). There can be multiple *addnname* DD statements for each area included in the extract request.

The input data set can be allocated dynamically in one of following ways:

- The IDSNMASK keyword parameter of the EXTRACT command
- When the GLOBAL DBRC=YES option is specified, the ADS information in the DBRC RECON data sets (only for VSAM ESDS data sets)
- The area name member in the IMS DFSMDA library (only for VSAM ESDS data sets)

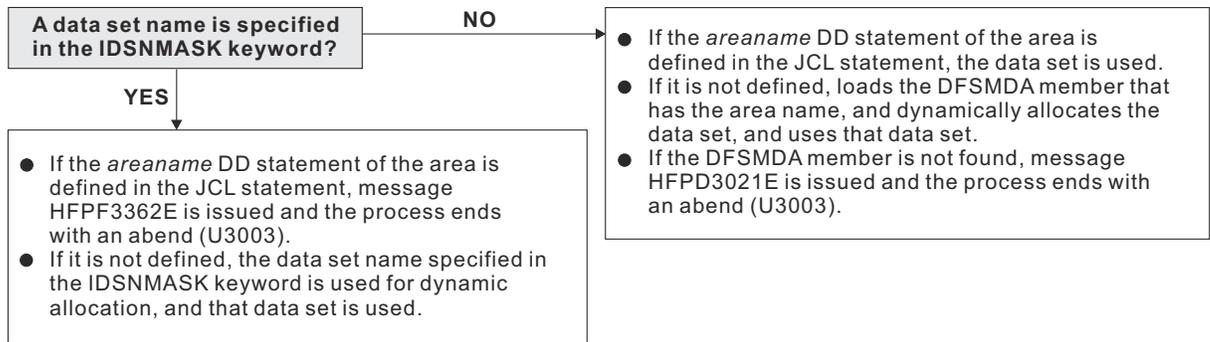
Or it can be specified with one of the following DD statements explicitly:

- *//addnname* DD JCL statement
- *//areaname* DD JCL statement

If an area data set is used for the EXTRACT command input, it must be error free (that is, it has no EQEs). And if the area is registered with DBRC, the data set must have no EEQEs.

The following figure shows how FPA determines area data set names.

When DBRC=NO:



When DBRC=YES:

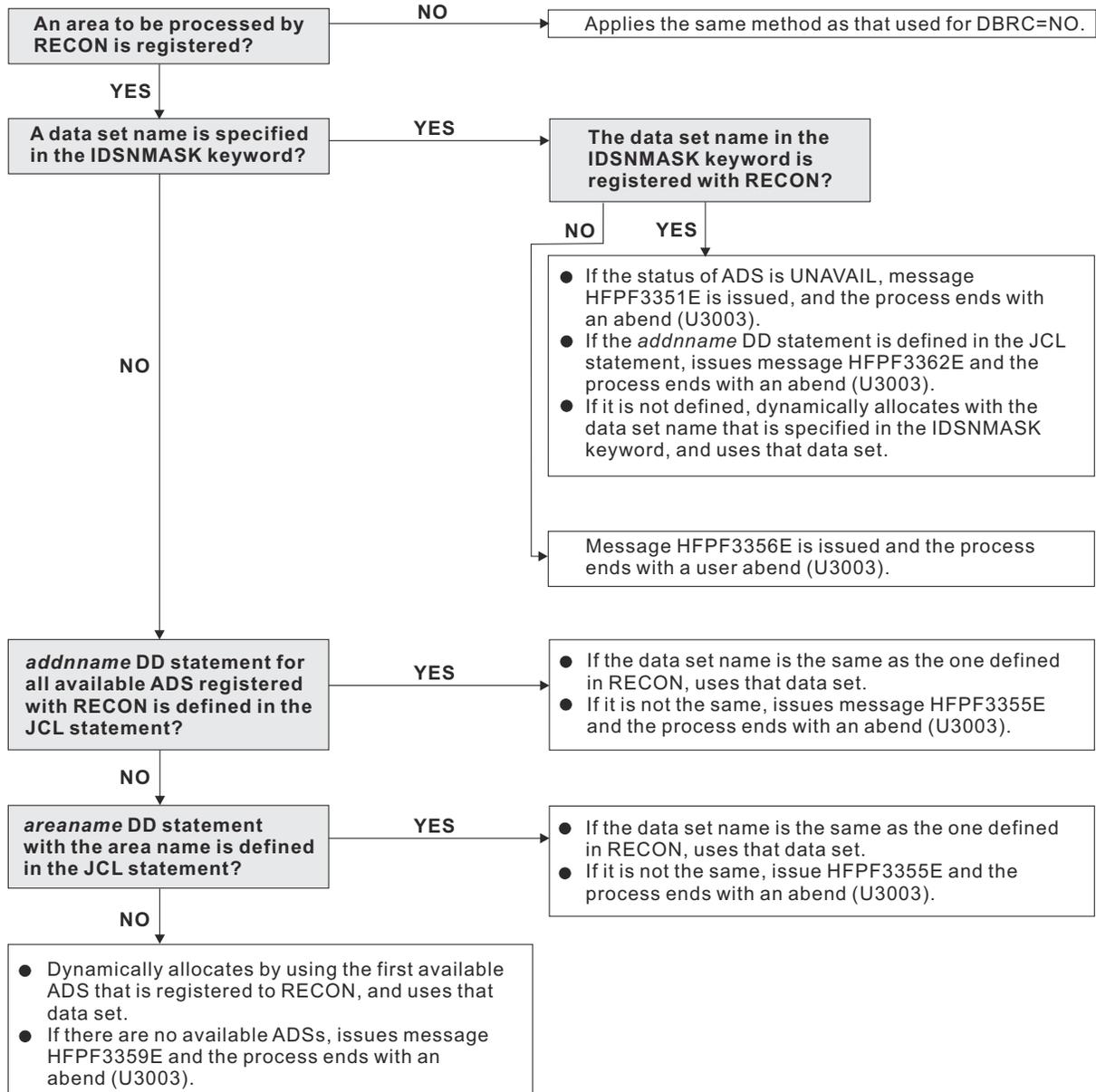


Figure 121. Input data set names for the EXTRACT command

HFPSYSIN DD

This DD statement is a required DD statement that defines the input control statement that specifies the FPA functions.

The HFPSYSIN DD statement can be coded as a standard SYSIN file, a sequential data set, or a PDS member. LRECL=80 is required for the DCB of this data set. Only one command, besides the GLOBAL command, can be specified in HFPSYSIN. FPA analyzes the command syntax in columns 1 - 72. Columns 73 - 80 are regarded as comment.

For details about coding the HFPSYSIN DD statement, see [“Utility control statements for the Extract function”](#) on page 272.

HFPCSITE DD

This DD statement is an optional DD statement. The HFPCSITE DD statement can be used to define the data set name of the site default table. When the SITE_DSNAME keyword is specified in the HFPSYSIN DD statement, the HFPCSITE DD statement is allocated dynamically. HFPCSITE DD in the JCL is mutually exclusive with the SITE_DSNAME keyword in the HFPSYSIN DD statement. The data set attribute of HFPCSITE DD is PDS.

Output DD statements

The following output DD statements are used for the Extract function.

OAREAnnn/OARnnnnn DD

This DD statement is a required DD statement when neither the ODSNMASK keyword nor the FILECTL subcommand is specified.

This statement defines the output data set that contains the extracted segment data records, each of which is composed of a prefix and segment data, for one or more areas. This statement defines an output data set for all of the database segment records that are produced for one or more of the areas defined in the "output" ACB library (IMSACB DD). This data set will be used for an input of the RELOAD command to reload areas when EFORMAT=DBT or EFORMAT=TFMT is specified.

This data set can be allocated dynamically by one of following methods:

- The ODSNMASK keyword parameter of the EXTRACT command
- The DSNAME keyword parameter of the FILECTL subcommand

Or it can be specified with one of the following DD statements explicitly:

- //OAREAnnn DD JCL statement
- /OARnnnnn DD JCL statement
- //ddname DD (The *ddname* is specified by the DDNAME keyword parameter of the FILECTL subcommand.)

This data set can be also defined by the FILECTL subcommand with the EXTRACT command.

An OAREAnnn DD statement is for areas in the range of 1 - 999, and an OARnnnnn DD statement is for areas in the range of 1 - 2048. If the area number of the extracted area is greater than 999, you must use the OARnnnnn DD statement. The value of *nnn* or *nnnnn* must be right-aligned, with leading zeros if necessary. Do not specify DISP=MOD for these DD statements.

For the format of an extracted segment record, see [“Extracted records layout”](#) on page 1186.

Note on dynamic allocation:

When this statement is not specified in the JCL statement, the data set is allocated dynamically by using the ODSNMASK keyword.

HFPPRINT DD

This DD statement is an optional DD statement that defines the processing log output data set. The data set contains the processing messages that are issued by FPA.

The following DCB attributes of the file are set by the FPA main task:

- RECFM=FB
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set. You cannot specify DUMMY.

The HFPPRINT data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPPRINT DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

The following figure shows an example of the processing messages that are issued for the EXTRACT command.

```
IMS HPFP UTILITIES                                "Processing messages"                                PAGE: 1
5698-FPP V2R1                                     2019-11-22 17:10:02
HFPC0001I COMMAND ANALYZE PROCESSING STARTED.
HFPC0005I DBRC=NO IS SPECIFIED.
- EEQE DETECTION IS NOT PERFORMED.
HFPC0002I COMMAND ANALYZE PROCESSING COMPLETED.
HFPF0001I EXTRACT PROCESSING STARTED FOR DATABASE: DEDBJN21
HFPJ0024I EXTRACT PROCESSING STARTED FOR EXTRACTED OUTPUT RECORD FILE.
- DDNAME: OAREA001, DSNAME: FPA.FPAUSR.DB21AR0
HFPJ0021I EXTRACT PROCESSING STARTED FOR AREA NO: 1, AREANAME: DB21AR0
HFPJ0022I EXTRACT PROCESSING COMPLETED FOR AREA NO: 1, AREANAME: DB21AR0
HFPJ0025I EXTRACT PROCESSING COMPLETED FOR EXTRACTED OUTPUT RECORD FILE.
- DDNAME: OAREA001, DSNAME: FPA.FPAUSR.DB21AR0
HFPP0002I EXTRACT PROCESSING COMPLETED FOR DATABASE: DEDBJN21
HFPL0001I REPORT STARTED FOR COMMAND: EXTRACT, DATABASE: DEDBJN21
HFPL0002I REPORT COMPLETED FOR COMMAND: EXTRACT, DATABASE: DEDBJN21
```

Figure 122. Processing messages issued for the EXTRACT command

HFPRPTS DD

This DD statement is an optional DD statement that defines the report output data set. The data set contains the reports that can be used to verify the process of the command. The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

The following reports are generated in this data set:

- Audit report
- Processing report
- DBD Definition report
- Extracted report
- Output File report

The HFPRPTS data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPRPTS DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

HFPTOTAL DD

This DD statement is an optional DD statement. This statement defines an output data set that contains the total number of extracted segments in an area. When this DD statement is present, the Extract process writes the total number of extracted segments of each segment type in an area.

The following DCB attributes of the file are set by the FPA main task:

- DSORG=PS
- LRECL=60
- BLKSIZE=6000

Do not code the DCB information in your JCL statement.

The format of this record is specified in the HFPZTOT macro, which is provided as a product-sensitive user interface. For details of the record format, see [“Total segments records layout” on page 1183](#).

HFPABEND DD

This DD statement is an optional DD statement. It defines a snap dump output data set that is used for diagnostic purposes. This data set might be used when the FPA job ends abnormally. Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

ctttSMMSG DD

This DD statement is an optional DD statement. However, this DD statement is required when DFSORT is used. This statement defines an output data set for storing DFSORT messages. For the condition for running DFSORT and the naming rules for the DD names, see the description of *ctttWKnn* DD.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

Input/Output DD statements

The following input/output DD statements are used for the Extract function.

RECON1/RECON2/RECON3 DD

These statements define the DBRC RECON data sets. These DD statements are required DD statements when both of the following conditions are met:

- DBRC=YES is specified for the GLOBAL command
- DFSMDA members for RECON n data sets are not used

The RECON n data set must be the same data set that is used in the IMS control region. Because FPA issues a signon, a signoff, an authorization call, and an unauthorization call to DBRC, it updates RECON n data sets.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is necessary, the data set is allocated dynamically by using a DFSMDA member.

ctttWKnn DD

These DD statements are optional DD statements. These data sets are available only when SORT=YES is specified as a keyword on the command. These data sets are intermediate storage data sets that are used by DFSORT. For more information about coding the SORTWK nn DD statements, see *DFSORT Application Programming Guide*.

The adequate size of storage data sets is 1.5 to 2.5 times the input file size. In the Extract function, the input file is the unloaded segment record file. The size of the file is computed by multiplying average length of the unloaded segment record by the number of segments. Allocating twice the space used by the SORTIN data set is usually adequate for each work data set. This data set is allocated dynamically by OS. When the SORTIN is too large for sort, a B37 abend occurs. In this case, this DD is a required DD.

Do not specify DISP=MOD for these DD statements.

The naming rules of the ddnames are as follows:

c:
The thousands digit of n for n -th extracted data set.

U:
If n is 999 or less, U must be specified.

V:
If n is 1000 - 1999, V must be specified.

W:
If n is 2000 - 2048, W must be specified.

ttt:
The hundreds, tenths, and units digits of n .

nn:
Specifies the number of the SORT work data sets. This value must be 01 - 03.

Note on dynamic allocation:

If this statement is not specified in the JCL statement when these data sets are necessary, FPA allocates them dynamically.

Utility control statements for the Extract function

The utility control statements for the Extract function define the user description of the GLOBAL command, the EXTRACT command, and the optional subcommands FILECTL, OUTPUT, SELECT, and USER_RECORD, all of which control the Extract process.

The utility control statements must be specified in the HFPSYSIN data set. This control statement data set usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte, fixed-length records. Block size, if coded, must be a multiple of 80.

The following figure shows a sample of how to code the HFPSYSIN data set.

```
//HFPSYSIN DD *  
GLOBAL  
DBRC=YES  
EXTRACT  
DBD=dbdname  
/*
```

Figure 123. EXTRACT HFPSYSIN data set

Subsections:

- [“GLOBAL command keywords” on page 272](#)
- [“EXTRACT command keywords” on page 273](#)
- [“FILECTL subcommand keywords” on page 274](#)
- [“OUTPUT subcommand keywords” on page 275](#)
- [“SELECT subcommand keywords” on page 275](#)
- [“USER_RECORD subcommand keywords” on page 276](#)

GLOBAL command keywords

For the GLOBAL command, you can specify the keywords that are listed in the following table. For the format of the GLOBAL command, see [“GLOBAL command” on page 394](#).

Table 49. GLOBAL command keywords

Keyword	Required or optional	Description	Topic
DBRC	Optional	Specifies to request or bypass the database recovery control (DBRC) facility process for the database that is specified by the DBD.	“DBRC keyword” on page 451
ICINCOMP	Optional	Specifies, when an image copy is used as an input data set, whether the image copy records are to be compressed by using the CSRCESTRV MVS macro when they are stored into a data space storage.	“ICINCOMP keyword” on page 474
IMSCATACB_INPUT	Optional	Specifies whether to retrieve the currently active ACB definition or the pending ACB definition from the IMS directory.	“IMSCATACB_INPUT keyword” on page 480
IMSCATHLQ	Optional	Specifies the high-level qualifier of the bootstrap data set of the IMS directory.	“IMSCATHLQ keyword” on page 481
MSTIMESTAMP	Optional	Specifies whether to include a time stamp in the messages.	“MSTIMESTAMP keyword” on page 497
SCAN	Optional	Specifies to verify the syntax of command, keywords, and parameters in command sets without running FPA. The Scan function also allocates some (but not all) internal resources for running the process function.	“SCAN keyword” on page 523
SITE_DSNAME	Optional	Specifies the data set name for the site default table.	“SITE_DSNAME keyword” on page 539
SITE_MEMBER	Optional	Specifies the member name for the site default table.	“SITE_MEMBER keyword” on page 540

EXTRACT command keywords

For the EXTRACT command, you can specify the keywords that are listed in the following table. For the format of the EXTRACT command, see [“EXTRACT command” on page 391](#).

Table 50. EXTRACT command keywords

Keyword	Required or optional	Description	Topic
COMPRESS	Optional	Specifies whether the extracted record should contain compressed segment if segment edit/compression routine is defined for the segment in DBDGEN. If COMPRTN is not specified for SEGM in the DBD statement, this keyword is ignored.	“COMPRESS keyword” on page 449
DBD	Required	Specifies the database description (DBD) that contains the areas to be processed.	“DBD keyword” on page 450
EFORMAT	Optional	Specifies the format of the extracted file.	“EFORMAT keyword” on page 460
IAREA	Optional	Specifies one or more input areas for the Extract process.	“IAREA keyword” on page 469
ICACHE	Optional	Specifies the type of control intervals that are to be preloaded into a data space storage for processing.	“ICACHE keyword” on page 472
ICHECK	Optional	Controls the concurrent number of input devices to access by using the ITASKCTL keyword.	“ICHECK keyword” on page 473
ICINCOMP	Optional	Specifies, when an image copy is used as an input data set, whether the image copy records are to be compressed data by using the CSRCESTRV MVS macro when they are stored into a data space storage.	“ICINCOMP keyword” on page 474
IDSNMASK	Optional	Specifies the data set name or the mask for dynamically allocating cataloged pre-existing input data sets.	“IDSNMASK keyword” on page 476

Table 50. EXTRACT command keywords (continued)

Keyword	Required or optional	Description	Topic
ITASKCTL	Optional	Specifies the maximum number of input areas to be processed concurrently.	“ITASKCTL keyword” on page 486
KEYSEQCHK	Optional	Specifies what to do if a key sequence error is detected.	“KEYSEQCHK keyword” on page 490
OCHECK	Optional	Controls the concurrent number of output devices to access.	“OCHECK keyword” on page 500
ODSNMASK	Optional	Specifies the data set name or the mask for dynamically allocating cataloged pre-existing output data sets.	“ODSNMASK keyword” on page 500
PTRCHK	Optional	Specifies what to do when a pointer error is detected.	“PTRCHK keyword” on page 505
SEGREC_PREFIX	Optional	Specifies the contents of the fields within the prefix portion of the segment records.	“SEGREC_PREFIX keyword” on page 531
SEGREC_SUFFIX	Optional	Specifies the contents of the fields within the suffix portion of the segment records.	“SEGREC_SUFFIX keyword” on page 533
SORT	Optional	Specifies whether to sort the extracted segment records on the basis of RAP (root anchor point).	“SORT keyword” on page 541
SORTOPT_FILSZ	Optional	Specifies that FPA adds the FILSZ parameter to the SORT control statement.	“SORTOPT_FILSZ keyword” on page 542

FILECTL subcommand keywords

The FILECTL subcommand defines the output files to be created during the Extract process. You can specify this subcommand to group the data targeted to one or more output areas into a single extracted data set. One or more extracted data sets are produced by each FILECTL subcommand that is specified. Do not specify the OAREA or the ODSNMASK keyword on both the EXTRACT command and the FILECTL subcommand. Specify the OAREA keyword on the FILECTL subcommand to specify the output areas to be placed into the extract data set.

For the FILECTL subcommand, you can specify the keywords that are listed in the following table. For the format of the FILECTL subcommand, see [“FILECTL subcommand” on page 425](#).

Table 51. FILECTL subcommand keywords

Keyword	Required or optional	Description	Topic
AVGREC	Optional	Identifies the space requirements of the output data set that is created by dynamic allocation.	“AVGREC keyword” on page 445
DATACLAS	Optional	Specifies the SMS data class of an output data set that is created by dynamic allocation.	“DATACLAS keyword” on page 449
DDNAME	Optional	Specifies the ddnames for the data sets to be processed by the subcommand.	“DDNAME keyword” on page 453
DISP	Optional	Specifies the status and the disposition of the data set that is accessed by dynamic allocation.	“DISP keyword” on page 453
DSNAME	Optional	Specifies a mask for the data set name that is used for dynamic allocation of the data set to be processed by the subcommand.	“DSNAME keyword” on page 456
EXPDT	Optional	Specifies the expiration date of an output data set that is created by dynamic allocation. Do not specify both the EXPDT keyword and the RETPD keyword because they are mutually exclusive.	“EXPDT keyword” on page 462
LIKE	Optional	Specifies the SMS allocation attributes of an output data set that is created by dynamic allocation. The model data set specified must be an existing SMS data set.	“LIKE keyword” on page 492

Table 51. FILECTL subcommand keywords (continued)

Keyword	Required or optional	Description	Topic
MGMTCLAS	Optional	Specifies the SMS management class of an output data set that is created by dynamic allocation.	“MGMTCLAS keyword” on page 496
OAREA	Optional	Specifies one or more output areas for the process.	“OAREA keyword” on page 497
RETPD	Optional	Specifies the retention period of an output data set that is created by dynamic allocation. Do not specify both the EXPDT keyword and the RETPD keyword because they are mutually exclusive.	“RETPD keyword” on page 519
SPACE	Optional	Specifies the space requirements of the output data set that is created by dynamic allocation.	“SPACE keyword” on page 543
STORCLAS	Optional	Specifies the SMS storage class of the output data set that is created by dynamic allocation.	“STORCLAS keyword” on page 546
UNIT	Optional	Specifies the physical device type to be used for a data set that is accessed by dynamic allocation. Use this keyword also to specify the number of physical devices to be allocated.	“UNIT keyword” on page 552
VOLCNT	Optional	Specifies the maximum number of volumes that will be required by an output data set that is created by dynamic allocation that uses nonspecific volumes.	“VOLCNT keyword” on page 556
VOLSER	Optional	Specifies one or more volume-serial identifiers for the data set that is accessed by dynamic allocation. If this keyword is omitted for an output data set, nonspecific (private) volumes are required.	“VOLSER keyword” on page 556

OUTPUT subcommand keywords

The OUTPUT subcommand controls the output selection and what data from a selected segment is to be written to the output file.

For the OUTPUT subcommand, you can specify the keywords that are listed in the following table. For the format of the OUTPUT subcommand, see [“OUTPUT subcommand” on page 431](#).

Table 52. OUTPUT subcommand keywords

Keyword	Required or optional	Description	Topic
FIELDS	Optional	Specifies the content of the segment data records that are written to the output file.	“FIELDS keyword” on page 463
SEGMENT	Required	Specifies the segment name to control the output selection and that specifies the content of the segment data records.	“SEGMENT keyword” on page 530
WHERE	Optional	Specifies how to control the output selection.	“WHERE keyword” on page 557

SELECT subcommand keywords

The SELECT subcommand specifies the segment name and the selection criteria for the segment.

For the SELECT subcommand, you can specify the keywords that are listed in the following table. For the format of the SELECT subcommand, see [“SELECT subcommand” on page 433](#).

Table 53. SELECT subcommand keywords

Keyword	Required or optional	Description	Topic
EVERY	Optional	Specifies to select every Nth occurrence of a segment starting from the first segment.	“EVERY keyword” on page 461

Table 53. SELECT subcommand keywords (continued)

Keyword	Required or optional	Description	Topic
SEGMENT	Required	Specifies the segment name to control segment selection.	“SEGMENT keyword” on page 530
SKIP	Optional	Specifies to skip a number of segment occurrences under a parent.	“SKIP keyword” on page 540
STOPAFT	Optional	Specifies the end of the selection of segments after a specific number of segment occurrences under a parent segment.	“STOPAFT keyword” on page 545
WHERE	Optional	Specifies to control segment selection.	“WHERE keyword” on page 557

USER_RECORD subcommand keywords

The USER_RECORD subcommand specifies the user-defined header and the trailer records that are to be written to an extract file.

For the USER_RECORD subcommand, you can specify the keywords that are listed in the following table. For the format of the USER_RECORD subcommand, see [“USER_RECORD subcommand” on page 437](#).

Table 54. USER_RECORD subcommand keywords

Keyword	Required or optional	Description	Topic
BREAK	Optional	Specifies the level and the timing for the trigger event for user-defined header and trailer of an extract file.	“BREAK keyword” on page 446
FIELDS	Required	Specifies the content of the header and trailer records that are to be written to the output file.	“FIELDS keyword” on page 463

Examples for the Extract function

Review examples and learn techniques for extracting DEDB areas.

In these examples, the following DEDB and the segment hierarchy are assumed.

```

DEDB name:  DEDBJN22 has four areas
  area name:  DB22AR0 (area number is 1 and has three ADSs)
    ADS DD name:  DB22AR01,      DS name:  HPFP.DB22AR0.ADS1
    ADS DD name:  DB22AR02,      DS name:  HPFP.DB22AR0.ADS2
    ADS DD name:  DB22AR03,      DS name:  HPFP.DB22AR0.ADS3
  area name:  DB22AR1 (area number is 2 and has two ADSs)
    ADS DD name:  DB22AR11,      DS name:  HPFP.DB22AR1.ADS1
    ADS DD name:  DB22AR12,      DS name:  HPFP.DB22AR1.ADS2
  area name:  DB22AR2 (area number is 3 and has two ADSs)
    ADS DD name:  DB22AR21,      DS name:  HPFP.DB22AR2.ADS1
    ADS DD name:  DB22AR22,      DS name:  HPFP.DB22AR2.ADS2
  area name:  DB22AR3 (area number is 4 and has one ADS)
    ADS DD name:  DB22AR31,      DS name:  HPFP.DB22AR3.ADS1
    
```

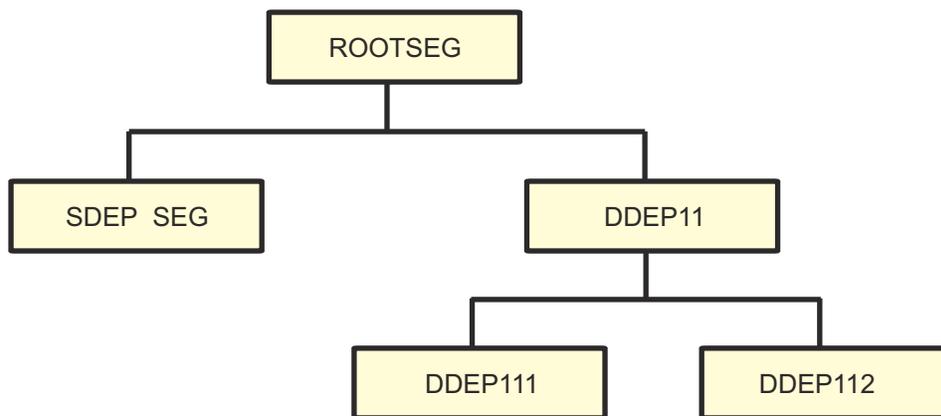


Figure 124. Example segment hierarchy structure

Example 1: Extracting segments with selection criteria

The following figure shows example JCL that is used for extracting segments with selection criteria.

In this example, segments with segment name ROOTSEG will be retrieved and written to the output file when:

- The character string 'Engineering' is found at position 3 for a length of 11.
- The number at position 15 for a length of 5 is greater than the number '10001216' in packed decimal.

The output file will be dynamically allocated by the FILECTL subcommand. The area name specified by the OAREA keyword of the FILECTL subcommand must be the same as the area name specified by the IAREA keyword of the EXTRACT command.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//         DD DISP=SHR,DSN=IMSVS.SDFSRESL
//         DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
EXTRACT
  DBD=DEDBJN22,
  IAREA=(DB22AR0)
SELECT SEG=ROOTSEG,
  WHERE=(3:11 EQ C'Engineering' AND 15:5 GT P'10001216')
OUTPUT SEG=ROOTSEG
FILECTL
  DSNAME='HPFP.OUTFILE',
  OAREA=DB22AR0,
  DISP=(NEW,CATLG,DELETE),
  SPACE=(CYL,5,1)
/*
  
```

Figure 125. Extracting segments with selection criteria

Example 2: Changing the style of the segment records

The following figure shows example JCL that is used to change the style of the segment records.

In this example, segments with segment name DDEP11 will be retrieved when the 3-byte data that starts at position 13 is equal to the character string 'J97'.

For each selected occurrence of the DDEP11 segment type, the 12-byte data that starts at offset 1 of the occurrence, and the 20-byte data that starts at offset 15 of the same occurrence will be concatenated and written to the output file that is specified by the DDNAME keyword of the FILECTL subcommand.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//        DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//OUTFILE DD DISP=(NEW,CATLG),DSN=HPFP.OUTFILE,SPACE=(CYL,(5,1)),
//        UNIT=SYSDA,VOL=SER=HFPVOL
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
EXTRACT
  DBD=DEDBJN22,
  IAREA=(DB22AR0)
SELECT
  SEG=DDEP11,
  WHERE=(13:3 EQ C'J97')
OUTPUT
  SEG=DDEP11,
  FIELDS=(1:12,15:20)
FILECTL
  OAREA=(DB22AR0),
  DDNAME=OUTFILE
/*

```

Figure 126. Changing the style of the segment records

Example 3: Creating a test database

The following figure shows example JCL that is used to extract segments with the selection criteria and to create a test database.

There are two steps in this example. The first step is an extract step to create an output file for the segments with the selection criteria. In this step, every fifth occurrence of the segment with segment name ROOTSEG and their dependent segments will be retrieved and written to the output file with the unloaded segment records file format. The second step is a reload step to create a test database by using the output file. In this step, the output ADS DD statement name (DB22AR0) and the DS name (HPFP.TEST.DB22AR0) are dynamically defined and allocated by using the ODSNMASK parameter and the ALLOCATE subcommand.

```

//*****
//*  EXTRACT STEP                               *
//*****
//EXTRACT EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//OAREA001 DD DISP=(NEW,PASS),DSN=&&OAREA001,SPACE=(CYL,(5,1)),
//          UNIT=SYSDA
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
EXTRACT
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  EFORMAT=DBT,
  SORT=YES
  SELECT SEG=ROOTSEG,
  EVERY=5
/*
//*****
//*  RELOAD STEP                               *
//*****
//RELOAD EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IAREA001 DD DISP=(OLD,DELETE),DSN=&&OAREA001
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=NO
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  OAREA=(DB22AR0),
  ODSNMASK='HPFP.TEST.&AREA'
ALLOCATE
  OAREA=(DB22AR0),
  SPACE=(CYL,5),
  VOLSER=HFPVOL
/*

```

Figure 127. Creating a test database

Example 4: Extracting segments in user-defined format

The following figure shows example JCL that is used to extract segments in user-defined format.

In this example, the following records will be written to the output file that is specified by the OAREA001 DD statement:

- The number of areas in a DBD, which is defined by the USER_RECORD subcommand, is written as the database-level header.
- The area name, which is defined by the USER_RECORD subcommand, is written as the area-level header.
- Each segment record with its segment code, which is defined by the SEGREC_PREFIX keyword, is written in the prefix portion of the segment records, and the segment concatenated key and its length, which is defined by the SEGREC_SUFFIX keyword, is written in the suffix portion of the segment records.
- The area name, which is defined by the USER_RECORD subcommand, is written as the area-level trailer.
- The number of areas in a DBD, which is defined by the USER_RECORD subcommand, is written as the database-level trailer.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//        DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//OAREA001 DD DISP=(NEW,CATLG),DSN=HPFP.OUTFILE,SPACE=(CYL,(5,1)),
//        UNIT=SYSDA,VOL=SER=HFPVOL
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
EXTRACT
  DBD=DEDBJN22,
  IAREA=(DB22AR0),
  EFORMAT=USER,
  SEGREC_PREFIX=(SEGMENT_CODE),
  SEGREC_SUFFIX=(SEGMENT_CKEY_LENGTH,SEGMENT_CKEY)
USER_RECORD
  BREAK=(DATABASE,BEFORE),
  FIELDS=(DBD_AREAS)
USER_RECORD
  BREAK=(AREA,BEFORE),
  FIELDS=(AREA_NAME)
USER_RECORD
  BREAK=(AREA,AFTER),
  FIELDS=(AREA_NAME)
USER_RECORD
  BREAK=(DATABASE,AFTER),
  FIELDS=(DBD_AREAS)
/*

```

Figure 128. Extracting segments in user-defined format

Example 5: Extracting segments into designated output data sets

The following figure shows example JCL that is used for designating the output data sets to write the extracted segments into.

In this example, every fifth occurrence of the segment with segment name ROOTSEG and their dependent segments will be retrieved and written to the output file that is specified by the DDNAME keyword on each FILECTL subcommand.

- The SELECT subcommand specifies to retrieve every fifth occurrence of the segment with segment ROOTSEG.
- The first FILECTL subcommand specifies to retrieve segments from area DB22AR0 and write them into area OAREA001.
- The second FILECTL subcommand specifies to retrieve segments from area DB22AR1, DB22AR2, and DB22AR3 and write them into area OAREA002.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB   DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//OAREA001 DD DISP=(NEW,CATLG),DSN=HPFP.OUTFILE,SPACE=(CYL,(5,1)),
//          UNIT=SYSDA,VOL=SER=HFPVOL
//OAREA002 DD DISP=(NEW,CATLG),DSN=HPFP.OUTFILE2,SPACE=(CYL,(5,1)),
//          UNIT=SYSDA,VOL=SER=HFPVOL
//HFPRPTS  DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
EXTRACT
  DBD=DEDBJN22,
  IAREA=ALL,
  FORMAT=STD
SELECT
  SEG=ROOTSEG,
  EVERY=5
FILECTL
  OAREA=DB22AR0,
  DDNAME=OAREA001
FILECTL
  OAREA=(DB22AR1,DB22AR2,DB22AR3),
  DDNAME=OAREA002
/*

```

Figure 129. Extracting segments into designated output data sets

Chapter 11. Printing DMAC blocks in a DEDB offline with FPA

You can print DMAC blocks by using the FPA DMAC Print function.

Topics:

- [“Functions of DMAC Print” on page 283](#)
- [“Input and output for the DMAC Print function” on page 283](#)
- [“Running the DMAC Print function” on page 284](#)
- [“Region size requirements for the DMAC Print function” on page 285](#)
- [“DD statements for the DMAC Print function” on page 286](#)
- [“Utility control statements for the DMAC Print function” on page 291](#)
- [“Examples for the DMAC Print function” on page 292](#)

Functions of DMAC Print

The DMAC Print function provides maps of the contents of the DMAC blocks for one or more areas by using the IMS mapping macro and the DMAC dump.

DMAC is an important CI that has the information for ROOT, UOW definition, and SDEP LB/LE values, and so on, for a DEDB area data set. The DMAC Print function maps the global section and the local section with offset, label, and values. It also provides the dump that is specified by the DMAC length, EQE list, CUSN, and RBA.

Input and output for the DMAC Print function

The DMAC Print function is invoked by the DMACPRT command in the HFPSYSIN DD control statement to print DMAC blocks in a DEDB.

Use the DMAC Print function to map the contents of the DMAC block for one or more areas by using the IMS mapping macro, and to generate the DMAC dump.

The DMAC Print function can process the following image copies of a DEDB as input data sets:

- Standard image copies that are created by IMS Image Copy utility (DFSUDUMP0) or IMS HP Image Copy
- Compressed image copies that are created by IMS HP Image Copy

The following figure shows the input and the output for the DMAC Print function.

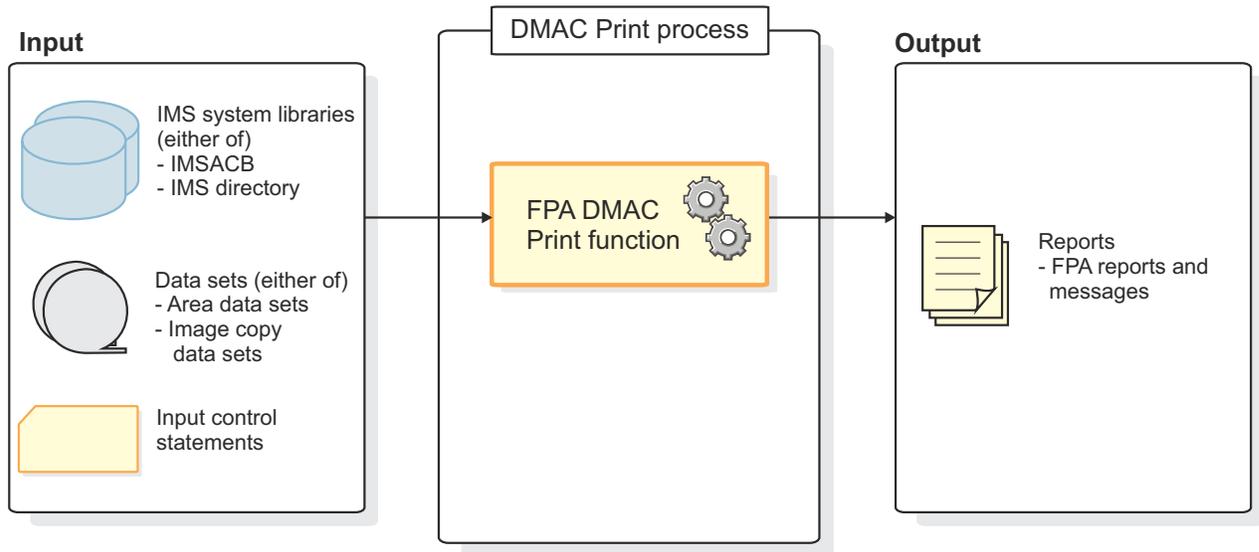


Figure 130. Input and output for the DMAC Print function

Running the DMAC Print function

FPA runs the DMAC Print function as a standard z/OS batch job. You must specify an EXEC statement and DD statements that define the input and output data sets in your JCL.

Procedure

1. Optional: Estimate the region size that is required to run the job.
See [“Region size requirements for the DMAC Print function”](#) on page 285.
2. Specify the EXEC statement in the following format:

```
//DMACPRT EXEC PGM=HFPMAIN0,REGION=rrrrM,
//          PARM='DBRCGRP=dbrcgrp,GSGNAME=gsgname,IMSPLEX=impsplex'
```

For the parameters, see the following table.

Parameter	Description
DBRCGRP=	DBRCGRP is an optional 1- to 3-character parameter that specifies the identifier assigned to a group of DBRC instances (DBRC sharing group) that access the same RECON data set in the IMSplex.
GSGNAME=	GSGNAME is an optional 1- to 8-character parameter that specifies the global service group.
IMSPLEX=	IMSPLEX is an optional 1- to 5-character parameter that specifies the IMSplex group names to share either databases or message queues.

3. Specify the DD statements to define input data sets and output data sets.
4. Run the JCL job.

Example

The following figure shows a JCL example for the DMACPRT command.

```

//*****
//**  DMACPR THE AREA(S)    **
//*****
//DMACPR EXEC PGM=HFPMAIN0
//*
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//        DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//HFPPRINT DD SYSOUT=A
//HFPRPTS DD SYSOUT=A
//HFPSYSIN DD *
GLOBAL DBRC=NO
DMACPR
DBD=IVPDB3
/*
//DFSIVD3A DD DISP=OLD,DSN=IMSVS.DFSIVD31
//DFSIVD3B DD DISP=OLD,DSN=IMSVS.DFSIVD33
//*

```

Figure 131. JCL example for the DMACPR command

Related concepts

[Region size requirements for the DMAC Print function](#)

The DMAC Print function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Related reference

[Examples for the DMAC Print function](#)

Review examples and learn techniques for printing DMAC blocks in DEDB areas.

[Tips and best practices for the FPA DMAC Print function](#)

The FPA DMAC Print function supports a keyword that helps you use the function more effectively and efficiently.

Region size requirements for the DMAC Print function

The DMAC Print function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Region size for the DMAC Print function depends on the number of input areas. Typically, database maintenance is a relatively important function. To increase the performance of the DMAC Print program, it would be advisable to run it in a region that has a relatively high dispatching priority (that is, at least equal to the dispatching priority of a message processing region).

Estimating the region size for DMAC Print

The region size required for the DMAC Print function is a function of the number of areas of which their DMAC blocks are printed.

In most cases, the specification on the EXEC statement for HFPMAIN0 of a region that is too small, is detected during the initialization processing of the main task. This causes processing to be terminated at that point. Under certain conditions, the region size shortfall does not become evident until the actual DMAC Print processing has begun. The job must be completely rerun. Thus, the region size calculations should always be performed prior to initiating DMAC Print processing.

The following worksheet is for estimating the region size that is required for a typical DMAC print job. The actual region size depends on the keywords that are specified for the job. Therefore, when you specify the region size in your JCL, use a higher value than the value calculated from this worksheet.

DMAC Print programs, internal tables, control blocks	3,020,000
Number of areas defined in DBD * 1,300	-----
Number of input areas * 4,000	-----
Total bytes of dmacprt storage	-----

Figure 132. Worksheet for estimating region size of the DMAC Print function

DD statements for the DMAC Print function

DD statements of the FPA DMAC Print function determine the input and output data sets and specify how to run the DMAC Print function.

Subsections:

- [“List of DD statements” on page 286](#)
- [“Input DD statements” on page 287](#)
- [“Output DD statements” on page 290](#)
- [“Input/Output DD statements” on page 291](#)

List of DD statements

The following table summarizes the DD statements for the DMAC Print function.

Table 55. DD statements for the DMAC Print function

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
JOBLIB or STEPLIB	Input	PDS	No	Required
IMSACB, IMSACBA, or IMSACBB	Input	PDS	No	Required when IMSCATHLQ=*NO
IMSDALIB	Input	PDS	No	Optional
MODSTAT	Input	PDS	Yes	Optional
MODSTAT2	Input	PDS	Yes	Optional
OLCSTAT	Input	PDS	Yes	Optional
<i>addnname</i> or <i>areaname</i>	Input	VSAM ESDS cluster or PS (Note)	Yes	Required under certain conditions (for details, see addnname/areaname DD)
HFPYSIN	Input	LRECL=80	No	Required
HFPPRINT	Output	SYSOOT	Yes	Optional
HFPRPTS	Output	LRECL=133	Yes	Optional
HFPABEND	Output	LRECL=125	Yes	Optional
RECON n	Input and output	KSDS	Yes	Required when both of the following conditions are met: <ul style="list-style-type: none"> • DBRC=YES • DFSMDA members for RECONn data sets are not used

Table 55. DD statements for the DMAC Print function (continued)

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
--------	-----	--------	-------------------------------	----------------------

Note: If image copy is used as an input data set.

Input DD statements

The following input DD statements are used for the DMAC Print function.

JOBLIB/STEPLIB DD

This DD statement is a required DD statement that defines the following input data sets:

- The FPA load module library (required).
- The library that contains DFSMDA dynamic allocation members for the RECON data set. When DBRC=YES is specified and the RECON n DD statement is omitted, this DD statement is required.
- The library that contains a default site default member (HFPCSITE) to determine the data set name and the member of the site default table.
- The IMS load module library (required).
- The IMS Tools Base library (SHKTLOAD), if you specify the ITKBSRVR keyword.
- The IMS Tools Base library (SGLXLOAD), if you specify the IMSCATHLQ keyword.

FPA must be run as an APF-authorized program. All of the data sets that are listed within the JOBLIB or the STEPLIB concatenation must also be APF-authorized. If you use DFSMDA members and the DFSMDA members are in a non-APF authorized library, you can specify the DFSMDA library on the IMSDALIB DD statement.

IMSACB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. The IMSACB DD statements (IMSACB, IMSACBA, IMSACBB, MODSTAT, MODSTAT2, and OLCSTAT) can be used to identify the active ACB library. When both OLCSTAT or MODSTAT and MODSTAT2 are found, FPA checks the time stamp and determines the latest one.

The active IMSACB library is determined according to the following rules:

1. Check IMSACB DD statement in the JCL statement.
2. Check OLCSTAT DD statement in the JCL statement.
3. Check MODSTAT and MODSTAT2 DD statement in the JCL statement.
4. Allocate OLCSTAT dynamically and check the OLCSTAT member.
5. Allocate MODSTAT and MODSTAT2 dynamically and check the MODSTAT and MODSTAT2 members.

If you specify the IMSCATHLQ keyword, IMSACB DD statements are not used. The IMS directory is used instead of the ACB library.

IMSACBA DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSACBB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSDALIB DD

This DD statement is an optional DD statement that defines the IMSDALIB data set. This DD statement is useful when the data set for dynamic allocation members is not APF-authorized.

MODSTAT DD

This DD statement is an optional DD statement that defines the MODSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

MODSTAT2 DD

This DD statement is an optional DD statement that defines the MODSTAT2 data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT2 library by using a TYPE=DATABASE statement of the DFSMDA macro.

OLCSTAT DD

This DD statement is an optional DD statement that defines the OLCSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the OLCSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

***addnname/areaname* DD**

This DD statement is a required DD statement. This statement defines an input data set, which is the VSAM ESDS or image copy for an area to be printed. The ddname used must be the ddname for the area that is defined in the input ACB library (IMSACB DD). There can be multiple *addnname* DD statements for each area included in the print request.

The input data set can be allocated dynamically in one of following ways:

- The IDSNMASK keyword parameter of the DMAPRT command
- When the GLOBAL DBRC=YES option is specified, the ADS information in the DBRC RECON data sets (only for VSAM ESDS data sets)
- The area name member in the IMS DFSMDA library (only for VSAM ESDS data sets)

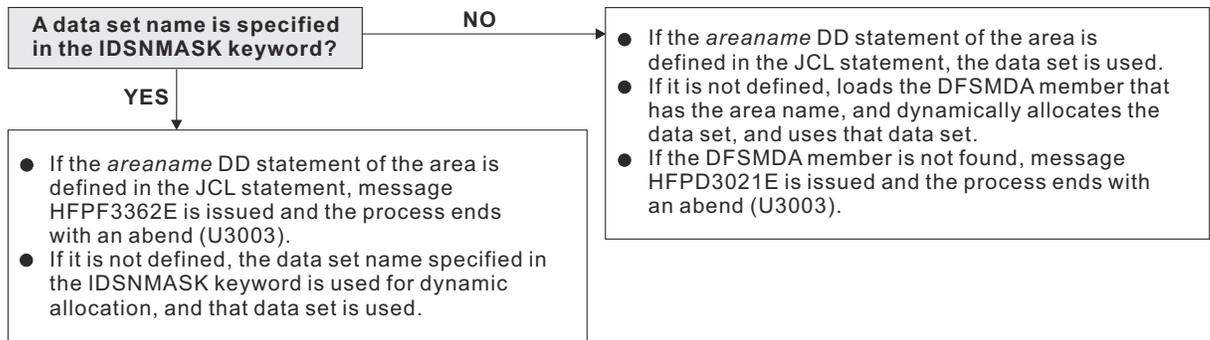
Or it can be specified with one of the following DD statements explicitly:

- *//addnname* DD JCL statement
- *//areaname* DD JCL statement

If an area data set is used for the DMAPRT command input, it must be error free (that is, it has no EEQEs). And if the area is registered with DBRC, the data set must have no EEQEs.

The following figure shows how FPA determines area data set names.

When DBRC=NO:



When DBRC=YES:

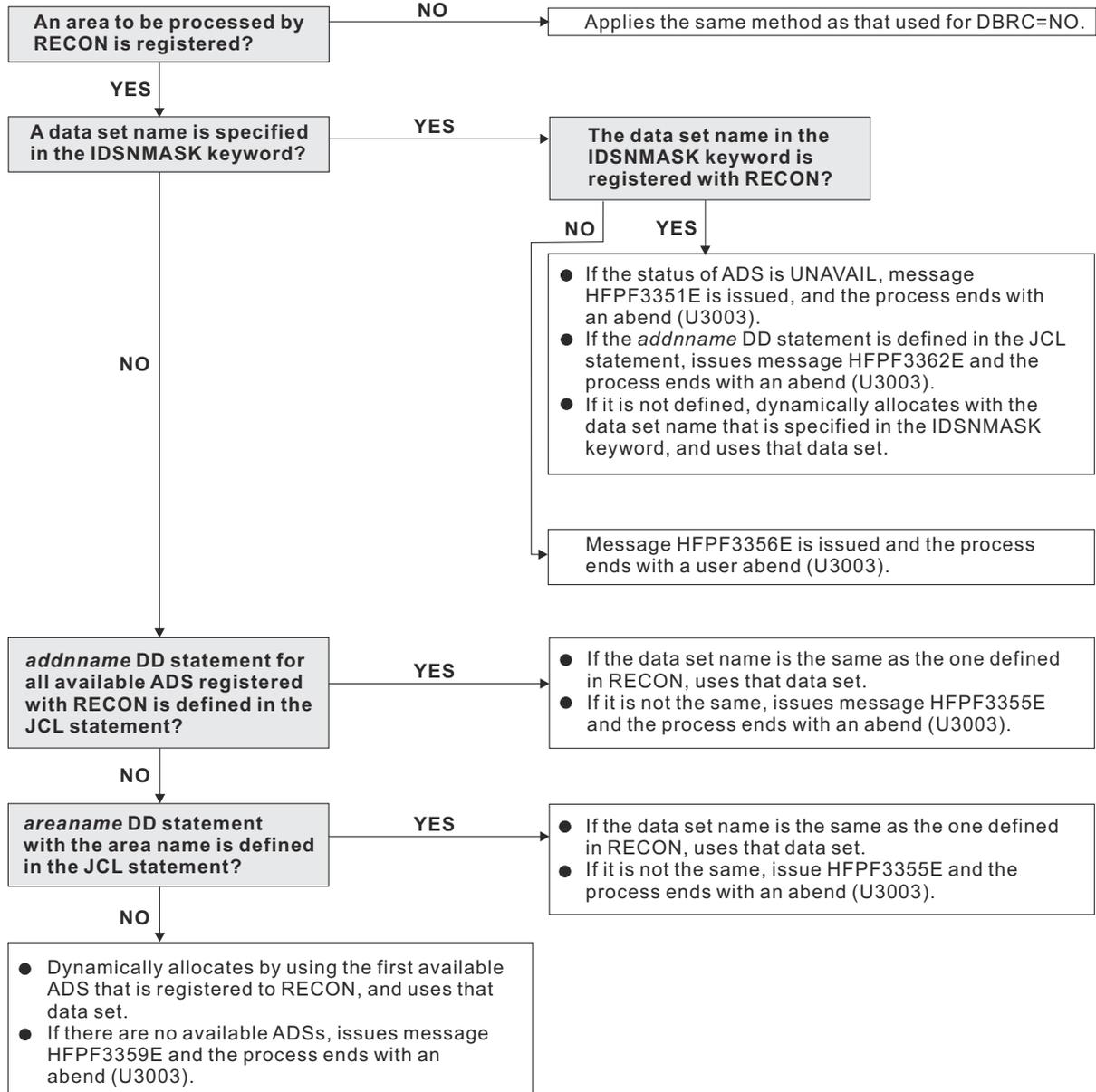


Figure 133. Input data set names for the DMACPRT command

HFPYSIN DD

This DD statement is a required DD statement that defines the input control statement that specifies the FPA functions.

The HFPSYSIN DD statement can be coded as a standard SYSIN file, a sequential data set, or a PDS member. LRECL=80 is required for the DCB of this data set. Only one command, besides the GLOBAL command, can be specified in HFPSYSIN. FPA analyzes the command syntax in columns 1 - 72. Columns 73 - 80 are regarded as comment.

For details about coding the HFPSYSIN DD statement, see [“Utility control statements for the DMAC Print function” on page 291](#).

Output DD statements

The following output DD statements are used for the DMAC Print function.

HFPPRINT DD

This DD statement is an optional DD statement that defines the processing log output data set. The data set contains the processing messages that are issued by FPA.

The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set. You cannot specify DUMMY.

The HFPPRINT data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPPRINT DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

The following figure shows an example of the processing messages that are issued for the DMACPRT command.

```
IMS HFPF UTILITIES                                "Processing messages"                                PAGE: 1
5698-FPP V2R1                                     2019-11-22 17:10:02

HFPC0001I COMMAND ANALYZE PROCESSING STARTED.
HFPC0005I DBRC=NO IS SPECIFIED.
- EEQE DETECTION IS NOT PERFORMED.
HFPC0002I COMMAND ANALYZE PROCESSING COMPLETED.
HFPF0001I DMACPRT PROCESSING STARTED FOR DATABASE: DEDBJN21
HFPF0002I DMACPRT PROCESSING COMPLETED FOR DATABASE: DEDBJN21
HFPL0001I REPORT STARTED FOR COMMAND: DMACPRT, DATABASE: DEDBJN21
HFPL0002I REPORT COMPLETED FOR COMMAND: DMACPRT, DATABASE: DEDBJN21
```

Figure 134. Processing messages issued for the DMACPRT command

HFPRTS DD

This DD statement is an optional DD statement that defines the report output data set. The data set contains the reports that can be used to verify the process of the command. The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

The following reports are generated in this data set:

- Audit report
- Processing report
- DBD Definition report

- DMACPRT report

The HFPRPTS data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPRPTS DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

HFPABEND DD

This DD statement is an optional DD statement. It defines a snap dump output data set that is used for diagnostic purposes. This data set might be used when the FPA job ends abnormally. Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

Input/Output DD statements

The following input/output DD statements are used for the DMAC Print function.

RECON1/RECON2/RECON3 DD

These statements define the DBRC RECON data sets. These DD statements are required DD statements when both of the following conditions are met:

- DBRC=YES is specified for the GLOBAL command
- DFSMDA members for RECONn data sets are not used

The RECONn data set must be the same data set that is used in the IMS control region. Because FPA issues a signon, a signoff, an authorization call, and an unauthorization call to DBRC, it updates RECONn data sets.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is necessary, the data set is allocated dynamically by using a DFSMDA member.

Utility control statements for the DMAC Print function

The utility control statements for the DMAC Print function define the user description of the GLOBAL command and the DMACPRT command.

The utility control statements must be specified in the HFPSYSIN data set. This control statement data set usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte, fixed-length records. Block size, if coded, must be a multiple of 80.

The following figure shows a sample of how to code the HFPSYSIN data set.

```
//HFPSYSIN DD *
GLOBAL
DBRC=YES
DMACPRT
DBD=dbdname
/*
```

Figure 135. DMACPRT HFPSYSIN data set

Subsections:

- [“GLOBAL command keywords” on page 292](#)
- [“DMACPRT command keywords” on page 292](#)

GLOBAL command keywords

For the GLOBAL command, you can specify the keywords that are listed in the following table. For the format of the GLOBAL command, see [“GLOBAL command”](#) on page 394.

Table 56. GLOBAL command keywords

Keyword	Required or optional	Description	Topic
DBRC	Optional	Specifies to request or bypass the database recovery control (DBRC) facility process for the database that is specified by the DBD.	“DBRC keyword” on page 451
ICINCOMP	Optional	Specifies, when an image copy is used as an input data set, whether the image copy records are to be compressed by using the CSRCESTRV MVS macro when they are stored into a data space storage.	“ICINCOMP keyword” on page 474
IMSCATACB_INPUT	Optional	Specifies whether to retrieve the currently active ACB definition or the pending ACB definition from the IMS directory.	“IMSCATACB_INPUT keyword” on page 480
IMSCATHLQ	Optional	Specifies the high-level qualifier of the bootstrap data set of the IMS directory.	“IMSCATHLQ keyword” on page 481
ITKBLOAD	Optional	Specifies the library name of IMS Tools KB.	“ITKBLOAD keyword” on page 487
ITKBSRVR	Optional	Specifies the name of the IMS Tools KB server XCF group.	“ITKBSRVR keyword” on page 488
MSTIMESTAMP	Optional	Specifies whether to include a time stamp in the messages.	“MSTIMESTAMP keyword” on page 497
SCAN	Optional	Specifies to verify the syntax of command, keywords, and parameters in command sets without running FPA. The Scan function also allocates some (but not all) internal resources for running the process function.	“SCAN keyword” on page 523
SITE_DSNAME	Optional	Specifies the data set name for the site default table.	“SITE_DSNAME keyword” on page 539
SITE_MEMBER	Optional	Specifies the member name for the site default table.	“SITE_MEMBER keyword” on page 540

DMACPRT command keywords

For the DMACPRT command, you can specify the keywords that are listed in the following table. For the format of the DMACPRT command, see [“DMACPRT command”](#) on page 390.

Table 57. DMACPRT command keywords

Keyword	Required or optional	Description	Topic
DBD	Required	Specifies the database description (DBD) that contains the areas to be processed.	“DBD keyword” on page 450
IAREA	Optional	Specifies one or more input areas for the DMAC Print process.	“IAREA keyword” on page 469
IDSNMASK	Optional	Specifies the data set name or the mask for dynamically allocating cataloged pre-existing input data sets.	“IDSNMASK keyword” on page 476

Examples for the DMAC Print function

Review examples and learn techniques for printing DMAC blocks in DEDB areas.

In the examples, the following DEDB is assumed:

```

DEDB name:  DEDBJN22 has four areas
  area name:  DB22AR0 (area number is 1 and has three ADSs)
    ADS DD name:  DB22AR01,      DS name:  HPFP.DB22AR0.ADS1
    ADS DD name:  DB22AR02,      DS name:  HPFP.DB22AR0.ADS2
    ADS DD name:  DB22AR03,      DS name:  HPFP.DB22AR0.ADS3
  area name:  DB22AR1 (area number is 2 and has two ADSs)
    ADS DD name:  DB22AR11,      DS name:  HPFP.DB22AR1.ADS1
    ADS DD name:  DB22AR12,      DS name:  HPFP.DB22AR1.ADS2
  area name:  DB22AR2 (area number is 3 and has two ADSs)
    ADS DD name:  DB22AR21,      DS name:  HPFP.DB22AR2.ADS1
    ADS DD name:  DB22AR22,      DS name:  HPFP.DB22AR2.ADS2
  area name:  DB22AR3 (area number is 4 and has one ADS)
    ADS DD name:  DB22AR31,      DS name:  HPFP.DB22AR3.ADS1

```

In DBRC, all ADSs are in the AVAILABLE status.

Example 1: Printing a DMAC block in an area

The following figure shows example JCL for printing a DMAC block in an area that is registered with DBRC.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS to be printed is dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//        DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
DMACPRT
  DBD=DEDBJN22,
  IAREA=(DB22AR0)
/*

```

Figure 136. Printing a DMAC block in an area

Example 2: Printing DMAC blocks in all areas of a DEDB

The following figure shows example JCL for printing DMAC blocks in all areas of a DEDB that is registered with DBRC.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADSs to be printed is dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
DMACPRT
  DBD=DEDBJN22,
  IAREA=ALL
/*
```

Figure 137. Printing DMAC blocks in all areas of a DEDB

Chapter 12. Building secondary index databases for a DEDB offline with FPA

You can build secondary index databases of a DEDB by using the FPA Build Index function.

Topics:

- [“Functions of Build Index” on page 295](#)
- [“Input and output for the Build Index function” on page 295](#)
- [“Running the Build Index function” on page 297](#)
- [“Region size requirements for the Build Index function” on page 299](#)
- [“DD statements for the Build Index function” on page 301](#)
- [“Utility control statements for the Build Index function” on page 310](#)
- [“Examples for the Build Index function” on page 314](#)

Functions of Build Index

The Build Index function of FPA efficiently builds one or more secondary index databases. The function concurrently scans multiple DEDB areas and builds multiple secondary index databases in a single job step.

IMS and the Build Index function of FPA support two database structures for Fast Path secondary indexes; HISAM and SHISAM. Both secondary index database structures offer sequential key access to primary DEDB databases. To learn about secondary indexes, see *IMS Database Administration*.

The Build Index function has the following characteristics:

- Rebuilds secondary index databases faster than initially loading data to a DEDB.
- Helps you build specific secondary index databases in case of system failures.
- Reduces the amount of time that it takes to build multiple secondary index databases by using both parallel sorting and parallel loading.

The Build Index function supports the following three modes for building indexes:

Scan load mode

Scans the DEDB areas and builds secondary index databases in one job step.

Scan mode

Scans the DEDB areas and then generates build pointer segment records. The generated records are used as input to a Build Index job that runs in load mode to build secondary indexes.

Load mode

Reads the build pointer segment records that were generated in scan mode, and then loads index pointer segments into the secondary index databases.

Input and output for the Build Index function

The Build Index function is invoked by the INDEXBLD command in the HFPSYSIN DD control statement.

The Build Index function, the Resync function, and the Freespace Reclaim function use the INDEXBLD command to start the Build Index process. During the Build Index process, FPA builds, synchronizes, or reclaims space in secondary index databases. To determine which task to perform, FPA evaluates the IDXPROC keyword on the INDEXBLD command as follows:

- When IDXPROC=BUILD, FPA starts the Build Index function. During the job, FPA scans one or more DEDB areas and builds one or more secondary index databases.

- When IDXPROC=RESYNC, FPA starts the Resync function. During the job, FPA scans one or more DEDB areas and secondary index databases and synchronizes secondary index databases.
- When IDXPROC=FS_RECLAIM, FPA starts the Freespace Reclaim function. During the job, FPA reclaims the CIs in secondary index database data sets.

The following figure shows the input and the output for the Build Index function.

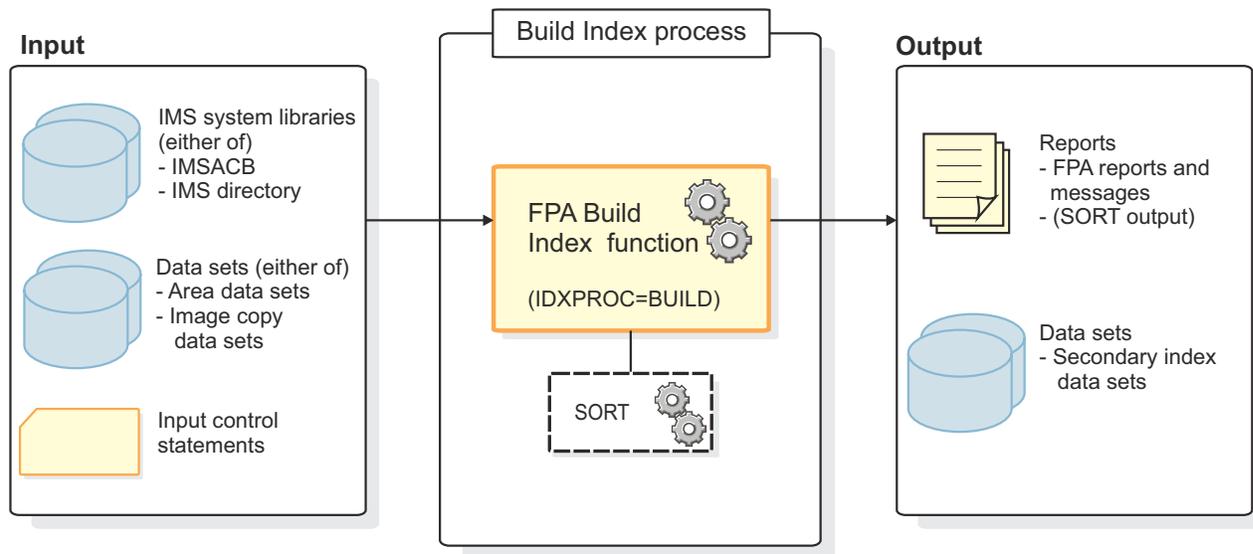


Figure 138. Input and output for the Build Index function

The Build Index function can also read and scan the following image copies of a DEDB:

- Standard image copies that are created by the IMS Image Copy utility (DFSUDMPO) or IMS HP Image Copy
- Compressed image copies that are created by IMS HP Image Copy

Restriction: Unlike other functions of FPA, the Build Index function cannot internally invoke IMS HP Image Copy to create image copies.

Build pointer segment records

Build pointer segment records are segment records for the index pointer segments that need to be loaded into the secondary index databases.

A build pointer segment record data set cannot contain records of multiple secondary index databases. Therefore, you must prepare one build pointer segment record data set for each secondary index database to build.

Build pointer segment record data sets do not require DCB information to be specified in JCL.

Subsections:

- [“How build pointer segment records are created” on page 296](#)
- [“Space requirement for build pointer segment record data sets” on page 297](#)

For the format of build pointer segment records, see [“Build pointer segment records layout” on page 1193](#).

How build pointer segment records are created

Build pointer segment records are created by the FPA Build Index function when a Build Index job scans DEDB areas in scan mode. Build pointer segment records are used by the FPA Build Index function as inputs when a Build Index job loads index pointer segments in load mode.

The following table summarizes the methods for specifying build pointer segment record data sets and the behavior of the Build Index function.

Build pointer segment record data sets specified with	Description
<code>S0nnnnn0</code> DD statements	<p>Scan mode Generates the records in the data sets that the <code>S0nnnnn0</code> DD statements point to.</p> <p>Load mode Obtains the records from the data sets that the <code>S0nnnnn0</code> DD statements point to.</p>
<code>BUILDSDSMASK</code> keyword	<p>Scan mode Generates the records in the data sets that the <code>BUILDSDSMASK</code> keyword specifies. Build pointer segment record data sets must be allocated and cataloged before the job.</p> <p>Load mode Obtains the records from the data sets that the <code>BUILDSDSMASK</code> keyword specifies.</p>
<code>BLDMFILECTL</code> subcommand with the <code>DISP=NEW</code> keyword	<p>Scan mode Dynamically generates build pointer segment record data sets and generates the records in the data sets. Build pointer segment record data sets do not need to be allocated before the job.</p> <p>Load mode This method is not applicable.</p>
<code>BLDMFILECTL</code> subcommand with the <code>DISP=OLD</code> or <code>DISP=SHR</code> keyword	<p>Scan mode Generates the records in the data sets that the <code>BLDMFILECTL</code> subcommand specifies. Build pointer segment data sets must be allocated before the job.</p> <p>Load mode This method is not applicable.</p>

Space requirement for build pointer segment record data sets

To estimate the space that is required for a build pointer segment record data set, use the following formula. The unit is in bytes.

$$A * (\text{Number of pointer segments that are loaded into the secondary index database})$$

where A is the length of the pointer segment + 4. If the value is smaller than 20, use 20 for A.

Running the Build Index function

By running the Build Index function, you can build secondary index databases from the DEDB database. FPA runs the Build Index function as a standard z/OS batch job. You must specify an EXEC statement and DD statements that define the input and output data sets in your JCL.

Procedure

- Optional: Estimate the region size that is required to run the job.
See [“Region size requirements for the Build Index function”](#) on page 299.
- Specify the EXEC statement in the following format:

```
//INDEXBLD EXEC PGM=HFPMAIN0,REGION=rrrrM,
//          PARM='DBRCGRP=dbrcgrp,GSGNAME=gsgname,IMSPLEX=imsplex'
```

Parameter	Description
DBRCGRP=	DBRCGRP is an optional 1- to 3-character parameter that specifies the identifier assigned to a group of DBRC instances (DBRC sharing group) that access the same RECON data set in the IMSplex.
GSGNAME=	GSGNAME is an optional 1- to 8-character parameter that specifies the global service group.
IMSPLEX=	IMSPLEX is an optional 1- to 5-character parameter that specifies the IMSplex group names to share either databases or message queues.

3. Specify the DD statements to define input data sets and output data sets.

For descriptions of the DD statements, see [“DD statements for the Build Index function” on page 301](#).

To build secondary indexes by running two Build Index jobs (one in scan mode and another in load mode), you must supply information about build pointer segment record data sets. You can do so by specifying DD statements (SO $nnnnn$ 0 DD) or utility control statements.

4. Specify the utility control statements.

- a) Specify the INDEXBLD command.
- b) Specify the database name on the DBD keyword.
- c) Either specify IDXPROC=BUILD, or specify no IDXPROC keyword.

The IDXPROC keyword must specify BUILD to build secondary index databases. The default parameter for the IDXPROC keyword is BUILD.

- d) Specify the runtime mode on the BUILDMODE keyword. You can specify one of the following parameters:

SCANLOAD

The job runs in scan load mode. It scans the DEDB areas and builds secondary index databases in one job step. BUILDMODE=SCANLOAD is the default.

SCAN

The job runs in scan mode. It scans the DEDB areas and then generates build pointer segment records. You must later run the job in load mode to build secondary index databases.

LOAD

The job runs in load mode. It reads the records from the build pointer segment record data sets and loads index pointer segments into the secondary index databases.

Considerations for selecting a mode: Running the job in scan load mode is the easiest method to build secondary index databases. However, in scan load mode, DFSORT is internally used during the job to sort the records in build pointer segment record data sets. If you want to use a different sort program, you can run the job in scan mode, sort the records manually by using your desired sort program, and then run the job in load mode.

See [“Utility control statements for the Build Index function” on page 310](#) and specify other command, subcommands, and keywords.

5. Run the JCL job.

6. If you run the job in scan mode, update the statements or create another JCL job to run in load mode. Then, run that job to load the secondary index databases.

In load mode, if you specify SORT=YES, the sort is done during the job before the pointer segments are loaded (SORT=YES is the default). If you specify SORT=NO, sort is not done in the job. In this case, you must manually sort the build pointer segment records *before* you run the job in load mode.

Tips for manually sorting the records:

- Secondary Index Processing report, which is generated in step “5” on page 298, contains auto-generated sort control statements for DFSORT. You can copy and paste the control statements into a DFSORT job to sort the records with DFSORT.
- To sort the records manually, use the first column as the starting position of the sort field, and "key_length + 1" as the length of the sort field.

Example

The following figure shows a JCL example for the INDEXBLD command for building secondary index databases in scan load mode.

```
//*****
//** BUILDING SECONDARY INDEX DB(S) **
//*****
//INDEXBLD EXEC PGM=HFPMAIN0
//*
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
//IMSACB DD DISP=SHR,DSN=IMSVS.FPSI.ACBLIB
//HFPPRINT DD SYSOUT=A
//HFPRPTS DD SYSOUT=A
//*
//HFPSYSIN DD *
GLOBAL DBRC=NO
INDEXBLD
DBD=IVPDB3
/*
//DFSIVD3A DD DISP=OLD,DSN=IMSVS.DFSIVD31
//DFSIVD3B DD DISP=OLD,DSN=IMSVS.DFSIVD33
//SINDEXA DD DISP=OLD,DSN=IMSVS.SINDEXA
//SINDEXB DD DISP=OLD,DSN=IMSVS.SINDEXB
//*
```

Figure 139. JCL example for the INDEXBLD command for building secondary index databases (scan load mode)

Related concepts

[Region size requirements for the Build Index function](#)

The Build Index function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Related reference

[Examples for the Build Index function](#)

Review examples and learn techniques for building indexes for DEDB areas.

[Tips and best practices for the FPA Build Index function](#)

The FPA Build Index function supports several keywords that improve the performance of FPA Build Index jobs and help you use the function more effectively and efficiently.

Region size requirements for the Build Index function

The Build Index function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Region size for the Build Index function depends on the size of a UOW, the number of input areas, the number of output secondary index database data sets, the ITASKCTL specification, and the OTASKCTL specification. Typically, database maintenance is a relatively important function. To increase the performance of the Build Index program, it would be advisable to run it in a region that has a relatively high dispatching priority (that is, at least equal to the dispatching priority of a message processing region).

Estimating the region size for the Build Index function

The region size required for the Build Index function is a function of the size of a UOW, the number of input areas, the number of output secondary index database data set, the ITASKCTL specification, and the OTASKCTL specification.

In most cases, the specification on the EXEC statement for HFPMAIN0 of a region that is too small, is detected during the initialization processing of the main task. This causes processing to be terminated at that point. Under certain conditions, the region size shortfall does not become evident until the actual Build Index process has begun. The job must be completely rerun. Thus, the region size calculations must always be performed prior to initiating Build Index processing.

In addition, the sort program also requires region space. For how to estimate the required region size, see the *DFSORT Application Programming Guide*.

The following worksheet is for estimating the region size that is required for a typical Build Index job. The actual region size depends on the keywords that are specified for the job. Therefore, when you specify the region size in your JCL, use a higher value than the value calculated from this worksheet.

FPA programs, internal tables, control blocks	3,020,000	
Number of areas defined in input DBD * 1,300	-----	
Number of secondary index databases defined in output DBD * 1,400	-----	
Internal tables, control blocks	18,200	
Maximum segment length * 1,000	-----	
Maximum CISZ * 4	-----	
Internal I/O buffer	1,000,000	(If ICHECK=DEFER)
subtotal	-----	
Value specified on ITASKCTL * subtotal	-----	
Internal tables, control blocks	21,000	
Internal I/O buffer	1,000,000	(If ICHECK=NODEFER)
Storage used by sort program	-----	
subtotal	-----	
Value specified on OTASKCTL * subtotal	-----	
Number of Build Index processed areas * 3,700	-----	
Total of CISZ * UOW1 of each area * 2	-----	
Total bytes of Build Index storage	-----	

Figure 140. Worksheet for estimating region size for the Build Index function

If ICACHE=YES is specified, the Build Index function preloads the IOVF part into the data space. The data space size required for the Build Index function is a total of the size of the IOVF part in each area. In addition, if the input data is an image copy data set, all image copy records are stored into the data space. In this case, the data space for the image copy data set must be added to the size that is calculated by the preceding worksheet.

The following figure shows the worksheet for estimating the data space size for Build Index function.

Total size of IOVF parts in all areas	-----	
Total of CISZ * UOW1 * (ROOT1 - ROOT2) in each area ICACHE=YES	-----	(If input data is IC and is specified)
Total byte of a data space storage	-----	

Figure 141. Worksheet for estimating data space size for the Build Index function

DD statements for the Build Index function

DD statements of the FPA Build Index function determine the input and output data sets and specify how to run the Build Index function.

Subsections:

- [“List of DD statements” on page 301](#)
- [“Input DD statements” on page 302](#)
- [“Output DD statements” on page 306](#)
- [“Input/Output DD statements” on page 309](#)

List of DD statements

The following table summarizes the DD statements for the Build Index function.

Table 58. DD statements for the Build Index function

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
JOBLIB or STEPLIB	Input	PDS	No	Required
IMSACB, IMSACBA, or IMSACBB	Input	PDS	No	Required when IMSCATHLQ=*NO
IMSDALIB	Input	PDS	No	Optional
MODSTAT	Input	PS	Yes	Optional
MODSTAT2	Input	PS	Yes	Optional
OLCSTAT	Input	PS	Yes	Optional
IMS	Input	PDS	No	Required when DBRC=YES
IMSRESLB	Input	PDS	No	Optional
<i>addnname</i> or <i>areaname</i>	Input	VSAM ESDS cluster or PS (Note 1)	Yes	Required under certain conditions (for details, see addnname/areaname DD)
<i>index</i>	Output	KSDS	Yes	Required when DFSMDA members for index databases are not used
<i>indexovrflw</i>	Output	ESDS	Yes	Required when DFSMDA members for index databases are not used
HFPSYSIN	Input	LRECL=80	No	Required
HFPCSITE	Input	PDS	Yes	Optional
HFPPRINT	Output	SYSOUT	Yes	Optional
HFPRTS	Output	LRECL=133	Yes	Optional
HFPABEND	Output	LRECL=125	Yes	Optional
RECON n	Input and output	KSDS	Yes	Required when both of the following conditions are met: <ul style="list-style-type: none"> • DBRC=YES • DFSMDA members for RECONn data sets are not used

Table 58. DD statements for the Build Index function (continued)

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
UtttWKnn	Input and output		Yes	Optional
UtttMSG	Output	LRECL=133	Yes	Optional
S0nnnnn0	Input and output		Yes	Optional
S0nnnnnR	Output		Yes	Optional

Notes:

1. If image copy is used as an input data set.

Input DD statements

The following input DD statements are used for the Build Index function.

JOBLIB/STEPLIB DD

This DD statement is a required DD statement that defines the following input data sets:

- The FPA load module library (required).
- The library that contains DFSMDA dynamic allocation members for the RECON data set. When DBRC=YES is specified and the RECONn DD statement is omitted, this DD statement is required.
- The library that contains a default site default member (HFPCSITE) to determine the data set name and the member of the site default table.
- The IMS load module library (required).
- The library that contains the user-defined data entry database partition selection exit routine that was specified on the XDFLD PSELRTN= parameter during DBD generation for the DEDB.
- The library that contains the secondary index database maintenance exit routine that was specified on the XDFLD EXTRTN= parameter during DBD generation for the DEDB.
- The IMS Tools Base library (SHKTLOAD), if you specify the ITKBSRVR keyword.
- The IMS Tools Base library (SGLXLOAD), if you specify the IMSCATHLQ keyword.
- The IMS Library Integrity Utilities library, if you want to activate Integrity Checker.

FPA must be run as an APF-authorized program. All the data sets that are listed within the JOBLIB or the STEPLIB concatenation must also be APF-authorized. If you use DFSMDA members and the DFSMDA members are in a non-APF authorized library, you can specify the DFSMDA library on the IMSDALIB DD statement.

For more information about user-defined data entry database partition selection exit routines and secondary index database maintenance exit routines, see *IMS Exit Routines*.

IMSACB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. The IMSACB DD statements (IMSACB, IMSACBA, IMSACBB, MODSTAT, MODSTAT2, and OLCSTAT) can be used to identify the active ACB library. When both OLCSTAT or MODSTAT and MODSTAT2 are found, FPA checks the time stamp and determines the latest one.

The active IMSACB library is determined according to the following rules:

1. Check IMSACB DD statement in the JCL statement.
2. Check OLCSTAT DD statement in the JCL statement.
3. Check MODSTAT and MODSTAT2 DD statement in the JCL statement.

4. Allocate OLCSTAT dynamically and check the OLCSTAT member.
5. Allocate MODSTAT and MODSTAT2 dynamically and check the MODSTAT and MODSTAT2 members.

If you specify the IMSCATHLQ keyword, IMSACB DD statements are not used. The IMS directory is used instead of the ACB library.

IMSACBA DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSACBB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSDALIB DD

This DD statement is an optional DD statement that defines the IMSDALIB data set. This DD statement is useful when the data set for dynamic allocation members is not APF-authorized.

MODSTAT DD

This DD statement is an optional DD statement that defines the MODSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

MODSTAT2 DD

This DD statement is an optional DD statement that defines the MODSTAT2 data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT2 library by using a TYPE=DATABASE statement of the DFSMDA macro.

OLCSTAT DD

This DD statement is an optional DD statement that defines the OLCSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the OLCSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

IMS DD

This DD statement is required when DBRC=YES is specified for the GLOBAL command. This statement defines the library that contains the DBD that describes the secondary index databases. This statement is used when the NOTIFY.REORG command is issued to DBRC.

If you specify the IMSCATHLQ keyword, IMS DD statement is not used. The IMS directory is used instead of the library that contains the DBD.

IMSRESLB DD

This DD statement is an optional DD statement that defines the libraries that contain the randomizer routines, compression routines, user-defined data entry database partition selection exit routine and secondary index database maintenance exit routine that are specified in the ACB member (specified by the IMSACB DD statements). If the IMSRESLB DD statement is specified in the JCL statement, only the randomizer and compression routines are loaded directly from the IMSRESLB DD statement. If the

IMSRESLB DD statement is not specified in the JCL statement, these resources are loaded from the JOBLIB or the STEPLIB DD statement.

The IMS version of IMSRESLB DD statement must be the same as those of IMSACB DD.

All the data sets that are listed within the IMSRESLB concatenation must be APF-authorized.

addname/areaname DD

This DD statement is a required DD statement that defines an input data set, which is the VSAM ESDS or image copy for an area to be processed. The ddname used must be the ddname for the area that is defined in the input ACB library (IMSACB DD). There can be multiple *addname* DD statements for each area included in the process request.

The input data set that is used by FPA is determined by the following elements:

- The IDSNMASK keyword parameter of the INDEXBLD command
- When the GLOBAL DBRC=YES option is specified, the ADS information in the DBRC RECON data sets (only for VSAM ESDS data sets)
- The area name member in the IMS DFSMDA library (only for VSAM ESDS data sets)

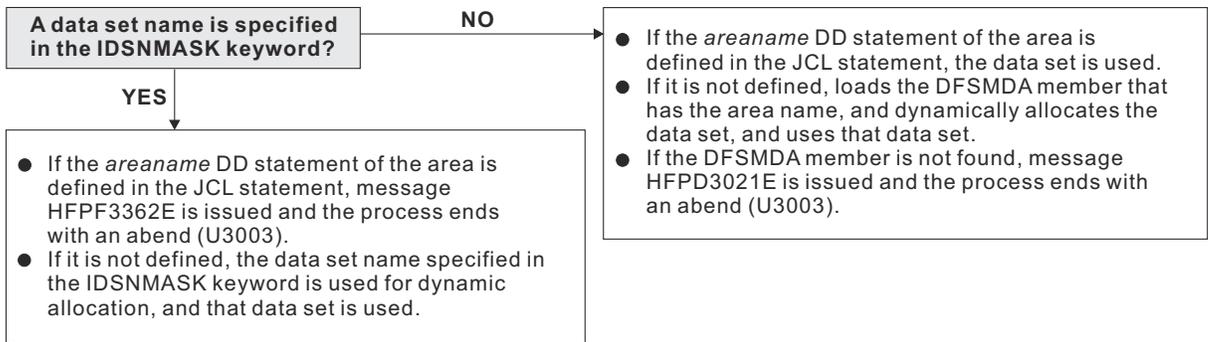
Or it can be specified with one of the following DD statements explicitly:

- *//addname* DD JCL statement
- *//areaname* DD JCL statement

If an area data set is used for the INDEXBLD command input, it must be error free (that is, it has no EQEs). And if the area is registered with DBRC, the data set must have no EEQEs.

The following figure shows how FPA determines area data set names.

When DBRC=NO:



When DBRC=YES:

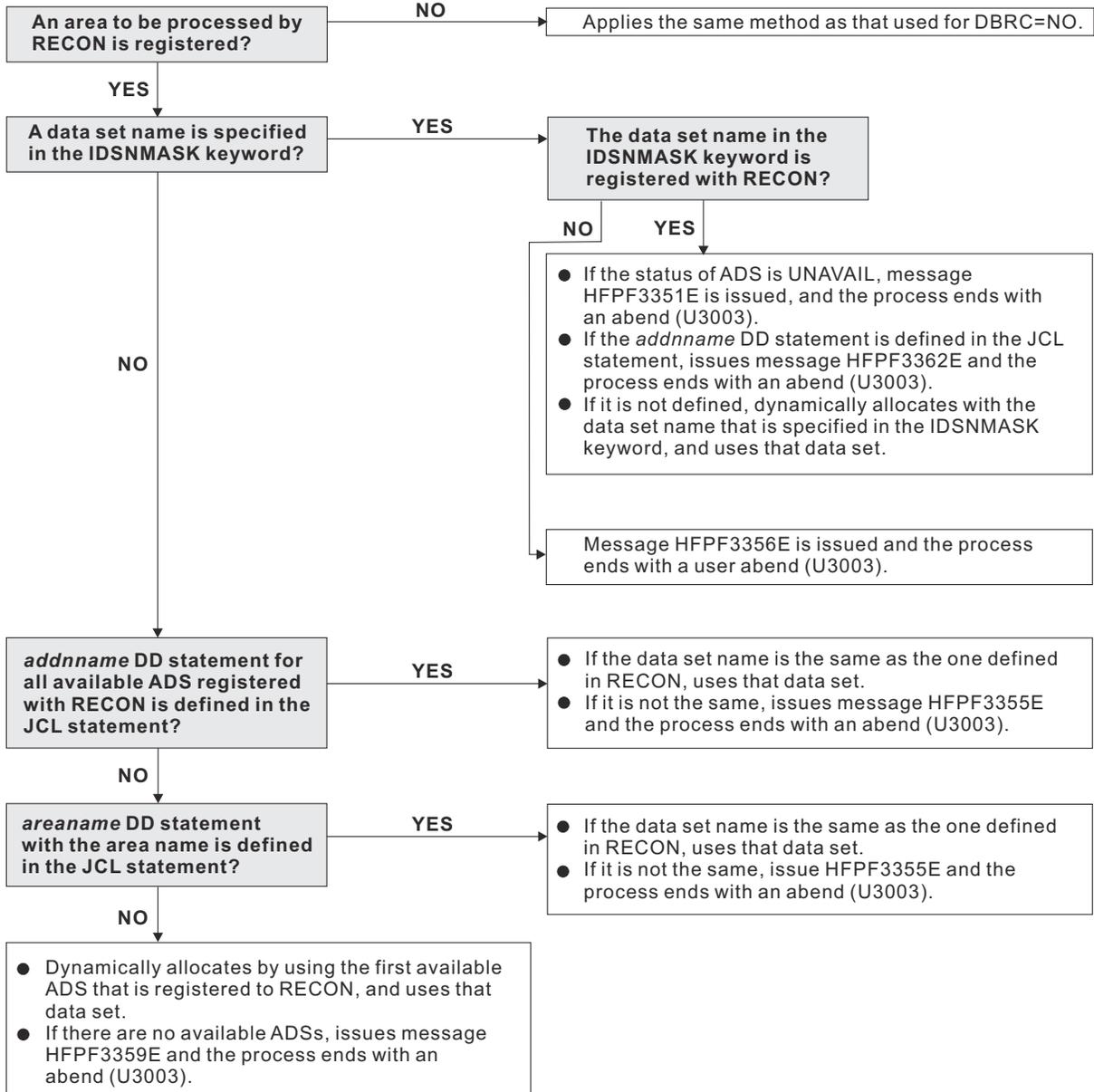


Figure 142. Input data set names for the INDEXBLD command

HFPSYSIN DD

This DD statement is a required DD statement that defines the input control statement that specifies the FPA functions.

The HFPSYSIN DD statement can be coded as a standard SYSIN file, a sequential data set, or a PDS member. LRECL=80 is required for the DCB of this data set. Only one command, besides the GLOBAL command, can be specified in HFPSYSIN. FPA analyzes the command syntax in columns 1 - 72. Columns 73 - 80 are regarded as comment.

For details about coding the HFPSYSIN DD statement, see [“Utility control statements for the Build Index function”](#) on page 310.

HFPCSITE DD

This DD statement is an optional DD statement. The HFPCSITE DD statement can be used to define the data set name of the site default table. When the SITE_DSNAME keyword is specified in the HFPSYSIN DD statement, the HFPCSITE DD statement is allocated dynamically. HFPCSITE DD in the JCL is mutually exclusive with the SITE_DSNAME keyword in the HFPSYSIN DD statement. The data set attribute of HFPCSITE DD is PDS.

Output DD statements

The following output DD statements are used for the Build Index function.

***index* DD**

This DD statement is a required output DD statement when DFSMDA members for index databases are not used. This DD statement defines the VSAM KSDS data set to be loaded.

The DD name must be the DD name for the secondary index database that is defined in the input ACB library (IMSACB DD). Multiple *index* DD statements can be specified for each secondary index database that is built by the Build Index function.

The output data set that is used by FPA is determined by the following elements:

- The name of the secondary index database member in the IMS DFSMDA library
- The *index* DD JCL statement

***indexovrflw* DD**

This DD statement is a required output DD statement when DFSMDA members for index databases are not used. This DD statement defines the VSAM ESDS data set to be loaded.

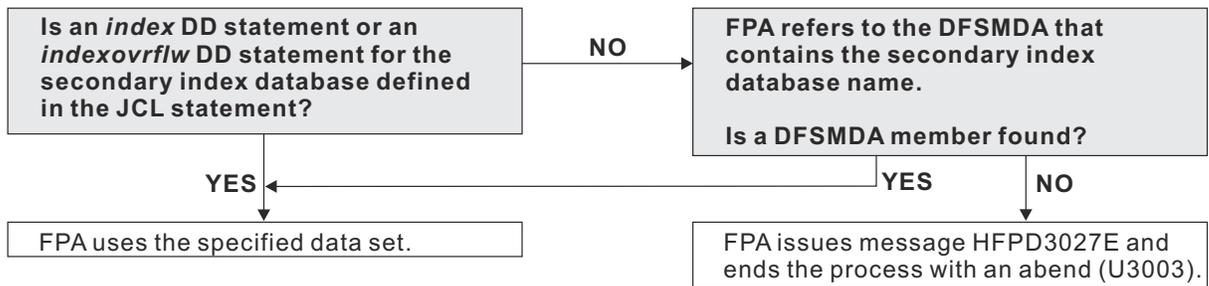
The DD name must be the DD name for the secondary index database that is defined in the input ACB library (IMSACB DD). Multiple *indexovrflw* DD statements can be specified for each secondary index database that is built by the Build Index function.

The output data set that is used by FPA is determined by the following elements:

- The name of the secondary index database member in the IMS DFSMDA library
- The *indexovrflw* DD JCL statement

The following figure shows how FPA determines data set names for secondary index databases.

When DBRC=NO:



When DBRC=YES:

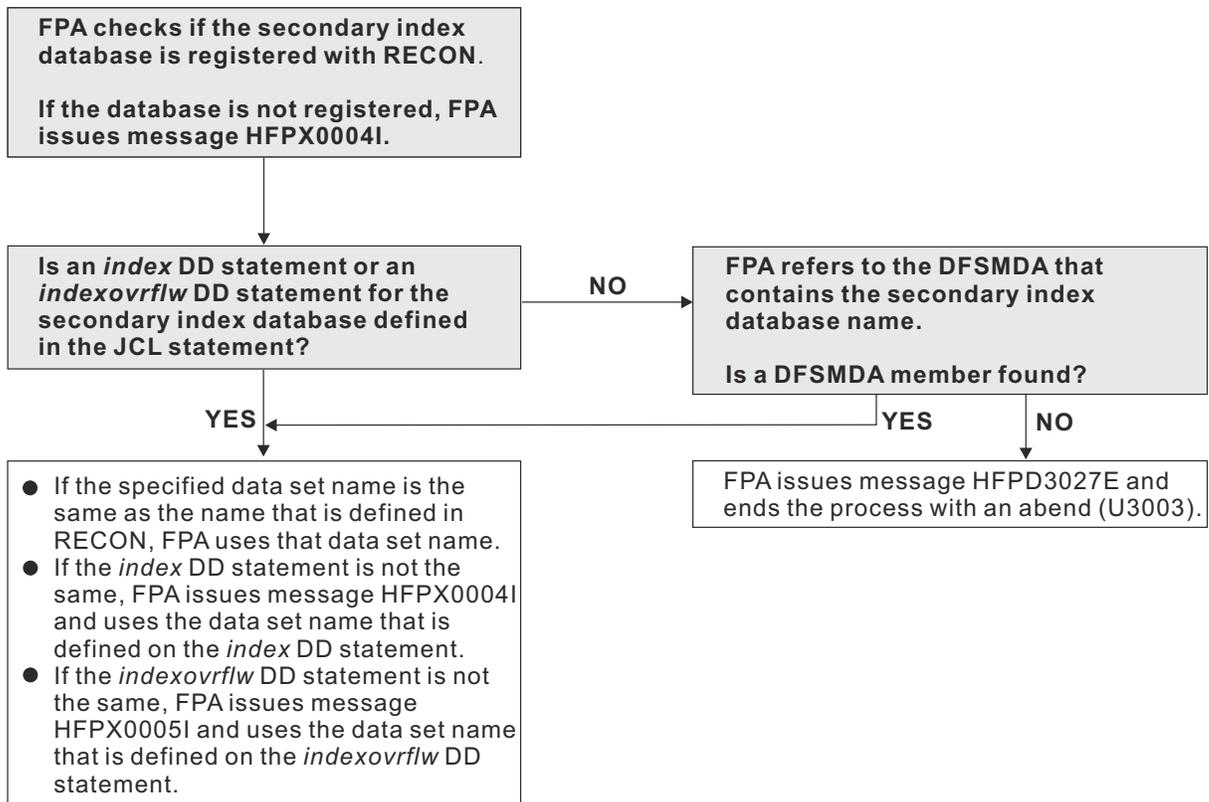


Figure 143. Output data set names for the INDEXBLD command

HFPPRINT DD

This DD statement is an optional DD statement that defines the processing log output data set. The data set contains the processing messages that are issued by FPA.

The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set. You cannot specify DUMMY.

The HFPPRINT data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPPRINT DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*.

The following figure shows an example of the processing messages that are issued for the INDEXBLD command.

```

IMS HPFP UTILITIES                                "Processing messages"                                PAGE: 1
5698-FPP V2R1                                     2019-11-22 13:03:37
HFPC0001I COMMAND ANALYZE PROCESSING STARTED.
HFPC0005I DBRC=NO IS SPECIFIED.
- EEQE DETECTION IS NOT PERFORMED.
HFPF0001I INDEXBLD PROCESSING STARTED FOR DATABASE: DEDBJN23
HFPB0001I INDEXBLD SCAN PROCESSING STARTED FOR AREA NO: 1, AREANAME: DB23AR0
HFPB0002I INDEXBLD SCAN PROCESSING COMPLETED FOR AREA NO: 1, AREANAME: DB23AR0
HFPB0003I INDEXBLD LOAD PROCESSING STARTED FOR SECONDARY INDEX DBDNAME: ITHISM02
HFPB0004I INDEXBLD LOAD PROCESSING COMPLETED FOR SECONDARY INDEX DBDNAME: ITHISM02
HFPF0002I INDEXBLD PROCESSING COMPLETED FOR DATABASE: DEDBJN23
HFPL0001I REPORT STARTED FOR COMMAND: INDEXBLD, DATABASE: DEDBJN23
HFPL0002I REPORT COMPLETED FOR COMMAND: INDEXBLD, DATABASE: DEDBJN23

```

Figure 144. Processing messages issued for the INDEXBLD command

HFPFRPTS DD

This DD statement is an optional DD statement that defines the report output data set. The data set contains the reports that can be used to verify the process of the command. The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

The following reports are generated in this data set:

- Audit report
- Processing report
- DBD Definition report
- Secondary Index Definition report
- Secondary Index Processing report
- Secondary Index Duplicate Key report (when duplicate keys are found)

The HFPFRPTS data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPFRPTS DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*.

HFPABEND DD

This DD statement is an optional DD statement. It defines a snap dump output data set that is used for diagnostic purposes. This data set might be used when the FPA job ends abnormally. Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*.

UttsMSG DD

This DD statement is an optional DD statement. However, this DD statement is required when DFSORT is used. This statement defines an output data set for storing DFSORT messages. For the condition for running DFSORT and the naming rules for the DD names, see the description of UttsWKnn DD.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

S0nnnnnR DD

These DD statements are optional DD statements. They point to the duplicate pointer segment record data sets, which are the output data sets for storing duplicate pointer segment records.

The DD statements are used only when you request to save duplicate pointer segment records (DUPKEYACT=SAVE) and the duplicate keys are found while building the secondary index databases.

nnnnn is a sequential number that FPA assigns to each secondary index database. FPA assigns a sequential number in the order the secondary index databases appear on the LCHILD statements in the DBDGEN source. The value can be in the range of 00001 - 25755.

These data sets can be allocated dynamically by one of the following methods:

- The DSNAME keyword parameter of the FILECTL subcommand
- The DUPKEYDSNMASK keyword parameter of the INDEXBLD command

Input/Output DD statements

The following input/output DD statements are used for the Build Index function.

RECON1/RECON2/RECON3 DD

These statements define the DBRC RECON data sets. These DD statements are required DD statements when both of the following conditions are met:

- DBRC=YES is specified for the GLOBAL command
- DFSMDA members for RECON n data sets are not used

The RECON n data set must be the same data set that is used in the IMS control region. Because FPA issues a signon, a signoff, an authorization call, and an unauthorization call to DBRC, it updates RECON n data sets.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is necessary, the data set is allocated dynamically by using a DFSMDA member.

UtttWKnn

These DD statements are optional DD statements. These are intermediate storage data sets that are used by DFSORT. For more information about coding the SORTWK nn DD statements, see *DFSORT Application Programming Guide*.

The adequate size of storage data sets is 1.5 to 2 times the input file size. In the Build Index function, the input file is the pointer segments that are put into the secondary index database. The size of the file is computed by multiplying the length of the pointer segments that are put into the secondary index database by the number of pointer segments. Allocating twice the space used by the SORTIN data set is usually adequate for each work data set. This data set is allocated dynamically by OS. When the SORTIN is too large for sort, a B37 abend occurs. In this case, this is a required DD.

Do not specify DISP=MOD for these DD statements.

The naming rules of the ddnames are as follows:

ttt:

The hundreds, tenths, and units digits of n for n -th output thread subtask

Note: When a subtask ends processing a secondary index database, it starts processing the next secondary index database. For this reason, the n -th output thread subtask does not always process the n -th output data set.

nn:

Specifies the number of the SORT work data sets. This value must be in the range of 01 - 03.

Note on dynamic allocation:

If this statement is not specified in the JCL statement when these data sets are necessary, FPA allocates them dynamically.

S0nnnnn0 DD

These DD statements are optional DD statements. They point to the build pointer segment record data sets that store build pointer segment records.

- In scan mode, these statements define the output data sets for generating build pointer segment records.
- In load mode, these statements define the input data sets that contain the generated build pointer segment records.
- In scan load mode, these statements define the intermediate work data sets for storing pointer segments until the segments are loaded to the secondary index databases.

nnnnn is a sequential number that FPA assigns to each secondary index database. FPA assigns a sequential number in the order the secondary index databases appear on the LCHILD statements in the DBDGEN source. The value can be in the range of 00001 - 25755.

These data sets can be allocated dynamically by one of the following methods:

- The BUILDDSNMASK keyword parameter of the INDEXBLD command
- The DSNAME keyword parameter of the BLDMFILECTL subcommand

For a description and space requirements for build pointer segment records, see [“Build pointer segment records”](#) on page 296.

Utility control statements for the Build Index function

The utility control statements for the Build Index function define the user description of the GLOBAL command and the INDEXBLD command. The utility control statements control the Build Index process.

The utility control statements must be specified in the HFPSYSIN data set. This control statement data set usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte, fixed-length records. Block size, if coded, must be a multiple of 80.

The following figure shows a sample of how to code the HFPSYSIN data set.

```
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
  INDEXBLD
  DBD=dbdname
/*
```

Figure 145. INDEXBLD HFPSYSIN data set

Subsections:

- [“GLOBAL command keywords”](#) on page 310
- [“INDEXBLD command keywords”](#) on page 311
- [“BLDMFILECTL subcommand keywords”](#) on page 312
- [“FILECTL subcommand keywords”](#) on page 313

GLOBAL command keywords

For the GLOBAL command, you can specify the keywords that are listed in the following table. For the format of the GLOBAL command, see [“GLOBAL command”](#) on page 394.

Table 59. GLOBAL command keywords

Keyword	Required or optional	Description	Topic
DBRC	Optional	Specifies to request or bypass the database recovery control (DBRC) facility process for the database that is specified by the DBD. When DBRC=Y is specified, a DBRC NOTIFY command is issued. If you set "NONRECOV NOICREQ" to the secondary index database, IC NEEDED flag is not turned on even if the DBRC NOTIFY command is issued in the Build Index process.	“DBRC keyword” on page 451
ICINCOMP	Optional	Specifies, when an image copy is used as an input data set, whether the image copy records are to be compressed by using the CSRCESTRV MVS macro when they are stored into a data space storage.	“ICINCOMP keyword” on page 474
IMSCATACB_INPUT	Optional	Specifies whether to retrieve the currently active ACB definition or the pending ACB definition from the IMS directory.	“IMSCATACB_INPUT keyword” on page 480
IMSCATHLQ	Optional	Specifies the high-level qualifier of the bootstrap data set of the IMS directory.	“IMSCATHLQ keyword” on page 481
ITKBLOAD	Optional	Specifies the library name of IMS Tools KB.	“ITKBLOAD keyword” on page 487
ITKBSRVR	Optional	Specifies the name of the IMS Tools KB server XCF group.	“ITKBSRVR keyword” on page 488
MSTIMESTAMP	Optional	Specifies whether to include a time stamp in the messages.	“MSTIMESTAMP keyword” on page 497
SCAN	Optional	Specifies to verify the syntax of command, keywords, and parameters in command sets without running FPA. The Scan function also allocates some (but not all) internal resources for running the process function.	“SCAN keyword” on page 523
SITE_DSNAME	Optional	Specifies the data set name for the site default table.	“SITE_DSNAME keyword” on page 539
SITE_MEMBER	Optional	Specifies the member name for the site default table.	“SITE_MEMBER keyword” on page 540

INDEXBLD command keywords

The Build Index function, the Resync function, and the Freespace Reclaim function use the INDEXBLD command to start the Build Index process. During the Build Index process, FPA builds, synchronizes, or reclaims space in secondary index databases. To determine which task to perform, FPA evaluates the IDXPROC keyword on the INDEXBLD command as follows:

- When IDXPROC=BUILD, FPA starts the Build Index function. During the job, FPA scans one or more DEDB areas and builds one or more secondary index databases.
- When IDXPROC=RESYNC, FPA starts the Resync function. During the job, FPA scans one or more DEDB areas and secondary index databases and synchronizes secondary index databases.
- When IDXPROC=FS_RECLAIM, FPA starts the Freespace Reclaim function. During the job, FPA reclaims the CIs in secondary index database data sets.

For the INDEXBLD command, you can specify the keywords that are listed in the following table. For the format of the INDEXBLD command, see [“INDEXBLD command \(for the Build Index function\)” on page 397](#).

Table 60. INDEXBLD command keywords

Keyword	Required or optional	Description	Topic
BUILDDSNMASK	Optional	Specifies the data set mask for dynamically allocating cataloged pre-existing build pointer segment record data sets.	“BUILDDSNMASK keyword” on page 446
BUILDMODE	Optional	Specifies the mode for building secondary index databases.	“ BUILDMODE keyword” on page 447
DBD	Required	Specifies the database description (DBD) that contains the areas to be processed.	“DBD keyword” on page 450
DUPKEYACT	Optional	Saves duplicate pointer segment records in duplicate pointer segment record data sets.	“DUPKEYACT keyword” on page 457
DUPKEYDSNMASK	Optional	Specifies the data set name or the mask to be used when you want the duplicate pointer segment record data set allocated dynamically.	“DUPKEYDSNMASK keyword” on page 458
DUPKEYMAX	Optional	Specifies the threshold for the number of duplicate keys per secondary index database. When the threshold is reached, FPA stops processing that secondary index database.	“DUPKEYMAX keyword” on page 458
IAREA	Optional	Specifies one or more input areas for the Build Index process.	“IAREA keyword” on page 469
ICACHE	Optional	Specifies the type of control intervals that are to be preloaded into a data space storage for processing.	“ICACHE keyword” on page 472
ICHECK	Optional	Controls the concurrent number of input devices to access by using the ITASKCTL keyword.	“ICHECK keyword” on page 473
ICINCOMP	Optional	Specifies, when an image copy is used as an input data set, whether the image copy records are to be compressed by using the CSRCESTRV MVS macro when they are stored into a data space storage.	“ICINCOMP keyword” on page 474
IDSNMASK	Optional	Specifies the data set name or the mask for dynamically allocating cataloged pre-existing input data sets.	“IDSNMASK keyword” on page 476
IDXPROC	Optional	Specifies the function that the Build Index process starts.	“IDXPROC keyword” on page 478
INDEXDBD	Optional	Specifies the database description (DBD) that contains information about the secondary index databases to be processed.	“INDEXDBD keyword” on page 482
ITASKCTL	Optional	Specifies the maximum number of input areas to be processed concurrently.	“ITASKCTL keyword” on page 486
OTASKCTL	Optional	Specifies the maximum number of output secondary index databases to be processed concurrently.	“OTASKCTL keyword” on page 503
PROC_AFT_DUPKEY	Optional	Specifies whether to end the job step when the number of duplicate keys reaches the threshold defined by the DUPKEYMAX keyword.	“PROC_AFT_DUPKEY keyword” on page 504
SORT	Optional	Specifies whether to sort the build pointer segment records based on the record type and the key of the index pointer segment.	“SORT keyword” on page 541

BLDMFILECTL subcommand keywords

The BLDMFILECTL subcommand defines the output files to be created during the Build Index process. One or more build pointer segment record data sets are produced based on the BLDMFILECTL subcommand when the Build Index function runs in scan mode.

For the BLDMFILECTL subcommand, you can specify the keywords that are listed in the following table. For the format of the BLDMFILECTL subcommand, see [“BLDMFILECTL subcommand” on page 422](#).

Table 61. *BLDMFILECTL* subcommand keywords

Keyword	Required or optional	Description	Topic
AVGREC	Optional	Identifies the space requirements of the output data set that is created by dynamic allocation.	“AVGREC keyword” on page 445
DATACLAS	Optional	Specifies the SMS data class of an output data set that is created by dynamic allocation.	“DATACLAS keyword” on page 449
DISP	Optional	Specifies the status and the disposition of the data set that is accessed by dynamic allocation.	“DISP keyword” on page 453
DSNAME	Optional	Specifies a mask for the data set name that is used for dynamic allocation of the data set to be processed by the subcommand.	“DSNAME keyword” on page 456
EXPDT	Optional	Specifies the expiration date of an output data set that is created by dynamic allocation. Do not specify both the EXPDT keyword and the RETPD keyword because they are mutually exclusive.	“EXPDT keyword” on page 462
INDEXDBD	Optional	Specifies the database description (DBD) that contains information about the secondary index databases to be processed.	“INDEXDBD keyword” on page 482
LIKE	Optional	Specifies the SMS allocation attributes of an output data set that is created by dynamic allocation. The model data set specified must be an existing SMS data set.	“LIKE keyword” on page 492
MGMTCLAS	Optional	Specifies the SMS management class of an output data set that is created by dynamic allocation.	“MGMTCLAS keyword” on page 496
RETPD	Optional	Specifies the retention period of an output data set that is created by dynamic allocation. Do not specify both the EXPDT keyword and the RETPD keyword because they are mutually exclusive.	“RETPD keyword” on page 519
SPACE	Optional	Specifies the space requirements of the output data set that is created by dynamic allocation.	“SPACE keyword” on page 543
STORCLAS	Optional	Specifies the SMS storage class of the output data set that is created by dynamic allocation.	“STORCLAS keyword” on page 546
UNIT	Optional	Specifies the physical device type to be used for a data set that is accessed by dynamic allocation. Use this keyword also to specify the number of physical devices to be allocated.	“UNIT keyword” on page 552
VOLCNT	Optional	Specifies the maximum number of volumes that will be required by an output data set that is created by dynamic allocation that uses nonspecific volumes.	“VOLCNT keyword” on page 556
VOLSER	Optional	Specifies one or more volume-serial identifiers for the data set that is accessed by dynamic allocation. If this keyword is omitted for an output data set, nonspecific (private) volumes are required.	“VOLSER keyword” on page 556

FILECTL subcommand keywords

The FILECTL subcommand defines the output files to create during the Build Index process. One or more duplicate pointer segment record data sets are produced based on the FILECTL subcommand when duplicate pointer segments are found during the process.

For the FILECTL subcommand, you can specify the keywords that are listed in the following table. For the format of the FILECTL subcommand, see [“FILECTL subcommand” on page 425](#).

Table 62. *FILECTL* subcommand keywords

Keyword	Required or optional	Description	Topic
AVGREC	Optional	Identifies the space requirements of the output data set that is created by dynamic allocation.	“AVGREC keyword” on page 445
DATACLAS	Optional	Specifies the SMS data class of an output data set that is created by dynamic allocation.	“DATACLAS keyword” on page 449

Table 62. FILECTL subcommand keywords (continued)

Keyword	Required or optional	Description	Topic
DISP	Optional	Specifies the status and the disposition of the data set that is accessed by dynamic allocation.	“DISP keyword” on page 453
DSNAME	Optional	Specifies a mask for the data set name that is used for dynamic allocation of the data set to be processed by the subcommand.	“DSNAME keyword” on page 456
EXPDT	Optional	Specifies the expiration date of an output data set that is created by dynamic allocation. Do not specify both the EXPDT keyword and the RETPD keyword because they are mutually exclusive.	“EXPDT keyword” on page 462
INDEXDBD	Optional	Specifies the database description (DBD) that contains information about the secondary index databases to be processed.	“INDEXDBD keyword” on page 482
LIKE	Optional	Specifies the SMS allocation attributes of an output data set that is created by dynamic allocation. The model data set specified must be an existing SMS data set.	“LIKE keyword” on page 492
MGMTCLAS	Optional	Specifies the SMS management class of an output data set that is created by dynamic allocation.	“MGMTCLAS keyword” on page 496
RETPD	Optional	Specifies the retention period of an output data set that is created by dynamic allocation. Do not specify both the EXPDT keyword and the RETPD keyword because they are mutually exclusive.	“RETPD keyword” on page 519
SPACE	Optional	Specifies the space requirements of the output data set that is created by dynamic allocation.	“SPACE keyword” on page 543
STORCLAS	Optional	Specifies the SMS storage class of the output data set that is created by dynamic allocation.	“STORCLAS keyword” on page 546
UNIT	Optional	Specifies the physical device type to be used for a data set that is accessed by dynamic allocation. Use this keyword also to specify the number of physical devices to be allocated.	“UNIT keyword” on page 552
VOLCNT	Optional	Specifies the maximum number of volumes that will be required by an output data set that is created by dynamic allocation that uses nonspecific volumes.	“VOLCNT keyword” on page 556
VOLSER	Optional	Specifies one or more volume-serial identifiers for the data set that is accessed by dynamic allocation. If this keyword is omitted for an output data set, nonspecific (private) volumes are required.	“VOLSER keyword” on page 556

Examples for the Build Index function

Review examples and learn techniques for building indexes for DEDB areas.

In the examples, the following DEDB is assumed:

```

DEDB name:  DEDBJN22 has four areas
  area name:  DB22AR0 (area number is 1 and has one ADS)
    ADS DD name:  DB22AR01,  DS name:  HPFP.DB22AR0.ADS1
  area name:  DB22AR1 (area number is 2 and has one ADS)
    ADS DD name:  DB22AR11,  DS name:  HPFP.DB22AR1.ADS1
  area name:  DB22AR2 (area number is 3 and has one ADS)
    ADS DD name:  DB22AR21,  DS name:  HPFP.DB22AR2.ADS1
  area name:  DB22AR3 (area number is 4 and has one ADS)
    ADS DD name:  DB22AR31,  DS name:  HPFP.DB22AR3.ADS1

Secondary index database name: INDEX1
  Primary DBDS DD name: INDEXK1, DS name: HPFP.INDEXK1
Secondary index database name: INDEX2
  Primary DBDS DD name: INDEXK2, DS name: HPFP.INDEXK2
  Overflow DBDS DD name: INDEXE2, DS name: HPFP.INDEXE2
Secondary index database name: INDEX3
  Primary DBDS DD name: INDEXK3, DS name: HPFP.INDEXK3
  Overflow DBDS DD name: INDEXE3, DS name: HPFP.INDEXE3

```

In DBRC, all ADSs are in the AVAILABLE status.

Example 1: Building all the secondary index databases for the DEDB areas

The following figure shows example JCL statement for building all the secondary index databases for the DEDB areas.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS to be validated is dynamically allocated. The allocation information (DD name and DS name of the ADS) is obtained from DBRC.
- The output secondary index database data sets to be validated are dynamically allocated. The allocation information (DD name and DS name of the secondary index database data sets) is obtained from DFSMDA members in the IMSDALIB DD data set. The output data sets must be defined before running this example.
- When the secondary index databases are built, the IC NEEDED status of all secondary index databases is changed to ON.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//        DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMS     DD DISP=SHR,DSN=IMSVS.DBDLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
          GLOBAL DBRC=YES
          INDEXBLD DBD=DEDBJN22,
                  IAREA=ALL,
                  INDEXDBD=ALL
/*
```

Figure 146. Building all the secondary index databases for the DEDB areas

Example 2: Building specific secondary index databases for the DEDB areas

The following figure shows example JCL statement for building specific secondary index databases for the DEDB areas.

When one or more secondary index databases are broken, you can use this method to rebuild only the broken secondary index databases.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS to be validated is dynamically allocated. The allocation information (DD name and DS name of the ADS) is obtained from DBRC.
- The output secondary index database data sets to be validated are dynamically allocated. The allocation information (DD name and DS name of the secondary index database data set) is obtained from DFSMDA members in the IMSDALIB DD data set. The output data sets must be defined before running this example.
- When the secondary index databases are built, the IC NEEDED status of secondary index databases INDEX1 and INDEX2 is changed to ON.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB   DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMS      DD DISP=SHR,DSN=IMSVS.DBDLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL DBRC=YES
INDEXBLD  DBD=DEDBJN22,
          IAREA=ALL,
          INDEXDBD=(INDEX1,INDEX2)
/*

```

Figure 147. Building specific secondary index databases for the DEDB areas

Example 3: Building a secondary index database that contains only certain pointer segments

The following figure shows example JCL statement for building one secondary index database for a DEDB area.

Generally, when building secondary index databases, all areas are used as input for the build process. However, if you want to build a secondary index database that contains only the pointer segments that point to a certain area, you can specify that area by using the IAREA keyword. This method is faster than building the secondary index databases by specifying all areas as input areas. Use this method when you know that the secondary index database to be created does not require pointer segments that point to all the areas.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADS to be validated is dynamically allocated. The allocation information (DD name and DS name of the ADS) is obtained from DBRC.
- The output secondary index database data sets to be validated are dynamically allocated. The allocation information (DD name and DS name of the secondary index database data set) is obtained from DFSMDA members in the IMSDALIB DD data set. The output data sets must be defined before running this example.
- When the secondary index database is built, the IC NEEDED status of secondary index database INDEX1 is changed to ON.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB   DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMS      DD DISP=SHR,DSN=IMSVS.DBDLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL DBRC=YES
INDEXBLD  DBD=DEDBJN22,
          IAREA=DB22AR0,
          INDEXDBD=INDEX1
/*

```

Figure 148. Building a secondary index database that contains only certain pointer segments

Example 4: Generating build pointer segment records and then concurrently building secondary indexes

The following figures show example JCL statements for building all secondary index databases for DEDB areas in three steps.

In the first step, Build Index jobs run in scan mode to generate build pointer segment records for all three secondary index databases. In the second step, DFSORT or an external sort program is used to sort the build pointer segment records. In the third step, Build Index jobs run in load mode to build all the three secondary index databases. In each step, multiple JCL jobs can be submitted simultaneously.

First step: Scanning and generating build pointer segment records (Scan mode)

The following two figures show JCL examples for the first step. These examples start Build Index processes and generate build pointer segment records.

In these examples:

- BUILDMODE=SCAN specifies to run the job in scan mode. FPA scans the DEDB areas and then generates the build pointer segment records for all index databases.
- BLDMFILECTL subcommand specifies to generate build pointer segment record data sets. Data sets HPFP.BUILD.SCANx.indexdbd are generated for the secondary index databases to be built.
- The only difference between the two jobs is the areas to scan, which are specified by the IAREA keyword.

```
//SCAN1 EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMS DD DISP=SHR,DSN=IMSVS.DBDLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL DBRC=YES
INDEXBLD DBD=DEDBJN22,
          IAREA=(DB22AR0,DB22AR1),
          INDEXDBD=ALL,
          IDXPROC=BUILD,
          BUILDMODE=SCAN
BLDMFILECTL DSNAME='HPFP.BUILD.SCAN1.&INDEX',
            DISP=(NEW,CATLG),
            UNIT=SYSDA
/*
```

Figure 149. Scanning areas 1 and 2, and generating build pointer segment records

```
//SCAN2 EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMS DD DISP=SHR,DSN=IMSVS.DBDLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL DBRC=YES
INDEXBLD DBD=DEDBJN22,
          IAREA=(DB22AR2,DB22AR3),
          INDEXDBD=ALL,
          IDXPROC=BUILD,
          BUILDMODE=SCAN
BLDMFILECTL DSNAME='HPFP.BUILD.SCAN2.&INDEX',
            DISP=(NEW,CATLG),
            UNIT=SYSDA
/*
```

Figure 150. Scanning areas 3 and 4, and generating build pointer segment records

Second step: Sorting the build pointer segment records

Records in the build pointer segment record data sets must be sorted before they are loaded into the secondary index databases in the load step.

The following JCL example is for DFSORT. The control statement cards for DFSORT are included in the Secondary Index Processing report that was generated in the preceding scan step.

In this example, the SORT1 step sorts the build pointer segment record data sets for secondary index INDEX1. The SORTIN DD statement specifies the build pointer segment record data sets that were generated by preceding Build Index jobs. For secondary index INDEX1, the SORTIN DD statement specifies two build pointer segment record data sets:

- HPPF.BUILD.SCAN1.INDEX1, which was created by the SCAN1 step in the preceding scan job.
- HPPF.BUILD.SCAN2.INDEX1, which was created by the SCAN2 step in the preceding scan job.

```
//SORT1 EXEC PGM=SORT
//SYSOUT DD SYSOUT=*
//SORTIN DD DSN=HPPF.BUILD.SCAN1.INDEX1,DISP=SHR
// DD DSN=HPPF.BUILD.SCAN2.INDEX1,DISP=SHR
//SORTOUT DD DSN=HPPF.BUILD.INDEX1.SORT,DISP=(NEW,CATLG),
// UNIT=SYSDA,SPACE=(CYL,(10,10)),DCB=(*.SORTIN)
//SYSIN DD *
SORT FIELDS=(1,00007,CH,A)
RECORD TYPE=F,LENGTH=00020
OPTION FILSZ=E0000441312
/*
//SORT2 EXEC PGM=SORT
//SYSOUT DD SYSOUT=*
//SORTIN DD DSN=HPPF.BUILD.SCAN1.INDEX2,DISP=SHR
// DD DSN=HPPF.BUILD.SCAN2.INDEX2,DISP=SHR
//SORTOUT DD DSN=HPPF.BUILD.INDEX2.SORT,DISP=(NEW,CATLG),
// UNIT=SYSDA,SPACE=(CYL,(10,10)),DCB=(*.SORTIN)
//SYSIN DD *
SORT FIELDS=(1,00010,CH,A)
RECORD TYPE=F,LENGTH=00020
OPTION FILSZ=E0000184434
/*
//SORT3 EXEC PGM=SORT
//SYSOUT DD SYSOUT=*
//SORTIN DD DSN=HPPF.BUILD.SCAN1.INDEX3,DISP=SHR
// DD DSN=HPPF.BUILD.SCAN2.INDEX3,DISP=SHR
//SORTOUT DD DSN=HPPF.BUILD.INDEX3.SORT,DISP=(NEW,CATLG),
// UNIT=SYSDA,SPACE=(CYL,(10,10)),DCB=(*.SORTIN)
//SYSIN DD *
SORT FIELDS=(1,00019,CH,A)
RECORD TYPE=F,LENGTH=00029
OPTION FILSZ=E0000166363
/*
```

Figure 151. Sorting the build pointer segment record data sets

Third step: Loading the build pointer segment records (Load mode)

The following three figures show JCL examples for the third step. These examples start Build Index processes and load index pointer segments into the secondary index databases.

In these examples:

- BUILDMODE=LOAD specifies to run the job in load mode. FPA reads the build pointer segment records and builds the secondary index databases.
- The build pointer segment records are obtained from the build pointer segment record data sets that are specified by BUILDDSNMASK='HPPF.BUILD.&INDEX.SORT'. The data set names are determined based on the SORTOUT DD statements of preceding sort steps. The data sets are allocated dynamically.
- SORT=NO, which specifies not to sort the records in the build pointer segment record data sets, is specified because the build pointer segment records are already sorted in preceding sort steps.
- The only difference among the three jobs is the secondary index DBD name, which is specified by the INDEXDBD keyword.

```

//LOAD1 EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMS DD DISP=SHR,DSN=IMSVS.DBDLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL DBRC=YES
INDEXBLD DBD=DEDBJN22,
          INDEXDBD=INDEX1,
          IDXPROC=BUILD,
          BUILDMODE=LOAD,
          BUILDDSNMASK='HPFP.BUILD.&INDEX.SORT',
          SORT=NO
/*

```

Figure 152. Building secondary index database INDEX1

```

//LOAD2 EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMS DD DISP=SHR,DSN=IMSVS.DBDLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL DBRC=YES
INDEXBLD DBD=DEDBJN22,
          INDEXDBD=INDEX2,
          IDXPROC=BUILD,
          BUILDMODE=LOAD,
          BUILDDSNMASK='HPFP.BUILD.&INDEX.SORT',
          SORT=NO
/*

```

Figure 153. Building secondary index database INDEX2

```

//LOAD3 EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMS DD DISP=SHR,DSN=IMSVS.DBDLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL DBRC=YES
INDEXBLD DBD=DEDBJN22,
          INDEXDBD=INDEX3,
          IDXPROC=BUILD,
          BUILDMODE=LOAD,
          BUILDDSNMASK='HPFP.BUILD.&INDEX.SORT',
          SORT=NO
/*

```

Figure 154. Building secondary index database INDEX3

Chapter 13. Synchronizing secondary index databases with a DEDB with FPA

Use the FPA Resync function to synchronize secondary index databases with their primary DEDB database.

Topics:

- [“Functions of Resync” on page 321](#)
- [“Input and output for the Resync function” on page 321](#)
- [“Running the Resync function” on page 326](#)
- [“Region size requirements for the Resync function” on page 332](#)
- [“DD statements for the Resync function” on page 333](#)
- [“Utility control statements for the Resync function” on page 343](#)
- [“Examples for the Resync function” on page 346](#)

Functions of Resync

The Resync function synchronizes index pointer segments in secondary index databases with the pointers in the primary DEDB database without bringing the DEDB database offline.

A DEDB and its secondary index databases might get out of synchronization if you enable the index suppression option in the BMP region and then update the DEDB. In this case, the DEDB is updated, but its secondary indexes are not. You can use the Resync function to synchronize such out-of-sync secondary index databases with their primary DEDB database.

Synchronization jobs of the Resync function run against online areas. The Resync function supports both online and offline synchronizations.

For online synchronization, FPA uses DL/I calls (ISRT and DLET) to update secondary index databases. For offline synchronization, FPA uses VSAM macro instructions to update secondary index databases. When offline synchronization is requested, FPA calls IMS Tools Online System Interface, which issues the /DBR, /DBD, and /STA commands to stop secondary index databases before the scan and start the databases after the synchronization.

IMS and the Resync function support two database structures for Fast Path secondary indexes, HISAM and SHISAM. Both structures offer sequential key access to primary DEDB databases. For more information about Fast Path secondary indexes, see *IMS Database Administration*.

Input and output for the Resync function

The INDEXBLD command in the HFPSYSIN DD control statement invokes the Resync function.

The Build Index function, the Resync function, and the Freespace Reclaim function use the INDEXBLD command to start the Build Index process. During the Build Index process, FPA builds, synchronizes, or reclaims space in secondary index databases. To determine which task to perform, FPA evaluates the IDXPROC keyword on the INDEXBLD command as follows:

- When IDXPROC=BUILD, FPA starts the Build Index function. During the job, FPA scans one or more DEDB areas and builds one or more secondary index databases.
- When IDXPROC=RESYNC, FPA starts the Resync function. During the job, FPA scans one or more DEDB areas and secondary index databases and synchronizes secondary index databases.
- When IDXPROC=FS_RECLAIM, FPA starts the Freespace Reclaim function. During the job, FPA reclaims the CIs in secondary index database data sets.

The input and output of a Resync job varies depending on how you synchronize secondary index databases. For the synchronization methods supported by the Resync function, see [“Three methods to synchronize secondary index databases”](#) on page 326.

When you use the Resync function, be aware of the following restrictions:

- The Resync function cannot read or scan image copies of a DEDB.
- Unlike other functions of FPA, the Resync function cannot invoke IMS HP Image Copy to create image copies.

Verify update method and direct update method

As shown in the following figure, the input consists of DEDB area data sets and secondary index database data sets, and the output consists of synchronized secondary index database data sets.

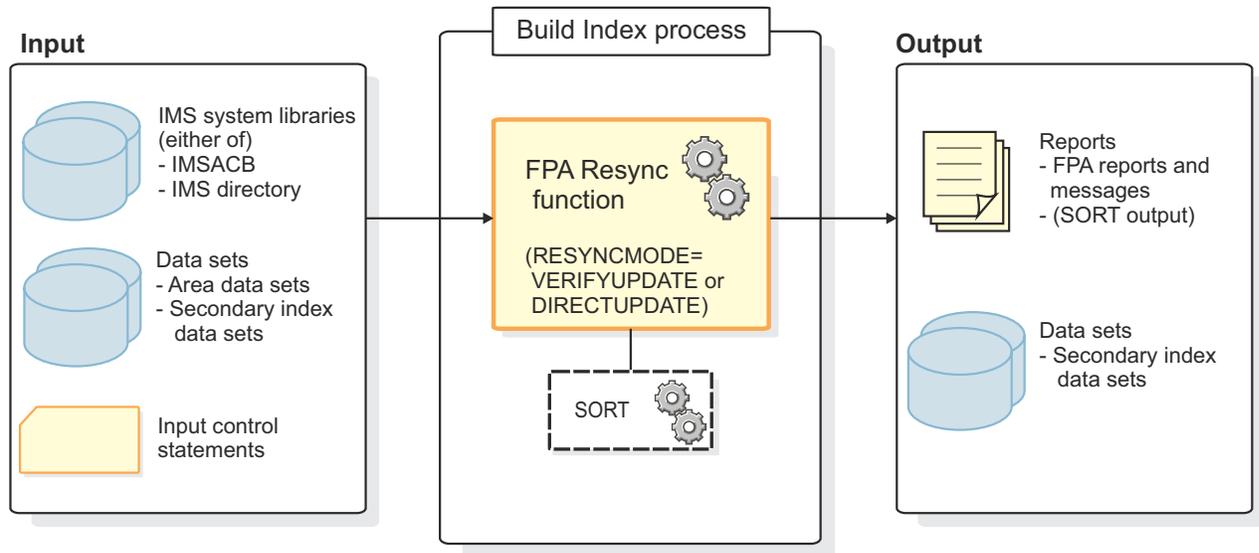


Figure 155. Input and output for the Resync function (verify update method and direct update method)

For JCL examples, see the following topics:

- [“Example 1: Synchronization using the verify update method”](#) on page 346
- [“Example 4: Synchronization using the verify update method \(offline\)”](#) on page 352

Verify update method and direct update method when area scan and sort are done in different jobs

The following figure shows the input and the output for the first Resync job, which runs the area scan process. When you specify the RESYNCMODE=AREASCAN option, the Resync function runs the area scan process only. The input consists of DEDB area data sets, and the output consists of unverified resync pointer segment record data sets.

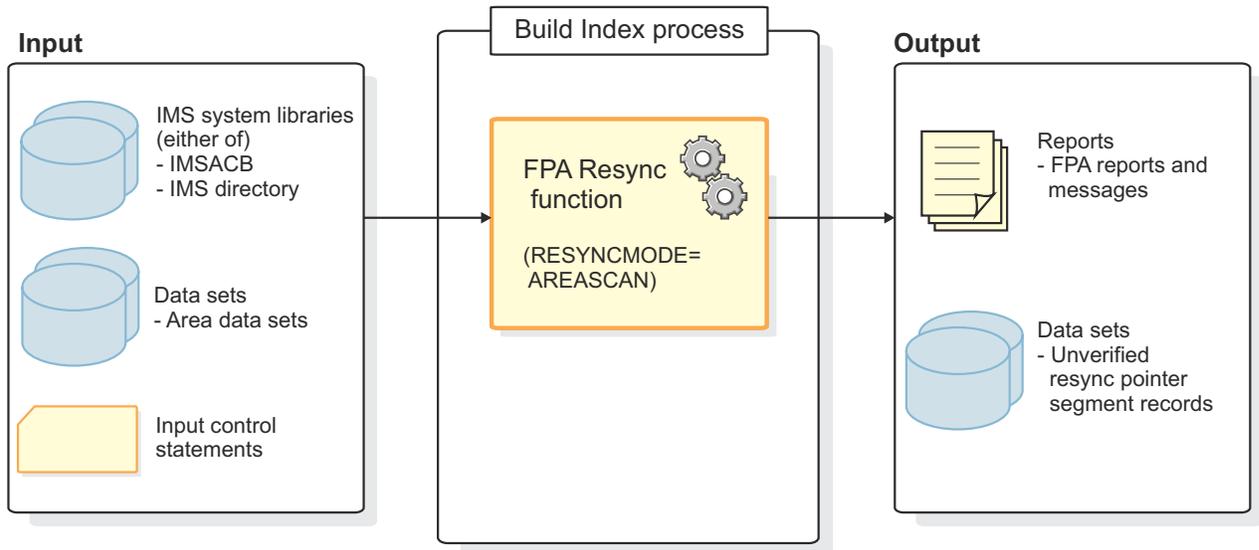


Figure 156. Input and output for the Resync function (area scan job)

The following figure shows the input and the output for the second Resync job. After the area scan job, you run a Resync job in verify update mode or direct update mode to update the secondary index databases. The input for this job consists of unverified resync pointer segment record data sets and secondary index database data sets, and the output consists of synchronized secondary index database data sets.

Note: Before you run the second Resync job in verify update mode, you must run a sort job to sort the unverified resync pointer segment records.

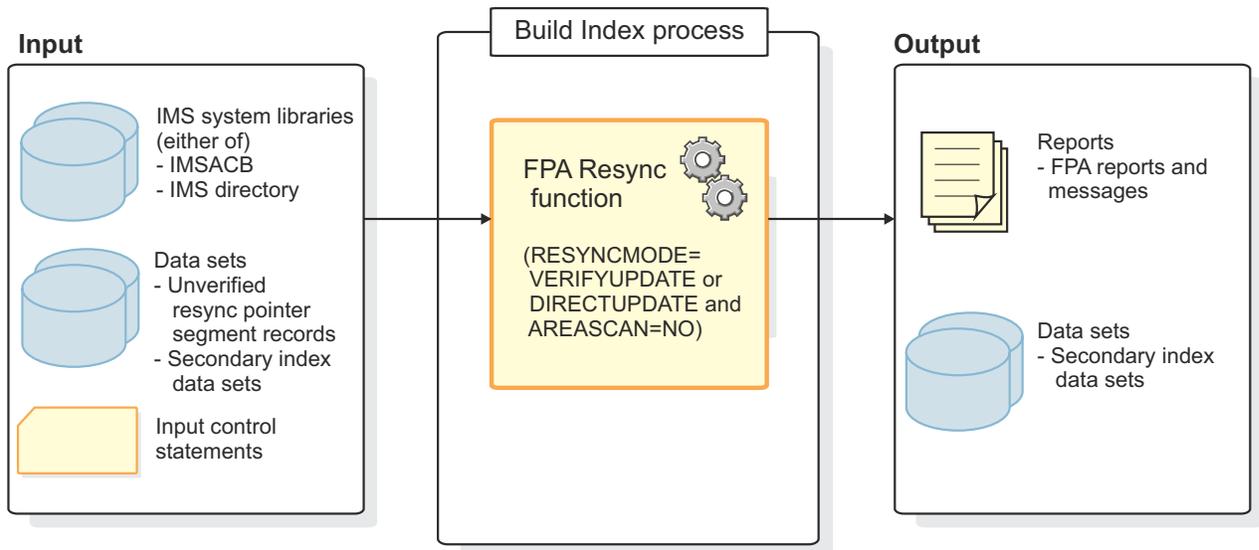


Figure 157. Input and output for the Resync function (verify update job or direct update job without the area scan process)

For JCL examples, see “[Example 3: Synchronization using the verify update method \(running area scan and sort processes separately\)](#)” on page 349.

Two-step method

The input for a Resync job in verify mode consists of DEDB area data sets and secondary index database data sets, and the output consists of resync pointer segment record data sets.

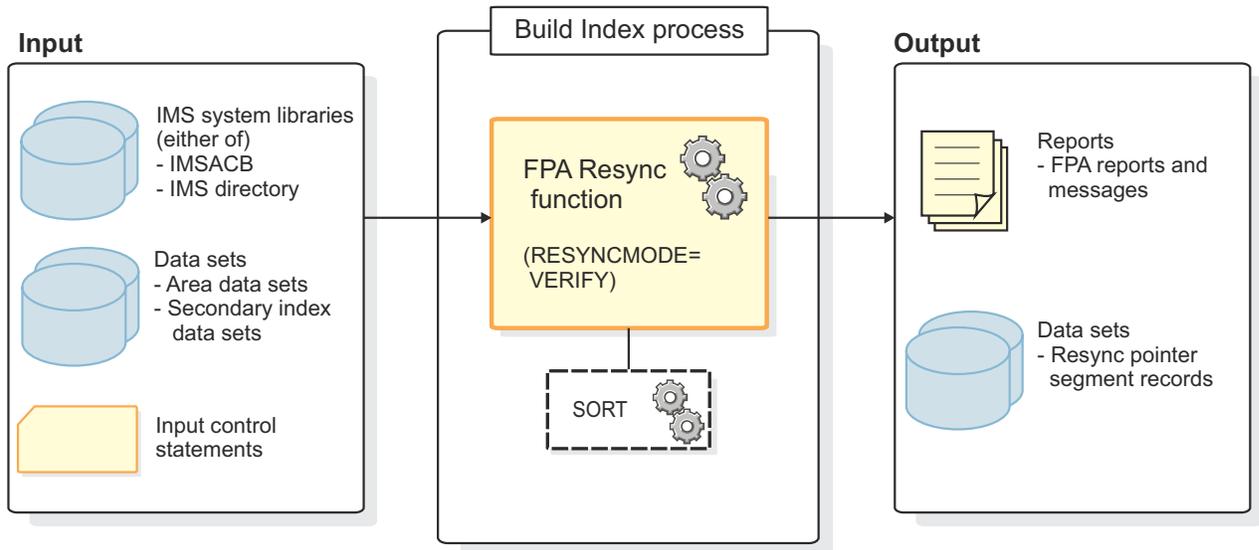


Figure 158. Input and output for the Resync function (verify mode)

The input for a Resync job in update mode consists of resync pointer segment record data sets and secondary index database data sets, and the output consists of synchronized secondary index database data sets.

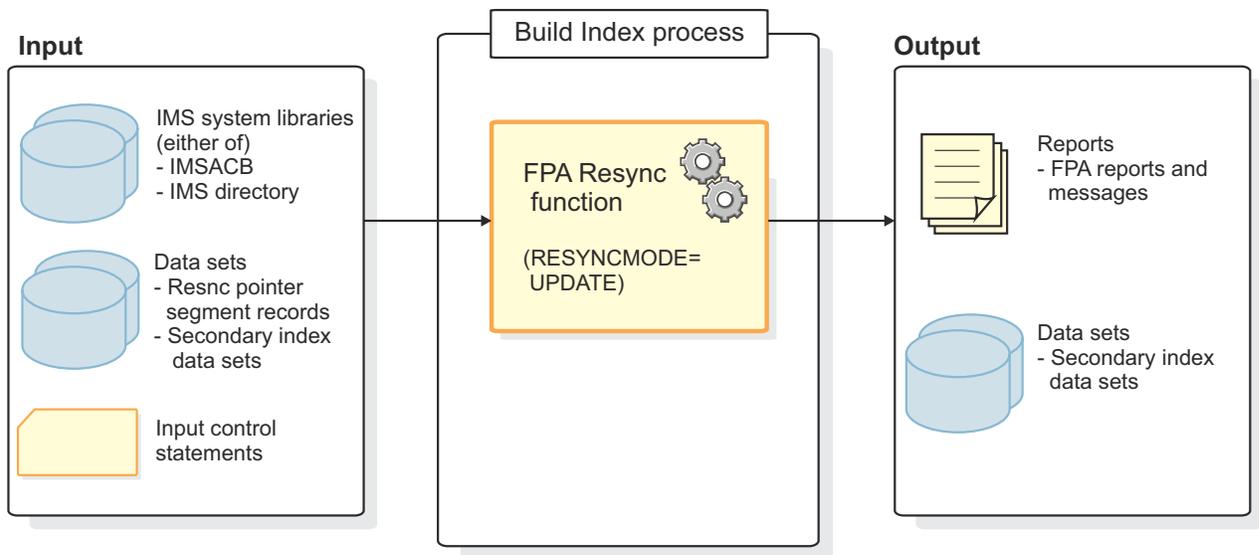


Figure 159. Input and output for the Resync function (update mode)

For JCL examples, see [“Example 2: Synchronization using the two-step method”](#) on page 347.

Resync work records

Resync pointer segment records and unverified resync pointer segment records are intermediate work records that the Resync function uses to pass data between Resync jobs.

Resync pointer segment records

Resync pointer segment records are segment records for the index pointer segments that require updating in the secondary index databases. These records are created in verify mode (RESYNCMODE=VERIFY) and are read in update mode (RESYNCMODE=UPDATE). FPA stores these records in resync pointer segment record data sets.

Unverified resync pointer segment records

Unverified resync pointer segment records are segment records for all index pointer segments for secondary index databases. These records are generated from the segments in DEDB areas. These records are created in area scan mode (RESYNCMODE=AREASCAN) and are read when a Resync job runs under one of the following conditions:

- Verify update mode without the area scan process (RESYNCMODE=VERIFYUPDATE and AREASCAN=NO)
- Direct update mode without the area scan process (RESYNCMODE=DIRECTUPDATE and AREASCAN=NO)

FPA stores these records in unverified resync pointer segment record data sets.

A Resync work data set cannot contain records for multiple secondary index databases. Therefore, you must use one Resync work data set for each secondary index database that requires synchronization.

For the format of Resync work records, see [“Resync pointer segment records and duplicate pointer segment records layout”](#) on page 1193.

Subsections:

- [“How to specify Resync work data sets to write data into”](#) on page 325
- [“How to specify Resync work data sets to read data from”](#) on page 326
- [“Space requirement for Resync work data sets”](#) on page 326

How to specify Resync work data sets to write data into

The following table summarizes the methods for specifying Resync work data sets to write data into and the behavior of the Resync function. Resync work data sets do not require DCB information to be specified in JCL.

Resync work data sets specified with	Description
<code>S0nnnnnR</code> DD statements or <code>S0nnnnn0</code> DD statements	Generates the records in the Resync work data sets that the following DD statements point to: <ul style="list-style-type: none">• <code>S0nnnnnR</code> DD statements for resync pointer segment record data sets• <code>S0nnnnn0</code> DD statements for unverified resync pointer segment record data sets
<code>RESYNCDSNMASK</code> keyword	Generates the records in the Resync work data sets that the <code>RESYNCDSNMASK</code> keyword specifies. You must allocate Resync work data sets before you run the job.
<code>FILECTL</code> subcommand with the <code>DISP=NEW</code> keyword	Dynamically generates Resync work data sets and the records in the data sets. You do not need to allocate Resync work data sets before you run the job.
<code>FILECTL</code> subcommand with the <code>DISP=OLD</code> or <code>DISP=SHR</code> keyword	Generates the records in the Resync work data sets that the <code>FILECTL</code> subcommand specifies. You must allocate Resync work data sets before you run the job.

How to specify Resync work data sets to read data from

The following table summarizes the methods for specifying Resync work data sets to read data from and the behavior of the Resync function.

Resync work data sets specified with	Description
S0nnnnnR DD statements or S0nnnnn0 DD statements	Obtains the records from the Resync work data sets that the following DD statements point to: <ul style="list-style-type: none">• S0nnnnnR DD statements for resync pointer segment record data sets• S0nnnnn0 DD statements for unverified resync pointer segment record data sets
RESYNCDSNMASK keyword	Obtains the records from the Resync work data sets that the RESYNCDSNMASK keyword specifies.

Space requirement for Resync work data sets

The space requirement for resync pointer segment record data sets depends on the length of the pointer segment and the number of segments that need to be updated in the secondary index databases. To estimate the space that is required for a resync pointer segment record data set, use the following formula. The unit is in bytes.

$$\begin{aligned} &(\text{Length of the pointer segment} + 11) \\ &\times (\text{Number of segments that are not reflected to the secondary index database}) \end{aligned}$$

The space requirement for unverified resync pointer segment record data sets depends on the length of the pointer segment and the number of pointer segments that are generated from the DEDB areas to scan. To estimate the space that is required for an unverified resync pointer segment record data set, use the following formula. The unit is in bytes.

$$\begin{aligned} &(\text{Length of the pointer segment} + 11) \\ &\times (\text{Number of pointer segments that are generated from scanned DEDB areas}) \end{aligned}$$

Running the Resync function

Use the Resync function to synchronize index pointer segments that are in secondary index databases with the pointers that are in the primary DEDB database. FPA runs the Resync function as a standard z/OS batch job. You must specify an EXEC statement and DD statements that define the input and output data sets in your JCL.

Considerations before running a Resync job:

- With online synchronization, if the DEDB is updated while the secondary index databases are being updated, the DEDB and the secondary index databases might be out of synchronization after the job. In such a case, run the Resync job again to synchronize the databases.
- Only the direct update method and offline synchronization support secondary index databases that contain non-unique keys. If the secondary index databases contain non-unique keys, use the direct update method or offline synchronization.

Three methods to synchronize secondary index databases

The Resync function supports three methods for synchronizing secondary index databases. Each method involves running one or more Resync jobs in corresponding Resync modes, which are controlled by the RESYNCMODE keyword of the INDEXBLD command.

For online synchronization, the following three methods are supported. For offline synchronization, only the verify update method is supported.

Verify update method (RESYNCMODE=VERIFYUPDATE)

The Resync function scans the DEDB areas and their secondary index databases to identify out-of-sync pointers in the secondary index databases. Then to update the secondary index databases, the function inserts and deletes the identified pointer segments. This method involves the following processes:

1. Area scan process: Scans the DEDB areas and generates records in the unverified resync pointer segment record data sets.
2. Sort process: Starts DFSORT and sorts the records in the unverified resync pointer segment record data sets.
3. Verify update process: Scans the secondary index databases, compares the pointer segment records in the secondary index databases to the records in the unverified resync pointer segment record data sets, and updates the secondary index databases.

By default, the Resync function runs the three processes in one job. However, if you use online synchronization, you can separate the area scan process and the sort process into different jobs. For the area scan process, you must use the Resync function. For the sort process, you can use DFSORT or a functionally equivalent sort program.

Direct update method (RESYNCMODE=DIRECTUPDATE)

The Resync function scans only the DEDB areas and inserts all pointer segments into the secondary index databases. The Resync function does not scan the secondary index databases because it assumes that the secondary index databases are empty or do not contain any index segments for the scanned DEDB areas. This method involves the following processes:

1. Area scan process: Scans the DEDB areas and generates records in the unverified resync pointer segment record data sets.
2. Direct update process: Reads the unverified resync pointer segment records and updates the secondary index databases.

The Resync function runs both processes in one job. However, you can separate the area scan process into a different Resync job.

Two-step method (RESYNCMODE=VERIFY and RESYNCMODE=UPDATE)

Using this method, you run two Resync jobs. The first job (verify step) scans the DEDB areas and their secondary index databases, and the second job (update step) updates the secondary index databases.

1. Verify step: Scans the DEDB areas and their secondary index databases, generates intermediate pointer segment records, uses DFSORT to sort the records, and identifies out-of-sync pointers. The records are stored in the resync pointer segment record data sets and used in the subsequent update step.
2. Update step: Reads the out-of-sync pointers in the resync pointer segment records, and updates the secondary index databases.

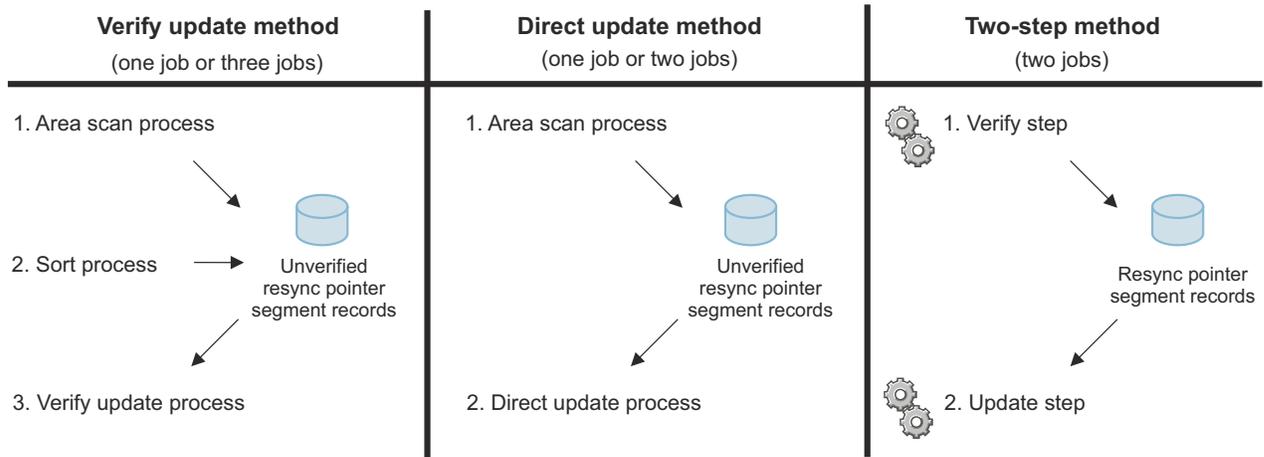


Figure 160. Resync methods

To synchronize secondary index databases, see the following topics:

- [“Synchronizing with the verify update method or the direct update method” on page 328](#)
- [“Synchronizing with the two-step method” on page 330](#)

Synchronizing with the verify update method or the direct update method

If you use the verify update method or the direct update method, you can use one job or multiple jobs to synchronize secondary index databases.

About this task

To use multiple jobs with the verify update method, you create three jobs: one for the area scan process, one for the sort process, and one for the verify update process. To use multiple jobs with the direct update method, you create two jobs: one for the area scan process and one for the direct update process.

Procedure

1. Optional: Estimate the region size that is required to run the job.
See [“Region size requirements for the Resync function” on page 332](#).
2. Specify the EXEC statement in the following format:

```
//RESYNC EXEC PGM=HFPMAIN0,REGION=rrrrM,
// PARM='DBRCGRP=dbrcgrp,GSGNAME=gsgname,IMSPLEX=imsplex,
IMSID=imsid'
```

Parameter	Description
DBRCGRP=	DBRCGRP is an optional 1- to 3-character parameter that specifies the identifier assigned to a group of DBRC instances (DBRC sharing group) that access the same RECON data set in the IMSplex.
GSGNAME=	GSGNAME is an optional 1- to 8-character parameter that specifies the global service group.
IMSPLEX=	IMSPLEX is an optional 1- to 5-character parameter that specifies the names of the IMSplex groups that share databases or message queues.

Parameter	Description
IMSID=	<p>IMSID is an optional 1- to 8-character parameter that specifies the IMS system identification name.</p> <p>If you specify this parameter, FPA generates the buffers for temporary ESCD and SCD. When FPA issues a call to a randomizing module, register 11 contains the address of the temporary ESCD. FPA sets only the ESCDSCD field, which contains the address of SCD, in the temporary ESCD. Also, FPA sets only the SSCDIMID field, which contains the IMS system ID, in the temporary SCD. Other fields of temporary ESCD and temporary SCD cannot be referred to.</p>

- Specify the DD statements that define input data sets and output data sets.

For descriptions of the DD statements, see [“DD statements for the Resync function” on page 333](#).

- Optional: To run the area scan process in a separate Resync job or use an external sort program, complete the steps in [“Scanning DEDB areas in area scan mode” on page 331](#).
- Specify the utility control statements.
 - Specify the INDEXBLD command in the HFPSYSIN DD.
 - Specify IDXPROC=RESYNC for the INDEXBLD command. If you do not specify RESYNC, the job runs the Build Index function.
 - Specify the RESYNCMODE keyword. To apply the verify update method, specify RESYNCMODE=VERIFYUPDATE. To apply the direct update method, specify RESYNCMODE=DIRECTUPDATE.

```
//HFPSYSIN DD *
GLOBAL DBRC=NO,
        IMSID=IMS1
INDEXBLD DBD=IVPDB3,
        IDXPROC=RESYNC,
        RESYNCMODE=VERIFYUPDATE
```

- Optional: To stop the secondary index databases during the job (offline synchronization), specify the TOICTL=DBR or TOICTL=DBD option to bring the secondary index databases offline. The TOICTL option is applicable only to verify update mode (RESYNCMODE=VERIFYUPDATE).
 - If you performed step [“4” on page 329](#), specify the AREASCAN=NO option to skip the area scan process and the sort process.
- If you performed step [“4” on page 329](#), use S0nnnnn0 DD statements or the RESYNCDSNMASK keyword to specify the generated unverified resync pointer segment data sets. The Resync function reads Resync work records from these data sets. For more information, see [“How to specify Resync work data sets to read data from” on page 326](#).
 - Submit the job.

Related concepts

Region size requirements for the Resync function

The Resync function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Related reference

[Example 1: Synchronization using the verify update method](#)

Review the example JCL statements that synchronize the DEDB areas and all secondary index databases with the verify update method.

[Example 3: Synchronization using the verify update method \(running area scan and sort processes separately\)](#)

Review the example JCL statements that synchronize the DEDB areas and all secondary index databases in three steps.

[Example 4: Synchronization using the verify update method \(offline\)](#)

Review the example JCL statements that synchronize the DEDB areas and all secondary index databases. During the job, IMS Tools Online System Interface issues IMS commands to bring the secondary index databases offline.

Tips and best practices for the FPA Resync function

The FPA Resync function supports several keywords that improve the performance of FPA Resync jobs and that further simplify management of DEDBs.

Synchronizing with the two-step method

Using the two-step method, you run two Resync jobs. The first job (verify step) scans the DEDB areas and their secondary index databases, and the second job (update step) updates the secondary index databases.

Procedure

1. Optional: Estimate the region size that is required to run the job.
See [“Region size requirements for the Resync function”](#) on page 332.
2. Specify the EXEC statement in the following format:

```
//RESYNC EXEC PGM=HFPMAIN0,REGION=rrrrM,
// PARM='DBRCGRP=dbrcgrp,GSGNAME=gsgname,IMSPLEX=imsplex,
IMSID=imsid'
```

Parameter	Description
DBRCGRP=	DBRCGRP is an optional 1- to 3-character parameter that specifies the identifier assigned to a group of DBRC instances (DBRC sharing group) that access the same RECON data set in the IMSplex.
GSGNAME=	GSGNAME is an optional 1- to 8-character parameter that specifies the global service group.
IMSPLEX=	IMSPLEX is an optional 1- to 5-character parameter that specifies the names of the IMSplex groups that share databases or message queues.
IMSID=	IMSID is an optional 1- to 8-character parameter that specifies the IMS system identification name. If you specify this parameter, FPA generates the buffers for temporary ESCD and SCD. When FPA issues a call to a randomizing module, register 11 contains the address of the temporary ESCD. FPA sets only the ESCDSCD field, which contains the address of SCD, in the temporary ESCD. Also, FPA sets only the SSCDIMID field, which contains the IMS system ID, in the temporary SCD. Other fields of temporary ESCD and temporary SCD cannot be referred to.

3. Specify the DD statements that define input data sets and output data sets.
For descriptions of the DD statements, see [“DD statements for the Resync function”](#) on page 333.
4. Code the S0nnnnR DD statements or utility control statements to specify the resync pointer segment record data sets. The Resync function creates Resync work records in these data sets. For more information, see [“How to specify Resync work data sets to write data into”](#) on page 325.
5. Specify the utility control statements.
 - a) Specify the INDEXBLD command in the HFPSYSIN DD.
 - b) Specify IDXPROC=RESYNC for the INDEXBLD command. If you do not specify RESYNC, the job runs the Build Index function.
 - c) Specify RESYNCMODE=VERIFY to run the job in verify mode.

```
//HFPSYSIN DD *
GLOBAL DBRC=NO
INDEXBLD DBD=IVPDB3,
          IDXPROC=RESYNC,
          RESYNCMODE=VERIFY
```

6. Submit the job. The Resync function runs in verify mode and generates resync pointer segment records in the resync pointer segment data sets.
7. Create another Resync job to run in update mode. You can copy and modify the JCL member that you created for verify mode.
8. Modify the utility control statements. Specify RESYNCMODE=UPDATE to run the job in update mode.

```
//HFPSYSIN DD *
GLOBAL DBRC=NO,
       IMSID=IMS1
INDEXBLD DBD=IVPDB3,
          IDXPROC=RESYNC,
          RESYNCMODE=UPDATE
```

9. Code the S0nnnnR DD statements or utility control statements to specify the resync pointer segment record data sets. The Resync function reads Resync work records from these data sets. For more information, see [“How to specify Resync work data sets to read data from”](#) on page 326.
10. Submit the job. The Resync function runs in update mode and updates the secondary index databases.

Related concepts

Region size requirements for the Resync function

The Resync function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Related reference

[Example 2: Synchronization using the two-step method](#)

Review the example JCL statements that synchronize the DEDB areas and all secondary index databases in two steps.

[Tips and best practices for the FPA Resync function](#)

The FPA Resync function supports several keywords that improve the performance of FPA Resync jobs and that further simplify management of DEDBs.

Scanning DEDB areas in area scan mode

Using the verify update method, you can separate the area scan process and the sort process into different jobs. Using the direct update method, you can separate the area scan process into a different job.

Before you begin

The following steps are for running a Resync job in area scan mode. These steps are a part of synchronizing secondary index databases with the verify update method or direct update method. See [“Synchronizing with the verify update method or the direct update method”](#) on page 328 for the entire instructions.

Procedure

1. Specify the INDEXBLD command in the HFPSYSIN DD.
2. Specify IDXPROC=RESYNC for the INDEXBLD command. If you do not specify RESYNC, the job runs the Build Index function.
3. Specify RESYNCMODE=AREASCAN to run the Resync function in area scan mode, which scans the DEDB areas and generates records in the unverified resync pointer segment record data sets.

```
//HFPSYSIN DD *
GLOBAL DBRC=NO
```

```
INDEXBLD DBD=IVPDB3,  
          IDXPROC=RESYNC,  
          RESYNCMODE=AREASCAN
```

4. Code the `SO n n n n 0` DD statements or utility control statements to specify the unverified resync pointer segment record data sets. The Resync function creates Resync work records in these data sets. For more information, see [“How to specify Resync work data sets to write data into”](#) on page 325.
5. Submit the job. The Resync function runs in area scan mode and generates unverified resync pointer segment records in the unverified resync pointer segment record data sets.
6. If you plan to run the subsequent job step in verify update mode (`RESYNCMODE=VERIFYUPDATE`), use DFSORT or a functionally equivalent sort program to run a sort job for the unverified resync pointer segment record data sets.

Tip: The Secondary Index Resync report, which is generated in step “5” on page 332, contains auto-generated sort control statements for DFSORT. When you use DFSORT to sort the records, you can copy and paste the control statements into a DFSORT job.

What to do next

See [“Synchronizing with the verify update method or the direct update method”](#) on page 328 and complete the synchronization.

Related concepts

[Region size requirements for the Resync function](#)

The Resync function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Related reference

[Example 3: Synchronization using the verify update method \(running area scan and sort processes separately\)](#)

Review the example JCL statements that synchronize the DEDB areas and all secondary index databases in three steps.

[Tips and best practices for the FPA Resync function](#)

The FPA Resync function supports several keywords that improve the performance of FPA Resync jobs and that further simplify management of DEDBs.

Region size requirements for the Resync function

The Resync function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Region size for the Resync function depends on the size of a UOW, the number of input areas, the number of output secondary index database data sets, the ITASKCTL specification, and the OTASKCTL specification. Typically, database maintenance is a relatively important function. To increase the performance of the Resync program, it would be advisable to run it in a region that has a relatively high dispatching priority (that is, at least equal to the dispatching priority of a message processing region).

Estimating the region size for the Resync function

The region size required for the Resync function is a function of the size of a UOW, the number of input areas, the number of output secondary index database data set, the ITASKCTL specification, and the OTASKCTL specification.

In most cases, the specification on the EXEC statement for HFPMAIN0 of a region that is too small, is detected during the initialization processing of the main task. This causes processing to be terminated at that point. Under certain conditions, the region size shortfall does not become evident until the actual Build Index process has begun. The job must be completely rerun. Thus, the region size calculations must always be performed prior to initiating Build Index processing.

In addition, the sort program also requires region space. For how to estimate the required region size, see the *DFSORT Application Programming Guide*.

List of DD statements

The following table summarizes the DD statements for the Resync function.

Table 63. DD statements for the Resync function

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
JOBLIB or STEPLIB	Input	PDS	No	Required
IMSACB, IMSACBA, or IMSACBB	Input	PDS	No	Required when IMSCATHLQ=*NO
IMSDALIB	Input	PDS	No	Optional
MODSTAT	Input	PS	Yes	Optional
MODSTAT2	Input	PS	Yes	Optional
OLCSTAT	Input	PS	Yes	Optional
IMSRESLB	Input	PDS	No	Optional
<i>addnname</i> or <i>areaname</i>	Input	VSAM ESDS cluster	Yes	Required under certain conditions (for details, see <i>addnname/areaname</i> DD)
<i>index</i>	Input and output	KSDS	Yes	Required when DFSMDA members for index databases are not used
<i>indexovrflw</i>	Input and output	ESDS	Yes	Required when DFSMDA members for index databases are not used
HFPSYSIN	Input	LRECL=80	No	Required
HFPCSITE	Input	PDS	Yes	Optional
HFPPRINT	Output	SYSOUT	Yes	Optional
HFPRPTS	Output	LRECL=133	Yes	Optional
HFPABEND	Output	LRECL=125	Yes	Optional
SYSPRINT	Output	LRECL=121	Yes	Optional
RECON <i>n</i>	Input and output	KSDS	Yes	Required when both of the following conditions are met: <ul style="list-style-type: none"> • DBRC=YES • DFSMDA members for RECON<i>n</i> data sets are not used
U <i>ttt</i> WK <i>n</i>	Input and output		Yes	Optional
U <i>ttt</i> MSG	Output	LRECL=133	Yes	Optional
S0 <i>nnnnn</i> 0	Input and output		Yes	Required under certain conditions (for details, see S0 <i>nnnnn</i> 0 DD)
S0 <i>nnnnn</i> R	Input and output		Yes	Required under certain conditions (for details, see S0 <i>nnnnn</i> R DD)
S0 <i>nnnnn</i> P	Input and output		Yes	Optional

Table 63. DD statements for the Resync function (continued)

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
S0nnnnD	Input and output		Yes	Optional

Input DD statements

The following input DD statements are used for the Resync function.

JOBLIB/STEPLIB DD

This DD statement is a required DD statement that defines the following input data sets:

- The FPA load module library (required).
- The library that contains DFSMDA dynamic allocation members for the RECON data set. When DBRC=YES is specified and the RECON n DD statement is omitted, this DD statement is required.
- The library that contains a default site default member (HFPCSITE) to determine the data set name and the member of the site default table.
- The IMS load module library (required).
- The library that contains the user-defined data entry database partition selection exit routine that was specified on the XDFLD PSELRTN= parameter during DBD generation for the DEDB.
- The library that contains the secondary index database maintenance exit routine that was specified on the XDFLD EXTRTN= parameter during DBD generation for the DEDB.
- The IMS Tools Base library (SHKTLOAD), if you specify the ITKBSRVR keyword.
- The IMS Tools Base libraries (SFOILOAD and SGLXLOAD), if you specify the TOICTL=DBR or DBD option.
- The IMS Tools Base library (SGLXLOAD), if you specify the IMSCATHLQ keyword.
- The IMS Library Integrity Utilities library, if you want to activate Integrity Checker.

FPA must be run as an APF-authorized program. All the data sets that are listed within the JOBLIB or the STEPLIB concatenation must also be APF-authorized. If you use DFSMDA members and the DFSMDA members are in a non-APF authorized library, you can specify the DFSMDA library on the IMSDALIB DD statement.

For more information about user-defined data entry database partition selection exit routines and secondary index database maintenance exit routines, see *IMS Exit Routines*.

IMSACB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. The IMSACB DD statements (IMSACB, IMSACBA, IMSACBB, MODSTAT, MODSTAT2, and OLCSTAT) can be used to identify the active ACB library. When both OLCSTAT or MODSTAT and MODSTAT2 are found, FPA checks the time stamp and determines the latest one.

The active IMSACB library is determined according to the following rules:

1. Check IMSACB DD statement in the JCL statement.
2. Check OLCSTAT DD statement in the JCL statement.
3. Check MODSTAT and MODSTAT2 DD statement in the JCL statement.
4. Allocate OLCSTAT dynamically and check the OLCSTAT member.
5. Allocate MODSTAT and MODSTAT2 dynamically and check the MODSTAT and MODSTAT2 members.

If you specify the IMSCATHLQ keyword, IMSACB DD statements are not used. The IMS directory is used instead of the ACB library.

IMSACBA DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSACBB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSDALIB DD

This DD statement is an optional DD statement that defines the IMSDALIB data set. This DD statement is useful when the data set for dynamic allocation members is not APF-authorized.

MODSTAT DD

This DD statement is an optional DD statement that defines the MODSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

MODSTAT2 DD

This DD statement is an optional DD statement that defines the MODSTAT2 data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT2 library by using a TYPE=DATABASE statement of the DFSMDA macro.

OLCSTAT DD

This DD statement is an optional DD statement that defines the OLCSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the OLCSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

IMSRESLB DD

This DD statement is an optional DD statement that defines the libraries that contain the randomizer routines, compression routines, user-defined data entry database partition selection exit routine and secondary index database maintenance exit routine that are specified in the ACB member (specified by the IMSACB DD statements). If the IMSRESLB DD statement is specified in the JCL statement, only the randomizer and compression routines are loaded directly from the IMSRESLB DD statement. If the IMSRESLB DD statement is not specified in the JCL statement, these resources are loaded from the JOBLIB or the STEPLIB DD statement.

The IMS version of IMSRESLB DD statement must be the same as those of IMSACB DD.

All the data sets that are listed within the IMSRESLB concatenation must be APF-authorized.

***addname/areaname* DD**

This DD statement is a required DD statement that defines an input data set, which is the VSAM ESDS or image copy for an area to be processed. The ddname used must be the ddname for the area that is defined in the input ACB library (IMSACB DD). There can be multiple *addname* DD statements for each area included in the process request.

The input data set that is used by FPA is determined by the following elements:

- The IDSNMASK keyword parameter of the INDEXBLD command

- When the GLOBAL DBRC=YES option is specified, the ADS information in the DBRC RECON data sets (only for VSAM ESDS data sets)
- The area name member in the IMS DFSMDA library (only for VSAM ESDS data sets)

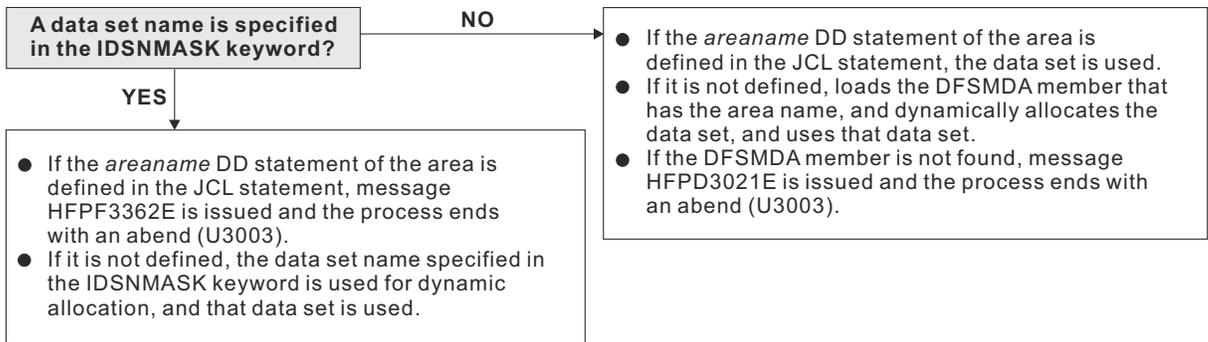
Or it can be specified with one of the following DD statements explicitly:

- *//addnname* DD JCL statement
- *//areaname* DD JCL statement

If an area data set is used for the INDEXBLD command input, it must be error free (that is, it has no EQEs). And if the area is registered with DBRC, the data set must have no EEQEs.

The following figure shows how FPA determines area data set names.

When DBRC=NO:



When DBRC=YES:

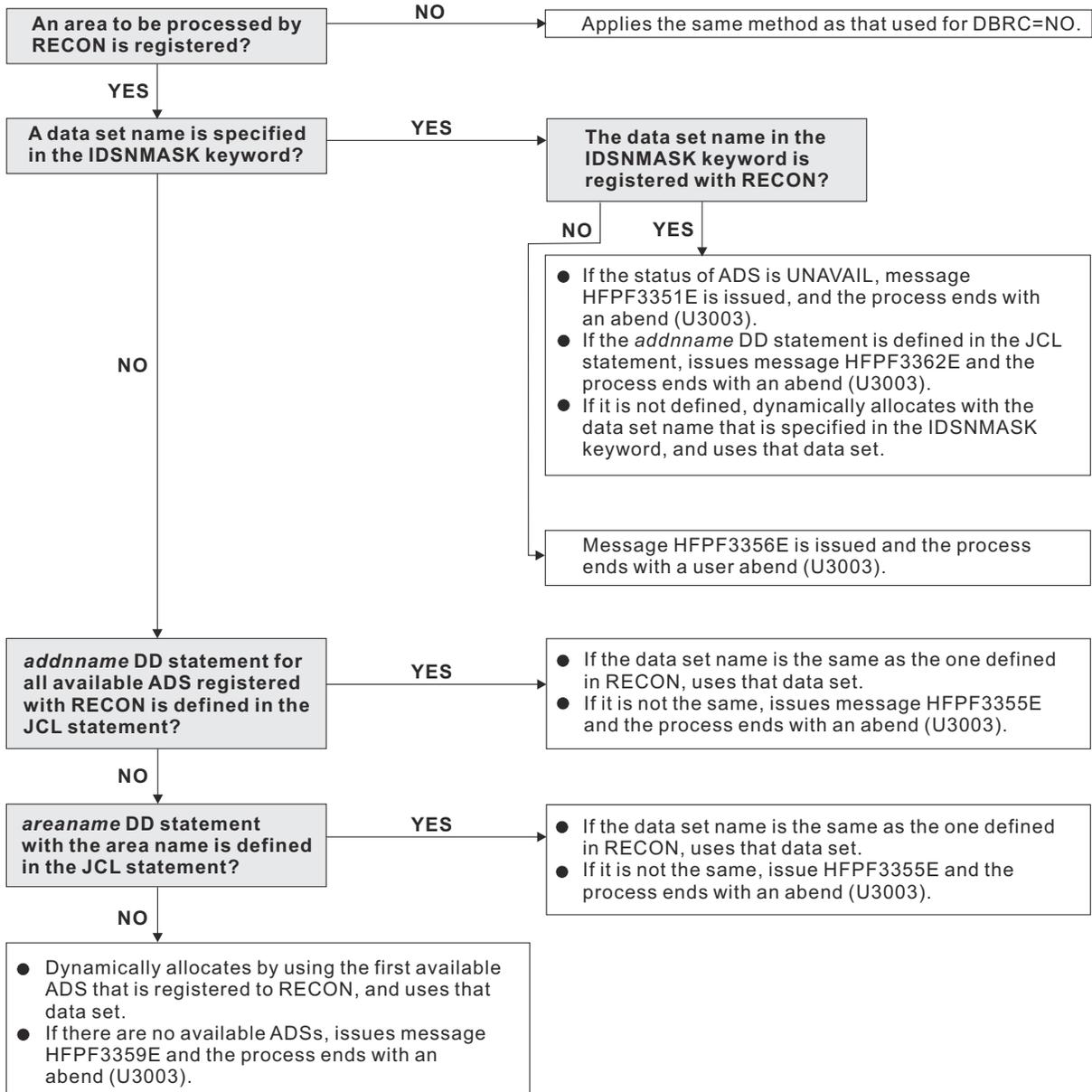


Figure 163. Input data set names for the INDEXBLD command

HFPSYSIN DD

This DD statement is a required DD statement that defines the input control statement that specifies the FPA functions.

The HFPSYSIN DD statement can be coded as a standard SYSIN file, a sequential data set, or a PDS member. LRECL=80 is required for the DCB of this data set. Only one command, besides the GLOBAL command, can be specified in HFPSYSIN. FPA analyzes the command syntax in columns 1 - 72. Columns 73 - 80 are regarded as comment.

For details about coding the HFPSYSIN DD statement, see [“Utility control statements for the Build Index function”](#) on page 310.

HFPCSITE DD

This DD statement is an optional DD statement. The HFPCSITE DD statement can be used to define the data set name of the site default table. When the SITE_DSNAME keyword is specified in the HFPSYSIN DD statement, the HFPCSITE DD statement is allocated dynamically. HFPCSITE DD in the JCL is mutually exclusive with the SITE_DSNAME keyword in the HFPSYSIN DD statement. The data set attribute of HFPCSITE DD is PDS.

Output DD statements

The following output DD statements are used for the Resync function.

HFPPRINT DD

This DD statement is an optional DD statement that defines the processing log output data set. The data set contains the processing messages that are issued by FPA.

The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set. You cannot specify DUMMY.

The HFPPRINT data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPPRINT DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

The following figure shows an example of the processing messages that are issued for the INDEXBLD command.

```
IMS HFPF UTILITIES                                "Processing messages"                                PAGE: 1
5698-FPP V2R1                                     2019-03-30 18:29:34
HFPC0001I COMMAND ANALYZE PROCESSING STARTED.
HFPC0005I DBRC=YES IS SPECIFIED.
- EEQE DETECTION IS PERFORMED.
HFPC0002I COMMAND ANALYZE PROCESSING COMPLETED.
HFPF0001I INDEXBLD PROCESSING STARTED FOR DATABASE: DEDBJN23
HFPB0001I INDEXBLD SCAN PROCESSING STARTED FOR AREA NO: 1, AREANAME: DB23AR0
HFPB0002I INDEXBLD SCAN PROCESSING COMPLETED FOR AREA NO: 1, AREANAME: DB23AR0 (ELAPSED TIME: 00:00:06.29)
HFPB0003I INDEXBLD COMPARE PROCESSING STARTED FOR SECONDARY INDEX DBD: ITHISM02
HFPB0004I INDEXBLD COMPARE PROCESSING COMPLETED FOR SECONDARY INDEX DBD: ITHISM02 (ELAPSED TIME: 00:00:00.41)
HFPB0003I INDEXBLD UPDATE PROCESSING STARTED FOR SECONDARY INDEX DBD: ITHISM02
HFPB0004I INDEXBLD UPDATE PROCESSING COMPLETED FOR SECONDARY INDEX DBD: ITHISM02 (ELAPSED TIME: 00:00:02.55)
HFPF0002I INDEXBLD PROCESSING COMPLETED FOR DATABASE: DEDBJN23
HFPL0001I REPORT STARTED FOR COMMAND: INDEXBLD, DATABASE: DEDBJN23
HFPL0002I REPORT COMPLETED FOR COMMAND: INDEXBLD, DATABASE: DEDBJN23
```

Figure 164. Processing messages issued for the INDEXBLD command

HFPFRPTS DD

This DD statement is an optional DD statement that defines the report output data set. The data set contains the reports that can be used to verify the process of the command. The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133

- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

The following reports are generated in this data set:

- Audit report
- Processing report
- DBD Definition report
- Secondary Index Definition report
- Secondary Index Resync report
- Resync Pointer Segment Dump report (when RESYNCDUMP=YES is specified)

The HFPRPTS data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPRPTS DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

HFPABEND DD

This DD statement is an optional DD statement. It defines a snap dump output data set that is used for diagnostic purposes. This data set might be used when the FPA job ends abnormally. Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

SYSPRINT DD

This DD statement is an optional DD statement that defines the processing log output data set, which contains the processing messages that are issued by the IMS system. This DD statement is used when you specify the TOICTL=NONE option.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

UttsMSG DD

This DD statement is an optional DD statement. However, this DD statement is required when DFSORT is used. This statement defines an output data set for storing DFSORT messages. For the condition for running DFSORT and the naming rules for the DD names, see the description of UttsWKnn DD.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

Input/Output DD statements

The following input/output DD statements are used for the Resync function.

index DD

This DD statement is a required DD statement when DFSMDA members for index databases are not used. This DD statement defines the VSAM KSDS data set to be synchronized.

The DD name must be the DD name of the secondary index database that is defined in the input ACB library (IMSACB DD). Multiple *index* DD statements can be specified for each secondary index database that is built by the Resync function.

The data set that is used by FPA is determined by the following elements:

- The name of the secondary index database member in the IMS DFSMDA library
- The *index* DD JCL statement

***indexovrflw* DD**

This DD statement is a required DD statement when DFSMDA members for index databases are not used. This DD statement defines the VSAM ESDS data set to be synchronized.

The DD name must be the DD name of the secondary index database that is defined in the input ACB library (IMSACB DD). Multiple *indexovrflw* DD statements can be specified for each secondary index database that is built by the Resync function.

The data set that is used by FPA is determined by the following elements:

- The name of the secondary index database member in the IMS DFSMDA library
- The *indexovrflw* DD JCL statement

RECON1/RECON2/RECON3 DD

These statements define the DBRC RECON data sets. These DD statements are required DD statements when both of the following conditions are met:

- DBRC=YES is specified for the GLOBAL command
- DFSMDA members for RECON n data sets are not used

The RECON n data set must be the same data set that is used in the IMS control region. Because FPA issues a signon, a signoff, an authorization call, and an unauthorization call to DBRC, it updates RECON n data sets.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is necessary, the data set is allocated dynamically by using a DFSMDA member.

UtttWKnn

These DD statements are optional DD statements. These are intermediate storage data sets that are used by DFSORT. For more information about coding the SORTWK nn DD statements, see *DFSORT Application Programming Guide*.

The adequate size of storage data sets is 1.5 to 2 times the input file size. In the Resync function, the input file is the pointer segments that are put into the secondary index database. The size of the file is computed by multiplying the length of the pointer segments that are put into the secondary index database by the number of pointer segments. Allocating twice the space used by the SORTIN data set is usually adequate for each work data set. This data set is allocated dynamically by OS. When the SORTIN is too large for sort, a B37 abend occurs. In this case, this is a required DD.

Do not specify DISP=MOD for these DD statements.

The naming rules of the ddnames are as follows:

ttt:

The hundreds, tenths, and units digits of n for n -th output thread subtask

Note: When a subtask ends processing a secondary index database, it starts processing the next secondary index database. For this reason, the n -th output thread subtask does not always process the n -th output data set.

nn:

Specifies the number of the SORT work data sets. This value must be in the range of 01 - 03.

Note on dynamic allocation:

If this statement is not specified in the JCL statement when these data sets are necessary, FPA allocates them dynamically.

S0nnnnn0 DD

These DD statements point to the unverified resync pointer segment record data sets, which are the data sets for storing resync pointer segment records.

These DD statements are used in Resync jobs that run in the following modes:

- Area scan mode (RESYNCMODE=AREASCAN), to generate records into the data sets
- Verify update mode with no area scan process (RESYNCMODE=VERIFYUPDATE and AREASCAN=NO), to read records from the data sets
- Direct update mode with no area scan process (RESYNCMODE=DIRECTUPDATE and AREASCAN=NO), to read records from the data sets

nnnnn is a sequential number that FPA assigns to each secondary index database. FPA assigns a sequential number in the order the secondary index databases appear on the LCHILD statements in the DBDGEN source. The value can be in the range of 00001 - 25755.

These data sets can be allocated dynamically by one of the following methods:

- The RESYNCDSNMASK keyword parameter of the INDEXBLD command
- The DSNAME keyword parameter of the FILECTL subcommand

For a description and space requirements for resync pointer segment records, see [“Resync work records” on page 324](#).

S0nnnnnR DD

These DD statements point to the resync pointer segment record data sets, which are the data sets for storing resync pointer segment records.

These DD statements are used only with the two-step method. In the verify step (RESYNCMODE=VERIFY), these statements define the output data sets for generating resync pointer segment records. In the update step (RESYNCMODE=UPDATE), these statements define the input data sets that contain the generated resync pointer segment records.

nnnnn is a sequential number that FPA assigns to each secondary index database. FPA assigns a sequential number in the order the secondary index databases appear on the LCHILD statements in the DBDGEN source. The value can be in the range of 00001 - 25755.

These data sets can be allocated dynamically by one of the following methods:

- The RESYNCDSNMASK keyword parameter of the INDEXBLD command
- The DSNAME keyword parameter of the FILECTL subcommand

For a description and space requirements for resync pointer segment records, see [“Resync work records” on page 324](#).

S0nnnnnP DD

These DD statements are optional DD statements. They point to the Duplicate Pointer Segment record data sets, which are the output data sets for storing duplicate pointer segment records. *nnnnn* is a sequential number that FPA assigns to each secondary index database. FPA assigns a sequential number in the order the secondary index databases appear on the LCHILD statements in the DBDGEN source. The value can be in the range of 00001 - 25755.

Note on dynamic allocation:

If this statement is not specified in the JCL statement when these data sets are necessary, FPA allocates them dynamically.

S0nnnnnD DD

These DD statements are optional DD statements that define intermediate work data sets that are used by FPA while it generates the Resync Pointer Segment Dump report. These DD statements are used in update mode (RESYNCMODE=UPDATE), verify update mode (RESYNCMODE=VERIFYUPDATE), and direct update mode (RESYNCMODE=DIRECTUPDATE) when the RESYNCDUMP=YES option, which specifies to generate the Resync Pointer Segment Dump report, is used.

nnnnn is a sequential number that FPA assigns to each secondary index database. FPA assigns a sequential number in the order the secondary index databases appear on the LCHILD statements in the DBDGEN source. The value can be in the range of 00001 - 25755.

You can estimate the space that is required for the intermediate data sets with the following formula. The unit is in bytes.

(Length of the pointer segment + 11) x (RESYNCDUMPMAX value)

Note on dynamic allocation:

If this statement is not specified in the JCL statement when these data sets are necessary, FPA allocates them dynamically.

Utility control statements for the Resync function

The utility control statements for the Resync function define the user description of the GLOBAL command, INDEXBLD command, and the FILECTL subcommand. The utility control statements control the Build Index process for the Resync function.

The utility control statements must be specified in the HFPSYSIN data set. This control statement data set usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte, fixed-length records. Block size, if coded, must be a multiple of 80.

The following figure shows a sample of how to code the HFPSYSIN data set.

```
//HFPSYSIN DD *
GLOBAL
  DBRC=YES,
  IMSID=IMS1
INDEXBLD
  DBD=dbdname,
  IDXPROC=RESYNC
/*
```

Figure 165. INDEXBLD HFPSYSIN data set

Subsections:

- [“GLOBAL command keywords” on page 343](#)
- [“INDEXBLD command keywords” on page 344](#)
- [“FILECTL subcommand keywords” on page 345](#)

GLOBAL command keywords

For the GLOBAL command, you can specify the keywords that are listed in the following table. For the format of the GLOBAL command, see [“GLOBAL command” on page 394](#).

Table 64. GLOBAL command keywords

Keyword	Required or optional	Description	Topic
DBRC	Optional	Specifies to request or bypass the database recovery control (DBRC) facility process for the database that is specified by the DBD. When DBRC=Y is specified, a DBRC NOTIFY command is issued. If you set "NONRECOV NOICREQ" to the secondary index database, IC NEEDED flag is not turned on even if the DBRC NOTIFY command is issued in the Build Index process.	“DBRC keyword” on page 451
IMSCATACB_INPUT	Optional	Specifies whether to retrieve the currently active ACB definition or the pending ACB definition from the IMS directory.	“IMSCATACB_INPUT keyword” on page 480
IMSCATHLQ	Optional	Specifies the high-level qualifier of the bootstrap data set of the IMS directory.	“IMSCATHLQ keyword” on page 481
IMSID	Conditionally required	Specifies the IMS system ID. The FPA process runs on the IMS system that is identified by this ID.	“IMSID keyword” on page 482
ITKBLOAD	Optional	Specifies the library name of IMS Tools KB.	“ITKBLOAD keyword” on page 487

Table 64. GLOBAL command keywords (continued)

Keyword	Required or optional	Description	Topic
ITKBSRVR	Optional	Specifies the name of the IMS Tools KB server XCF group.	“ITKBSRVR keyword” on page 488
MSTIMESTAMP	Optional	Specifies whether to include a time stamp in the messages.	“MSTIMESTAMP keyword” on page 497
SCAN	Optional	Specifies to verify the syntax of command, keywords, and parameters in command sets without running FPA. The Scan function also allocates some (but not all) internal resources for running the process function.	“SCAN keyword” on page 523
SITE_DSNAME	Optional	Specifies the data set name for the site default table.	“SITE_DSNAME keyword” on page 539
SITE_MEMBER	Optional	Specifies the member name for the site default table.	“SITE_MEMBER keyword” on page 540
SYNC	Optional	Specifies the interval between the synchronization points for updating secondary index databases with DL/I calls.	“SYNC keyword” on page 547
TOINUM	Optional	Specifies the number of retries for issuing the command to obtain a response from the IMS Tools Online System Interface.	“TOINUM keyword” on page 551
TOITIME	Optional	Specifies the time limit for obtaining a response from the IMS Tools Online System Interface.	“TOITIME keyword” on page 551
TOIXCFGRP	Optional	Specifies the XCF group that the IMS Tools Online System Interface uses to interface with FPA.	“TOIXCFGRP keyword” on page 552

INDEXBLD command keywords

The Build Index function, the Resync function, and the Freespace Reclaim function use the INDEXBLD command to start the Build Index process. During the Build Index process, FPA builds, synchronizes, or reclaims space in secondary index databases. To determine which task to perform, FPA evaluates the IDXPROC keyword on the INDEXBLD command as follows:

- When IDXPROC=BUILD, FPA starts the Build Index function. During the job, FPA scans one or more DEDB areas and builds one or more secondary index databases.
- When IDXPROC=RESYNC, FPA starts the Resync function. During the job, FPA scans one or more DEDB areas and secondary index databases and synchronizes secondary index databases.
- When IDXPROC=FS_RECLAIM, FPA starts the Freespace Reclaim function. During the job, FPA reclaims the CIs in secondary index database data sets.

For the INDEXBLD command, you can specify the keywords that are listed in the following table. For the format of the INDEXBLD command, see [“INDEXBLD command \(for the Resync function\)” on page 400](#).

Table 65. INDEXBLD command keywords

Keyword	Required or optional	Description	Topic
AREASCAN	Optional	Specifies whether to scan the DEDB areas during the Resync process.	“AREASCAN keyword” on page 442
DBD	Required	Specifies the database description (DBD) that contains the areas to be processed.	“DBD keyword” on page 450
DUPKEYMAX	Optional	Specifies the threshold for the number of duplicate keys per secondary index database. When the threshold is reached, FPA stops processing that secondary index database.	“DUPKEYMAX keyword” on page 458
IAREA	Optional	Specifies one or more input areas for the Resync process.	“IAREA keyword” on page 469
ICACHE	Optional	Specifies the type of control intervals that are to be preloaded into a data space storage for processing.	“ICACHE keyword” on page 472

Table 65. INDEXBLD command keywords (continued)

Keyword	Required or optional	Description	Topic
ICHECK	Optional	Controls the concurrent number of input devices to access by using the ITASKCTL keyword.	“ICHECK keyword” on page 473
IDSNMASK	Optional	Specifies the data set name or the mask for dynamically allocating cataloged pre-existing input data sets.	“IDSNMASK keyword” on page 476
IDXPROC	Required	Specifies the function that the Build Index process starts. You must specify IDXPROC=RESYNC to use the Resync function.	“IDXPROC keyword” on page 478
INDEXDBD	Optional	Specifies the database description (DBD) that contains information about the secondary index databases to be processed.	“INDEXDBD keyword” on page 482
ITASKCTL	Optional	Specifies the maximum number of input areas to be processed concurrently.	“ITASKCTL keyword” on page 486
OTASKCTL	Optional	Specifies the maximum number of output secondary index databases to be processed concurrently.	“OTASKCTL keyword” on page 503
PROC_AFT_DUPKEY	Optional	Specifies whether to end the job step when the number of duplicate keys reaches the threshold defined by the DUPKEYMAX keyword.	“PROC_AFT_DUPKEY keyword” on page 504
RESYNCDNSMASK	Optional	Specifies a data set mask for dynamically allocating cataloged pre-existing resync pointer segment record data sets or unverified resync pointer segment record data sets.	“RESYNCDNSMASK keyword” on page 515
RESYNCDUMP	Optional	Specifies whether to generate the Resync Pointer Segment Dump report. The index pointer segments that were or that will be updated to synchronize the secondary index databases with the primary DEDB database are written to the report.	“RESYNCDUMP keyword” on page 516
RESYNCDUMPMAX	Optional	Specifies the maximum number of index pointer segments to be printed for each index database in the Resync Pointer Segment Dump report.	“RESYNCDUMPMAX keyword” on page 517
RESYNCMODE	Optional	Specifies the synchronization mode for synchronizing secondary index databases with their primary DEDB database.	“RESYNCMODE keyword” on page 517
TOICTL	Optional	Specifies to temporarily stop the secondary index databases by using IMS Tools Online System Interface while the secondary index databases are being synchronized with the primary DEDB database.	“TOICTL keyword” on page 550

FILECTL subcommand keywords

The FILECTL subcommand defines the output files to create during the Build Index process. Each FILECTL subcommand produces one or more resync pointer segment record data sets or unverified resync pointer segment record data sets.

For the FILECTL subcommand, you can specify the keywords that are listed in the following table. For the format of the FILECTL subcommand, see [“FILECTL subcommand” on page 425](#).

Table 66. FILECTL subcommand keywords

Keyword	Required or optional	Description	Topic
AVGREC	Optional	Identifies the space requirements of the output data set that is created by dynamic allocation.	“AVGREC keyword” on page 445
DATACLAS	Optional	Specifies the SMS data class of an output data set that is created by dynamic allocation.	“DATACLAS keyword” on page 449
DISP	Optional	Specifies the status and the disposition of the data set that is accessed by dynamic allocation.	“DISP keyword” on page 453
DSNAME	Optional	Specifies a mask for the data set name that is used for dynamic allocation of the data set to be processed by the subcommand.	“DSNAME keyword” on page 456

Table 66. FILECTL subcommand keywords (continued)

Keyword	Required or optional	Description	Topic
EXPDT	Optional	Specifies the expiration date of an output data set that is created by dynamic allocation. Do not specify both the EXPDT keyword and the RETPD keyword because they are mutually exclusive.	“EXPDT keyword” on page 462
INDEXDBD	Optional	Specifies the database description (DBD) that contains information about the secondary index databases to be processed.	“INDEXDBD keyword” on page 482
LIKE	Optional	Specifies the SMS allocation attributes of an output data set that is created by dynamic allocation. The model data set specified must be an existing SMS data set.	“LIKE keyword” on page 492
MGMTCLAS	Optional	Specifies the SMS management class of an output data set that is created by dynamic allocation.	“MGMTCLAS keyword” on page 496
RETPD	Optional	Specifies the retention period of an output data set that is created by dynamic allocation. Do not specify both the EXPDT keyword and the RETPD keyword because they are mutually exclusive.	“RETPD keyword” on page 519
SPACE	Optional	Specifies the space requirements of the output data set that is created by dynamic allocation.	“SPACE keyword” on page 543
STORCLAS	Optional	Specifies the SMS storage class of the output data set that is created by dynamic allocation.	“STORCLAS keyword” on page 546
UNIT	Optional	Specifies the physical device type to be used for a data set that is accessed by dynamic allocation. Use this keyword also to specify the number of physical devices to be allocated.	“UNIT keyword” on page 552
VOLCNT	Optional	Specifies the maximum number of volumes that will be required by an output data set that is created by dynamic allocation that uses nonspecific volumes.	“VOLCNT keyword” on page 556
VOLSER	Optional	Specifies one or more volume-serial identifiers for the data set that is accessed by dynamic allocation. If this keyword is omitted for an output data set, nonspecific (private) volumes are required.	“VOLSER keyword” on page 556

Examples for the Resync function

Review examples and learn techniques for synchronizing DEDB areas and secondary index databases.

In the examples, the following DEDB is assumed:

```

DEDB name:  DEDBJN22 has four areas
  area name:  DB22AR0 (area number is 1 and has one ADS)
    ADS DD name:  DB22AR01,    DS name:  HPFP.DB22AR0.ADS1
  area name:  DB22AR1 (area number is 2 and has one ADS)
    ADS DD name:  DB22AR11,   DS name:  HPFP.DB22AR1.ADS1
  area name:  DB22AR2 (area number is 3 and has one ADS)
    ADS DD name:  DB22AR21,   DS name:  HPFP.DB22AR2.ADS1
  area name:  DB22AR3 (area number is 4 and has one ADS)
    ADS DD name:  DB22AR31,   DS name:  HPFP.DB22AR3.ADS1

Secondary index database name: INDEX1
  Primary DBDS DD name: INDEXK1, DS name: HPFP.INDEXK1
Secondary index database name: INDEX2
  Primary DBDS DD name: INDEXK2, DS name: HPFP.INDEXK2
Secondary index database name: INDEX3
  Primary DBDS DD name: INDEXK3, DS name: HPFP.INDEXK3
    
```

In DBRC, all ADSs are in the AVAILABLE status.

Example 1: Synchronization using the verify update method

Review the example JCL statements that synchronize the DEDB areas and all secondary index databases with the verify update method.

In this example:

- FPA uses the DFSMDA members in the IMSDALIB DD data set and dynamically allocates RECON data sets.
- FPA dynamically allocates the input ADS to validate. FPA obtains the allocation information (DD name and DS name of the ADS) from DBRC.
- FPA dynamically allocates the secondary index databases to synchronize. FPA obtains the allocation information (DD name and DS name of the secondary index database data sets) from the DFSMDA members in the IMSDALIB DD data set.
- During the synchronization, the DEDB areas and all secondary index databases remain online because TOICTL=NONE. Pointer segments to synchronize are inserted to or deleted from the secondary index databases by DL/I ISRT or DLET calls.

```
//HFP EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL DBRC=YES,
        IMSID=IMS1
INDEXBLD DBD=DEDBJN22,
        IAREA=ALL,
        INDEXDBD=ALL,
        IDXPROC=RESYNC,
        TOICTL=NONE,
        RESYNCMODE=VERIFYUPDATE
/*
```

Figure 166. Using the verify update method to synchronize the DEDB areas and all secondary index databases

Example 2: Synchronization using the two-step method

Review the example JCL statements that synchronize the DEDB areas and all secondary index databases in two steps.

In the first step, the Resync job runs in verify mode to generate resync pointer segment records for the three secondary index databases. In the second step, each Resync job runs in update mode to update the index pointer segments in the specified index segment database. For the second step, you can submit multiple JCL jobs simultaneously and process multiple secondary index databases concurrently.

In this example:

- RESYNCMODE=VERIFY specifies to run the job in verify mode. FPA scans the DEDB areas and all secondary indexes and generates the resync pointer segment records for all index databases.
- RESYNCDSNMASK='HPFP.RSNC.&INDEX' specifies the resync pointer segment record data sets. The resync pointer segment records are generated in resync pointer segment record data sets named HPFP.RSNC.*index_dbdname*. These data sets are allocated dynamically. Before you run the job, you must preallocate and catalog the data sets.

```

//HFP EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL DBRC=YES,
        IMSID=IMS1
INDEXBLD DBD=DEDBJN22,
        IAREA=ALL,
        INDEXDBD=ALL,
        IDXPROC=RESYNC,
        TOICTL=NONE,
        RESYNCMODE=VERIFY,
        RESYNCDSNMASK='HPFP.RSNC.&INDEX'
/*

```

Figure 167. First step: Generating resync pointer segment records

In the second step:

- RESYNCMODE=UPDATE specifies to run the job in update mode. FPA reads the resync pointer segment records and updates the secondary index databases to synchronize them with the primary DEDB database.
- The resync pointer segment records are obtained from the resync pointer segment record data sets that are specified by RESYNCDSNMASK='HPFP.RSNC.&INDEX'. The data sets are allocated dynamically.
- RESYNCDUMP=YES generates the Resync Pointer Segment Dump report.
- RESYNCDUMPMAX=500 specifies that information about a maximum of 500 pointer segments is written to the Resync Pointer Segment Dump report.
- The only difference among the three jobs is the secondary index DBD name that you specify on the INDEXDBD keyword.

```

//HFP EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL DBRC=YES,
        IMSID=IMS1
INDEXBLD DBD=DEDBJN22,
        INDEXDBD=INDEX1,
        IDXPROC=RESYNC,
        TOICTL=NONE,
        RESYNCMODE=UPDATE,
        RESYNCDSNMASK='HPFP.RSNC.&INDEX',
        RESYNCDUMP=YES,
        RESYNCDUMPMAX=500
/*

```

Figure 168. Second step: Reading resync pointer segment records and updating index pointer segments (example JCL 1)

```

//HFP EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL DBRC=YES,
        IMSID=IMS1
INDEXBLD DBD=DEDBJN22,
          INDEXDBD=INDEX2,
          IDXPROC=RESYNC,
          TOICTL=NONE,
          RESYNCMODE=UPDATE,
          RESYNCDSNMASK='HPFP.RSNC.&INDEX',
          RESYNCDUMP=YES,
          RESYNCDUMPMAX=500
/*

```

Figure 169. Second step: Reading resync pointer segment records and updating index pointer segments (example JCL 2)

```

//HFP EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL DBRC=YES,
        IMSID=IMS1
INDEXBLD DBD=DEDBJN22,
          INDEXDBD=INDEX3,
          IDXPROC=RESYNC,
          TOICTL=NONE,
          RESYNCMODE=UPDATE,
          RESYNCDSNMASK='HPFP.RSNC.&INDEX',
          RESYNCDUMP=YES,
          RESYNCDUMPMAX=500
/*

```

Figure 170. Second step: Reading resync pointer segment records and updating index pointer segments (example JCL 3)

Example 3: Synchronization using the verify update method (running area scan and sort processes separately)

Review the example JCL statements that synchronize the DEDB areas and all secondary index databases in three steps.

In the first step, the Resync job runs in area scan mode to generate unverified resync pointer segment records for the three secondary index databases. In the second step, DFSORT or an external sort program sorts the unverified resync pointer segment records. In the third step, each Resync job runs in verify update mode to compare the unverified resync pointer segment records with the secondary index databases and update the index pointer segments in the specified index segment database. For the third step, you can submit multiple JCL jobs simultaneously to process the secondary index databases concurrently.

First step: Scanning DEDB areas and generating unverified resync pointer segment records (Area scan mode)

The following two figures show JCL examples for the first step. These examples start Resync jobs in area scan mode and generate unverified resync pointer segment records.

In these examples:

- RESYNCMODE=AREASCAN specifies to run the job in area scan mode. FPA scans the DEDB areas and generates the unverified resync pointer segment records for all index databases.
- FILECTL subcommand specifies the unverified resync pointer segment record data sets. For each secondary index database, a data set named HPFP.RSNC.SCANx.indexdbd is generated.
- The only differences between the two jobs are the areas to scan, which are specified by the IAREA keyword, and the unverified resync pointer segment record data sets, which are specified by the DSNAME keyword.

```
//SCAN1 EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL DBRC=YES
INDEXBLD DBD=DEDBJN22,
          IAREA=(DB22AR0,DB22AR1),
          INDEXDBD=ALL,
          IDXPROC=RESYNC,
          TOICTL=NONE,
          RESYNCMODE=AREASCAN
FILECTL
          DSNAME='HPFP.RSNC.SCAN1.&INDEX',
          DISP=(NEW,CATLG),
          SPACE=(CYL,10,10)
/*
```

Figure 171. Scanning areas 1 and 2, and generating unverified resync pointer segment records

```
//SCAN2 EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL DBRC=YES
INDEXBLD DBD=DEDBJN22,
          IAREA=(DB22AR2,DB22AR3),
          INDEXDBD=ALL,
          IDXPROC=RESYNC,
          TOICTL=NONE,
          RESYNCMODE=AREASCAN
FILECTL
          DSNAME='HPFP.RSNC.SCAN2.&INDEX',
          DISP=(NEW,CATLG),
          SPACE=(CYL,10,10)
/*
```

Figure 172. Scanning areas 3 and 4, and generating unverified resync pointer segment records

Second step: Sorting the unverified resync pointer segment records

Records in the unverified resync pointer segment record data sets must be sorted before they are used in verify update mode. The following JCL example is for DFSORT. The control statement cards for DFSORT are printed in the Secondary Index Resync report that was generated in the preceding area scan step.

```

//SORT1 EXEC PGM=SORT
//SYSOUT DD SYSOUT=*
//SORTIN DD DSN=HPFP.RSNC.SCAN1.INDEX1,DISP=SHR
// DD DSN=HPFP.RSNC.SCAN2.INDEX1,DISP=SHR
//SORTOUT DD DSN=HPFP.RSNC.INDEX1.SORT,DISP=(NEW,CATLG),
// UNIT=SYSDA,SPACE=(CYL,(10,10)),DCB=(*.SORTIN)
//SYSIN DD *
SORT FIELDS=(1,00007,CH,A)
RECORD TYPE=F,LENGTH=00020
OPTION FILSZ=E0000441312
/*
//SORT2 EXEC PGM=SORT
//SYSOUT DD SYSOUT=*
//SORTIN DD DSN=HPFP.RSNC.SCAN1.INDEX2,DISP=SHR
// DD DSN=HPFP.RSNC.SCAN2.INDEX2,DISP=SHR
//SORTOUT DD DSN=HPFP.RSNC.INDEX2.SORT,DISP=(NEW,CATLG),
// UNIT=SYSDA,SPACE=(CYL,(10,10)),DCB=(*.SORTIN)
//SYSIN DD *
SORT FIELDS=(1,00010,CH,A)
RECORD TYPE=F,LENGTH=00020
OPTION FILSZ=E0000184434
/*
//SORT3 EXEC PGM=SORT
//SYSOUT DD SYSOUT=*
//SORTIN DD DSN=HPFP.RSNC.SCAN1.INDEX3,DISP=SHR
// DD DSN=HPFP.RSNC.SCAN2.INDEX3,DISP=SHR
//SORTOUT DD DSN=HPFP.RSNC.INDEX3.SORT,DISP=(NEW,CATLG),
// UNIT=SYSDA,SPACE=(CYL,(10,10)),DCB=(*.SORTIN)
//SYSIN DD *
SORT FIELDS=(1,00019,CH,A)
RECORD TYPE=F,LENGTH=00029
OPTION FILSZ=E0000166363
/*

```

Figure 173. Sorting the unverified resync pointer segment record data sets

Third step: Reading unverified resync pointer segment records and updating index pointer segments (Verify update mode)

The following three figures show JCL examples for the third step. These examples start Resync jobs in verify update mode. During the job, the Resync function reads the records in the unverified resync pointer segment record data sets, compares the records with the pointer segments in the secondary index databases, and updates the secondary index databases.

In these examples:

- RESYNCMODE=VERIFYUPDATE specifies to run the job in verify update mode.
- AREASCAN=NO specifies to skip the area scan process and the sort process. When this option is specified, FPA reads the records in the unverified resync pointer segment record data sets.
- The unverified resync pointer segment records are obtained from the unverified resync pointer segment record data sets that are specified by RESYNCDSNMASK='HPFP.RSNC.&INDEX.SORT'. The data sets are allocated dynamically.
- The only differences among the three jobs are the secondary index DBD name, which is specified by the INDEXDBD keyword, and the IMS system ID, which is specified by the IMSID keyword.

```

//VUPD1 EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL DBRC=YES,
IMSID=IMS1
INDEXBLD DBD=DEDBJN22,
INDEXDBD=INDEX1,
IDXPROC=RESYNC,
TOICTL=NONE,
RESYNCMODE=VERIFYUPDATE,
AREASCAN=NO,
RESYNCDSNMASK='HPFP.RSNC.&INDEX.SORT'
/*

```

Figure 174. Updating secondary index database INDEX1

```

//VUPD2 EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL DBRC=YES,
IMSID=IMS2
INDEXBLD DBD=DEDBJN22,
INDEXDBD=INDEX2,
IDXPROC=RESYNC,
TOICTL=NONE,
RESYNCMODE=VERIFYUPDATE,
AREASCAN=NO,
RESYNCDSNMASK='HPFP.RSNC.&INDEX.SORT'
/*

```

Figure 175. Updating secondary index database INDEX2

```

//VUPD3 EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL DBRC=YES,
IMSID=IMS3
INDEXBLD DBD=DEDBJN22,
INDEXDBD=INDEX3,
IDXPROC=RESYNC,
TOICTL=NONE,
RESYNCMODE=VERIFYUPDATE,
AREASCAN=NO,
RESYNCDSNMASK='HPFP.RSNC.&INDEX.SORT'
/*

```

Figure 176. Updating secondary index database INDEX3

Example 4: Synchronization using the verify update method (offline)

Review the example JCL statements that synchronize the DEDB areas and all secondary index databases. During the job, IMS Tools Online System Interface issues IMS commands to bring the secondary index databases offline.

In this example:

- FPA uses the DFSMDA members in the IMSDALIB DD data set and dynamically allocates RECON data sets.

- FPA dynamically allocates the input ADS to be validated. The allocation information (DD name and DS name of the ADS) is obtained from DBRC.
- FPA dynamically allocates the secondary index database data sets to be validated. The allocation information (DD name and DS name of the secondary index database data sets) is obtained from DFSMDA members in the IMSDALIB DD data set.
- TOICTL=DBR specifies to use IMS Tools Online System Interface to stop and restart the secondary index databases. The interface issues a /DBRECOVERY command to stop the secondary index databases and a /START command to start the secondary index databases.

```
//HFP EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=TOSI.SFOILOAD
//          DD DISP=SHR,DSN=TOSI.SGLXLOAD
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL DBRC=YES
INDEXBLD DBD=DEDBJN22,
          IAREA=ALL,
          INDEXDBD=ALL,
          IDXPROC=RESYNC,
          TOICTL=DBR
/*
```

Figure 177. Synchronizing the DEDB areas and all secondary index databases (offline)

Chapter 14. Reclaiming CIs in secondary index databases with FPA

Use the FPA Freespace Reclaim function to reclaim the control intervals (CIs) in secondary index database data sets.

Topics:

- [“Functions of Freespace Reclaim” on page 355](#)
- [“Input and output for the Freespace Reclaim function” on page 355](#)
- [“Running the Freespace Reclaim function” on page 356](#)
- [“Region size requirements for the Freespace Reclaim function” on page 357](#)
- [“DD statements for the Freespace Reclaim function” on page 358](#)
- [“Utility control statements for the Freespace Reclaim function” on page 363](#)
- [“Examples for the Freespace Reclaim function” on page 364](#)

Functions of Freespace Reclaim

The Freespace Reclaim function reclaims CIs in secondary index database data sets without bringing the DEDB and secondary index databases offline. This function is effective for secondary indexes that use a HISAM database with no overflow data set and when the HISAM database contains many index pointer segments whose delete byte flag is turned on.

When IMS issues DL/I calls to delete the source segments in the primary DEDB, the index pointer segments in the secondary index databases are not physically deleted but their delete byte flag is turned on to indicate that the segment is no longer used. This results in many index pointer segments whose delete byte is turned on, using up the space in the database. The Freespace Reclaim function issues DL/I GN calls to secondary index database to physically delete such index pointer segments.

The Freespace Reclaim function is more effective if used in combination with the CA Reclaim function of z/OS. If CA Reclaim is enabled, after the Freespace Reclaim function deletes all index pointer segments in a VSAM Control Area (CA), CA Reclaim reclaims the CA for storing new index pointer segments. For more information about z/OS CA Reclaim, see *z/OS DFSMS Using Data Sets*.

The Freespace Reclaim function is not effective for secondary indexes that use a SHISAM database or a HISAM database with overflow data set due to the following reasons:

- In secondary indexes that use SHISAM databases, index pointer segments do not have the delete byte flag. Index pointer segments are physically deleted when the corresponding source segments are removed from the DEDB.
- In secondary indexes that use HISAM databases with an overflow data set, index pointer segments are physically deleted when the corresponding source segments are removed from the DEDB. However, index pointer segments that have a pointer to its duplicate key in the overflow data set and have their delete byte flag turned on are not deleted by the Freespace Reclaim function because those index pointer segments must be kept in the database.

Input and output for the Freespace Reclaim function

The INDEXBLD command in the HFPSYSIN DD control statement invokes the Freespace Reclaim function.

The Build Index function, the Resync function, and the Freespace Reclaim function use the INDEXBLD command to start the Build Index process. During the Build Index process, FPA builds, synchronizes, or reclaims space in secondary index databases. To determine which task to perform, FPA evaluates the IDXPROC keyword on the INDEXBLD command as follows:

- When IDXPROC=BUILD, FPA starts the Build Index function. During the job, FPA scans one or more DEDB areas and builds one or more secondary index databases.
- When IDXPROC=RESYNC, FPA starts the Resync function. During the job, FPA scans one or more DEDB areas and secondary index databases and synchronizes secondary index databases.
- When IDXPROC=FS_RECLAIM, FPA starts the Freespace Reclaim function. During the job, FPA reclaims the CIs in secondary index database data sets.

The following figure shows the input and the output for the Freespace Reclaim function.

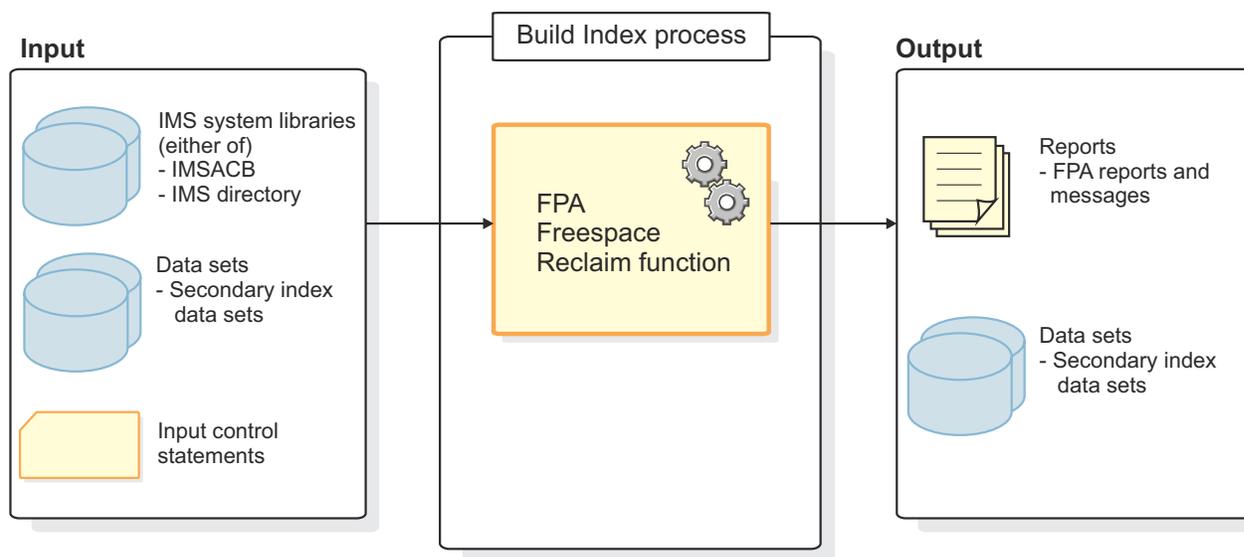


Figure 178. Input and output for the Freespace Reclaim function

Running the Freespace Reclaim function

FPA runs the Freespace Reclaim function as a standard z/OS batch job. You must specify an EXEC statement and DD statements that define the input and output data sets in your JCL.

Procedure

1. Optional: Estimate the region size that is required to run the job.
See [“Region size requirements for the Freespace Reclaim function”](#) on page 357.
2. Specify the EXEC statement in the following format:

```
//RECLAIM EXEC PGM=HFPMAIN0,REGION=rrrrM,
//          PARM='DBRCGRP=dbrcgrp,GSGNAME=gsgname,IMSPLEX=imsplex'
```

For the parameters, see the following table

Parameter	Description
DBRCGRP=	DBRCGRP is an optional 1- to 3-character parameter that specifies the identifier assigned to a group of DBRC instances (DBRC sharing group) that access the same RECON data set in the IMSplex.
GSGNAME=	GSGNAME is an optional 1- to 8-character parameter that specifies the global service group.
IMSPLEX=	IMSPLEX is an optional 1- to 5-character parameter that specifies the IMSplex group names to share either databases or message queues.

3. Specify the DD statements to define input data sets and output data sets.

For descriptions of the DD statements, see [“DD statements for the Freespace Reclaim function”](#) on page 358.

4. Specify the utility control statements.
 - a) Specify the INDEXBLD command in the HFPSYSIN DD.
 - b) Specify IDXPROC=FS_RECLAIM for the INDEXBLD command. If you do not specify FS_RECLAIM, the job runs the Build Index function.

See [“Utility control statements for the Freespace Reclaim function”](#) on page 363 and specify other command and keywords.

5. Run the JCL job.

Example

The following figure shows a JCL example for the INDEXBLD command for reclaiming CIs in secondary index database data sets.

```
//*****  
//** RECLAIMING CIS IN INDEX DB(S) **  
//*****  
//INDEXBLD EXEC PGM=HFPMAIN0  
//*  
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0  
// DD DISP=SHR,DSN=IMSVS.SDFSRESL  
//IMSACB DD DISP=SHR,DSN=IMSVS.FPSI.ACBLIB  
//HFPPRINT DD SYSOUT=A  
//HFPRPTS DD SYSOUT=A  
//*  
//HFPSYSIN DD *  
GLOBAL DBRC=NO  
INDEXBLD  
DBD=IVPDB3,  
IDXPROC=FS_RECLAIM  
/*
```

Figure 179. JCL example for the INDEXBLD command for reclaiming CIs in secondary index database data sets

Related concepts

[Region size requirements for the Freespace Reclaim function](#)

The Freespace Reclaim function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Related reference

[Examples for the Freespace Reclaim function](#)

Review examples and learn techniques for reclaiming CIs in secondary index database data sets.

[Tips and best practices for the FPA Freespace Reclaim function](#)

The FPA Freespace Reclaim function supports several keywords that improve the performance of FPA Freespace Reclaim jobs and help you use the function more effectively and efficiently.

Region size requirements for the Freespace Reclaim function

The Freespace Reclaim function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Region size for the Freespace Reclaim function depends on the number of input secondary indexes. Typically, database maintenance is a relatively important function. To increase the performance of the Freespace Reclaim program, it would be advisable to run it in a region that has a relatively high dispatching priority (that is, at least equal to the dispatching priority of a message processing region).

Estimating the region size for the Freespace Reclaim function

The region size required for the Freespace Reclaim function is a function of the number of secondary indexes whose CIs will be reclaimed.

In most cases, the specification on the EXEC statement for HFPMAIN0 of a region that is too small, is detected during the initialization processing of the main task. This causes processing to be terminated at that point. Under certain conditions, the region size shortfall does not become evident until the actual Build Index processing has begun. The job must be completely rerun. Thus, the region size calculations should always be performed prior to initiating Build Index processing.

The following worksheet is for estimating the region size that is required for a typical Freespace Reclaim job. The actual region size depends on the keywords that are specified for the job. Therefore, when you specify the region size in your JCL, use a higher value than the value calculated from this worksheet.

FPA programs, internal tables, control blocks	3,020,000
Number of secondary indexes processed by Freespace Reclaim * 1,400	-----
Maximum length of the defined index pointer segment among the secondary indexes to be processed	-----
Total bytes of Freespace Reclaim storage	-----

Figure 180. Worksheet for estimating region size for the Freespace Reclaim function

DD statements for the Freespace Reclaim function

DD statements of the FPA Freespace Reclaim function determine the input and output data sets and specify how to run the Freespace Reclaim function.

Subsections:

- [“List of DD statements” on page 358](#)
- [“Input DD statements” on page 359](#)
- [“Output DD statements” on page 361](#)
- [“Input/Output DD statements” on page 362](#)

List of DD statements

The following table summarizes the DD statements for the Freespace Reclaim function.

Table 67. DD statements for the Freespace Reclaim function

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
JOBLIB or STEPLIB	Input	PDS	No	Required
IMSACB, IMSACBA, or IMSACBB	Input	PDS	No	Required when IMSCATHLQ=*NO
IMSDALIB	Input	PDS	No	Optional
MODSTAT	Input	PS	Yes	Optional
MODSTAT2	Input	PS	Yes	Optional
OLCSTAT	Input	PS	Yes	Optional
<i>index</i>	Input and output	KSDS	Yes	Required when DFSMDA members for index databases are not used
<i>indexovrflw</i>	Input and output	ESDS	Yes	Required when DFSMDA members for index databases are not used

Table 67. DD statements for the Freespace Reclaim function (continued)

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
HFPSYSIN	Input	LRECL=80	No	Required
HFPCSITE	Input	PDS	Yes	Optional
HFPPRINT	Output	SYSOUT	Yes	Optional
HFPPTS	Output	LRECL=133	Yes	Optional
HFPABEND	Output	LRECL=125	Yes	Optional
SYSRINT	Output	LRECL=121	Yes	Optional
RECON n	Input and output	KSDS	Yes	Required when both of the following conditions are met: <ul style="list-style-type: none"> • DBRC=YES • DFSMDA members for RECONn data sets are not used

Input DD statements

The following input DD statements are used for the Freespace Reclaim function.

JOBLIB/STEPLIB DD

This DD statement is a required DD statement that defines the following input data sets:

- The FPA load module library (required).
- The library that contains DFSMDA dynamic allocation members for the RECON data set. When DBRC=YES is specified and the RECON n DD statement is omitted, this DD statement is required.
- The library that contains a default site default member (HFPCSITE) to determine the data set name and the member of the site default table.
- The IMS load module library (required).
- The IMS Tools Base library (SHKTLOAD), if you specify the ITKBSRVR keyword.
- The IMS Tools Base library (SGLXLOAD), if you specify the IMSCATHLQ keyword.

FPA must be run as an APF-authorized program. All the data sets that are listed within the JOBLIB or the STEPLIB concatenation must also be APF-authorized. If you use DFSMDA members and the DFSMDA members are in a non-APF authorized library, you can specify the DFSMDA library on the IMSDALIB DD statement.

IMSACB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. The IMSACB DD statements (IMSACB, IMSACBA, IMSACBB, MODSTAT, MODSTAT2, and OLCSTAT) can be used to identify the active ACB library. When both OLCSTAT or MODSTAT and MODSTAT2 are found, FPA checks the time stamp and determines the latest one.

The active IMSACB library is determined according to the following rules:

1. Check IMSACB DD statement in the JCL statement.
2. Check OLCSTAT DD statement in the JCL statement.
3. Check MODSTAT and MODSTAT2 DD statement in the JCL statement.
4. Allocate OLCSTAT dynamically and check the OLCSTAT member.
5. Allocate MODSTAT and MODSTAT2 dynamically and check the MODSTAT and MODSTAT2 members.

If you specify the IMSCATHLQ keyword, IMSACB DD statements are not used. The IMS directory is used instead of the ACB library.

IMSACBA DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSACBB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSDALIB DD

This DD statement is an optional DD statement that defines the IMSDALIB data set. This DD statement is useful when the data set for dynamic allocation members is not APF-authorized.

MODSTAT DD

This DD statement is an optional DD statement that defines the MODSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

MODSTAT2 DD

This DD statement is an optional DD statement that defines the MODSTAT2 data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT2 library by using a TYPE=DATABASE statement of the DFSMDA macro.

OLCSTAT DD

This DD statement is an optional DD statement that defines the OLCSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the OLCSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

HFPSYSIN DD

This DD statement is a required DD statement that defines the input control statement that specifies the FPA functions.

The HFPSYSIN DD statement can be coded as a standard SYSIN file, a sequential data set, or a PDS member. LRECL=80 is required for the DCB of this data set. Only one command, besides the GLOBAL command, can be specified in HFPSYSIN. FPA analyzes the command syntax in columns 1 - 72. Columns 73 - 80 are regarded as comment.

For details about coding the HFPSYSIN DD statement, see [“Utility control statements for the Freespace Reclaim function” on page 363.](#)

HFPCSITE DD

This DD statement is an optional DD statement. The HFPCSITE DD statement can be used to define the data set name of the site default table. When the SITE_DSNAME keyword is specified in the HFPSYSIN DD statement, the HFPCSITE DD statement is allocated dynamically. HFPCSITE DD in the JCL is mutually exclusive with the SITE_DSNAME keyword in the HFPSYSIN DD statement. The data set attribute of HFPCSITE DD is PDS.

Output DD statements

The following output DD statements are used for the Freespace Reclaim function.

HFPPRINT DD

This DD statement is an optional DD statement that defines the processing log output data set. The data set contains the processing messages that are issued by FPA.

The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set. You cannot specify DUMMY.

The HFPPRINT data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPPRINT DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

The following figure shows an example of the processing messages that are issued for the INDEXBLD command.

```
IMS HFPF UTILITIES                                "Processing messages"                                PAGE: 1
5698-FPP V2R1                                     2019-07-07 0:20:14
HFPC0001I COMMAND ANALYZE PROCESSING STARTED.
HFPC0005I DBRC=YES IS SPECIFIED.
- EEQE DETECTION IS PERFORMED.
HFPC0002I COMMAND ANALYZE PROCESSING COMPLETED.
HFPP0001I INDEXBLD PROCESSING STARTED FOR DATABASE: DEDBJN23
HFPB0003I INDEXBLD FS_RECLAIM PROCESSING STARTED FOR SECONDARY INDEX DBD: ITHISM02
HFPB0004I INDEXBLD FS_RECLAIM PROCESSING COMPLETED FOR SECONDARY INDEX DBD: ITHISM02 (ELAPSED TIME: 00:00:01.83)
HFPP0002I INDEXBLD PROCESSING COMPLETED FOR DATABASE: DEDBJN23
HFPL0001I REPORT STARTED FOR COMMAND: INDEXBLD, DATABASE: DEDBJN23
HFPL0002I REPORT COMPLETED FOR COMMAND: INDEXBLD, DATABASE: DEDBJN23
```

Figure 181. Processing messages issued for the INDEXBLD command

HFPFRPTS DD

This DD statement is an optional DD statement that defines the report output data set. The data set contains the reports that can be used to verify the process of the command. The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

The following reports are generated in this data set:

- Audit report
- Processing report
- DBD Definition report
- Secondary Index Definition report
- Secondary Index Freespace Reclaim report (when FS_RECLAIM_REPORT=YES is specified)
- Result of Access Method Services report (when FS_RECLAIM_REPORT=YES is specified)

The HFPFRPTS data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPRPTS DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

HFPABEND DD

This DD statement is an optional DD statement. It defines a snap dump output data set that is used for diagnostic purposes. This data set might be used when the FPA job ends abnormally. Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

SYSPRINT DD

This DD statement is an optional DD statement that defines the processing log output data set, which contains the processing messages that are issued by the IMS system.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

Input/Output DD statements

The following input/output DD statements are used for the Freespace Reclaim function.

***index* DD**

This DD statement is a required DD statement when DFSMDA members for index databases are not used. This DD statement defines the VSAM KSDS data set to be processed.

The DD name must be the DD name of the secondary index database that is defined in the input ACB library (IMSACB DD). Multiple *index* DD statements can be specified for each secondary index database that is processed by the Freespace Reclaim function.

The data set that is used by FPA is determined by the following elements:

- The name of the secondary index database member in the IMS DFSMDA library
- The *index* DD JCL statement

***indexovrflw* DD**

This DD statement is a required DD statement when DFSMDA members for index databases are not used. This DD statement defines the VSAM ESDS data set to be processed.

The DD name must be the DD name of the secondary index database that is defined in the input ACB library (IMSACB DD). Multiple *indexovrflw* DD statements can be specified for each secondary index database that is processed by the Freespace Reclaim function.

The data set that is used by FPA is determined by the following elements:

- The name of the secondary index database member in the IMS DFSMDA library
- The *indexovrflw* DD JCL statement

RECON1/RECON2/RECON3 DD

These statements define the DBRC RECON data sets. These DD statements are required DD statements when both of the following conditions are met:

- DBRC=YES is specified for the GLOBAL command
- DFSMDA members for RECON n data sets are not used

The RECON n data set must be the same data set that is used in the IMS control region. Because FPA issues a signon, a signoff, an authorization call, and an unauthorization call to DBRC, it updates RECON n data sets.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is necessary, the data set is allocated dynamically by using a DFSMDA member.

Utility control statements for the Freespace Reclaim function

The utility control statements for the Freespace Reclaim function define the user description of the GLOBAL command and the INDEXBLD command. The utility control statements control the Build Index process for the Freespace Reclaim function.

The utility control statements must be specified in the HFPSYSIN data set. This control statement data set usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte, fixed-length records. Block size, if coded, must be a multiple of 80.

The following figure shows a sample of how to code the HFPSYSIN data set.

```
//HFPSYSIN DD *
GLOBAL
  DBRC=YES,
  IMSID=IMS1
INDEXBLD
  DBD=dbdname,
  IDXPROC=FS_RECLAIM
/*
```

Figure 182. INDEXBLD HFPSYSIN data set

Subsections:

- [“GLOBAL command keywords” on page 363](#)
- [“INDEXBLD command keywords” on page 364](#)

GLOBAL command keywords

For the GLOBAL command, you can specify the keywords that are listed in the following table. For the format of the GLOBAL command, see [“GLOBAL command” on page 394](#).

Table 68. GLOBAL command keywords

Keyword	Required or optional	Description	Topic
DBRC	Optional	Specifies to request or bypass the database recovery control (DBRC) facility process for the database that is specified by the DBD.	“DBRC keyword” on page 451
IMSCATACB_INPUT	Optional	Specifies whether to retrieve the currently active ACB definition or the pending ACB definition from the IMS directory.	“IMSCATACB_INPUT keyword” on page 480
IMSCATHLQ	Optional	Specifies the high-level qualifier of the bootstrap data set of the IMS directory.	“IMSCATHLQ keyword” on page 481
IMSID	Required	Specifies the IMS system ID. The FPA process runs on the IMS system that is identified by this ID.	“IMSID keyword” on page 482
ITKBLOAD	Optional	Specifies the library name of IMS Tools KB.	“ITKBLOAD keyword” on page 487
ITKBSRVR	Optional	Specifies the name of the IMS Tools KB server XCF group.	“ITKBSRVR keyword” on page 488
MSTIMESTAMP	Optional	Specifies whether to include a time stamp in the messages.	“MSTIMESTAMP keyword” on page 497

Table 68. GLOBAL command keywords (continued)

Keyword	Required or optional	Description	Topic
SCAN	Optional	Specifies to verify the syntax of command, keywords, and parameters in command sets without running FPA. The Scan function also allocates some (but not all) internal resources for running the process function.	“SCAN keyword” on page 523
SITE_DSNAME	Optional	Specifies the data set name for the site default table.	“SITE_DSNAME keyword” on page 539
SITE_MEMBER	Optional	Specifies the member name for the site default table.	“SITE_MEMBER keyword” on page 540

INDEXBLD command keywords

The Build Index function, the Resync function, and the Freespace Reclaim function use the INDEXBLD command to start the Build Index process. During the Build Index process, FPA builds, synchronizes, or reclaims space in secondary index databases. To determine which task to perform, FPA evaluates the IDXPROC keyword on the INDEXBLD command as follows:

- When IDXPROC=BUILD, FPA starts the Build Index function. During the job, FPA scans one or more DEDB areas and builds one or more secondary index databases.
- When IDXPROC=RESYNC, FPA starts the Resync function. During the job, FPA scans one or more DEDB areas and secondary index databases and synchronizes secondary index databases.
- When IDXPROC=FS_RECLAIM, FPA starts the Freespace Reclaim function. During the job, FPA reclaims the CIs in secondary index database data sets.

For the INDEXBLD command, you can specify the keywords that are listed in the following table. For the format of the INDEXBLD command, see [“INDEXBLD command \(for the Freespace Reclaim function\)” on page 403](#).

Table 69. INDEXBLD command keywords

Keyword	Required or optional	Description	Topic
DBD	Required	Specifies the database description (DBD) of the primary DEDB.	“DBD keyword” on page 450
FS_RECLAIM_REPORT	Optional	Specifies whether to generate the Secondary Index Freespace Reclaim report.	“FS_RECLAIM_REPORT keyword” on page 468
IDXPROC	Required	Specifies the function that the Build Index process starts. You must specify IDXPROC=FS_RECLAIM to use the Freespace Reclaim function.	“IDXPROC keyword” on page 478
INDEXDBD	Optional	Specifies the database description (DBD) that contains information about the secondary index databases to be processed.	“INDEXDBD keyword” on page 482

Examples for the Freespace Reclaim function

Review examples and learn techniques for reclaiming CIs in secondary index database data sets.

In the examples, the following DEDB is assumed:

```

DEDB name:  DEDBJN22 has four areas
  area name:  DB22AR0 (area number is 1 and has one ADS)
    ADS DD name:  DB22AR01,      DS name:  HPFP.DB22AR0.ADS1
  area name:  DB22AR1 (area number is 2 and has one ADS)
    ADS DD name:  DB22AR11,     DS name:  HPFP.DB22AR1.ADS1
  area name:  DB22AR2 (area number is 3 and has one ADS)
    ADS DD name:  DB22AR21,     DS name:  HPFP.DB22AR2.ADS1
  area name:  DB22AR3 (area number is 4 and has one ADS)
    ADS DD name:  DB22AR31,     DS name:  HPFP.DB22AR3.ADS1

Secondary index database name: INDEX1
  Primary DBDS DD name: INDEXK1, DS name: HPFP.INDEXK1
Secondary index database name: INDEX2
  Primary DBDS DD name: INDEXK2, DS name: HPFP.INDEXK2
Secondary index database name: INDEX3
  Primary DBDS DD name: INDEXK3, DS name: HPFP.INDEXK3

```

In DBRC, all ADSs are in the AVAILABLE status.

Example 1: Reclaiming CIs in a secondary index database data set

The following figure shows example JCL for reclaiming CIs in a secondary index database data set.

In this example:

- FPA dynamically allocates the secondary index database data set to reclaim CIs. FPA obtains the allocation information (DD name and DS name of the secondary index database data set) from the DFSMDA member in the IMSDALIB DD data set.
- INDEXDBD keyword specifies the secondary index database data set.
- IDXPROC=FS_RECLAIM starts the Freespace Reclaim function.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//        DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES,
  IMSID=IMS1
INDEXBLD
  DBD=DEDBJN22,
  INDEXDBD=INDEX1,
  IDXPROC=FS_RECLAIM
/*

```

Figure 183. Reclaiming CIs in a secondary index database data set

Example 2: Reclaiming CIs in all secondary index database data sets of a DEDB and generating reports

The following figure shows example JCL for reclaiming CIs in all secondary index database data sets of a DEDB and generating the Secondary Index Freespace Reclaim report and the Result of Access Method Services report.

In this example:

- FPA dynamically allocates the secondary index database data sets to reclaim CIs. FPA obtains the allocation information (DD name and DS name of the secondary index database data sets) from the DFSMDA members in the IMSDALIB DD data set.
- INDEXDBD=ALL specifies to process all secondary index database data sets of the DEDB that the DBD keyword specifies.
- IDXPROC=FS_RECLAIM starts the Freespace Reclaim function.
- FS_RECLAIM_REPORT=YES specifies to print the Secondary Index Freespace Reclaim report and the Result of Access Method Services report.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES,
  IMSID=IMS1
INDEXBLD
  DBD=DEDBJN22,
  INDEXDBD=ALL,
  IDXPROC=FS_RECLAIM,
  FS_RECLAIM_REPORT=YES
/*

```

Figure 184. Reclaiming CIs in all secondary index database data sets of a DEDB and generating reports

Chapter 15. Initializing a DEDB offline with FPA

You can initialize a DEDB by using the FPA Initialize function.

Topics:

- [“Functions of Initialize” on page 367](#)
- [“Input and output for the Initialize function” on page 367](#)
- [“Running the Initialize function” on page 367](#)
- [“Region size requirements for the Initialize function” on page 369](#)
- [“DD statements for the Initialize function” on page 369](#)
- [“Utility control statements for the Initialize function” on page 375](#)
- [“Examples for the Initialize function” on page 377](#)

Functions of Initialize

The Initialize function can be used to initialize one or more area data sets of a DEDB database. The data sets are initialized and the DEDB areas are formatted with the DBDGEN specifications.

Input and output for the Initialize function

The Initialize function is invoked by the INITIALIZE command in the HFPSYSIN DD control statement to initialize multiple DEDB areas.

The following figure shows the input and the output for the Initialize function.

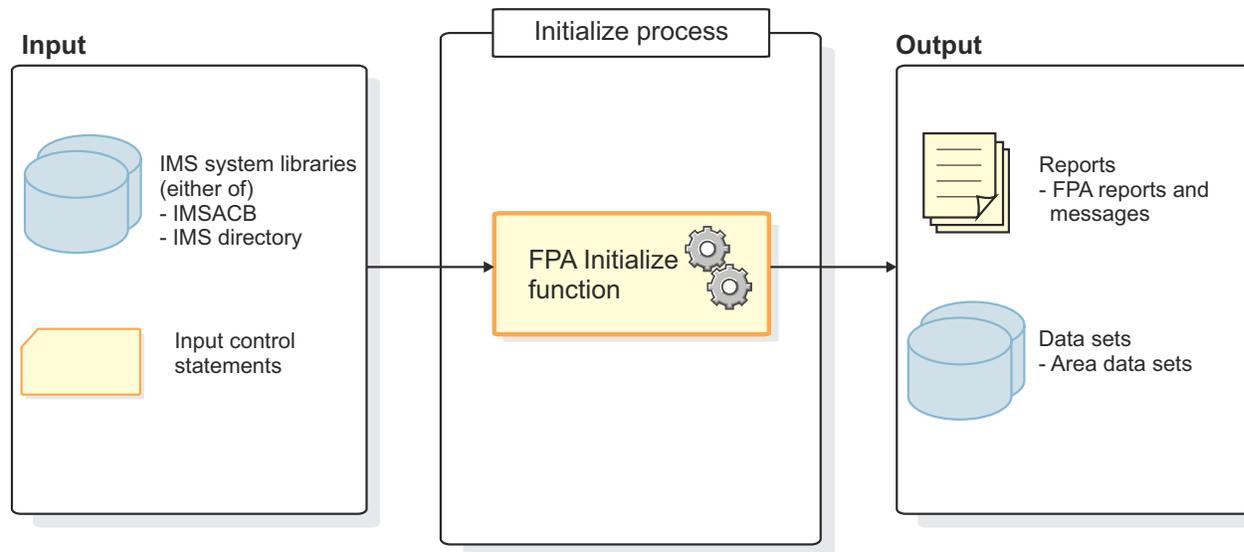


Figure 185. Input and output for the Initialize function

Running the Initialize function

FPA runs the Initialize function as a standard z/OS batch job. You must specify an EXEC statement and DD statements that define the input and output data sets in your JCL.

Procedure

1. Optional: Estimate the region size that is required to run the job.

See “[Region size requirements for the Initialize function](#)” on page 369.

2. Specify the EXEC statement in the following format:

```
//INIT      EXEC PGM=HFPMAIN0,REGION=TTTTM,  
//          PARM='DBRCGRP=dbrcgrp,GSGNAME=gsgname,IMSPLEX=imsplex'
```

For the parameters, see the following table.

Parameter	Description
DBRCGRP=	DBRCGRP is an optional 1- to 3-character parameter that specifies the identifier assigned to a group of DBRC instances (DBRC sharing group) that access the same RECON data set in the IMSplex.
GSGNAME=	GSGNAME is an optional 1- to 8-character parameter that specifies the global service group.
IMSPLEX=	IMSPLEX is an optional 1- to 5-character parameter that specifies the IMSplex group names to share either databases or message queues.

3. Specify the DD statements to define input data sets and output data sets.

4. Run the JCL job.

Example

The following figure shows a JCL example for the INITIALIZE command.

```
/******  
/** INITIALIZE THE AREA(S) **  
/******  
//INIT      EXEC PGM=HFPMAIN0  
//*  
//STEPLIB  DD DISP=SHR,DSN=HPFP.SHFPLMD0  
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL  
//          DD DISP=SHR,DSN=IMSVS.PGMLIB  
//IMSACB   DD DISP=SHR,DSN=IMSVS.ACBLIB  
//IMSRESLB DD DISP=SHR,DSN=IMSVS.SDFSRESL  
//HFPPRINT DD SYSOUT=A  
//HFPRPTS  DD SYSOUT=A  
//HFPSYSIN DD *  
          GLOBAL DBRC=NO  
          INITIALIZE DBD=IVPDB3  
//*  
//DFSIVD3A DD DISP=OLD,DSN=IMSVS.DFSIVD31.NEW  
//DFSIVD3B DD DISP=OLD,DSN=IMSVS.DFSIVD33.NEW
```

Figure 186. JCL example for the INITIALIZE command

Related concepts

[Region size requirements for the Initialize function](#)

The Initialize function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Related reference

[Examples for the Initialize function](#)

Review examples and learn techniques for initializing DEDB areas.

[Tips and best practices for the FPA Initialize function](#)

The FPA Initialize function supports several keywords that improve the performance of FPA Initialize jobs and that further simplify management of DEDBs.

Region size requirements for the Initialize function

The Initialize function of FPA operates in a virtual storage region as z/OS batch job steps. HFPMAIN0 runs in 31-bit addressing mode.

Region size for the Initialize function depends on the size of a UOW, the number of files being initialized, the number of output areas, and the OTASKCTL specification. Typically, database maintenance is a relatively important function. To increase the performance of the initialize program, it would be advisable to run it in a region that has a relatively high dispatching priority (that is, at least equal to the dispatching priority of a message processing region).

Estimating the region size for the Initialize function

The region size required for the Initialize function is a function of the size of a UOW, the number of areas being initialized, the number of output files, and the OTASKCTL specification.

In most cases, the specification on the EXEC statement for HFPMAIN0 of a region that is too small, is detected during the initialization processing of the main task. This causes processing to be terminated at that point. Under certain conditions, the region size shortfall does not become evident until the actual initialization has begun. The job must be completely rerun. Thus, the region size calculations must always be performed prior to initiating initialize processing.

The following worksheet is for estimating the region size that is required for a typical initialize job. The actual region size depends on the keywords that are specified for the job. Therefore, when you specify the region size in your JCL, use a higher value than the value calculated from this worksheet.

```

FPA programs, internal tables, control blocks 3,020,000
Number of areas defined in DBD * 1,300          -----
    Internal tables, control blocks           18,200
    Maximum CISZ * 6                          -----
                                     subtotal -----
    Value specified on OTASKCTL * subtotal -----
    Internal tables, control blocks           3,700
                                     subtotal -----
    Number of initialized areas * subtotal -----
Total of CISZ * UOW1 of each area * 2         -----
Total bytes of initialize storage              -----
  
```

Figure 187. Worksheet for estimating region size for the Initialize function

DD statements for the Initialize function

DD statements of the FPA Initialize function determine the input and output data sets and specify how to run the Initialize function.

Subsections:

- [“List of DD statements” on page 370](#)
- [“Input DD statements” on page 370](#)
- [“Output DD statements” on page 372](#)
- [“Input/Output DD statements” on page 375](#)

List of DD statements

The following table summarizes the DD statements for the Initialize function.

Table 70. DD statements for the Initialize function

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
JOBLIB or STEPLIB	Input	PDS	No	Required
IMSACB, IMSACBA, or IMSACBB	Input	PDS	No	Required when IMSCATHLQ=*NO
IMSDALIB	Input	PDS	No	Optional
MODSTAT	Input	PS	Yes	Optional
MODSTAT2	Input	PS	Yes	Optional
OLCSTAT	Input	PS	Yes	Optional
IMSRESLB	Input	PDS	No	Optional
<i>addnname</i> or <i>areaname</i>	Output	ESDS	Yes	Required under certain conditions (for details, see <i>addnname/areaname</i> DD)
HFPSYSIN	Input	LRECL=80	No	Required
HFPCSITE	Input	PDS	Yes	Optional
HFPPRINT	Output	SYSOUT	Yes	Optional
HFPRPTS	Output	LRECL=133	Yes	Optional
HFPABEND	Output	LRECL=125	Yes	Optional
RECON n	Input and output	KSDS	Yes	Required when both of the following conditions are met: <ul style="list-style-type: none"> • DBRC=YES • DFSMDA members for RECONn data sets are not used

Input DD statements

The following input DD statements are used for the Initialize function.

JOBLIB/STEPLIB DD

This DD statement is a required DD statement that defines the following input data sets:

- The FPA load module library (required).
- The library that contains DFSMDA dynamic allocation members for the RECON data set. When DBRC=YES is specified and the RECON n DD statement is omitted, this DD statement is required.
- The library that contains randomizer/edit compression members that were specified in the IMSACB data set, if IMSRESLB DD statement is not specified.
- The library that contains a default site default member (HFPCSITE) to determine the data set name and the member of the site default table.
- The IMS load module library (required).
- The IMS Tools Base library (SHKTLOAD), if you specify the ITKBSRVR keyword.
- The IMS Tools Base library (SGLXLOAD), if you specify the IMSCATHLQ keyword.

FPA must be run as an APF-authorized program. All of the data sets that are listed within the JOBLIB or the STEPLIB concatenation must also be APF-authorized. If you use DFSMDA members and the DFSMDA members are in a non-APF authorized library, you can specify the DFSMDA library on the IMSDALIB DD statement.

IMSACB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. The IMSACB DD statements (IMSACB, IMSACBA, IMSACBB, MODSTAT, MODSTAT2, and OLCSTAT) can be used to identify the active ACB library. When both OLCSTAT or MODSTAT and MODSTAT2 are found, FPA checks the time stamp and determines the latest one.

The active IMSACB library is determined according to the following rules:

1. Check IMSACB DD statement in the JCL statement.
2. Check OLCSTAT DD statement in the JCL statement.
3. Check MODSTAT and MODSTAT2 DD statement in the JCL statement.
4. Allocate OLCSTAT dynamically and check the OLCSTAT member.
5. Allocate MODSTAT and MODSTAT2 dynamically and check the MODSTAT and MODSTAT2 members.

If you specify the IMSCATHLQ keyword, IMSACB DD statements are not used. The IMS directory is used instead of the ACB library.

IMSACBA DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSACBB DD

This DD statement is an optional DD statement that defines the library that contains the DMB for the database. This statement must be provided if MODSTAT and MODSTAT2, or the OLCSTAT DD statement is specified.

IMSDALIB DD

This DD statement is an optional DD statement that defines the IMSDALIB data set. This DD statement is useful when the data set for dynamic allocation members is not APF-authorized.

MODSTAT DD

This DD statement is an optional DD statement that defines the MODSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

MODSTAT2 DD

This DD statement is an optional DD statement that defines the MODSTAT2 data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the MODSTAT2 library by using a TYPE=DATABASE statement of the DFSMDA macro.

OLCSTAT DD

This DD statement is an optional DD statement that defines the OLCSTAT data set. When this statement is specified, the IMSACBA and IMSACBB DD statements must be specified instead of the ACBLIB DD statement.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is required, the data set is allocated dynamically by using a DFSMDA member. You can build the DFSMDA member for the OLCSTAT library by using a TYPE=DATABASE statement of the DFSMDA macro.

IMSRESLB DD

This DD statement is an optional DD statement that defines the libraries that contain the randomizer routines and compression routines that are specified in the ACB member in the IMSACB DD statements. If the IMSRESLB DD statement is specified in the JCL statement, only the randomizer and compression routines are loaded directly from the IMSRESLB DD statement. If the IMSRESLB DD statement is not specified in the JCL statement, they are loaded from the JOBLIB or the STEPLIB DD statement.

The IMS version of IMSRESLB DD statement must be the same as those of IMSACB DD.

All the data sets that are listed within the IMSRESLB concatenation must be APF-authorized.

HFPSYSIN DD

This DD statement is a required DD statement that defines the input control statement that specifies the FPA functions.

The HFPSYSIN DD statement can be coded as a standard SYSIN file, a sequential data set, or a PDS member. LRECL=80 is required for the DCB of this data set. Only one command, besides the GLOBAL command, can be specified in HFPSYSIN. FPA analyzes the command syntax in columns 1 - 72. Columns 73 - 80 are regarded as comment.

For details about coding the HFPSYSIN DD statement, see [“Utility control statements for the Initialize function” on page 375.](#)

HFPCSITE DD

This DD statement is an optional DD statement. The HFPCSITE DD statement can be used to define the data set name of the site default table. When the SITE_DSNAME keyword is specified in the HFPSYSIN DD statement, the HFPCSITE DD statement is allocated dynamically. HFPCSITE DD in the JCL is mutually exclusive with the SITE_DSNAME keyword in the HFPSYSIN DD statement. The data set attribute of HFPCSITE DD is PDS.

Output DD statements

The following output DD statements are used for the Initialize function.

***addnname/areaname* DD**

This DD statement is a required DD statement. This statement defines an output area data set, which is the VSAM ESDS for an area to be initialized. The ddname used must be the ddname for the area that is defined in the output ACB library (IMSACB DD).

This data set can be allocated dynamically by one of following methods:

- The ODSNMASK keyword parameter of the INITIALIZE command
- When the GLOBAL DBRC=YES option is specified, the ADS information in the DBRC RECON data sets
- The area name member in the IMS DFSMDA library

Or it can be specified with one of the following DD statements explicitly:

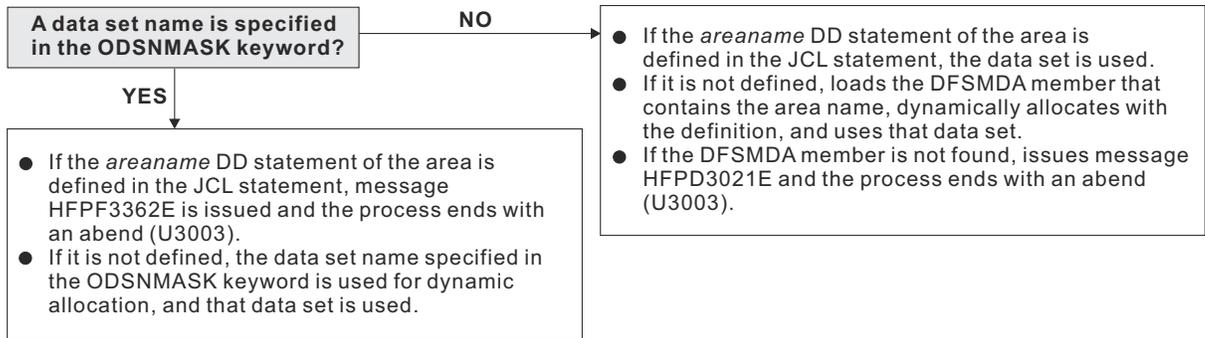
- *//addnname* DD JCL statement
- *//areaname* DD JCL statement

The area data set can be also defined by the ALLOCATE subcommand with the INITIALIZE command.

For the INITIALIZE command output, if the area is registered with DBRC, it must be in the RECOV NEEDED status; that is, all ADSs of the area must be in the UNAVAILABLE status in DBRC.

The following figure shows how FPA determines area data set names.

When DBRC=NO:



When DBRC=YES:

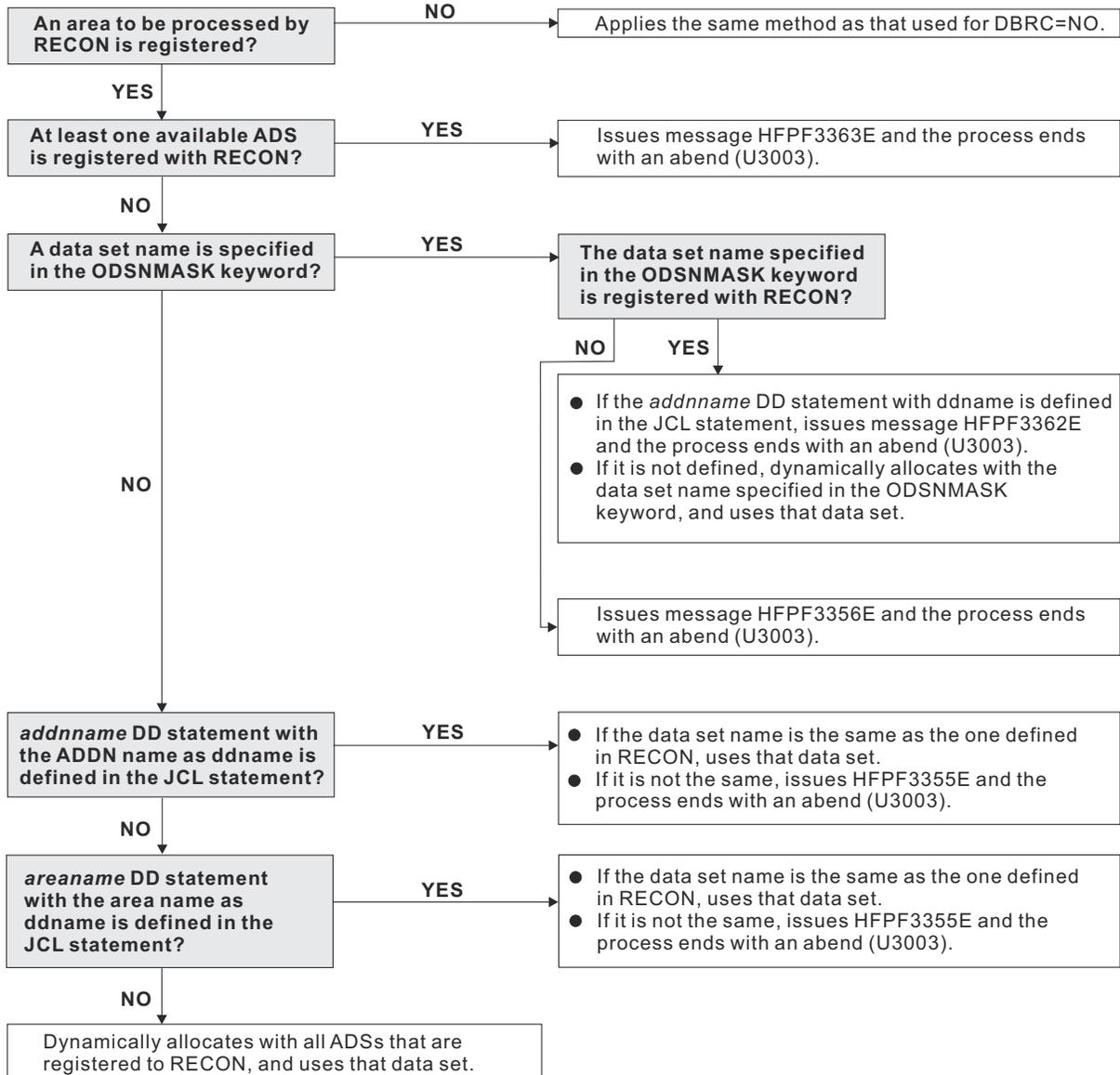


Figure 188. Output data set names for the INITIALIZE command

HFPPRINT DD

This DD statement is an optional DD statement that defines the processing log output data set. The data set contains the processing messages that are issued by FPA.

The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set. You cannot specify DUMMY.

The HFPPRINT data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPPRINT DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*.

The following figure shows an example of the processing messages that are issued for the INITIALIZE command.

```
IMS HPFP UTILITIES                                "Processing messages"                                PAGE: 1
5698-FPP V2R1                                     2019-11-22 18:54:26
HFPC0001I COMMAND ANALYZE PROCESSING STARTED.
HFPC0005I DBRC=NO IS SPECIFIED.
- EEQE DETECTION IS NOT PERFORMED.
HFPC0002I COMMAND ANALYZE PROCESSING COMPLETED.
HFPF0001I INITIALIZE PROCESSING STARTED FOR DATABASE: DEDBJN21
HFPK0001I INITIALIZE PROCESSING STARTED FOR AREA NO: 1, AREANAME: DB21AR0
HFPK0001I INITIALIZE PROCESSING STARTED FOR AREA NO: 2, AREANAME: DB21AR1
HFPK0002I INITIALIZE PROCESSING COMPLETED FOR AREA NO: 1, AREANAME: DB21AR0
HFPK0002I INITIALIZE PROCESSING COMPLETED FOR AREA NO: 2, AREANAME: DB21AR1
HFPF0002I INITIALIZE PROCESSING COMPLETED FOR DATABASE: DEDBJN21
HFPL0001I REPORT STARTED FOR COMMAND: INITIALIZE, DATABASE: DEDBJN21
HFPL0002I REPORT COMPLETED FOR COMMAND: INITIALIZE, DATABASE: DEDBJN21
```

Figure 189. Processing messages issued for the INITIALIZE command

HFPRPTS DD

This DD statement is an optional DD statement that defines the report output data set. The data set contains the reports that can be used to verify the process of the command. The following DCB attributes of the file are set by the FPA main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

The following reports are generated in this data set:

- Audit report
- Processing report
- DBD Definition report
- Initialization Processing Summary report

The HFPRPTS data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPRPTS DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*.

HFABEND DD

This DD statement is an optional DD statement. It defines a snap dump output data set that is used for diagnostic purposes. This data set might be used when the FPA job ends abnormally. Do not code

the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*

Input/Output DD statements

The following input/output DD statements are used for the Initialize function.

RECON1/RECON2/RECON3 DD

These statements define the DBRC RECON data sets. These DD statements are required DD statements when both of the following conditions are met:

- DBRC=YES is specified for the GLOBAL command
- DFSMDA members for RECON n data sets are not used

The RECON n data set must be the same data set that is used in the IMS control region. Because FPA issues a signon, a signoff, an authorization call, and an unauthorization call to DBRC, it updates RECON n data sets.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when the data set is necessary, the data set is allocated dynamically by using a DFSMDA member.

Utility control statements for the Initialize function

The utility control statements for the Initialize function define the user description of the GLOBAL command, the INITIALIZE command, and the optional ALLOCATE subcommands, all of which control the Initialize process.

The utility control statements must be specified in the HFPSYSIN data set. This control statement data set usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte, fixed-length records. Block size, if coded, must be a multiple of 80.

The following figure shows a sample of how to code the HFPSYSIN data set.

```
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
  INITIALIZE
  DBD=dbdname
/*
```

Figure 190. INITIALIZE HFPSYSIN data set

Subsections:

- [“GLOBAL command keywords” on page 375](#)
- [“INITIALIZE command keywords” on page 376](#)
- [“ALLOCATE subcommand keywords” on page 376](#)

GLOBAL command keywords

For the GLOBAL command, you can specify the keywords that are listed in the following table. For the format of the GLOBAL command, see [“GLOBAL command” on page 394](#).

Table 71. GLOBAL command keywords

Keyword	Required or optional	Description	Topic
DBRC	Optional	Specifies to request or bypass the database recovery control (DBRC) facility process for the database that is specified by the DBD.	“DBRC keyword” on page 451
IMSCATACB_INPUT	Optional	Specifies whether to retrieve the currently active ACB definition or the pending ACB definition from the IMS directory.	“IMSCATACB_INPUT keyword” on page 480
IMSCATHLQ	Optional	Specifies the high-level qualifier of the bootstrap data set of the IMS directory.	“IMSCATHLQ keyword” on page 481
ITKBLOAD	Optional	Specifies the library name of IMS Tools KB.	“ITKBLOAD keyword” on page 487
ITKBSRVR	Optional	Specifies the name of the IMS Tools KB server XCF group.	“ITKBSRVR keyword” on page 488
MSTIMESTAMP	Optional	Specifies whether to include a time stamp in the messages.	“MSTIMESTAMP keyword” on page 497
SCAN	Optional	Specifies to verify the syntax of command, keywords, and parameters in command sets without running FPA. The Scan function also allocates some (but not all) internal resources for running the process function.	“SCAN keyword” on page 523
SITE_DSNAME	Optional	Specifies the data set name for the site default table.	“SITE_DSNAME keyword” on page 539
SITE_MEMBER	Optional	Specifies the member name for the site default table.	“SITE_MEMBER keyword” on page 540

INITIALIZE command keywords

For the INITIALIZE command, you can specify the keywords that are listed in the following table. For the format of the INITIALIZE command, see [“INITIALIZE command” on page 404](#).

Table 72. INITIALIZE command keywords

Keyword	Required or optional	Description	Topic
ALLOCONLY	Optional	Suppresses the initialization of output area data sets.	“ALLOCONLY keyword” on page 440
DBD	Required	Specifies the database description (DBD) that contains the areas to be processed.	“DBD keyword” on page 450
OAREA	Optional	Specifies one or more output areas to process.	“OAREA keyword” on page 497
OCHECK	Optional	Controls the concurrent number of output devices to access.	“OCHECK keyword” on page 500
ODSNMASK	Optional	Specifies the data set name or mask for dynamically allocating cataloged pre-existing output data sets.	“ODSNMASK keyword” on page 500
OTASKCTL	Optional	Specifies the maximum number of output areas to be processed concurrently.	“OTASKCTL keyword” on page 503

ALLOCATE subcommand keywords

The ALLOCATE subcommand dynamically defines a VSAM cluster that is used for the processing of the INITIALIZE command.

For the ALLOCATE subcommand, you can specify the keywords that are listed in the following table. For the format of the ALLOCATE subcommand, see “ALLOCATE subcommand” on page 419.

Table 73. ALLOCATE subcommand keywords

Keyword	Required or optional	Description	Topic
AMSOPT	Optional	Passes optional parameters to the Access Method Services (AMS) to define VSAM clusters.	“AMSOPT keyword” on page 440
ATTRIBUTE	Optional	Requests optional processing related to allocation of a VSAM cluster.	“ATTRIBUTE keyword” on page 444
AVGREC	Optional	Identifies the space requirements of the output data set that is created by dynamic allocation.	“AVGREC keyword” on page 445
DATACLAS	Optional	Specifies the SMS data class of an output data set that is created by dynamic allocation.	“DATACLAS keyword” on page 449
KEYLABEL	Optional	Specifies the key label for encrypting a new VSAM cluster.	“KEYLABEL keyword” on page 489
MGMTCLAS	Optional	Specifies the SMS management class of an output data set that is created by dynamic allocation.	“MGMTCLAS keyword” on page 496
OAREA	Required	Specifies one or more output areas for the process.	“OAREA keyword” on page 497
SPACE	Optional	Specifies the space requirements of an output data set that is created by dynamic allocation.	“SPACE keyword” on page 543
STORCLAS	Optional	Specifies the SMS storage class of an output data set that is created by dynamic allocation.	“STORCLAS keyword” on page 546
VOLSER	Optional	Specifies one or more volume-serial identifiers for the data set that is accessed by dynamic allocation. If this keyword is omitted for an output data set, nonspecific (private) volumes are required.	“VOLSER keyword” on page 556

Examples for the Initialize function

Review examples and learn techniques for initializing DEDB areas.

In the examples, the following DEDB is assumed:

```

DEDB name:  DEDBJN22 has four areas
  area name:  DB22AR0 (area number is 1 and has three ADSs)
    ADS DD name: DB22AR01, DS name: HPFP.DB22AR0.ADS1
    ADS DD name: DB22AR02, DS name: HPFP.DB22AR0.ADS2
    ADS DD name: DB22AR03, DS name: HPFP.DB22AR0.ADS3
  area name:  DB22AR1 (area number is 2 and has two ADSs)
    ADS DD name: DB22AR11, DS name: HPFP.DB22AR1.ADS1
    ADS DD name: DB22AR12, DS name: HPFP.DB22AR1.ADS2
  area name:  DB22AR2 (area number is 3 and has two ADSs)
    ADS DD name: DB22AR21, DS name: HPFP.DB22AR2.ADS1
    ADS DD name: DB22AR22, DS name: HPFP.DB22AR2.ADS2
  area name:  DB22AR3 (area number is 4 and has one ADS)
    ADS DD name: DB22AR31, DS name: HPFP.DB22AR3.ADS1

```

In DBRC, all ADSs are in the AVAILABLE status.

Example 1: Initializing an area that is registered with DBRC

The following figure shows example JCL for initializing an area that is registered with DBRC.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The output ADSs to be initialized are dynamically allocated. The allocation information (DD statement name and DS name of the ADS) is obtained from DBRC. The output ADSs must be defined before running this example.

- When the initialize process ends, the RECOV NEEDED status of areas DB22AR0, DB22AR1, DB22AR2, and DB22AR3 are changed to OFF, and all ADSs of these areas are made AVAILABLE with DBRC.

```
//HFP EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
DBRC=YES
INITIALIZE
DBD=DEDBJN22
/*
```

Figure 191. Initializing an area that is registered with DBRC

Example 2: Initializing multiple areas in SMS environment

The following figure shows example JCL for initializing multiple areas in an SMS environment.

In this example:

- The DBRC=IGNORE option specifies not to check the area status in DBRC, so RECON data sets are not needed. Even if DBRC=FORCE is set for IMS, DBRC is not activated when the DBRC=IGNORE option is specified.
- The output ADSs to be initialized for specified areas are dynamically allocated by using the ODSNMASK parameter and the ALLOCATE subcommand.
- The output ADSs to be reloaded for specified areas are encrypted with a key label associated with the area data sets. The key label is specified on the KEYLABEL keyword of the ALLOCATE subcommand.
- The following ADSs are allocated:

Area	Allocated ADS
DB22AR0	HPFP.DB22AR0.ADS1
DB22AR1	HPFP.DB22AR1.ADS1
DB22AR2	HPFP.DB22AR2.ADS1

```
//HFP EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//HFPRPTS DD SYSOUT=*
//HFPPRINT DD SYSOUT=*
//HFPSYSIN DD *
GLOBAL
DBRC=IGNORE
INITIALIZE
DBD=DEDBJN22,
OAREA=(DB22AR0,DB22AR1,DB22AR2),
ODSNMASK='HPFP.&AREA.ADS1',
OTASKCTL=2
ALLOCATE
OAREA=(DB22AR0,DB22AR1,DB22AR2),
SPACE=(CYL,100),
STORCLAS=STORCLAS,
DATACLAS=ataclas,
KEYLABEL=keylabel,
ATTRIBUTE=(DELETE)
/*
```

Figure 192. Initializing multiple areas in an SMS environment

Chapter 16. Command reference for FPA

These topics provide reference information for the commands and keywords of FPA, and describe them in detail.

Topics:

- [“Command set” on page 381](#)
- [“FPA commands” on page 382](#)
- [“FPA subcommands” on page 419](#)
- [“FPA keywords” on page 438](#)

Command set

The main program of FPA is HFPMAIN0. This program receives a *command set* that determines the function to be processed.

A command set includes a command, subcommands, and keywords that are specified by the HFPSYSIN DD statement.

Commands

The HFPSYSIN DD statement must be followed by a valid FPA command. You define a command set by specifying a command and the following:

- Command keywords and parameters
- Any subcommands
- Subcommand keywords and parameters

A command name can be abbreviated if the first three characters are unique to the command. For example, the UNLOAD command can be coded as UNL because these characters are unique to the UNLOAD command.

Subcommands

You can use subcommands with certain commands to modify or complement the command. You define a subcommand set by specifying a subcommand, its keywords, and keyword parameters.

A subcommand name can be abbreviated if the first three characters are unique to the subcommand. For example, the FILECTL subcommand can be coded as FIL because these characters are unique to the FILECTL subcommand.

Keywords

Keywords follow a command or a subcommand and invoke the specified or the default parameters. Keywords are not positional and so can be specified in any order. Keywords have their parameters, which are separated by an equal sign, for example *keyword=parameter*, or by using parentheses, *keyword(parameter)*. If more than one parameter is specified, the list of parameters must be enclosed within parentheses, for example *keyword=(parameter1, parameter2, parameter3)*; the equal sign (=) is optional, but the keywords must be separated by commas.

Some keywords can be abbreviated if the first three or more characters are unique to the keyword for each command or a subcommand. For example, subcommand FILECTL has two keywords: VOLSER and VOLCNT. These keywords can be abbreviated as VOLS and VOLC because the first four characters are unique to the keywords. Similarly, the keyword UNIT can be abbreviated as UNI because the three characters are unique to the keyword. See the topic for each keyword to determine whether a keyword can be abbreviated.

Parameters

Some keywords require only one parameter and some require more. Some keywords can take a list of parameters. Parameters must be numeric, character, character string, or hexadecimal values. Lists of

parameters must be contained within parentheses. The equal sign (=) is optional, but the keywords must be separated by commas.

Comments

Comments in the HFPSYSIN command set are marked by an asterisk (*) in column 1. If two consecutive slash characters (//) appear anywhere in a command line, the remainder of that command line is treated as a comment. Columns 73 - 80 are also comments.

Separator characters

Commas are used to separate parameters. One or more commas can appear between keywords.

Parentheses are used to group listed parameters that follow a keyword. Commas are used to separate listed parameter entries within parentheses, for example, as in IAREA=(a,b,c,d,...).

Continuation characters

There are no continuation characters. All commands, subcommands, keywords can be continued on the next line.

FPA commands

The following topics explain the FPA commands.

ANALYZE command

Use the ANALYZE command to analyze one or more areas of a DEDB that uses the area data set.

ANALYZE is an optional command for the HFPSYSIN DD statement.

Related keywords:

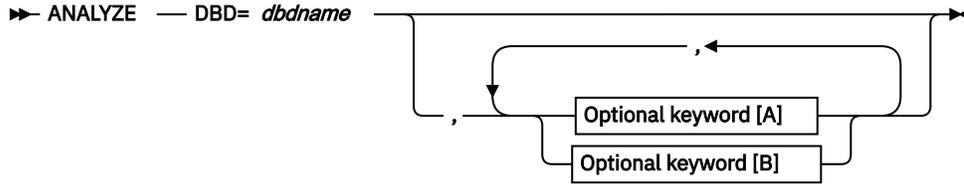
- [“ATASKCTL keyword” on page 443](#)
- [“CISNAP keyword” on page 448](#)
- [“DBD keyword” on page 450 \(required keyword\)](#)
- [“DOVNULLPTR_MSGLVL keyword” on page 455](#)
- [“HISTORYDD keyword” on page 469](#)
- [“IAREA keyword” on page 469](#)
- [“ICACHE keyword” on page 472](#)
- [“ICHECK keyword” on page 473](#)
- [“ICINCOMP keyword” on page 474](#)
- [“ICNOTIFY keyword” on page 475](#)
- [“ICOUT keyword” on page 476](#)
- [“IDSNMASK keyword” on page 476](#)
- [“IDX DUMP keyword” on page 477](#)
- [“IDX MAX ERROR keyword” on page 478](#)
- [“IDXTASKCTL keyword” on page 479](#)
- [“INDEXDBD keyword” on page 482](#)
- [“KEYSEQCHK keyword” on page 490](#)
- [“MAXERROR keyword” on page 495](#)
- [“PTRCHKLVL keyword” on page 506](#)
- [“SDEPCHKLVL keyword” on page 526](#)
- [“SDEP_ORPHAN_MSGLVL keyword” on page 527](#)
- [“SDEP_PTRIOVF_MSGLVL keyword” on page 528](#)
- [“SENSOR keyword” on page 535](#)
- [“SENSOR_DBREC keyword” on page 536](#)

- [“SENSOR_PTRERR keyword” on page 537](#)
- [“SENSOR_UOW keyword” on page 537](#)
- [“SENSOR_UOW_REPORT keyword” on page 538](#)
- [“SORTOPT_FILSZ keyword” on page 542](#)

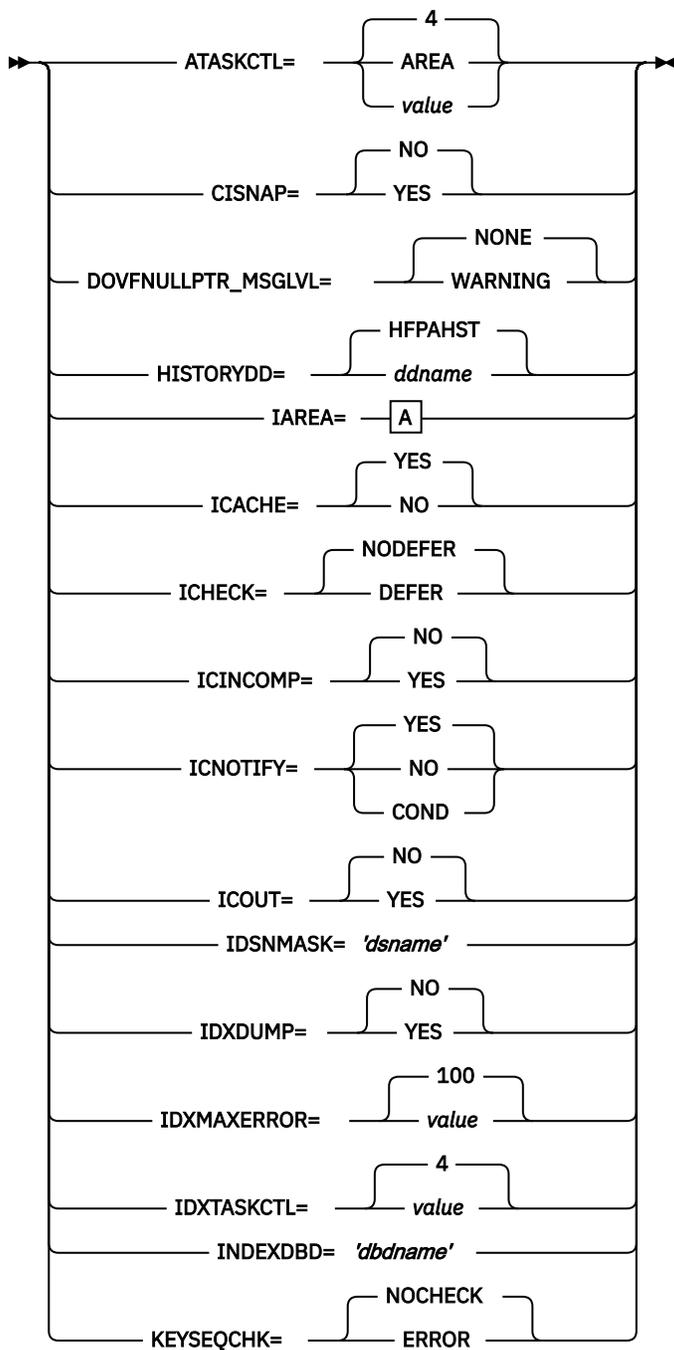
Format:

```
ANALYZE [keyword=parameter[,...]] [subcommand [keyword=parameter[,...]]]
```

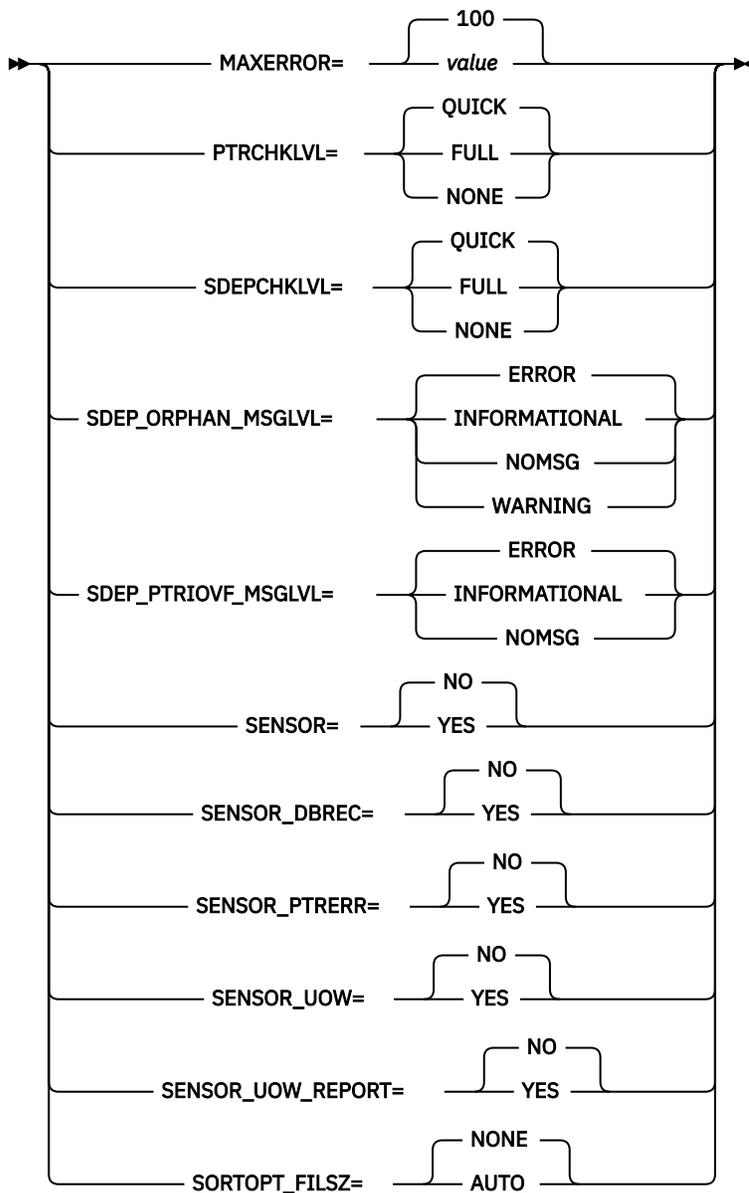
Syntax:



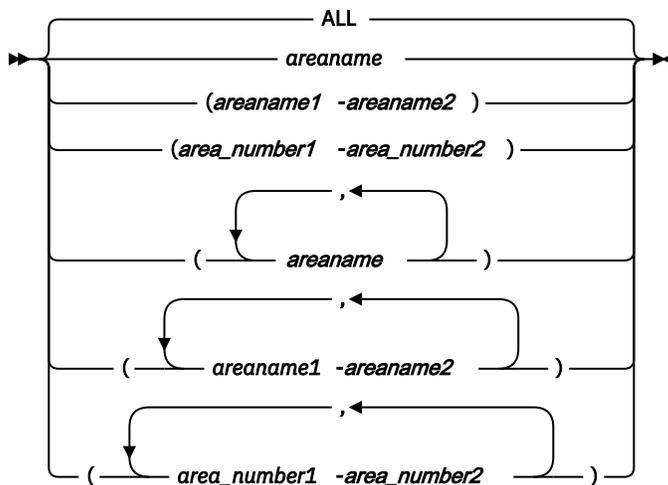
Optional keywords [A]:



Optional keywords [B]:



A:



Subcommands available with ANALYZE:



Default:

N/A

CHANGE command

Use the CHANGE command to restructure or reorganize multiple DEDB areas.

To reorganize areas of a DEDB, only the IMSACB library is required. If you want to change the definition of a DEDB (for example, to extend an area by increasing any of the sizes of UOW, IOVF, and CI), you also need to specify the OLDACB library that has the DMCB member of the new DEDB definition.

CHANGE is an optional command for the HFPSYSIN DD statement.

Related keywords:

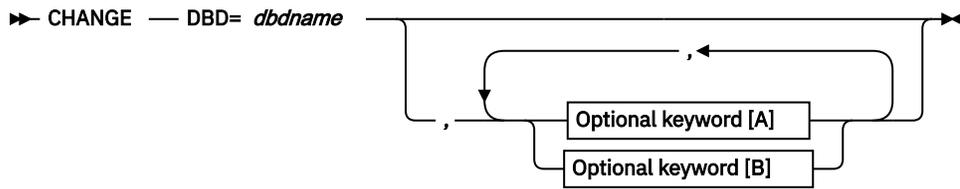
- [“ATASKCTL keyword” on page 443](#)
- [“CISNAP keyword” on page 448](#)
- [“DBD keyword” on page 450 \(required keyword\)](#)
- [“DBREC_DISCARD_MSGLVL keyword” on page 452](#)
- [“EDITCOMP_SKIP keyword” on page 459](#)
- [“HISTORYDD keyword” on page 469](#)
- [“IAREA keyword” on page 469](#)
- [“ICACHE keyword” on page 472](#)
- [“ICHECK keyword” on page 473](#)
- [“ICINCOMP keyword” on page 474](#)
- [“ICNOTIFY keyword” on page 475](#)
- [“ICOUT keyword” on page 476](#)
- [“IDSNMASK keyword” on page 476](#)
- [“IEXITRTN keyword” on page 479](#)
- [“IMSCATACB_OUTPUT keyword” on page 481](#)
- [“ITASKCTL keyword” on page 486](#)
- [“KEYCHGCHK keyword” on page 489](#)
- [“KEYSEQCHK keyword” on page 490](#)
- [“MAXERROR keyword” on page 495](#)
- [“OAREA keyword” on page 497](#)
- [“OCACHE keyword” on page 499](#)
- [“ODSNMASK keyword” on page 500](#)
- [“OEXITRTN keyword” on page 502](#)
- [“OTASKCTL keyword” on page 503](#)
- [“PAD keyword” on page 504](#)
- [“PTRCHK keyword” on page 505](#)
- [“PTRCHKLVL keyword” on page 506](#)
- [“RELOCATE_SDEP keyword” on page 512](#)
- [“SDEP keyword” on page 524](#)
- [“SDEP_BYPASS_MSGLVL keyword” on page 525](#)
- [“SDEPCHKLVL keyword” on page 526](#)

- [“SDEP_ORPHAN_MSGLVL keyword” on page 527](#)
- [“SENSOR keyword” on page 535](#)
- [“SENSOR_DBREC keyword” on page 536](#)
- [“SENSOR_PTRERR keyword” on page 537](#)
- [“SENSOR_UOW keyword” on page 537](#)
- [“SENSOR_UOW_REPORT keyword” on page 538](#)
- [“SORTOPT_FILSZ keyword” on page 542](#)
- [“SSP keyword” on page 544](#)

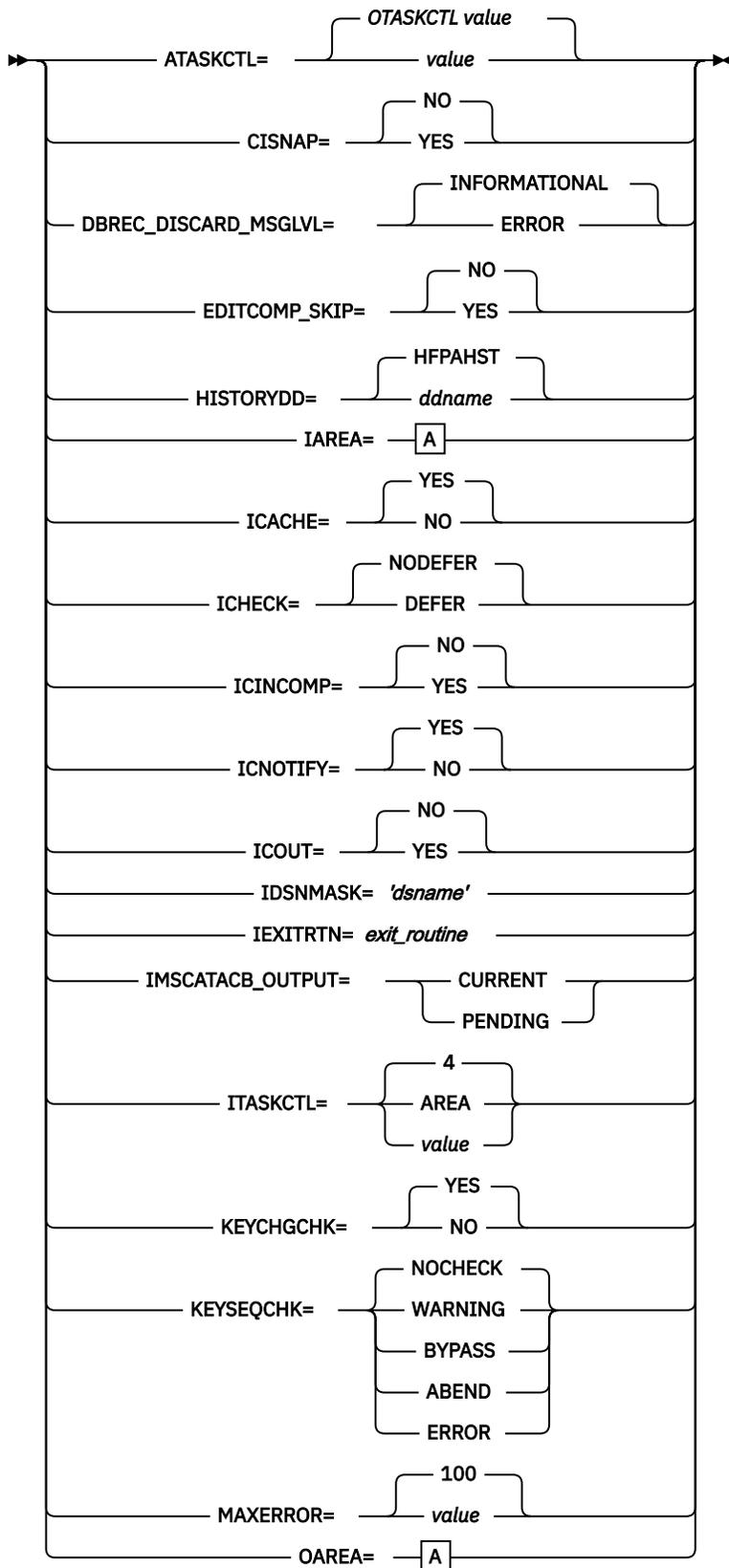
Format:

```
CHANGE [keyword=parameter[,...]] [subcommand [keyword=parameter[,...]]]
```

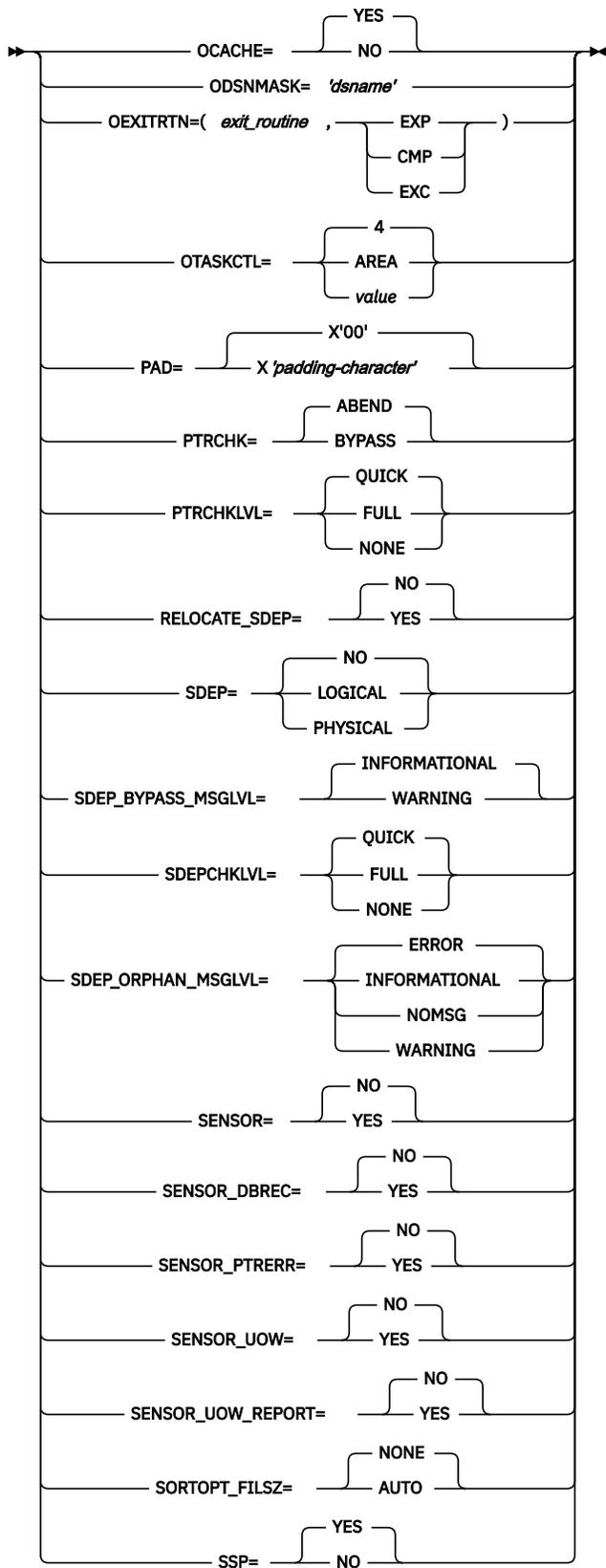
Syntax:



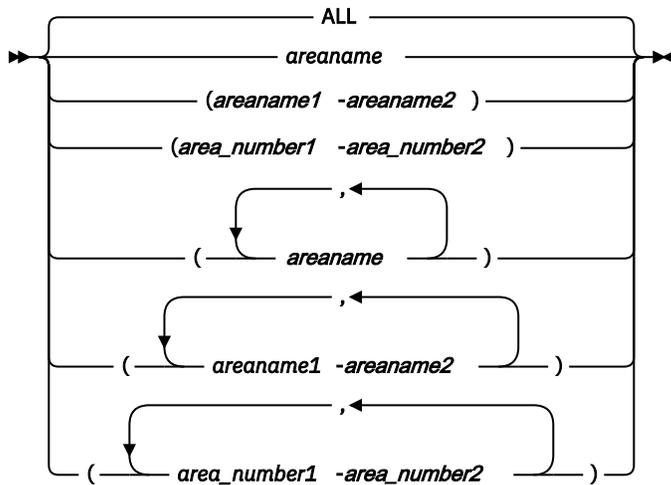
Optional keywords [A]:



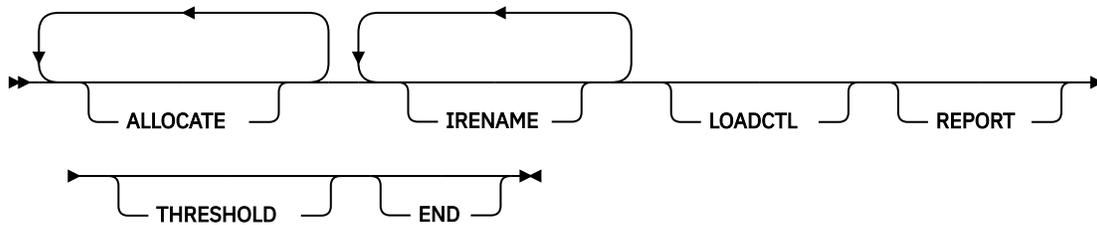
Optional keywords [B]:



A:



Subcommands available with CHANGE:



Default:
N/A

DMACPRT command

Use the DMACPRT command to print contents of the DMAC block in one or more areas.

DMACPRT is an optional command for the HFPSYSIN DD statement.

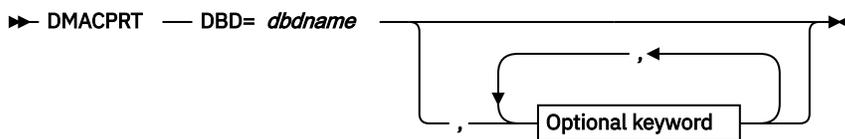
Related keywords:

- [“DBD keyword” on page 450](#) (required keyword)
- [“IAREA keyword” on page 469](#)
- [“IDSNMASK keyword” on page 476](#)

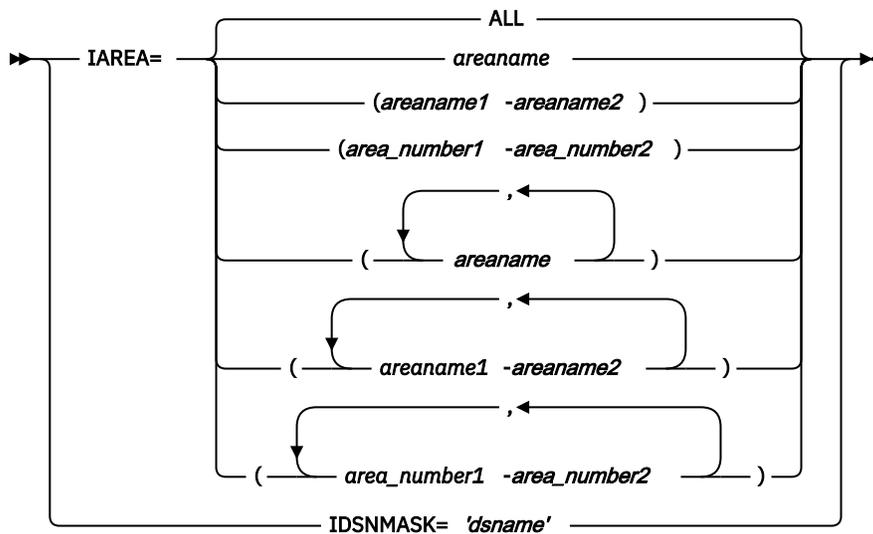
Format:

```
DMACPRT [keyword=parameter[,...]]
```

Syntax:



Optional keywords:



Default:
N/A

END command

Use the END command to specify the completion of a command set.

END is an optional command for all input sources.

Related keywords:

None.

Format:

END

Syntax:

➤ END ➤

Default:

N/A

EXTRACT command

Use the EXTRACT command to extract segment data from a DEDB and write the output to a sequential file.

EXTRACT is an optional command for the HFPSYSIN DD statement.

Related keywords:

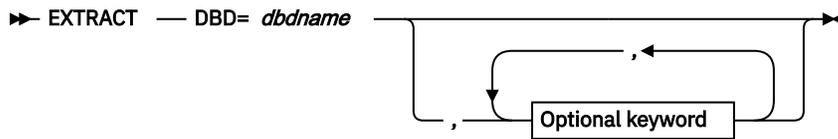
- [“COMPRESS keyword” on page 449](#)
- [“DBD keyword” on page 450](#) (required keyword)
- [“EFORMAT keyword” on page 460](#)
- [“IAREA keyword” on page 469](#)
- [“ICACHE keyword” on page 472](#)
- [“ICHECK keyword” on page 473](#)
- [“ICINCOMP keyword” on page 474](#)
- [“IDSNMASK keyword” on page 476](#)
- [“ITASKCTL keyword” on page 486](#)

- [“KEYSEQCHK keyword” on page 490](#)
- [“OCHECK keyword” on page 500](#)
- [“ODSNMASK keyword” on page 500](#)
- [“PTRCHK keyword” on page 505](#)
- [“SEGREC_PREFIX keyword” on page 531](#)
- [“SEGREC_SUFFIX keyword” on page 533](#)
- [“SORT keyword” on page 541](#)
- [“SORTOPT_FILSZ keyword” on page 542](#)

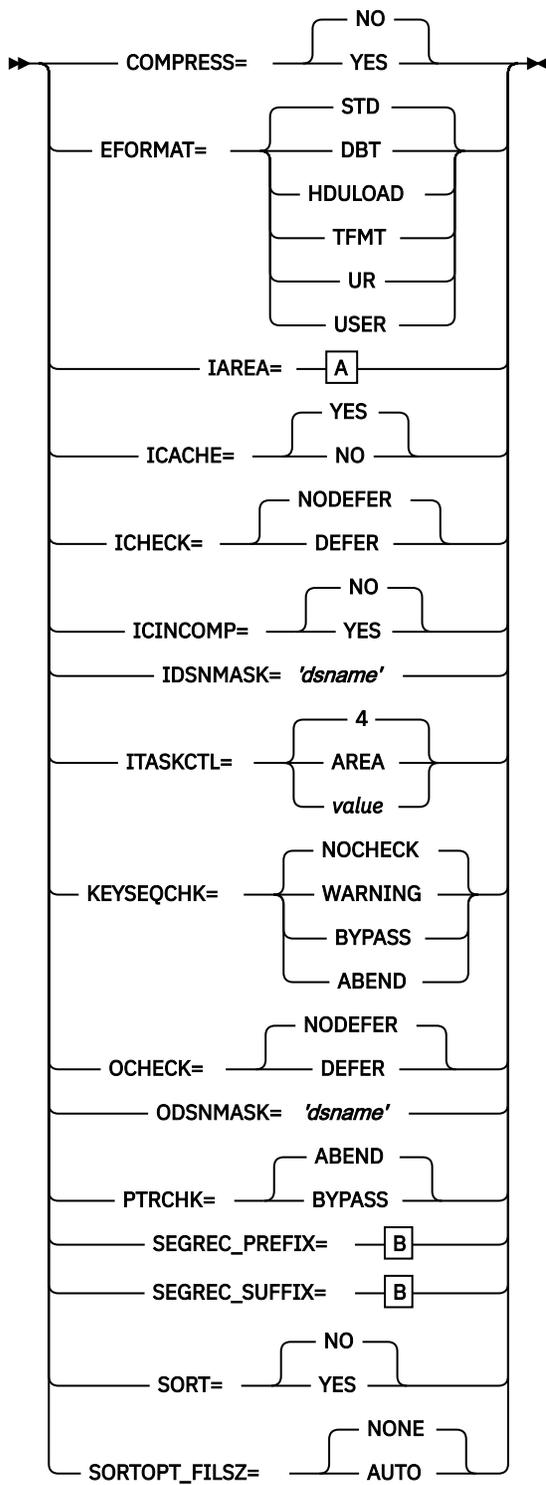
Format:

```
EXTRACT [keyword=parameter[,...]] [subcommand [keyword=parameter[,...]]]
```

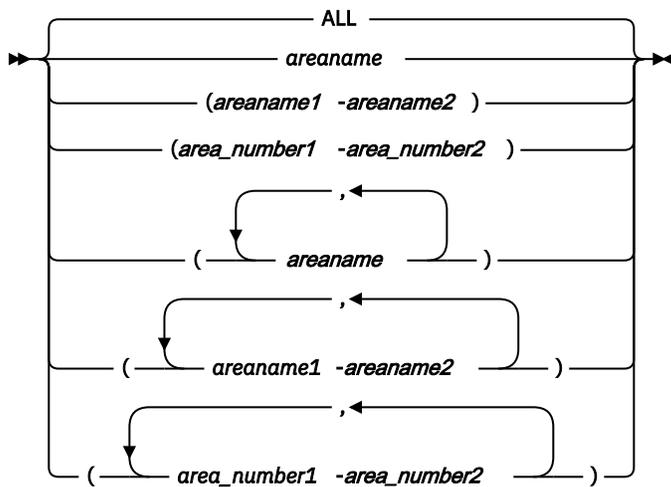
Syntax:



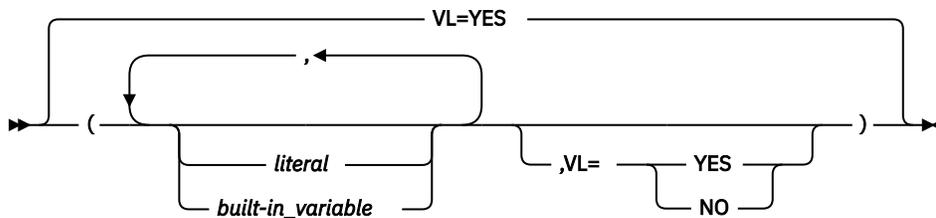
Optional keywords:



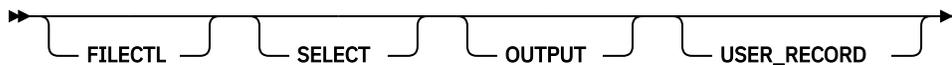
A:



B:



Subcommands available with EXTRACT:



Default:
N/A

GLOBAL command

Use the GLOBAL command to establish implicit job step keyword values.

The implicit keyword values that are set with the GLOBAL command will be in effect unless they are overridden. To override an implicitly set keyword value for a particular command, specify the keyword explicitly. If specified, GLOBAL must be the first command in the input stream.

GLOBAL is an optional command for the HFPSYSIN DD statement.

Related keywords:

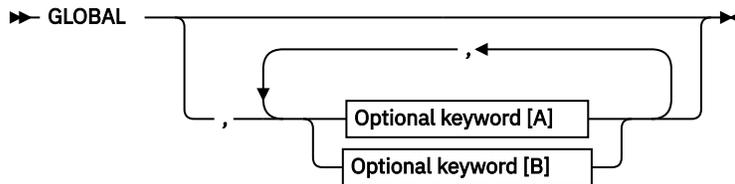
- [“ADXCFGRP keyword” on page 439](#)
- [“AREADSN keyword” on page 441](#)
- [“CISNAP keyword” on page 448](#)
- [“DBRC keyword” on page 451](#)
- [“HISTORYDD keyword” on page 469](#)
- [“ICINCOMP keyword” on page 474](#)
- [“ICNOTIFY keyword” on page 475](#)
- [“ICOUT keyword” on page 476](#)
- [“IMSCATACB_INPUT keyword” on page 480](#)
- [“IMSCATHLQ keyword” on page 481](#)
- [“IMSID keyword” on page 482](#)

- [“ITKBDATA keyword” on page 487](#)
- [“ITKBLOAD keyword” on page 487](#)
- [“ITKBSRVR keyword” on page 488](#)
- [“LIUGOPT keyword” on page 493](#)
- [“MSTIMESTAMP keyword” on page 497](#)
- [“PTRCHKLVL keyword” on page 506](#)
- [“RESTART keyword” on page 514](#)
- [“RESTART_COUNT keyword” on page 515](#)
- [“SCAN keyword” on page 523](#)
- [“SDEPCHKLVL keyword” on page 526](#)
- [“SDEP_ORPHAN_MSGLVL keyword” on page 527](#)
- [“SITE_DSNAME keyword” on page 539](#)
- [“SITE_MEMBER keyword” on page 540](#)
- [“SYNC keyword” on page 547](#)
- [“TOINUM keyword” on page 551](#)
- [“TOITIME keyword” on page 551](#)
- [“TOIXCFGRP keyword” on page 552](#)

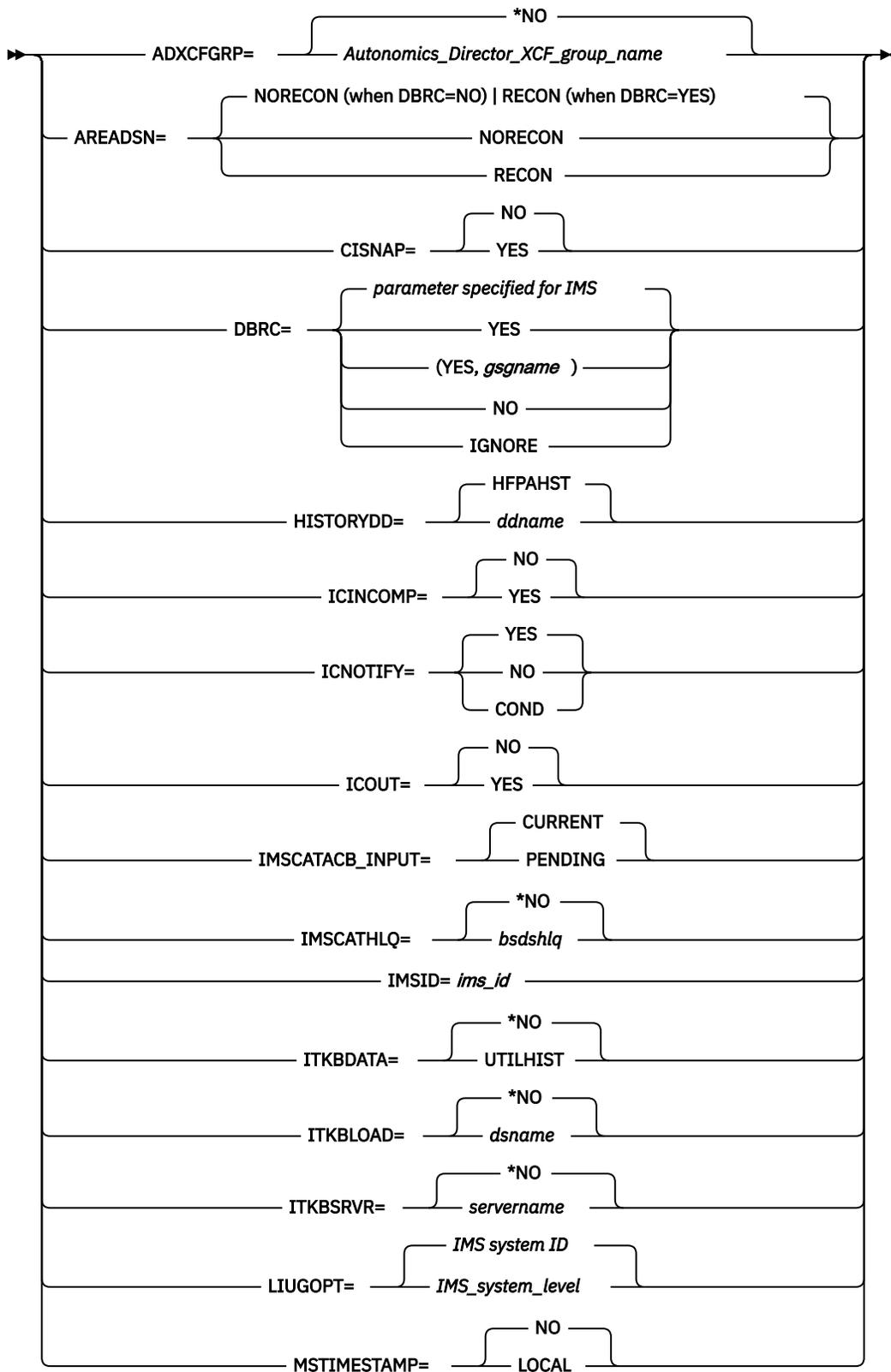
Format:

```
GLOBAL [keyword=parameter[,...]] [subcommand [keyword=parameter[,...]]]
```

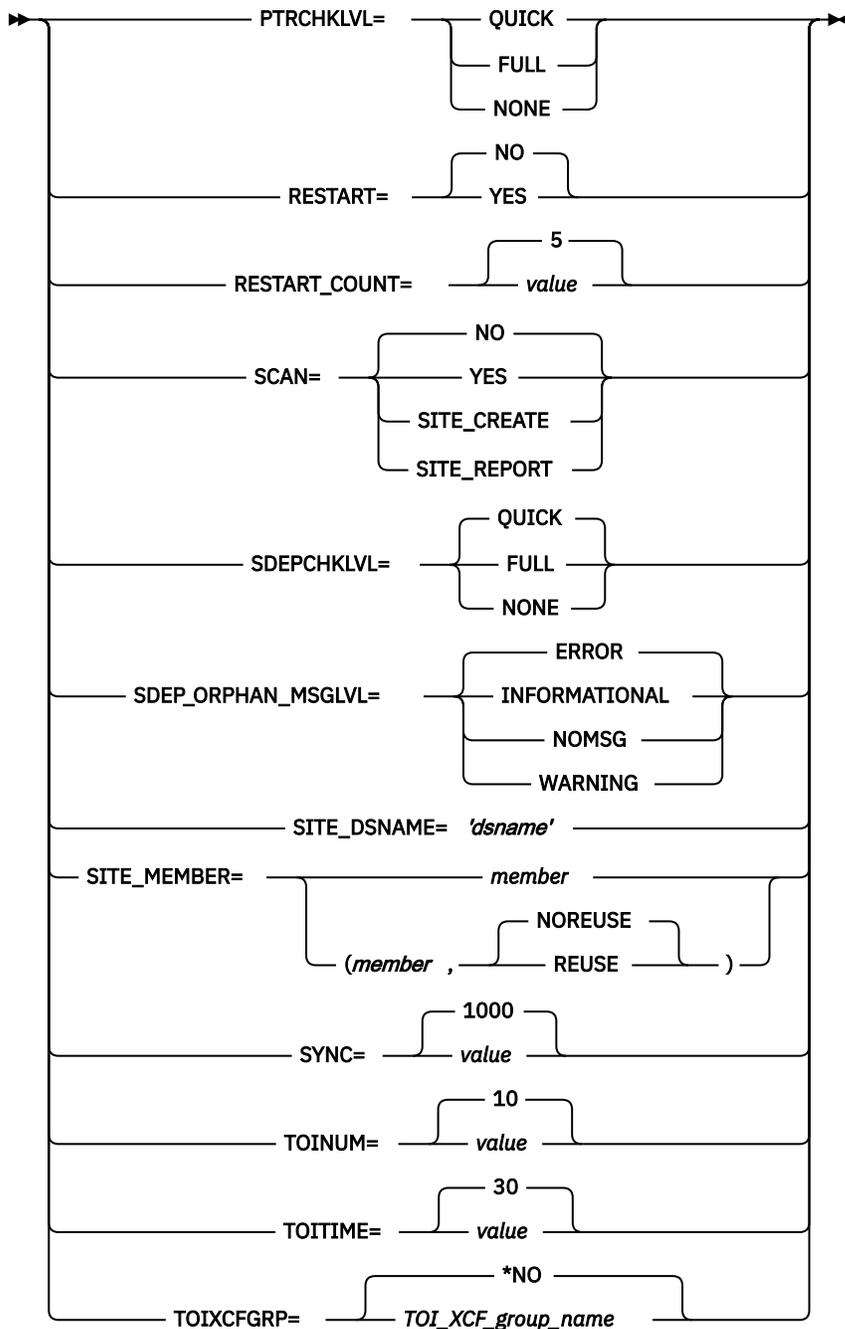
Syntax:



Optional keywords [A]:



Optional keywords [B]:



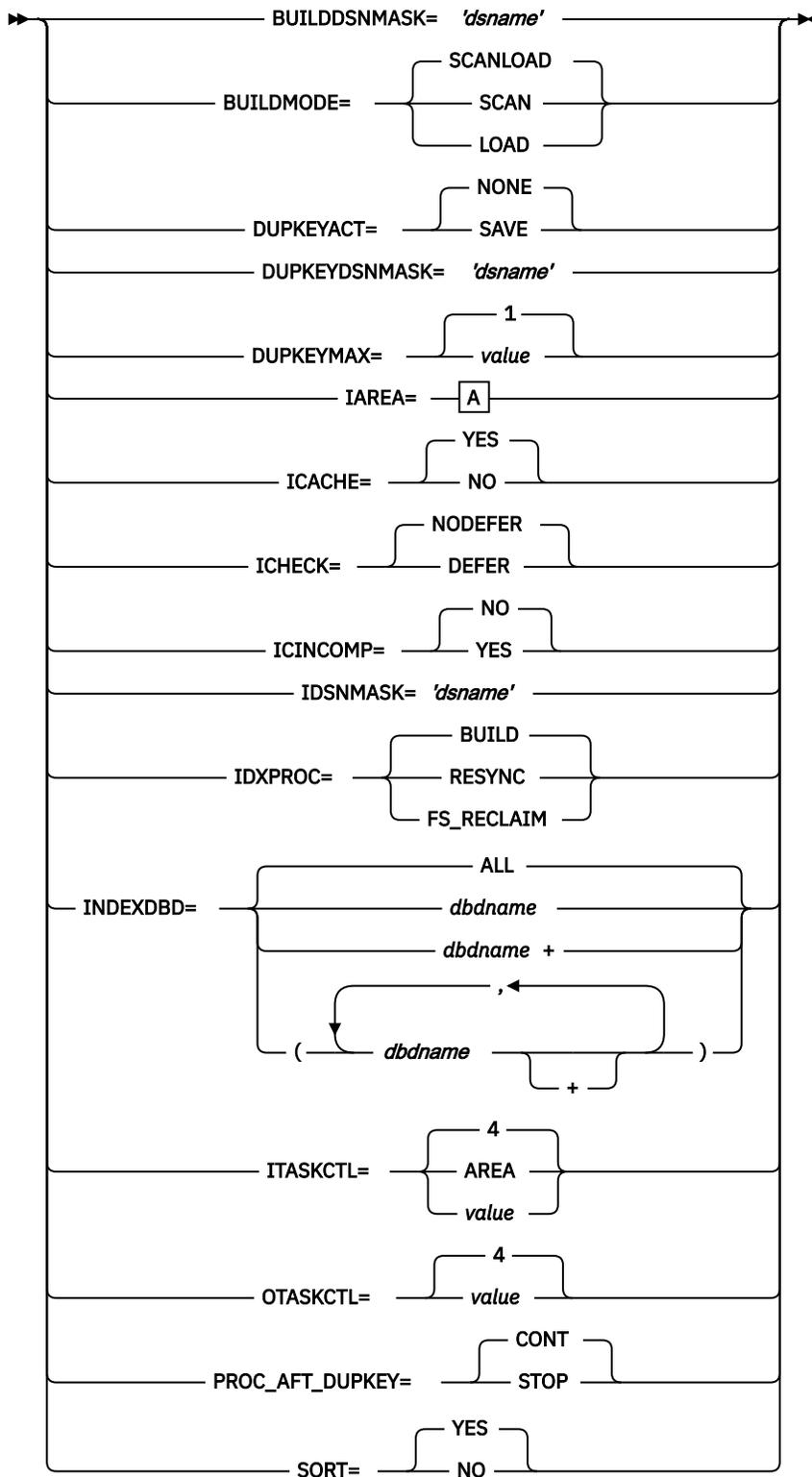
Default:
N/A

INDEXBLD command (for the Build Index function)

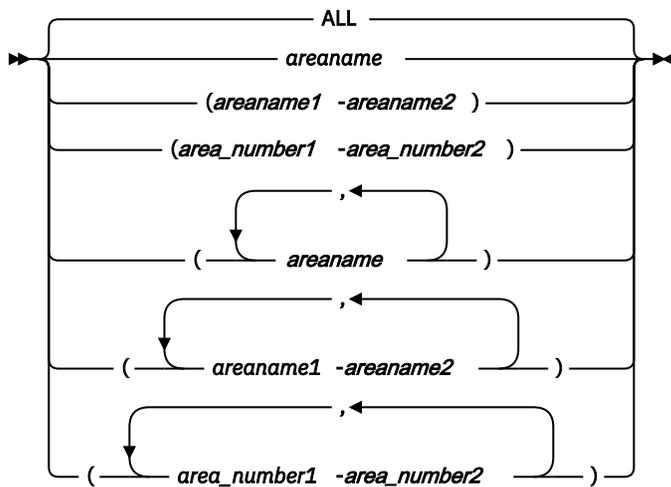
Use the INDEXBLD command to build one or more secondary index databases.

Note: The INDEXBLD command starts the Build Index function, the Resync function, or the Freespace Reclaim function. If you want to synchronize secondary indexes with the Resync function, see “INDEXBLD command (for the Resync function)” on page 400. If you want to reclaim CIs in secondary index database data sets, see “INDEXBLD command (for the Freespace Reclaim function)” on page 403.

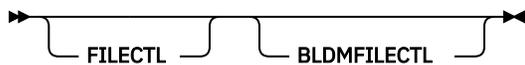
INDEXBLD is an optional command for the HFPSYSIN DD statement.



A:



Subcommands available with INDEXBLD:



Default:
N/A

INDEXBLD command (for the Resync function)

Use the INDEXBLD command to synchronize secondary index databases with their primary DEDB database.

Note: The INDEXBLD command starts the Build Index function, the Resync function, or the Freespace Reclaim function. If you want to build secondary indexes with the Build Index function, see [“INDEXBLD command \(for the Build Index function\)”](#) on page 397. If you want to reclaim CIs in secondary index database data sets, see [“INDEXBLD command \(for the Freespace Reclaim function\)”](#) on page 403.

INDEXBLD is an optional command for the HFPSYSIN DD statement.

Important: To use the Resync function, you must specify `IDXPROC=RESYNC`.

Related keywords:

- [“AREASCAN keyword”](#) on page 442
- [“DBD keyword”](#) on page 450 (required keyword)
- [“DUPKEYMAX keyword”](#) on page 458
- [“IAREA keyword”](#) on page 469
- [“ICACHE keyword”](#) on page 472
- [“ICHECK keyword”](#) on page 473
- [“IDSNMASK keyword”](#) on page 476
- [“IDXPROC keyword”](#) on page 478 (required keyword)
- [“INDEXDBD keyword”](#) on page 482
- [“ITASKCTL keyword”](#) on page 486
- [“OTASKCTL keyword”](#) on page 503
- [“PROC_AFT_DUPKEY keyword”](#) on page 504
- [“RESYNCDSNMASK keyword”](#) on page 515
- [“RESYNCDUMP keyword”](#) on page 516
- [“RESYNCDUMPMAX keyword”](#) on page 517

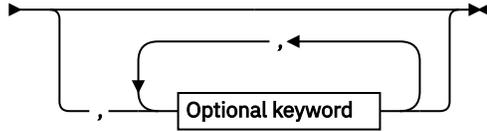
- [“RESYNCMODE keyword” on page 517](#)
- [“TOICTL keyword” on page 550](#)

Format:

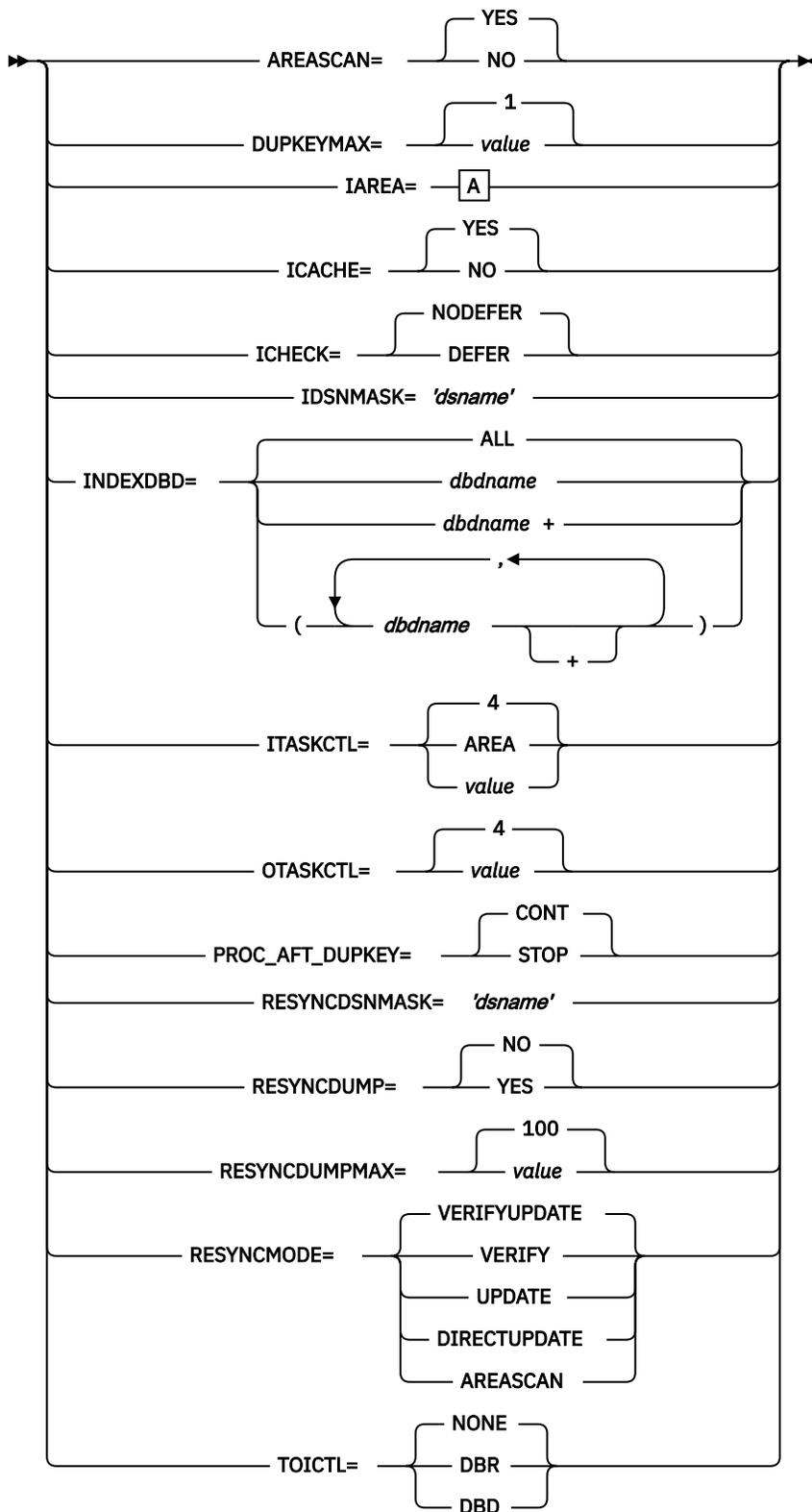
```
INDEXBLD [keyword=parameter[,...]] [subcommand [keyword=parameter[,...]]]
```

Syntax:

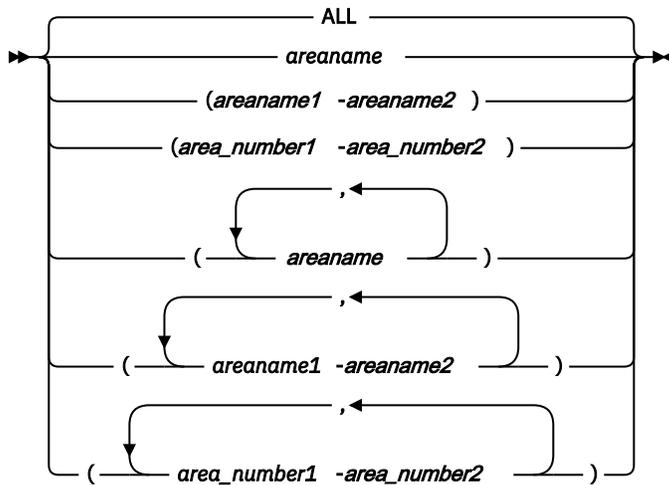
► INDEXBLD — DBD= *dbdname* — , — IDXPROC=RESYNC →



Optional keywords:



A:



Subcommands available with INDEXBLD:



Default:
N/A

INDEXBLD command (for the Freespace Reclaim function)

Use the INDEXBLD command to reclaim CIs in secondary index database data sets.

Note: The INDEXBLD command starts the Build Index function, the Resync function, or the Freespace Reclaim function. If you want to build secondary indexes with the Build Index function, see “INDEXBLD command (for the Build Index function)” on page 397. If you want to synchronize secondary indexes with the Resync function, see “INDEXBLD command (for the Resync function)” on page 400.

INDEXBLD is an optional command for the HFPSYSIN DD statement.

Important: To use the Freespace Reclaim function, you must specify IDXPROC=FS_RECLAIM.

Related keywords:

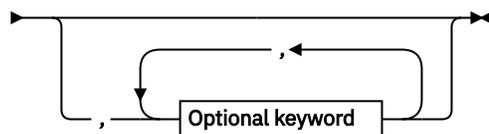
- “DBD keyword” on page 450 (required keyword)
- “FS_RECLAIM_REPORT keyword” on page 468
- “IDXPROC keyword” on page 478 (required keyword)
- “INDEXDBD keyword” on page 482

Format:

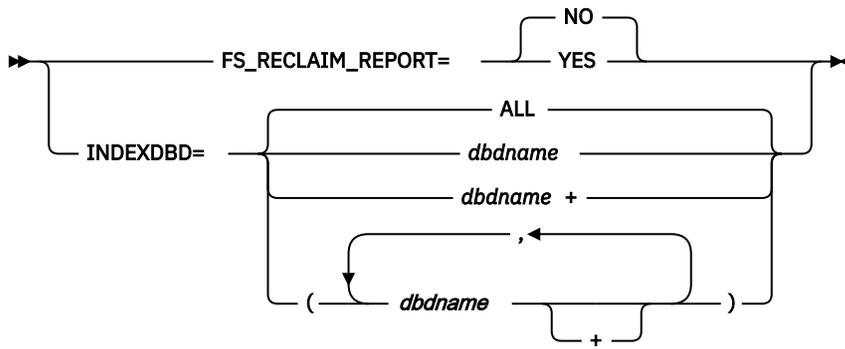
```
INDEXBLD [keyword=parameter[,...]]
```

Syntax:

➤ INDEXBLD — DBD= *dbdname* — , — IDXPROC=FS_RECLAIM ➔



Optional keywords:



Default:
N/A

INITIALIZE command

Use the INITIALIZE command to initialize one or more area data sets of a DEDB.

INITIALIZE is an optional command for the HFPSYSIN DD statement.

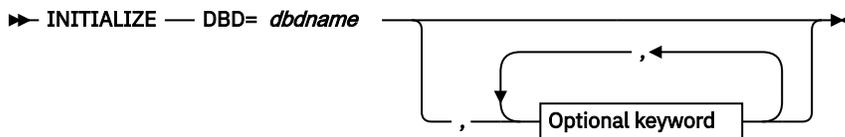
Related keywords:

- [“ALLOCONLY keyword” on page 440](#)
- [“DBD keyword” on page 450 \(required keyword\)](#)
- [“OAREA keyword” on page 497](#)
- [“OCHECK keyword” on page 500](#)
- [“ODSNMASK keyword” on page 500](#)
- [“OTASKCTL keyword” on page 503](#)

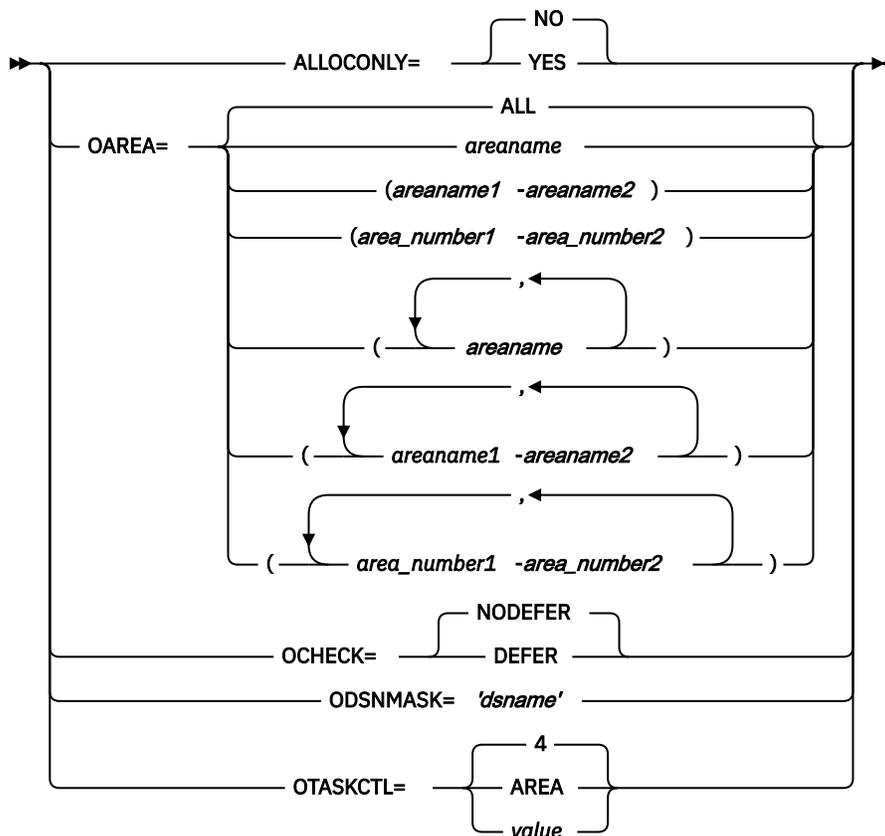
Format:

```
INITIALIZE [keyword=parameter[,...]] [subcommand [keyword=parameter[,...]]]
```

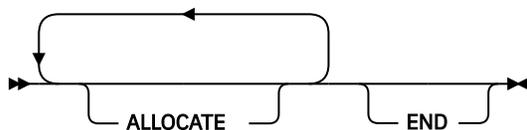
Syntax:



Optional keywords:



Subcommand available with INITIALIZE:



Default:

N/A

RELOAD command

Use the RELOAD command to reload one or more areas from a DEDB unloaded records file.

The ACB of the database that is being reloaded is used to describe the areas that are being loaded and is referred to by the IMSACB DD statement.

RELOAD is an optional command for the HFPSYSIN DD statement.

Related keywords:

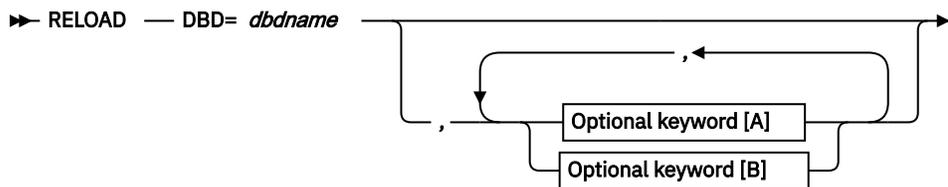
- [“ACBCHK keyword” on page 438](#)
- [“ATASKCTL keyword” on page 443](#)
- [“CISNAP keyword” on page 448](#)
- [“DBD keyword” on page 450 \(required keyword\)](#)
- [“EXITRTN keyword” on page 461](#)
- [“FORMAT keyword” on page 465](#)
- [“HISTORYDD keyword” on page 469](#)
- [“IAREA keyword” on page 469](#)
- [“ICHECK keyword” on page 473](#)

- [“ICNOTIFY keyword” on page 475](#)
- [“ICOUT keyword” on page 476](#)
- [“IDSNMASK keyword” on page 476](#)
- [“ITASKCTL keyword” on page 486](#)
- [“KEYCHGCHK keyword” on page 489](#)
- [“KEYSEQCHK keyword” on page 490](#)
- [“MAXERROR keyword” on page 495](#)
- [“OAREA keyword” on page 497](#)
- [“OCACHE keyword” on page 499](#)
- [“OCHECK keyword” on page 500](#)
- [“ODSNMASK keyword” on page 500](#)
- [“PTRCHKLVL keyword” on page 506](#)
- [“SDEPCHKLVL keyword” on page 526](#)
- [“SDEP_ORPHAN_MSGLVL keyword” on page 527](#)
- [“SEGREC_PREFIX keyword” on page 531](#)
- [“SEGREC_SUFFIX keyword” on page 533](#)
- [“SENSOR keyword” on page 535](#)
- [“SENSOR_DBREC keyword” on page 536](#)
- [“SENSOR_PTRERR keyword” on page 537](#)
- [“SENSOR_UOW keyword” on page 537](#)
- [“SENSOR_UOW_REPORT keyword” on page 538](#)
- [“SORT keyword” on page 541](#)
- [“SORTOPT_FILSZ keyword” on page 542](#)
- [“SSP keyword” on page 544](#)

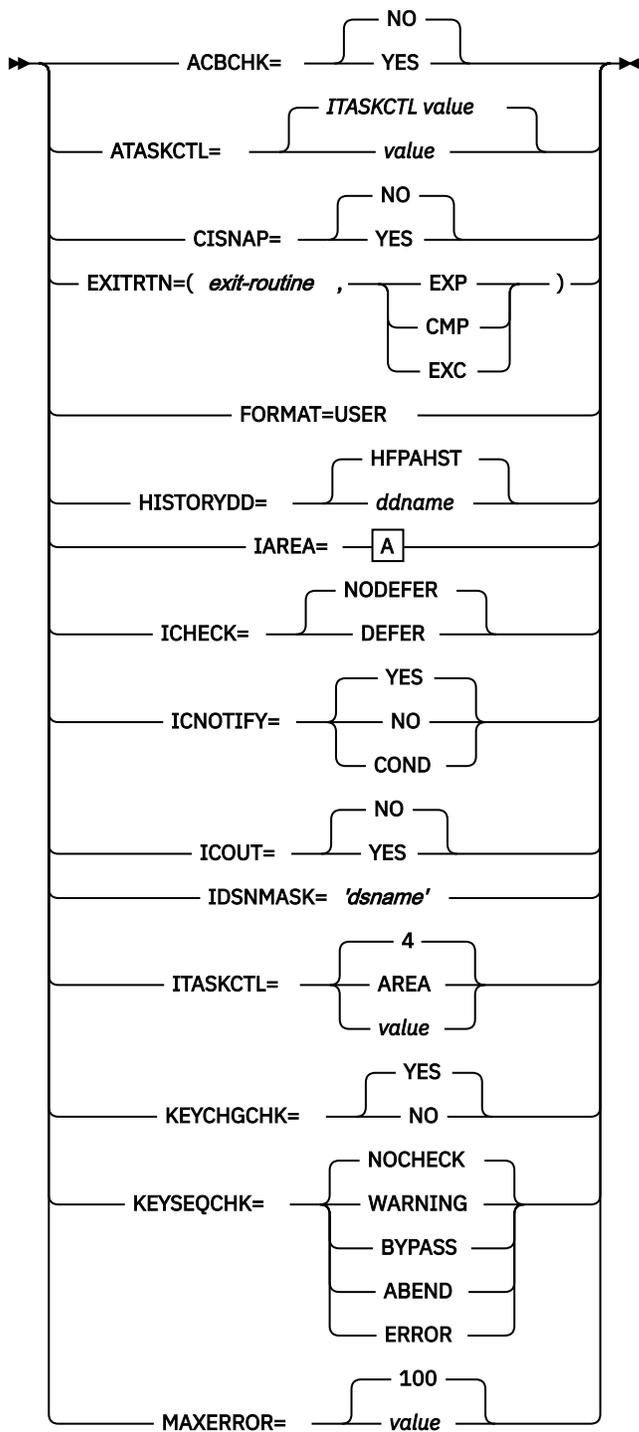
Format:

```
RELOAD [keyword=parameter[,...]] [subcommand [keyword=parameter[,...]]]
```

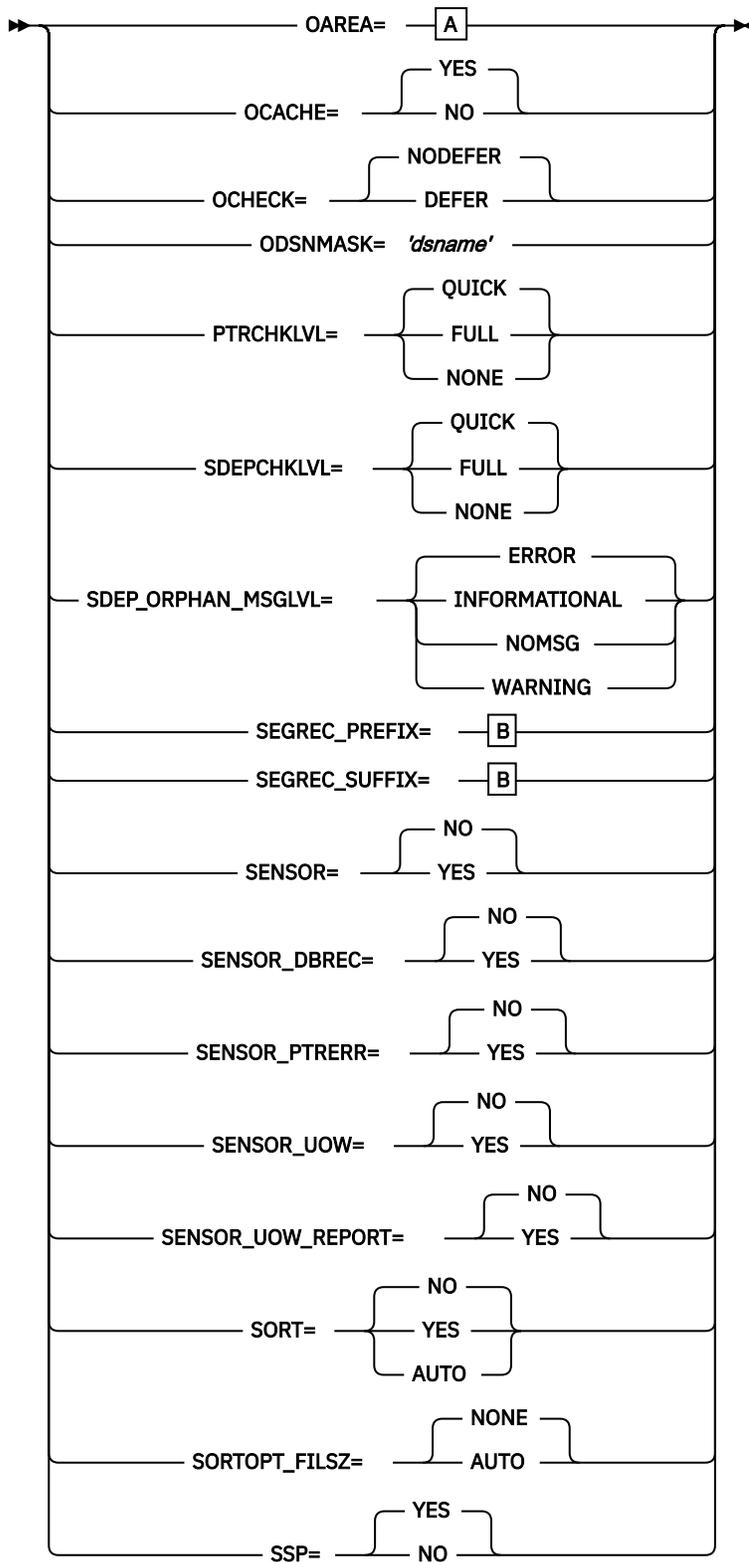
Syntax:



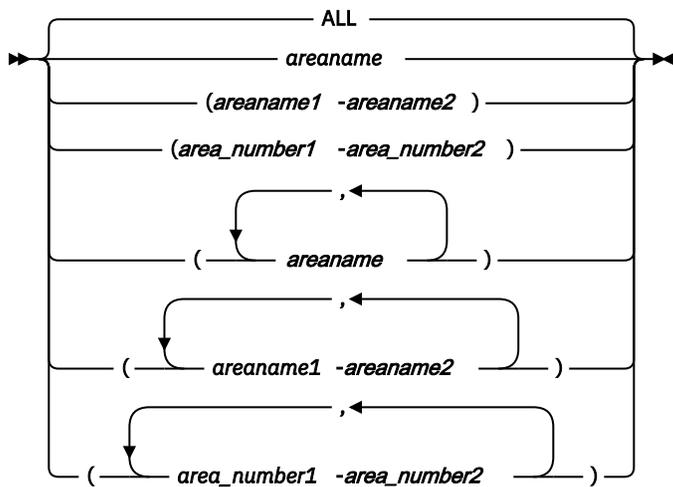
Optional keywords [A]:



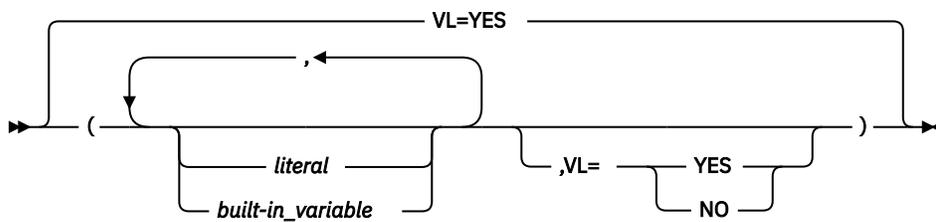
Optional keywords [B]:



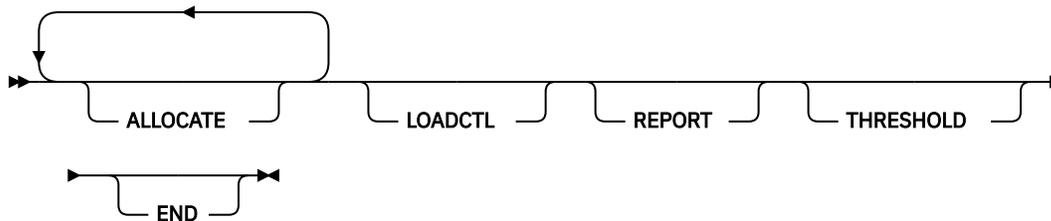
A:



B:



Subcommands available with RELOAD:



Default:
N/A

REORG command

Use the REORG command to reorganize one or more off-line DEDB areas.

REORG is an optional command for the HFPSYSIN DD statement.

Related keywords:

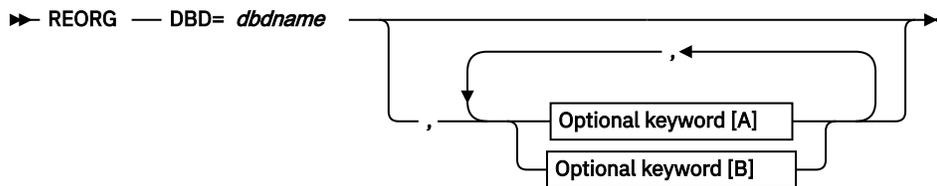
- [“ATASKCTL keyword” on page 443](#)
- [“CISNAP keyword” on page 448](#)
- [“DBD keyword” on page 450](#) (required keyword)
- [“HISTORYDD keyword” on page 469](#)
- [“IAREA keyword” on page 469](#)
- [“ICACHE keyword” on page 472](#)
- [“ICNOTIFY keyword” on page 475](#)
- [“ICOUT keyword” on page 476](#)
- [“IDSNMASK keyword” on page 476](#)
- [“ITASKCTL keyword” on page 486](#)

- [“MAXFAILS keyword” on page 496](#)
- [“PTRCHKLVL keyword” on page 506](#)
- [“RBASEFS keyword” on page 509](#)
- [“RDOVFFS keyword” on page 509](#)
- [“RECOVOPT keyword” on page 511](#)
- [“REORGLIM keyword” on page 513](#)
- [“RFSSCAN keyword” on page 519](#)
- [“RFSSTAT keyword” on page 520](#)
- [“SDEPCHKLVL keyword” on page 526](#)
- [“SDEP_ORPHAN_MSGLVL keyword” on page 527](#)
- [“SDEP_PTRIOVF_MSGLVL keyword” on page 528](#)
- [“SENSOR keyword” on page 535](#)
- [“SENSOR_DBREC keyword” on page 536](#)
- [“SENSOR_PTRERR keyword” on page 537](#)
- [“SENSOR_UOW keyword” on page 537](#)
- [“SENSOR_UOW_REPORT keyword” on page 538](#)
- [“SORTOPT_FILSZ keyword” on page 542](#)

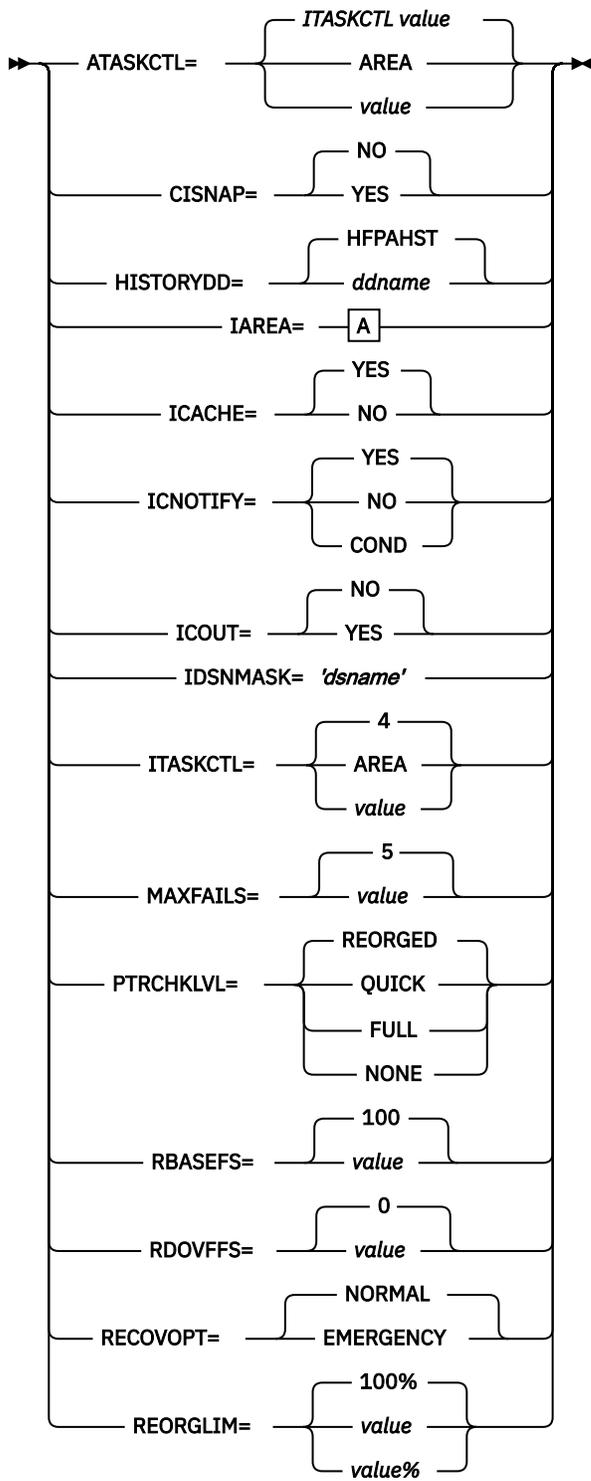
Format:

```
REORG [keyword=parameter[,...]] [subcommand [keyword=parameter[,...]]]
```

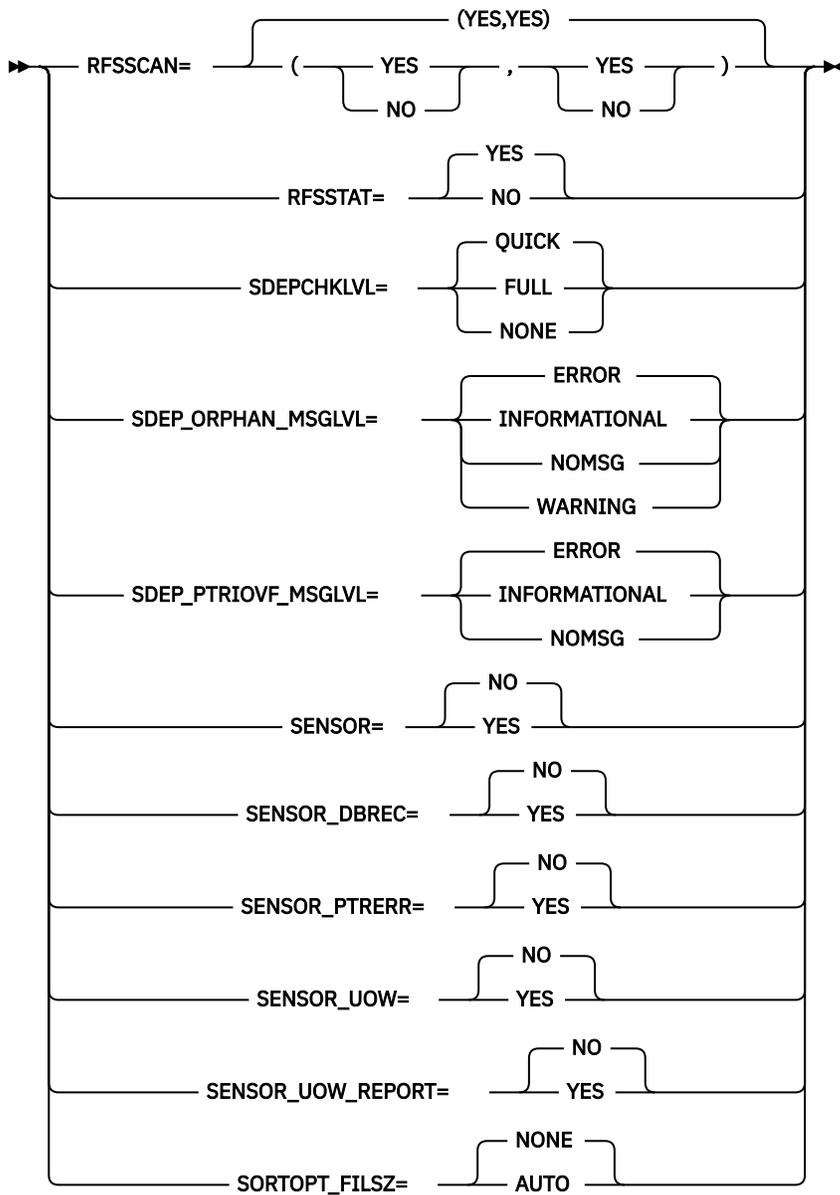
Syntax:



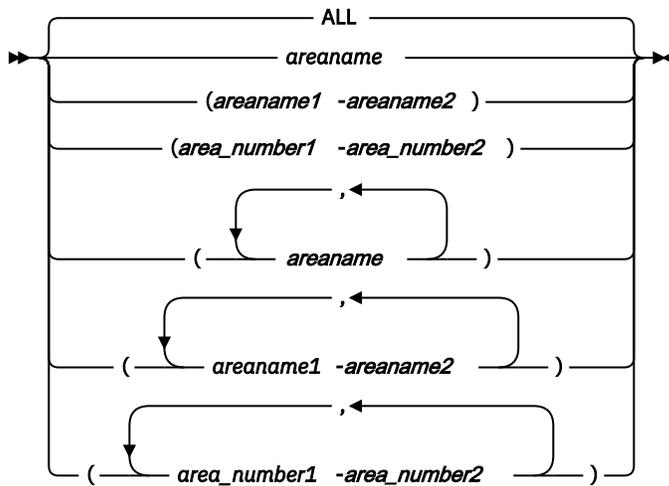
Optional keywords [A]:



Optional keywords [B]:



A:



Subcommands available with REORG:



Default:
N/A

TUNE command

Use the TUNE command to simulate changes in database attributes.

TUNE is an optional command for the HFPSYSIN DD statement.

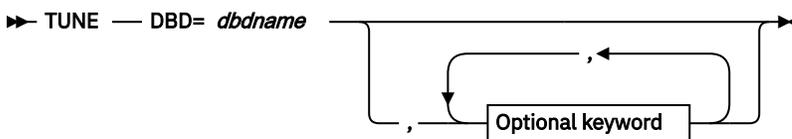
Related keywords:

- [“AREASCAN keyword” on page 442](#)
- [“CISNAP keyword” on page 448](#)
- [“DBD keyword” on page 450 \(required keyword\)](#)
- [“IAREA keyword” on page 469](#)
- [“ICACHE keyword” on page 472](#)
- [“ICHECK keyword” on page 473](#)
- [“ICINCOMP keyword” on page 474](#)
- [“IDSNMASK keyword” on page 476](#)
- [“ITASKCTL keyword” on page 486](#)
- [“MAXERROR keyword” on page 495](#)
- [“OTASKCTL keyword” on page 503](#)
- [“SORTOPT_FILSZ keyword” on page 542](#)
- [“TARDSNMASK keyword” on page 549](#)

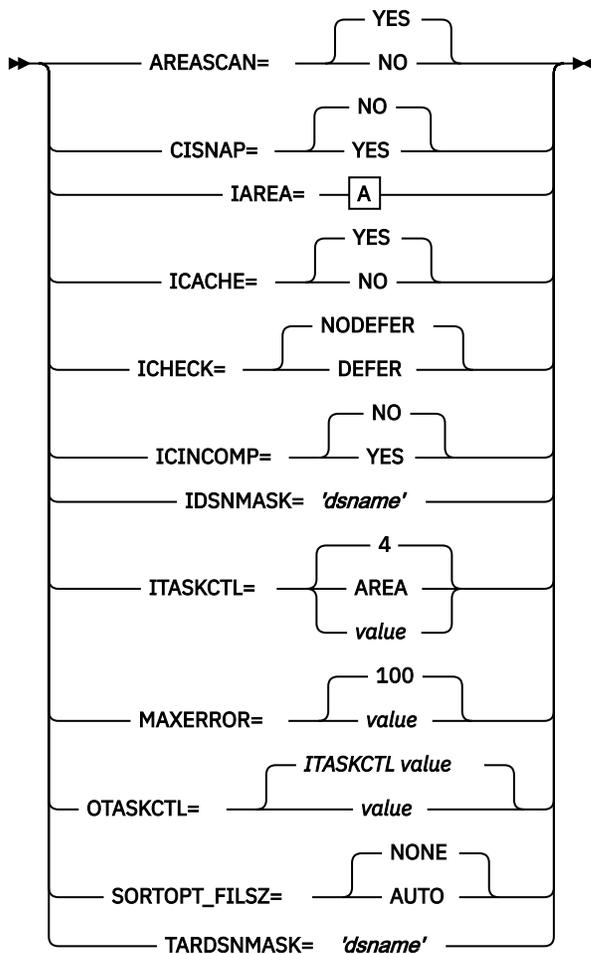
Format:

```
TUNE [keyword=parameter[,...]] [subcommand [keyword=parameter[,...]]]
```

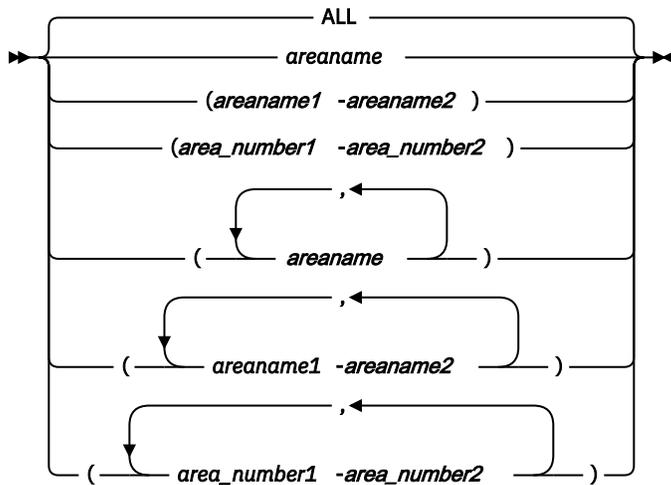
Syntax:



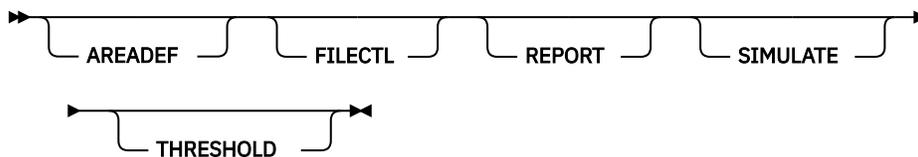
Optional keywords:



A:



Subcommands available with TUNE:



Default:

N/A

UNLOAD command

Use the UNLOAD command to unload one or more areas from a DEDB in a file format.

The ACB of the database being unloaded is used to describe the areas and is referred to by the IMSACB DD statement.

UNLOAD is an optional command for the HFPSYSIN DD statement.

Related keywords:

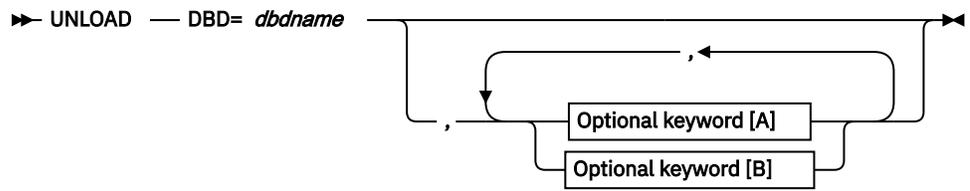
- [“ACBCHK keyword” on page 438](#)
- [“AREA_INFORMATION_RECORD keyword” on page 443](#)
- [“CISNAP keyword” on page 448](#)
- [“COMPRESS keyword” on page 449](#)
- [“DBD keyword” on page 450 \(required keyword\)](#)
- [“DBREC_DISCARD_MSGLVL keyword” on page 452](#)
- [“DOVNULLPTR_MSGLVL keyword” on page 455](#)
- [“EDITCOMP_SKIP keyword” on page 459](#)
- [“EXITRTN keyword” on page 461](#)
- [“FORMAT keyword” on page 465](#)
- [“IAREA keyword” on page 469](#)
- [“ICACHE keyword” on page 472](#)
- [“ICHECK keyword” on page 473](#)
- [“ICINCOMP keyword” on page 474](#)
- [“ICNOTIFY keyword” on page 475](#)
- [“ICOUT keyword” on page 476](#)
- [“IDSNMASK keyword” on page 476](#)
- [“IMSCATACB_OUTPUT keyword” on page 481](#)
- [“ITASKCTL keyword” on page 486](#)
- [“KEYCHGCHK keyword” on page 489](#)
- [“KEYSEQCHK keyword” on page 490](#)
- [“LRECL keyword” on page 494](#)
- [“OAREA keyword” on page 497](#)
- [“OCHECK keyword” on page 500](#)
- [“ODSNMASK keyword” on page 500](#)
- [“PAD keyword” on page 504](#)
- [“PTRCHK keyword” on page 505](#)
- [“RELOCATE_SDEP keyword” on page 512](#)
- [“SDEP keyword” on page 524](#)
- [“SDEP_BYPASS_MSGLVL keyword” on page 525](#)
- [“SEGREC_PREFIX keyword” on page 531](#)
- [“SEGREC_SUFFIX keyword” on page 533](#)
- [“SORT keyword” on page 541](#)
- [“SORTOPT_FILSZ keyword” on page 542](#)

- [“SSP keyword” on page 544](#)

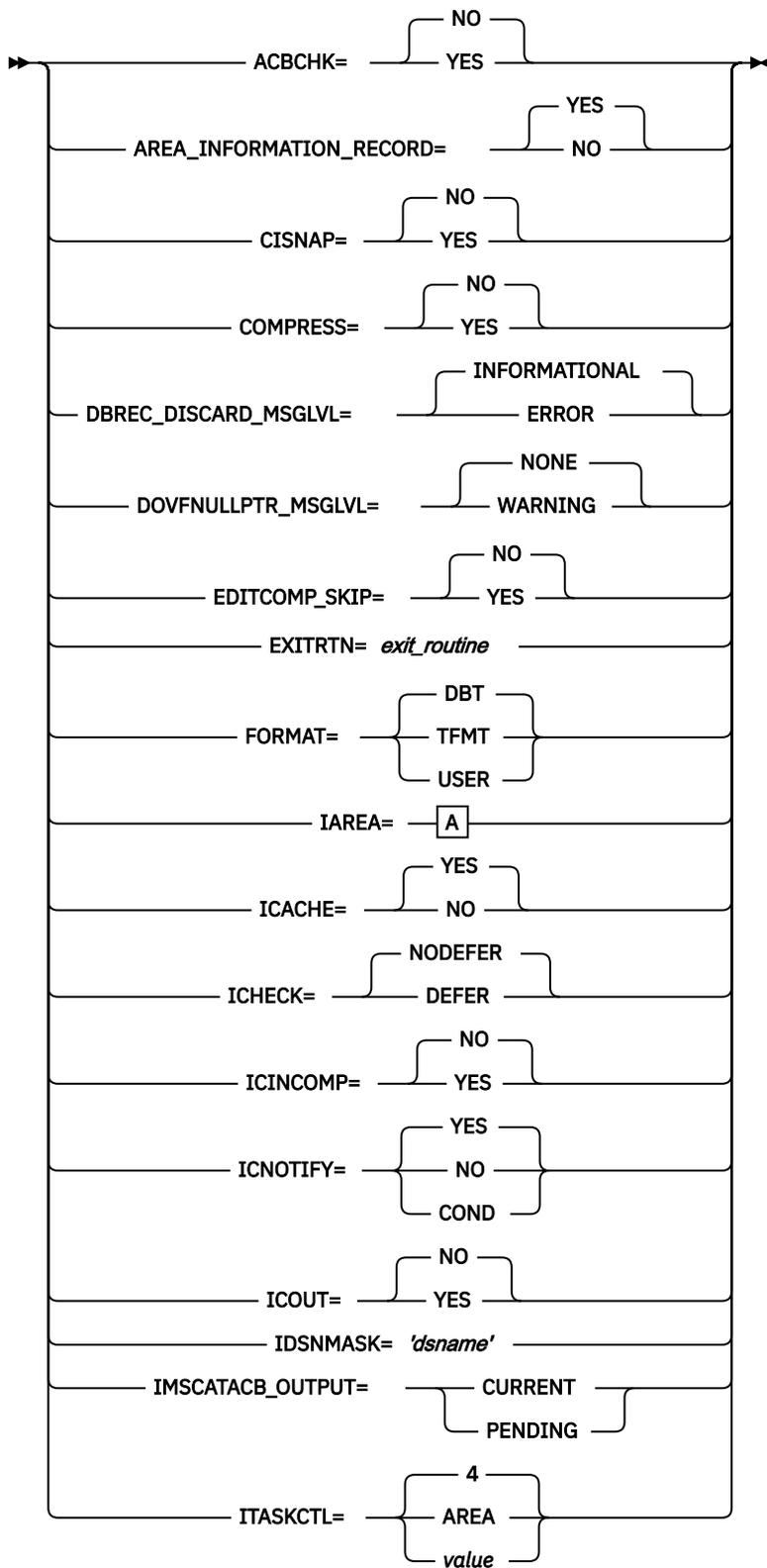
Format:

```
UNLOAD [keyword=parameter[,...]] [subcommand [keyword=parameter[,...]]]
```

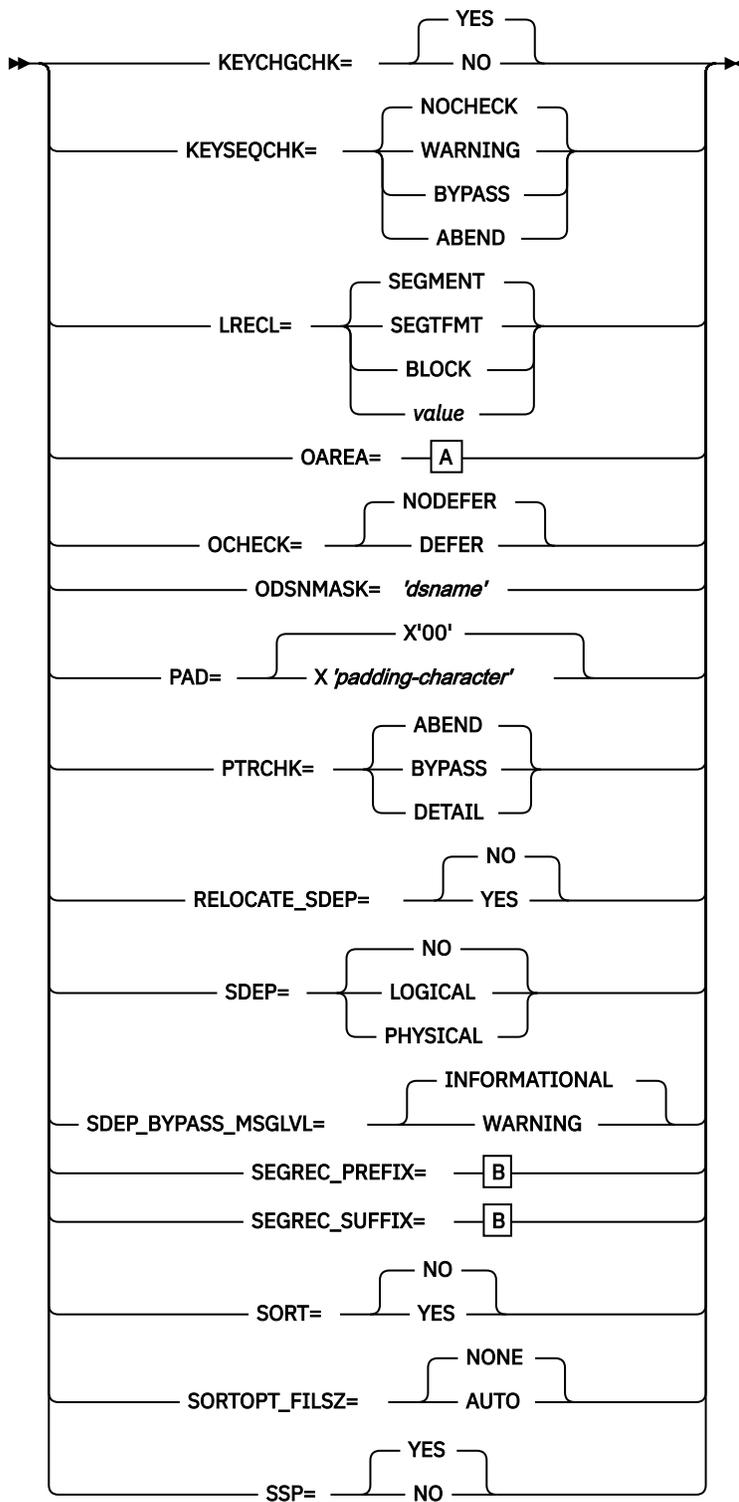
Syntax:



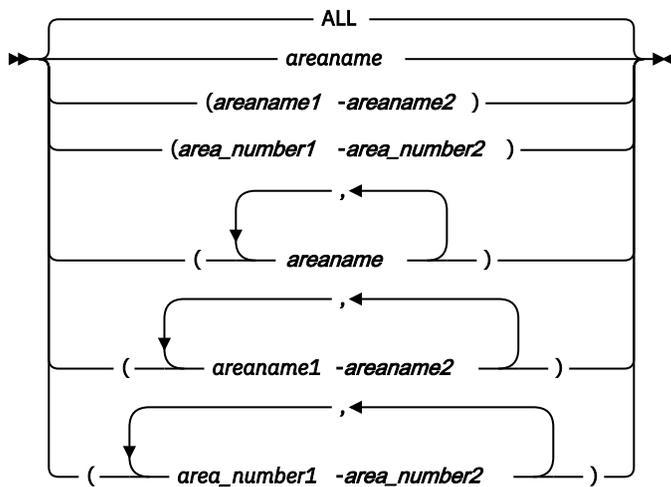
Optional keywords [A]:



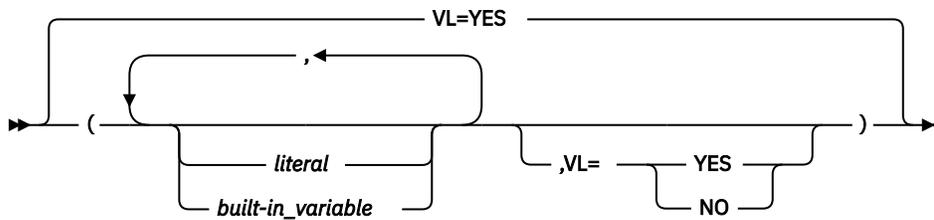
Optional keywords [B]:



A:



B:



Subcommands available with UNLOAD:



Default:
N/A

FPA subcommands

The following topics explain the FPA subcommands.

ALLOCATE subcommand

Use the ALLOCATE subcommand to dynamically define a VSAM cluster that is used for the processing of the FPA command.

ALLOCATE is an optional subcommand for the following commands:

- CHANGE
- INITIALIZE
- RELOAD

Related keywords:

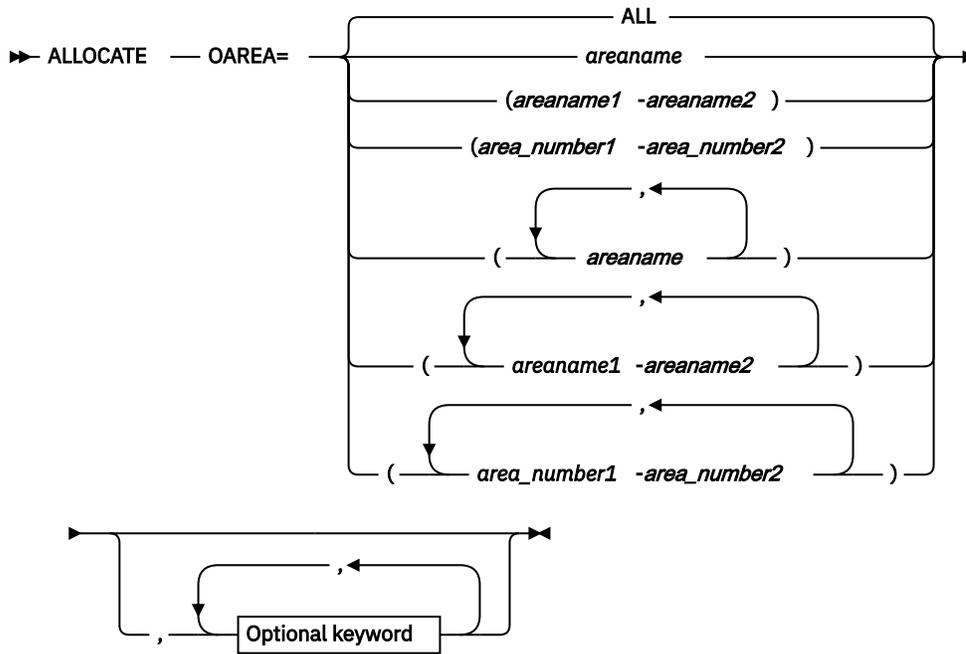
- [“AMSOPT keyword” on page 440](#)
- [“ATTRIBUTE keyword” on page 444](#)
- [“AVGREC keyword” on page 445](#)
- [“DATACLAS keyword” on page 449](#)
- [“KEYLABEL keyword” on page 489](#)

- [“MGMTCLAS keyword” on page 496](#)
- [“OAREA keyword” on page 497 \(required keyword\)](#)
- [“SPACE keyword” on page 543](#)
- [“STORCLAS keyword” on page 546](#)
- [“VOLSER keyword” on page 556](#)

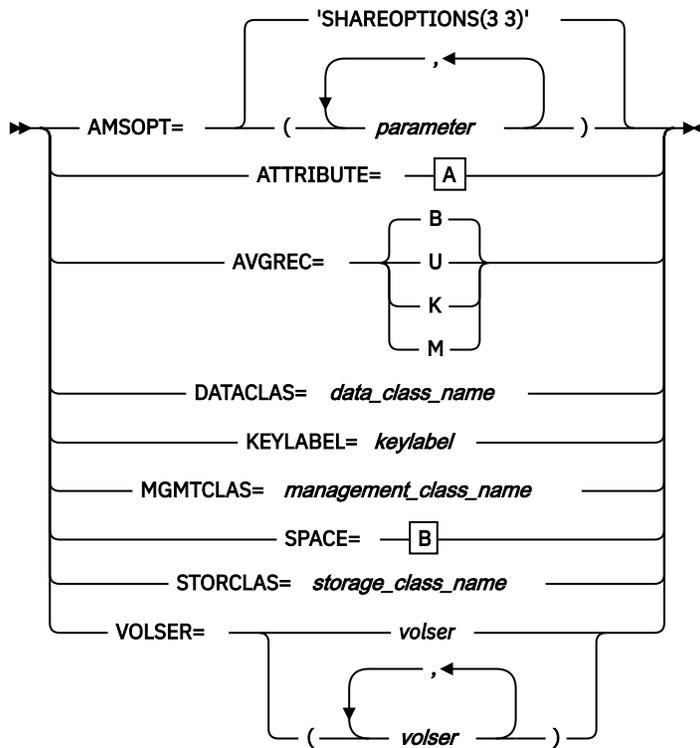
Format:

```
ALLOCATE [keyword=parameter[,...]]
```

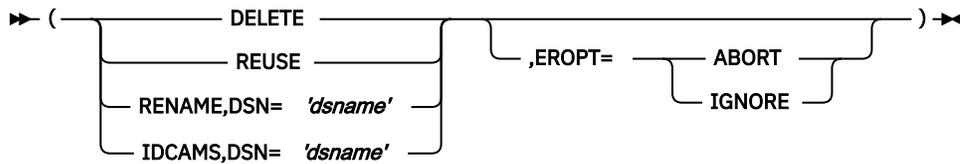
Syntax:



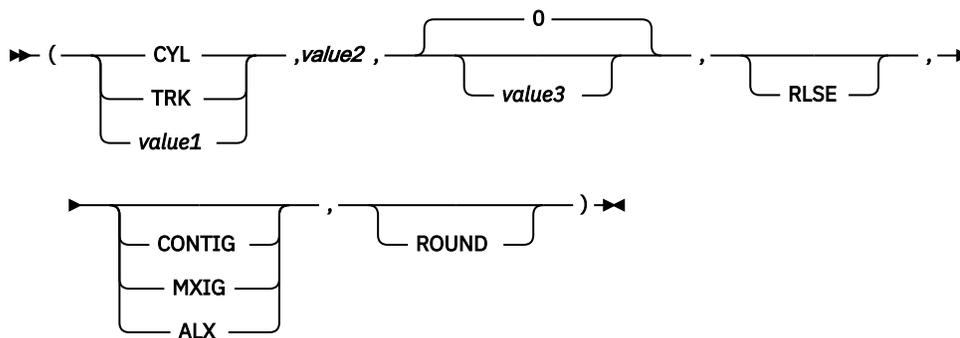
Optional keywords:



A:



B:



Default:
N/A

AREADEF subcommand

Use the `AREADEF` subcommand to define the new area definitions to simulate. If you provide new area definitions with the `AREADEF` subcommand, you do not need to customize the `ACBLIB` library.

Restriction: If you want to simulate the effect of adding or deleting areas, you cannot use the `AREADEF` subcommand to provide the new definitions. You must customize the `ACBLIB` library.

`AREADEF` is an optional subcommand for the `TUNE` command.

To specify the AREADEF subcommand, you must also specify the SIMULATE subcommand for the TUNE command.

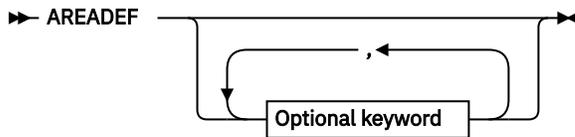
Related keywords:

- [“CISIZE keyword” on page 448](#)
- [“OAREA keyword” on page 497](#)
- [“ROOT keyword” on page 522](#)
- [“UOW keyword” on page 553](#)

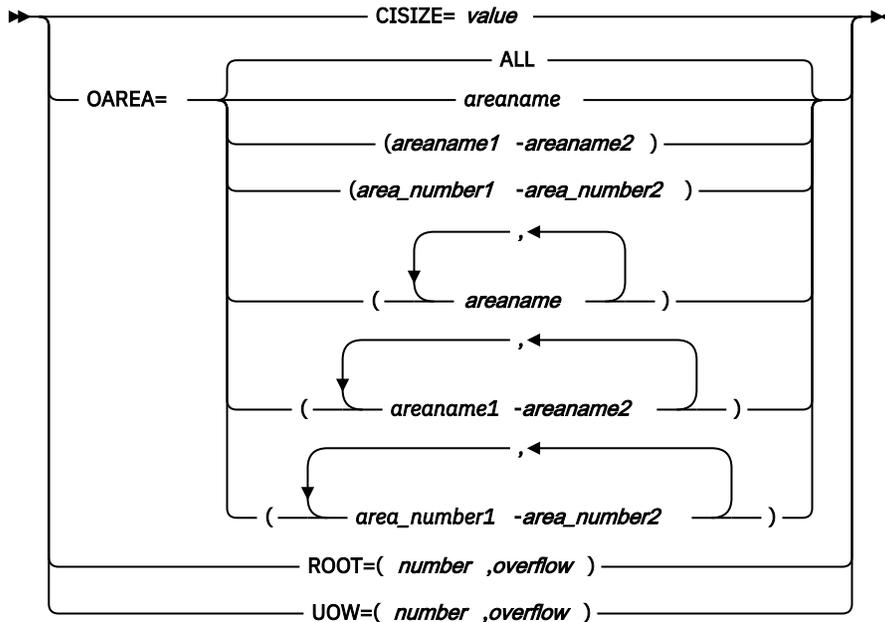
Format:

```
AREADEF [keyword=parameter[,...]]
```

Syntax:



Optional keywords:



Default:

N/A

BLDMFILECTL subcommand

Use the BLDMFILECTL subcommand to define the build pointer segment record data sets (S0nnnnn0) to create during the Build Index process. This subcommand is effective only when the Build Index function runs in scan load mode (BUILDMODE=SCANLOAD) or scan mode (BUILDMODE=SCAN).

Do not specify the BLDMFILECTL subcommand if you specify BUILDDDSNMASK keyword for the INDEXBLD command.

BLDMFILECTL is an optional subcommand for the INDEXBLD command.

Related keywords:

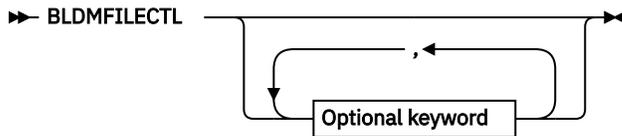
- [“AVGREC keyword” on page 445](#)
- [“DATACLAS keyword” on page 449](#)

- [“DISP keyword” on page 453](#)
- [“DSNAME keyword” on page 456](#)
- [“EXPDT keyword” on page 462](#)
- [“INDEXDBD keyword” on page 482](#)
- [“LIKE keyword” on page 492](#)
- [“MGMTCLAS keyword” on page 496](#)
- [“RETPD keyword” on page 519](#)
- [“SPACE keyword” on page 543](#)
- [“STORCLAS keyword” on page 546](#)
- [“UNIT keyword” on page 552](#)
- [“VOLCNT keyword” on page 556](#)
- [“VOLSER keyword” on page 556](#)

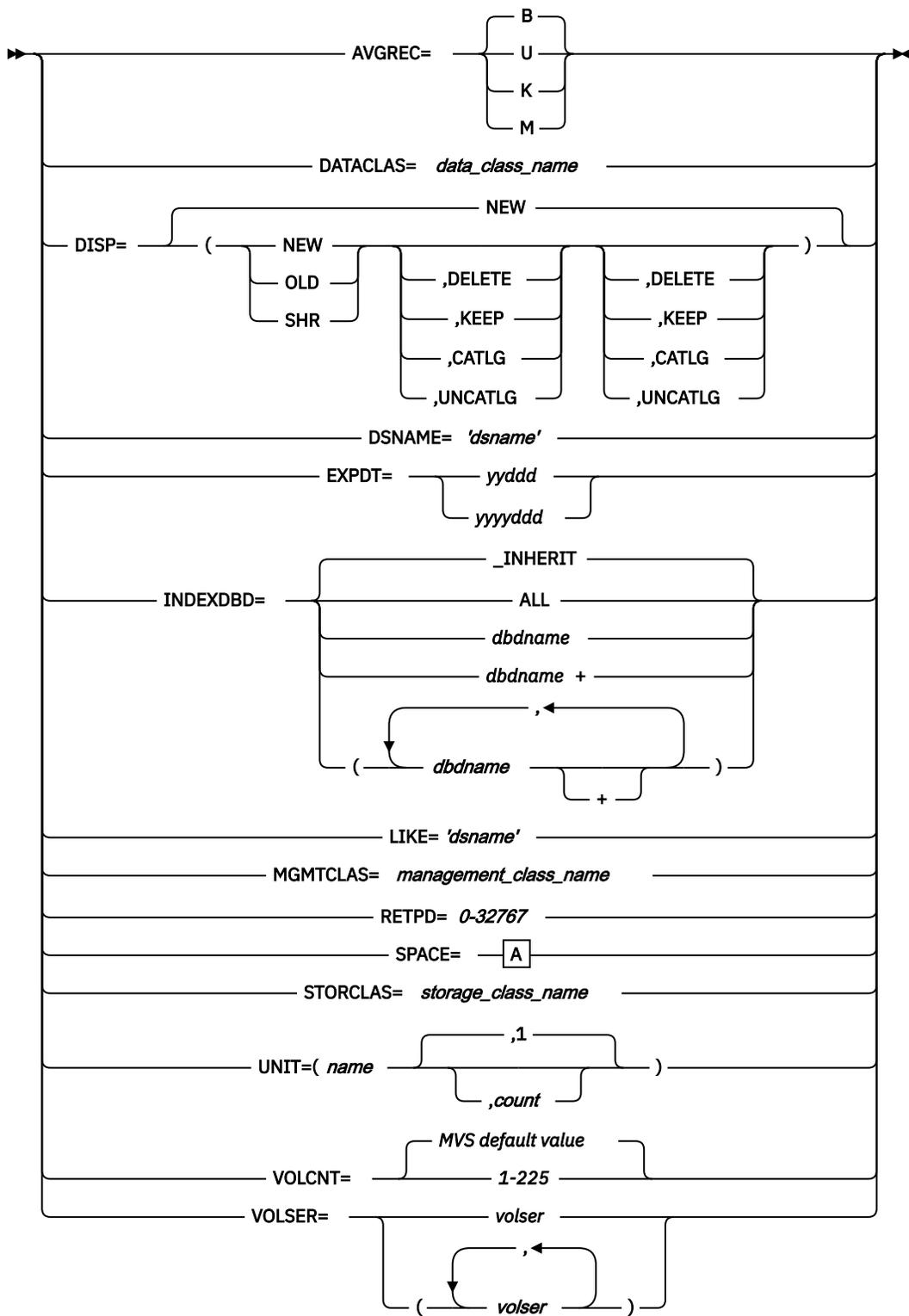
Format:

```
BLDMFILECTL [keyword=parameter[,...]]
```

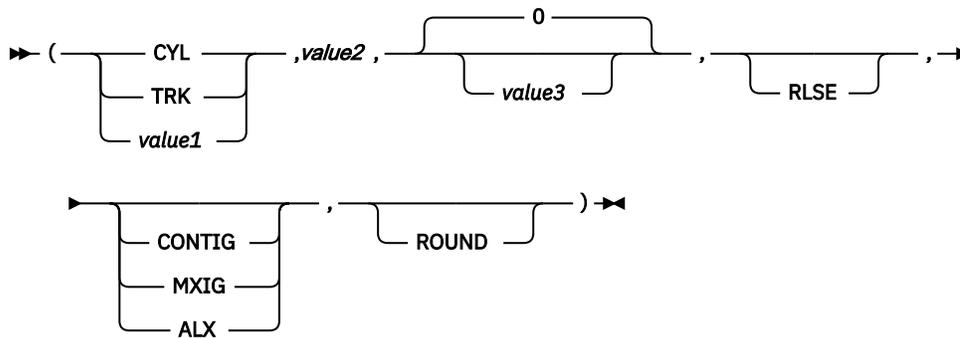
Syntax:



Optional keywords:



A:



Default:
N/A

FILECTL subcommand

Use the FILECTL subcommand to define the output files to be created during the Unload, Extract, Tune, or the Build Index process.

Usage note for the Unload or the Extract function

You can specify this subcommand to group the targeted data that are in one or more output areas into a single output data set. One or more output data sets are produced by each FILECTL subcommand that is specified.

Specify the OAREA keyword on the FILECTL subcommand to specify the output areas to be placed into the output data set. The same area name cannot be specified on multiple OAREA keywords on the FILECTL subcommand. When DDNAME is specified in FILECTL, keywords that are related to dynamic allocation (DISP, SPACE and UNIT) are ignored.

Do not specify the OAREA or the ODSNMASK keyword on the FILECTL subcommand when you have it specified on the UNLOAD or the EXTRACT command.

Usage note for the Build Index function

You can define the duplicate pointer segment record data sets (S0nnnnnR) if you specify both the FILECTL subcommand and the DUPKEYACT=SAVE keyword.

Usage note for the Tune function

You can define the tuning aid record data sets (TARnnnnn) if you specify both the FILECTL subcommand and the AREASCAN=YES keyword. Use the IAREA keyword to specify the areas where the tuning aid record data sets are created.

Do not specify the FILECTL subcommand when one or more of the following conditions apply:

- The TARDSNMASK keyword is specified for the TUNE command.
- The RESYNCDNSNMASK keyword is specified for the INDEXBLD command.
- The DUPKEYDSNMASK keyword is specified for the INDEXBLD command.

FILECTL is an optional subcommand for the following commands:

- EXTRACT
- INDEXBLD
- TUNE
- UNLOAD

Related keywords:

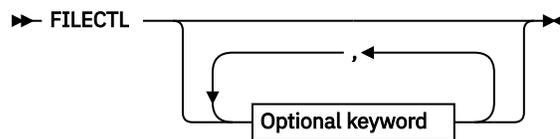
- [“AVGREC keyword” on page 445](#)
- [“DATACLAS keyword” on page 449](#)
- [“DDNAME keyword” on page 453](#)
- [“DISP keyword” on page 453](#)

- [“DSNAME keyword” on page 456](#)
- [“EXPDT keyword” on page 462](#)
- [“IAREA keyword” on page 469](#)
- [“INDEXDBD keyword” on page 482](#)
- [“LIKE keyword” on page 492](#)
- [“MGMTCLAS keyword” on page 496](#)
- [“OAREA keyword” on page 497](#)
- [“RETPD keyword” on page 519](#)
- [“SPACE keyword” on page 543](#)
- [“STORCLAS keyword” on page 546](#)
- [“UNIT keyword” on page 552](#)
- [“VOLCNT keyword” on page 556](#)
- [“VOLSER keyword” on page 556](#)

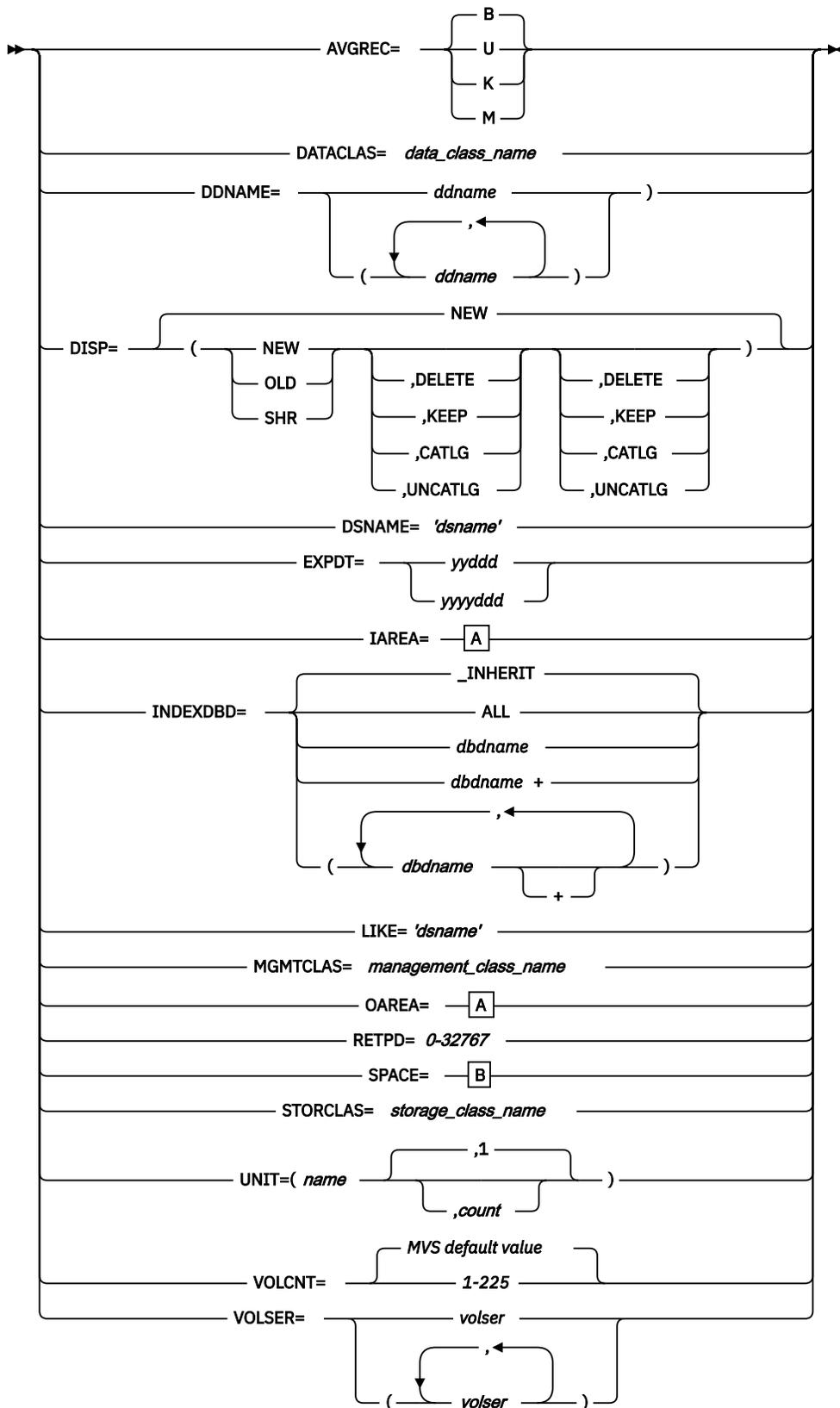
Format:

```
FILECTL [keyword=parameter[,...]]
```

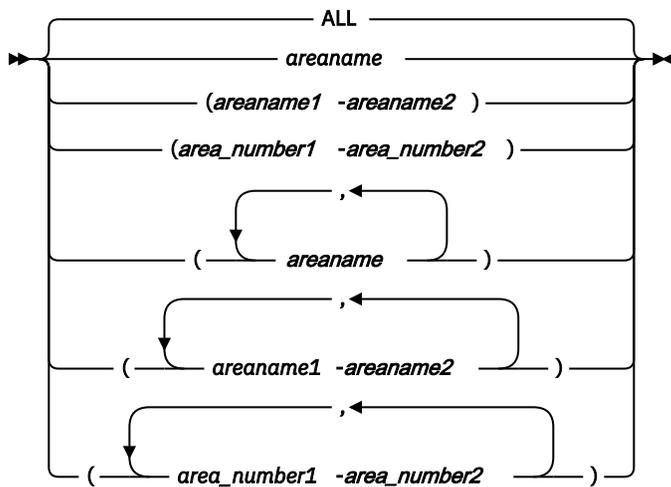
Syntax:



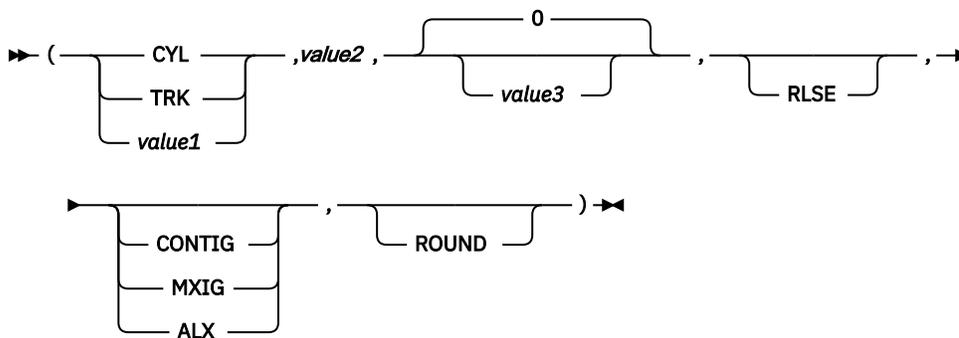
Optional keywords:



A:



B:



Default:
N/A

IRENAME subcommand

Use the IRENAME subcommand to rename the input area data set that is used for the processing of the FPA command.

Note: FPA invokes IDCAMS to rename the input area data set. You might not be able to rename the input area data set if you are changing the high-level qualifiers of the data set name and those qualifiers are an alias name of a catalog. For details, see *DFSMS Access Method Services for the Integrated Catalog Facility*.

IRENAME is an optional subcommand for the CHANGE command.

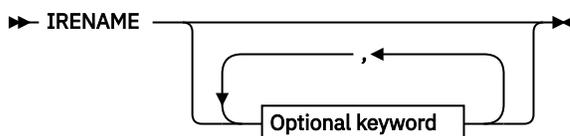
Related keywords:

- [“IAREA keyword” on page 469](#)
- [“DSNAME keyword” on page 456](#)

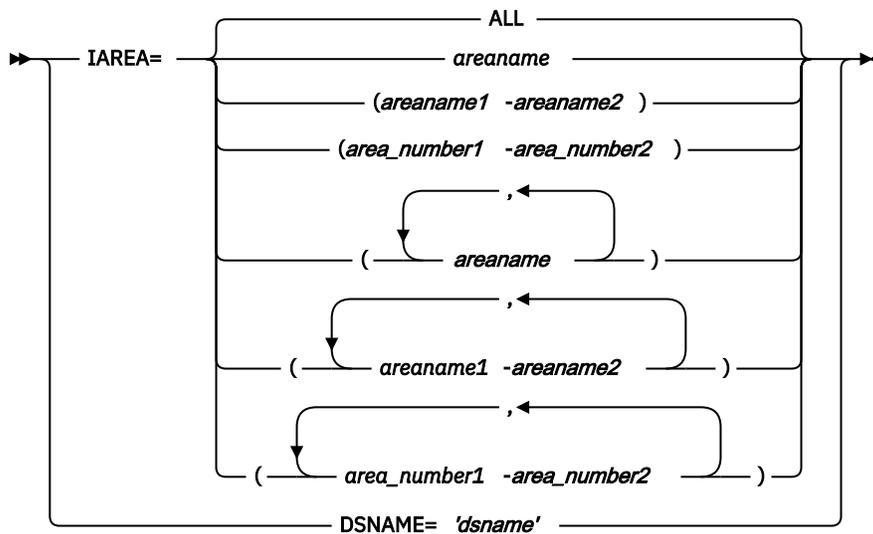
Format:

```
IRENAME [keyword=parameter[,...]]
```

Syntax:



Optional keywords:



Default:
N/A

LOADCTL subcommand

Use the LOADCTL subcommand to specify the load control information.

You can use this subcommand to insert dependent segments into the place that is specified by the LOCATION keyword, which is either DOVF or IOVF. Specify an Insert Limit Count (ILC) keyword for the maximum number of occurrences of segments that are inserted close to their root.

Performance considerations:

Database performance can be adversely affected by long dependent segment twin chains. The normal segment insertion strategy, whether loading or reorganizing a database, results in the insertion into the RAP CI, DOVF, or IOVF of all the dependent segments for a given root.

If long dependent segment twin chains are present, the subsequent database records are likely to be placed in the DOVF or IOVF part of the database. This situation tends to increase the number of I/Os that are required to access these subsequent database records.

Segment placement control using the LOADCTL subcommand preserves space in the RAA blocks (or DOVF) for more active data.

Note: The RBA values for segment allocation differ due to design differences when the LOADCTL subcommand is specified on the RELOAD command and when the LOADCTL control statement is specified with FPB Unload.

LOADCTL is an optional subcommand for the following commands:

- CHANGE
- RELOAD
- REORG

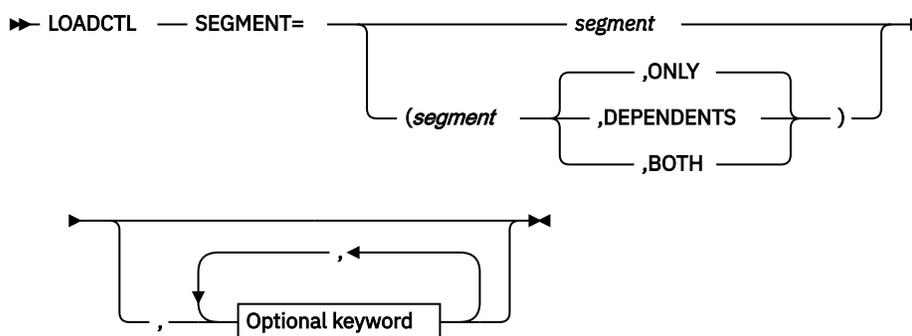
Related keywords:

- [“INSERT_LIMIT_COUNT keyword” on page 484](#)
- [“LOCATION keyword” on page 494](#)
- [“SEGMENT keyword” on page 530](#) (required keyword)

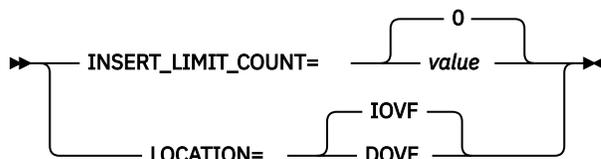
Format:

```
LOADCTL [keyword=parameter[,...]]
```

Syntax:



Optional keywords:



Default:

N/A

Examples of LOADCTL subcommand

The LOADCTL statement can be specified as follows:

Example 1:

```
LOADCTL SEGMENT=(SEGA,ONLY),LOCATION=IOVF,INSERT_LIMIT_COUNT=10
LOADCTL SEGMENT=(SEGA,ONLY),LOCATION=DOVF,INSERT_LIMIT_COUNT=5
```

- Because the first five SEGAs in a DEDB record do not exceed the INSERT_LIMIT_COUNT in the previous two LOADCTL statements, they are inserted ordinarily.
- Because the sixth to tenth SEGAs do not exceed the INSERT_LIMIT_COUNT in the first LOADCTL statement, they are checked with the INSERT_LIMIT_COUNT conditions of the second LOADCTL statement. If they exceed the INSERT_LIMIT_COUNT, the segments are inserted into DOVF.
- Because SEGAs after the 11th do exceed the INSERT_LIMIT_COUNT of the first LOADCTL statement, they are inserted into IOVF.

Example 2:

```
LOADCTL SEGMENT=(SEGA,ONLY),LOCATION=IOVF,INSERT_LIMIT_COUNT=10
LOADCTL SEGMENT=(SEGA,DEPENDENTS),LOCATION=DOVF,INSERT_LIMIT_COUNT=5
```

- Because the first five SEGAs do not exceed the INSERT_LIMIT_COUNT in the previous two LOADCTL statements, they are inserted ordinarily.
- Because the sixth to tenth SEGAs do not exceed the INSERT_LIMIT_COUNT in the first LOADCTL statement, they are checked with the INSERT_LIMIT_COUNT conditions of the second LOADCTL statement. If they exceed the INSERT_LIMIT_COUNT, the dependent segments are inserted into DOVF.
- Because SEGAs after the 11th do exceed the INSERT_LIMIT_COUNT of the first LOADCTL statement, they are again inserted into IOVF. The dependent segments of those segments are inserted ordinarily.

OUTPUT subcommand

Use the OUTPUT subcommand to select the output and to specify what data from a selected segment is to be written to the output file.

OUTPUT is an optional subcommand for the EXTRACT command.

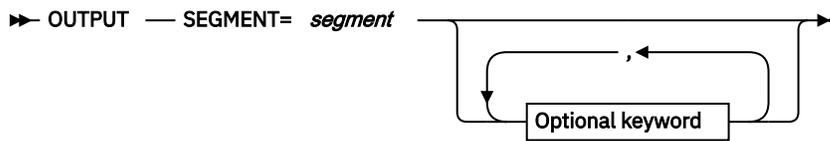
Related keywords:

- “FIELDS keyword” on page 463
- “SEGMENT keyword” on page 530 (required keyword)
- “WHERE keyword” on page 557

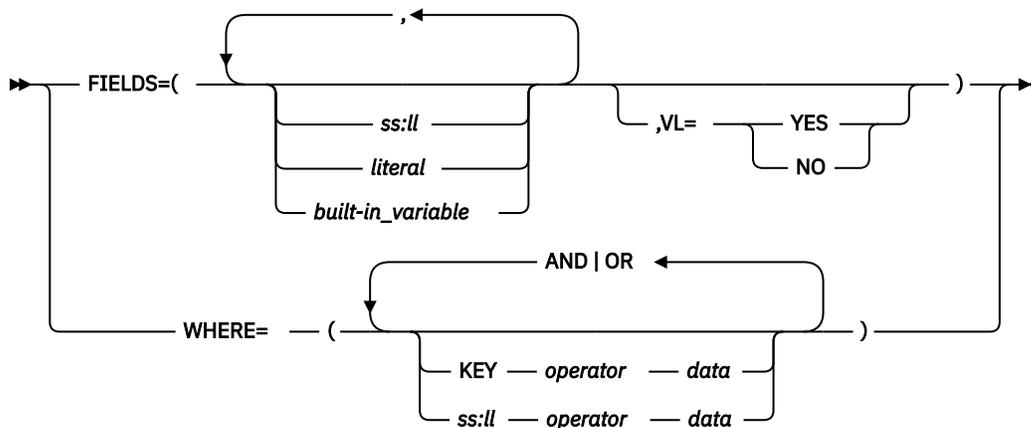
Format:

```
OUTPUT [keyword=parameter[,...]]
```

Syntax:



Optional keywords:



Default:

If the OUTPUT subcommand is not specified, one record is written for each selected segment.

Examples of OUTPUT subcommand

The OUTPUT subcommand can be specified as follows:

Example 1:

```
OUTPUT SEGMENT=SEGA , FIELDS=(1:10, 15:20)
OUTPUT SEGMENT=SEGB, FIELDS=(5:200)
```

In this example, only segments SEGA and SEGB will be included in the output. Data beginning at offset 1 for a length of 10, and offset 15 for a length of 20 will be included in SEGA. Data beginning at offset 5 for a length of 200 will be included in SEGB.

Example 2:

```
OUTPUT SEGMENT=SEGA, WHERE=(1:11 EQ C'Engineering' AND 15:5 GT P'10001216')
```

In this example, segments with the segment name SEGA will be written to the output file when:

- The character string 'Engineering' is found at position 1 for a length of 11

- The number at position 15 for a length of 5 is greater than the number '10001216' in packed decimal

REPORT subcommand

Use the REPORT subcommand to request specific reports.

The REPORT subcommand requests printing of the following reports:

- DEDB Area Analysis reports
- UOW report
- UOW Exception report
- Largest Database Records report
- Consolidated Largest Database Records report
- DB Record Length Distribution report
- Segment Length Distribution report
- RBA of Used IOVF CI report

REPORT is an optional subcommand for the following commands:

- ANALYZE
- CHANGE
- RELOAD
- REORG
- TUNE

If you specify the ANALYZE, CHANGE, RELOAD, or REORG command without the PTRCHKLVL=FULL or the PTRCHKLVL=QUICK keyword, the REPORT subcommand is ignored.

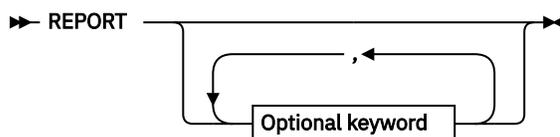
Related keywords:

- [“DBRLDIST keyword” on page 452](#)
- [“IOVFCI_RBA_REPORT keyword” on page 484](#)
- [“LARGEREC_COUNT keyword” on page 491](#)
- [“LARGEREC_REPORT keyword” on page 491](#)
- [“REPORTDD keyword” on page 513](#)
- [“SEGLDIST keyword” on page 529](#)
- [“SEGLSTAT keyword” on page 529](#)
- [“STARTUOW keyword” on page 545](#)
- [“STOPUOW keyword” on page 546](#)
- [“UOW_REPORT keyword” on page 555](#)

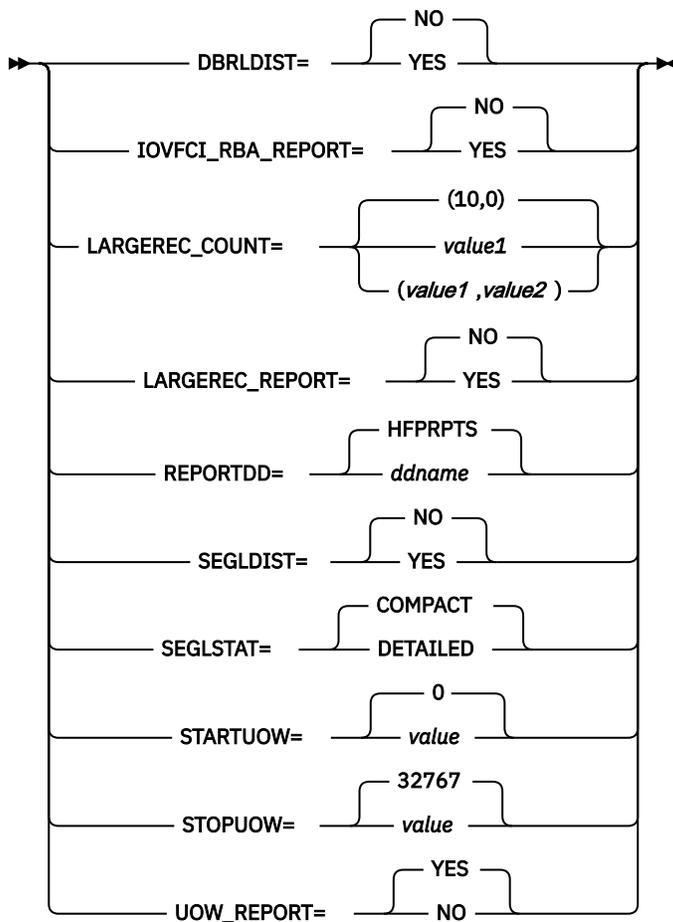
Format:

```
REPORT [keyword=parameter]
```

Syntax:



Optional keywords:



Default:
N/A

SELECT subcommand

Use the SELECT subcommand to specify the segment name and the selection criteria for the segment. SELECT is an optional subcommand for the EXTRACT command.

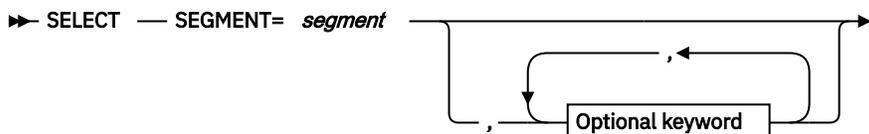
Related keywords:

- [“EVERY keyword” on page 461](#)
- [“SEGMENT keyword” on page 530 \(required keyword\)](#)
- [“SKIP keyword” on page 540](#)
- [“STOPAFT keyword” on page 545](#)
- [“WHERE keyword” on page 557](#)

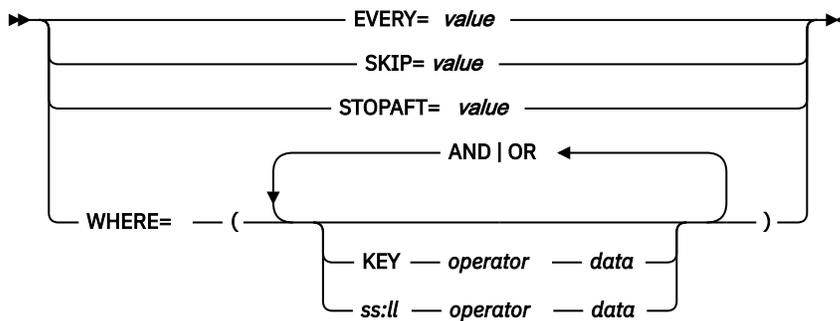
Format:

```
SELECT [keyword=parameter[,...]]
```

Syntax:



Optional keywords:



Default:

If the SELECT subcommand is not specified, all segments are selected.

Examples of SELECT subcommand

The SELECT subcommand can be specified as follows:

Example 1:

```
SELECT SEGMENT=SEGA,WHERE=(1:11 EQ C'Engineering' AND 15:5 GT P'10001216')
```

In this example, segments with the segment name SEGA will be retrieved when:

- The character string 'Engineering' is found at position 1 for a length of 11
- The number at position 15 for a length of 5 is greater than the number '10001216' in packed decimal.

Example 2:

```
SELECT SEGMENT=SEGA,STOPAFT=10,SKIP=5,EVERY=12
```

In this example, the SELECT control criteria will be applied only to SEGA. Before processing, five matching occurrences of SEGA will be skipped under each parent. During subsequent processing, every 12th occurrence of SEGA under each parent will be processed until 10 matching occurrences have been processed.

SIMULATE subcommand

Use the SIMULATE subcommand to provide resources for simulating databases.

SIMULATE is an optional subcommand for the TUNE command.

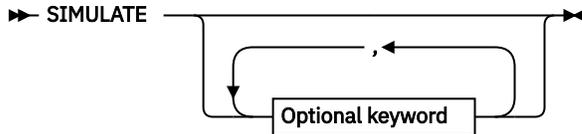
Related keywords:

- [“ACBLIB keyword” on page 439](#)
- [“IMSCATACB_OUTPUT keyword” on page 481](#)
- [“OAREA keyword” on page 497](#)
- [“REPORTDD keyword” on page 513](#)
- [“RMOD keyword” on page 521](#)
- [“RMODLIB keyword” on page 521](#)

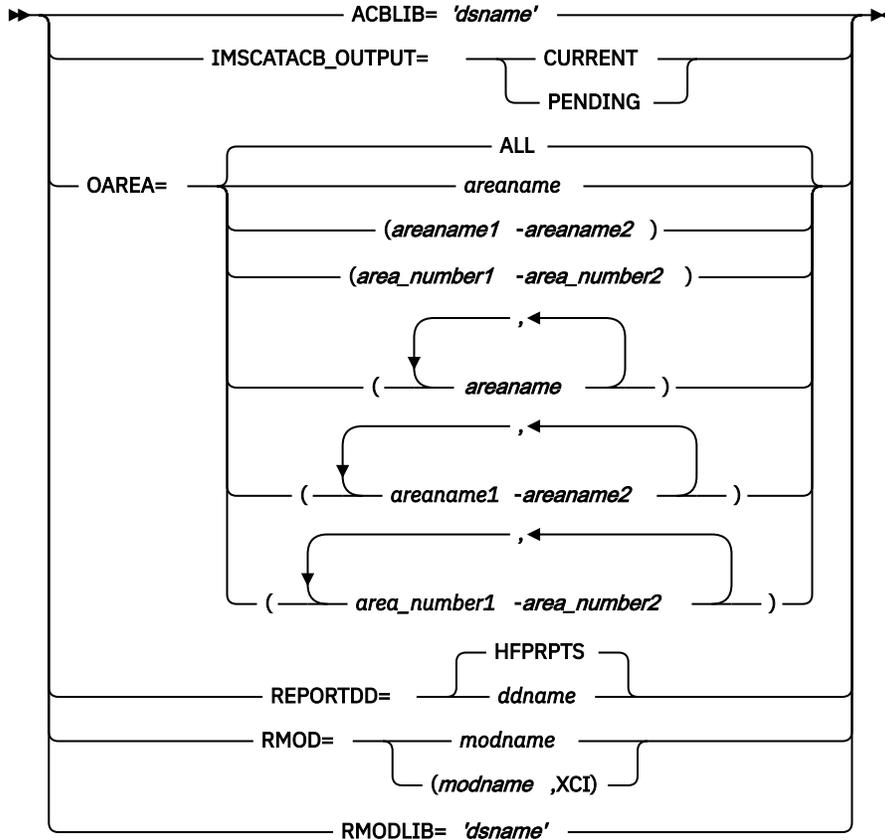
Format:

```
SIMULATE [keyword=parameter[,...]]
```

Syntax:



Optional keywords:



Default:
N/A

THRESHOLD subcommand

Use the THRESHOLD subcommand to specify the threshold parameter values, which issue warning messages when the actual values do not meet the specified criteria.

THRESHOLD is an optional subcommand for the following commands:

- ANALYZE
- CHANGE
- RELOAD
- REORG
- TUNE

If you specify the ANALYZE, CHANGE, RELOAD, or REORG command without the PTRCHKLV=FULL keyword, the THRESHOLD subcommand is ignored.

Except for the TUNE command, if you do not specify the REPORT subcommand, the THRESHOLD subcommand is ignored.

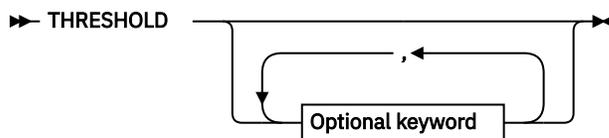
Related keywords:

- [“DOVFFS_PCT keyword” on page 455](#)
- [“FS_DOVFIOVF keyword” on page 466](#)
- [“FS_RAADOVF keyword” on page 467](#)
- [“FS_RAADOVFIOVF keyword” on page 467](#)
- [“FS_RAAIOVF keyword” on page 468](#)
- [“IOVFFS_PCT keyword” on page 485](#)
- [“IOVFUSED_PCT keyword” on page 485](#)
- [“RAAFS_PCT keyword” on page 507](#)
- [“RAP_LONGSYN_PCT keyword” on page 507](#)
- [“RAPOVFL_PCT keyword” on page 508](#)
- [“RECIO_AVG keyword” on page 510](#)
- [“RECIO_MAX keyword” on page 511](#)
- [“RECIOVF_PCT keyword” on page 510](#)
- [“ROOTIO_AVG keyword” on page 522](#)
- [“ROOTIO_MAX keyword” on page 523](#)
- [“SYN_LEN_AVG keyword” on page 547](#)
- [“SYN_MAX keyword” on page 548](#)
- [“SYN_PCT_LEN keyword” on page 548](#)
- [“UOWDOVF_PCT keyword” on page 553](#)
- [“UOWIOVF_AVG keyword” on page 554](#)
- [“UOWIOVF_MAX keyword” on page 554](#)
- [“UOWIOVF_PCT keyword” on page 555](#)

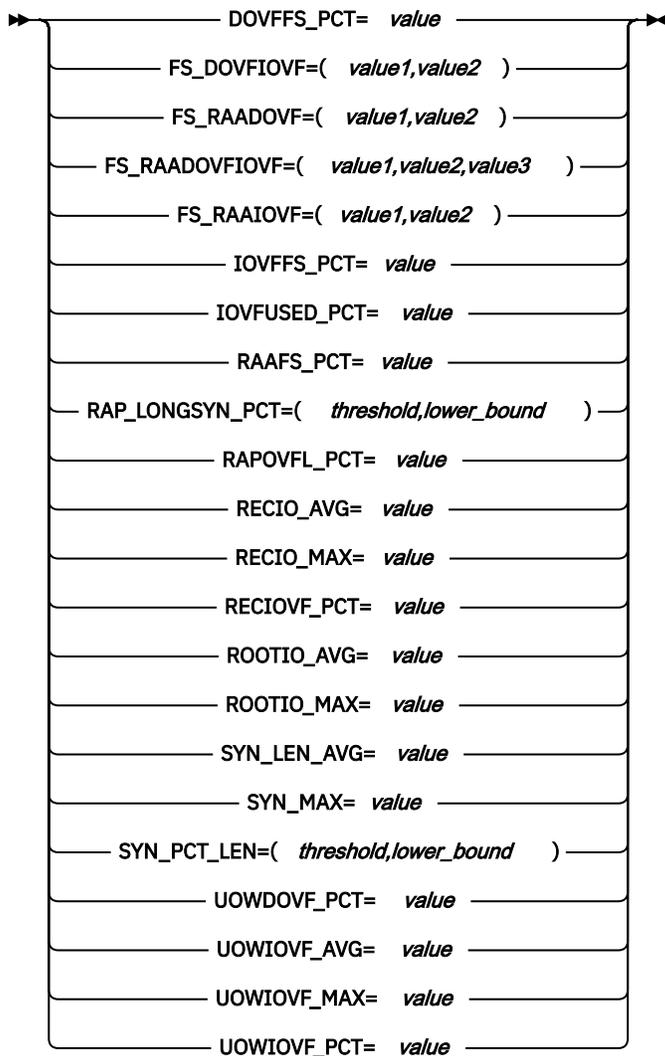
Format:

```
THRESHOLD [keyword=parameter[, ...]]
```

Syntax:



Optional keywords:



Default:
N/A

USER_RECORD subcommand

Use the USER_RECORD subcommand to specify the user-defined header and the trailer records that are to be written to an extract or an unload file.

This keyword is valid only when EFORMAT=USER or FORMAT=USER is specified.

USER_RECORD is an optional subcommand for the following commands:

- EXTRACT
- UNLOAD

Related keywords:

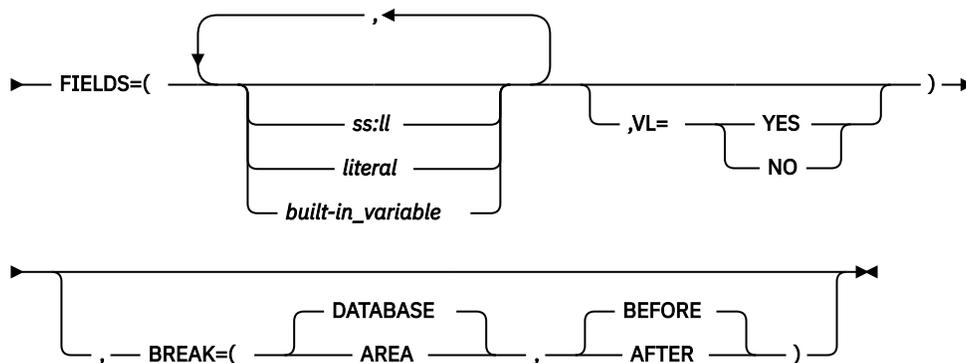
- [“BREAK keyword” on page 446](#)
- [“FIELDS keyword” on page 463](#) (required keyword)

Format:

```
USER_RECORD [keyword=parameter[,...]]
```

Syntax:

► USER_RECORD →



Default:

None.

FPA keywords

The following topics explain the FPA keywords.

ACBCHK keyword

Use the ACBCHK keyword to specify whether to check the ACB during the reload job.

For the UNLOAD command, this keyword specifies to write the DBD information in the USR files. For the RELOAD command, this keyword specifies to check the DBD information in the library that is specified on the IMSACB DD statement or in the IMS directory that is specified by the IMSCATHLQ keyword with the DBD information in the USR files.

ACBCHK is an optional keyword for the following commands:

- RELOAD
- UNLOAD

Related keywords:

[“FORMAT keyword” on page 465](#)

Format:



YES

- When ACBCHK=YES is specified for the UNLOAD command, the unload process writes DBD information records into the USR files.
- When ACBCHK=YES is specified for the RELOAD command, the reload process checks the consistency between the DBD information in the library that is specified on the IMSACB DD statement or in the IMS directory that is specified by the IMSCATHLQ keyword and the DBD information in the USR files that were created during the Unload process.

NO

- When ACBCHK=NO is specified for the UNLOAD command, no DBD information record is written into the USR files.
- When ACBCHK=NO is specified for the RELOAD command, the Reload process does not check the ACB.

Default:

ACBCHK=NO

Site default support:

Supported.

Keyword abbreviation:

Supported.

Notes:

- When both ACBCHK=YES and FORMAT=USER are specified for the UNLOAD command or the RELOAD command, ACBCHK=YES is ignored.
- When ACBCHK=NO is specified or the ACBCHK keyword is not specified for the RELOAD command, and when the DBD information records are included in the USR file, the Reload process skips the DBD information records and continues processing.

ACBLIB keyword

Use the ACBLIB keyword to specify the name of the data set that contains the DMB that describes the database configuration to simulate. Normally, use a test library and not IMSVS.ACBLIB.

If the ACBLIB keyword is not specified, the DMB is obtained from the IMSACB DD data set.

The ACBLIB keyword cannot be specified with the IMSCATACB_OUTPUT keyword.

ACBLIB is an optional keyword for the SIMULATE subcommand.

Related keywords:

None.

Format:

```
➤ ACBLIB= 'dsname' ➤
```

dsname

Specify a 1- to 44-character data set name. Enclose the *dsname* in single quotation marks.

Default:

If the IMSCATHLQ_OUTPUT keyword is not specified, the ACB library that the IMSACB DD statement specifies.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

ADXCFGRP keyword

Use the ADXCFGRP keyword to specify the Autonomics Director XCF group name. The name is used to send a sensor data notification to Autonomics Director when the sensor data is stored in the Sensor Data repository of IMS Tools KB.

ADXCFGRP is an optional keyword for the GLOBAL command.

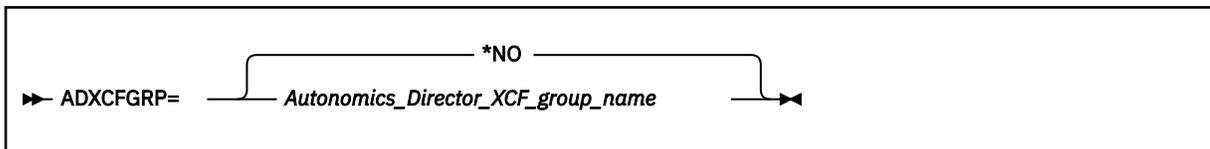
This keyword is used only when SENSOR=YES is specified.

The sensor data notification is needed for Autonomics Director to schedule a follow-on action of evaluating sensor data. Consider specifying the Autonomics Director XCF group name if you want Autonomics Director to schedule the follow-on action.

Related keywords:

[“SENSOR keyword” on page 535](#)

Format:



Autonomics_Director_XCF_group_name

Send a sensor data notification to Autonomics Director of the specified XCF group name.

If you specify the Autonomics Director XCF group, you must specify DBRC=YES.

***NO**

Do not send a sensor data notification to Autonomics Director.

Default:

ADXCFGRP=*NO

Site default support:

Supported.

Keyword abbreviation:

Supported.

ALLOCONLY keyword

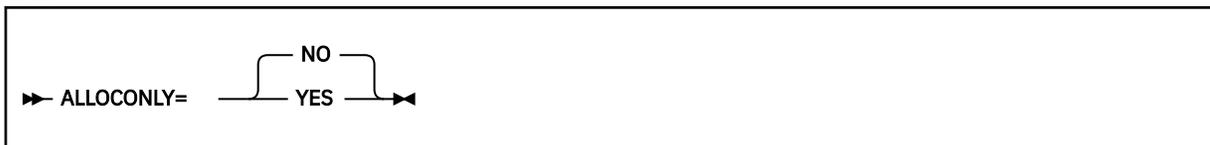
Use the ALLOCONLY keyword to suppress the initialization of output area data sets.

ALLOCONLY keyword is an optional keyword for the INITIALIZE command.

Related keywords:

None.

Format:



YES

Do not initialize the allocated output area data sets. If the ALLOCATE subcommand is not specified, ALLOCONLY=YES is ignored.

NO

Initialize the allocated output area data sets.

Default:

ALLOCONLY=NO

Site default support:

Supported.

Keyword abbreviation:

Not supported.

AMSOPT keyword

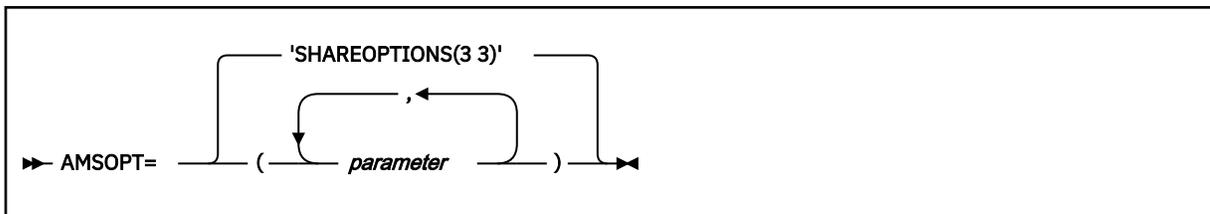
Use the AMSOPT keyword to pass optional parameters to the Access Method Services (AMS) to define VSAM clusters.

AMSOPT is an optional keyword for the ALLOCATE subcommand.

Related keywords:

None.

Format:



parameter

Many parameters are available for use with this keyword to enable you to control the values that are used by AMS to define VSAM clusters. For more information, see *DFSMS Access Method Services for the Integrated Catalog Facility*.

A parameter that contains blanks or other special characters must be enclosed in either single or double quotation marks.

Default:

AMSOPT='SHAREOPTIONS(3 3)'

Site default support:

Not supported.

Keyword abbreviation:

Supported.

AREADSN keyword

Use the AREADSN keyword to specify how to obtain area data set names.

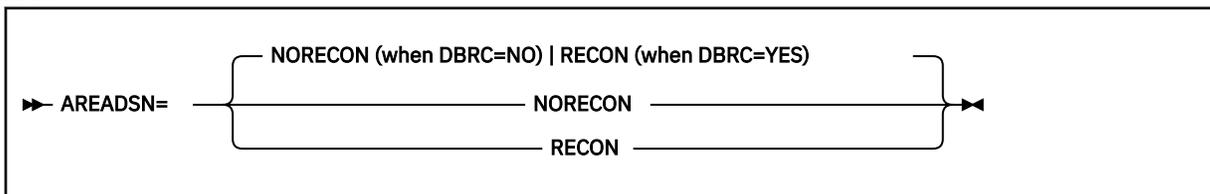
AREADSN is an optional keyword for the GLOBAL command.

AREADSN keyword is effective only when the ANALYZE or the TUNE command is specified.

Related keywords:

None.

Format:



NORECON

The area data set names are obtained from *addnname/areaname* DD statements, or from the specification of the utility control statements.

AREADSN=NORECON cannot be specified when DBRC=YES is specified.

RECON

The area data set names are obtained from the DBRC RECON data sets.

When DBRC=NO and AREADSN=RECON are specified, FPA does not issue authorization and unauthorization calls to DBRC but instead FPA issues sign-on and sign-off calls to DBRC to obtain the area data set names.

Default:

- When DBRC=NO is specified, AREADSN=NORECON.
- When DBRC=YES is specified, AREADSN=RECON.

Site default support:

Supported.

Keyword abbreviation:

Supported.

Related reference[DD statements for the Analyze function](#)

DD statements of the FPA Analyze function determine the input and output data sets and specify how to run the Analyze function.

[DD statements for the Tune function](#)

DD statements of the FPA Tune function determine the input and output data sets and specify how to run the Tune function.

AREASCAN keyword

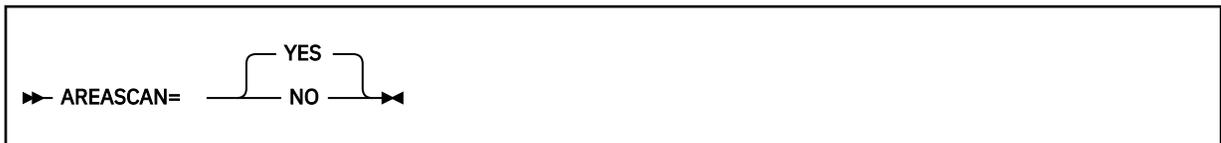
Use the AREASCAN keyword to specify whether to scan the DEDB areas during the FPA job.

AREASCAN is an optional keyword for the following commands:

- INDEXBLD
- TUNE

Related keywords:

- [“RESYNCDSDNMASK keyword” on page 515](#)
- [“TARDSNMASK keyword” on page 549](#)
- [“TARDSNMASK keyword” on page 932](#)
- [“TARECGEN keyword” on page 933](#)

Format:**YES**

Scan the DEDB areas.

For the TUNE command, if you specify AREASCAN=YES, the Tune function generates the tuning aid records and also does a cross-reference validation of pointers between the root and direct dependent segments. This validation is the same as the validation performed when PTRCHKLVL=FULL and SDEPCHKLVL=NONE are specified for the Analyze process.

NO

Do not scan the DEDB areas.

For the INDEXBLD command, AREASCAN=NO is effective only when you also specify RESYNCMODE=VERIFYUPDATE or RESYNCMODE=DIRECTUPDATE. You must also specify the unverified resync pointer segment records that were generated in a previous run of the FPA Resync job with the RESYNCMODE=AREASCAN option.

For the TUNE command, you must specify the tuning aid records that were generated in a previous run of the FPA Tune job or in an FPO job with TARECGEN=YES.

Default:

AREASCAN=YES

Site default support:

Supported.

Keyword abbreviation:

Supported.

AREA_INFORMATION_RECORD keyword

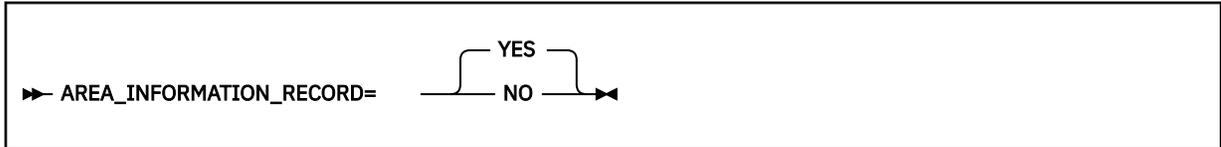
Use the AREA_INFORMATION_RECORD keyword to include the area information record in the unloaded segment record data set.

AREA_INFORMATION_RECORD is an optional keyword for the UNLOAD command.

Related keywords:

None.

Format:



YES

The area information record is included in the unloaded segment record data set. If FORMAT=USER is specified for the UNLOAD command, this parameter is ignored.

NO

The area information record is not included in the unloaded segment record data set.

Default:

AREA_INFORMATION_RECORD=YES

Site default support:

Supported.

Alias:

AIR

Keyword abbreviation:

Supported.

ATASKCTL keyword

Use the ATASKCTL keyword to specify the maximum number of input areas to be analyzed concurrently.

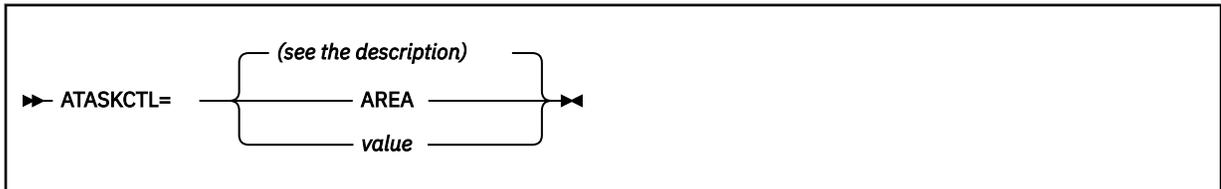
ATASKCTL is an optional keyword for the following commands:

- ANALYZE
- CHANGE
- RELOAD
- REORG

Related keywords:

None.

Format:



AREA

The number of input areas.

AREA cannot be specified for the CHANGE, RELOAD, and REORG commands when PTRCHKLVL=QUICK or PTRCHKLVL=FULL is specified.

value

Specify a value in the range of 1 - 2048.

You must specify this parameter in the following manner:

- For the ANALYZE command, specify a value less than or equal to the number of input areas.
- For the CHANGE command with PTRCHKLVL=QUICK or FULL, specify a value less than or equal to the OTASKCTL value.
- For the RELOAD and REORG commands with PTRCHKLVL=QUICK or FULL, specify a value less than or equal to the ITASKCTL value.

If you specify a value that is greater than these values, it will be replaced by an appropriate value.

Default:

- For the ANALYZE command, ATASKCTL=4.
- For the CHANGE command, the OTASKCTL value.
- For the RELOAD and REORG commands, the ITASKCTL value.

Site default support:

- For the ANALYZE command, site default is supported.
- For the CHANGE, RELOAD, and REORG commands, site default is not supported.

Keyword abbreviation:

Supported.

ATTRIBUTE keyword

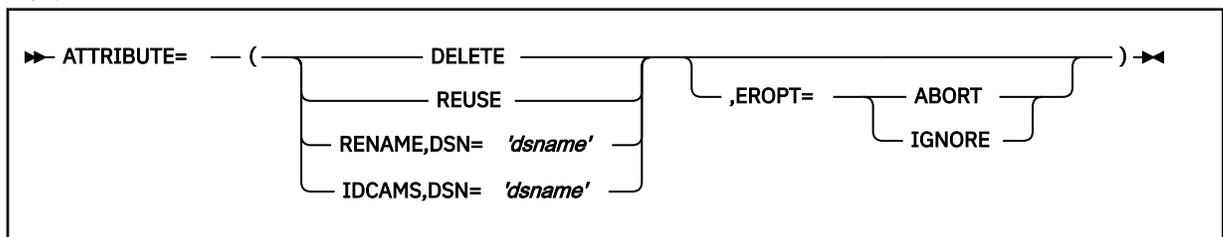
Use the ATTRIBUTE keyword to request optional processing related to allocation of a VSAM cluster that is used for the output of the ALLOCATE subcommand.

ATTRIBUTE is an optional keyword for the ALLOCATE subcommand.

Related keywords:

None.

Format:



DELETE

The current VSAM cluster is deleted before the new VSAM cluster is allocated.

REUSE

The current VSAM cluster is allocated and overwritten. If it does not exist, a new VSAM cluster is allocated.

RENAME

The current VSAM cluster is renamed before the new VSAM cluster is allocated.

IDCAMS

A sequential or partitioned data set is passed to Access Method Services as input for allocating the new VSAM cluster. Whether the new VSAM cluster is allocated or not is verified.

DSN

If you specified RENAME or IDCAMS as the primary parameter on the ATTRIBUTE keyword, you must specify a data set name mask for the DSN parameter. Enclose the data set name in single quotation marks. For details about the data set mask, read about the DSNNAME keyword.

EROPT

Specifies whether to continue the ALLOCATE subcommand processing when an error occurs during the processing that is related to allocation of a VSAM cluster requested by the ATTRIBUTE keyword. When Define Cluster command fails, FPA abends, even if EROPT=IGNORE is specified.

ABORT

Aborts the allocate function when an ATTRIBUTE process error is encountered.

IGNORE

Except when an error occurs in the Define Cluster function, the process continues the ALLOCATE function, even if an ATTRIBUTE process error occurs.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

AVGREC keyword

Use the AVGREC keyword to identify the space requirements of the output data set that is created by dynamic allocation.

This keyword is meaningful only when the allocation units (specified using the SPACE keyword) are given as a number of bytes.

AVGREC is an optional keyword for the following subcommands:

- ALLOCATE
- BLDMFILECTL
- FILECTL

Related keywords:

["SPACE keyword" on page 543](#)

Format:**B**

The allocation units that are specified by the unit subparameter of the SPACE keyword represents an average block size. This is default.

U

The allocation units that are specified by the unit subparameter of the SPACE keyword represents an average record size. The multiplier of the primary and secondary quantities is 1.

K

The allocation units that are specified by the unit subparameter of the SPACE keyword represents an average record size. The multiplier of the primary and secondary quantities is 1024.

M

The allocation units that are specified by the unit subparameter of the SPACE keyword represents an average record size. The multiplier of the primary and secondary quantities is 1048576.

Default:

AVGREC=B

Site default support:

Supported.

Keyword abbreviation:

Supported.

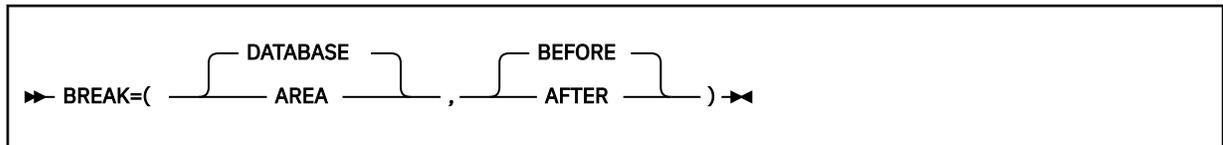
BREAK keyword

Use the BREAK keyword to specify the level and the timing for the trigger event for user-defined header and trailer of an extract or an unloaded file.

BREAK is an optional keyword for the USER_RECORD subcommand.

Related keywords:

None.

Format:

The first parameter specifies the level:

DATABASE

The trigger event occurs at the database processing level. This is the default value.

AREA

The trigger event occurs at the area processing level. When the UNLOAD command is processed in the non-REORG mode, the AREA parameter cannot be specified.

The second parameter specifies when to trigger the user record:

BEFORE

The user record is triggered when processing begins for the specified level. This is the default value.

AFTER

The user record is triggered after processing is complete for the specified level.

Default:

BREAK=(DATABASE,BEFORE)

Site default support:

Not supported.

Keyword abbreviation:

Supported.

BUILDSDSNMASK keyword

Use the BUILDSDSNMASK keyword to specify the data set mask for dynamically allocating cataloged pre-existing build pointer segment record data sets.

You can make a relative reference to a GDG by specifying a relative generation number in the mask, a number enclosed in brackets such as in '*gdg-name* (-2)'

When the data set name contains a reference to a partitioned data set (PDS) member name and it is used for dynamic allocation, you must specify DISP=SHR or OLD.

You cannot use the BLDMFILECTL subcommand if you specify the BUILDSDSNMASK keyword.

BUILDSDSNMASK is an optional keyword for the INDEXBLD command.

Related keywords:

None.

Format:

```
➤➤ BUILDDSNMASK= 'dsname' ➤➤
```

dsname

Specify a data set mask. Enclose *dsname* in single quotation marks.

Symbolic variable &INDEX must be specified in the data set mask. &INDEX defines the secondary index DBD name. For example, if you specify BUILDDSNMASK= 'HFP.&INDEX' and INDEXDBD=INDEXA, a data set named HFP.INDEXA is used for the build pointer segment record data set. &INDEX cannot be used to specify a member name of partitioned data set (PDS).

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

BUILDMODE keyword

Use the BUILDMODE keyword to specify the mode for building secondary index databases.

BUILDMODE is an optional keyword for the INDEXBLD command.

BUILDMODE keyword is effective only when IDXPROC=BUILD is specified.

Related keywords:

None.

Format:



SCANLOAD

The job runs in scan load mode. FPA scans the DEDB areas and builds secondary index databases in one job step.

SCAN

The job runs in scan mode. FPA scans the DEDB areas and then generates build pointer segment records. The records are later used in Build Index jobs that run in load mode to build the secondary indexes.

LOAD

The job runs in load mode. FPA reads the build pointer segment records that were generated in preceding Build Index jobs that ran in scan mode, and then builds secondary index databases.

Default:

BUILDMODE=SCANLOAD

Site default support:

Supported.

Keyword abbreviation:

Supported.

CISIZE keyword

Use the CISIZE keyword to simulate the areas with a new CI size.

When this keyword is specified, the Tune function replaces the CI size that is defined in the DMB in the ACBLIB data set and simulates the area condition with the new CI size. Use this keyword to tune the size of CIs.

CISIZE is an optional keyword for the AREADEF subcommand.

Related keywords:

- “ROOT keyword” on page 522
- “UOW keyword” on page 553

Format:

```
▶▶ CISIZE= value ▶▶
```

value

Specify the CI size for the area in bytes. The value can be 512, 1024, 2048, 4096, 8192, or multiples of 4096 up to 28672.

Default:

None.

Site default support:

Not supported.

Alias:

SIZE

Keyword abbreviation:

Supported.

CISNAP keyword

Use the CISNAP keyword to specify whether to generate the CI Map/CI Dump report and the Pointer Chain Reconstruction report when an incorrect pointer is detected.

CISNAP is an optional keyword for the following commands:

- ANALYZE
- CHANGE
- GLOBAL
- RELOAD
- REORG
- TUNE
- UNLOAD

Related keywords:

None.

Format:

```
▶▶ CISNAP=            NO             
                     YES           ▶▶
```

YES

Generate CI Map/CI Dump report and Pointer Chain Reconstruction report in the HFPASNAP data set.

NO

Do not generate CI Map/CI Dump report and Pointer Chain Reconstruction report.

Default:

CISNAP=NO

Site default support:

- For the ANALYZE, CHANGE, RELOAD, REORG, and UNLOAD commands, site default is supported.
- For the GLOBAL command, site default is not supported.

Keyword abbreviation:

Supported.

COMPRESS keyword

Use the COMPRESS keyword to specify whether the output record for the EXTRACT or the UNLOAD function should contain compressed segment if segment edit/compression routine is defined for the segment in DBDGEN.

If COMPRTN is not specified for SEGM in the DBD statement, this keyword is ignored. If you specify the FIELDS or the WHERE keyword for a compressed segment in the EXTRACT function, you must also specify the COMPRESS=NO keyword.

COMPRESS is an optional keyword for the following commands:

- EXTRACT
- UNLOAD

Related keywords:

None.

Format:**YES**

Specifies that output record should contain compressed segment if segment edit/compression routine is defined for the segment in DBDGEN.

NO

Specifies that output record should contain segment that is expanded if segment edit/compression routine is defined for the segment in DBDGEN.

Default:

COMPRESS=NO

Site default support:

Supported.

Keyword abbreviation:

Supported.

Related concepts

[Segment edit/compression facility support](#)

This topic helps you produce the unloaded data set records with expanded segment data from compressed segments. It contains the functional capabilities of this facility.

DATACLAS keyword

Use the DATACLAS keyword to specify the SMS data class of an output data set that was created by dynamic allocation.

For more information about SMS data class, see the *MVS JCL Reference*.

DATACLAS is an optional keyword for the following subcommands:

- ALLOCATE
- BLDMFILECTL
- FILECTL

Related keywords:

[“DSNAME keyword” on page 456](#)

Format:

```
▶▶ DATACLAS= data_class_name ◀◀
```

data_class_name

Specify a 1- to 8-character data class name.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

DBD keyword

Use the DBD keyword to identify the database description (DBD) that contains the areas to be processed.

The DBD name must correspond to the member name in the IMSACB library or in the IMS directory.

DBD is a required keyword for the following commands:

- ANALYZE
- CHANGE
- DMACPRT
- EXTRACT
- INDEXBLD
- INITIALIZE
- RELOAD
- REORG
- TUNE
- UNLOAD

Related keywords:

- [“IAREA keyword” on page 469](#)
- [“OAREA keyword” on page 497](#)

Format:

```
▶▶ DBD= dbdname ◀◀
```

dbdname

Specify a 1- to 8-character database name.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

DBRC keyword

Use the DBRC keyword to request or bypass the database recovery control (DBRC) facility process for the database that is specified by the DBD.

Processing includes DBRC sign-on and sign-off, area authorization, and event notification. To process multiple area data sets (MADS), DBRC=YES must be specified.

If you specify DBRC=YES for the CHANGE, the INITIALIZE, or the RELOAD command, the RECOV NEEDED status of the area is changed to OFF for that command.

If you specify DBRC=YES for the INDEXBLD command with the IDXPROC=BUILD option, FPA issues the NOTIFY.REORG command to DBRC only for the secondary index databases that were built successfully.

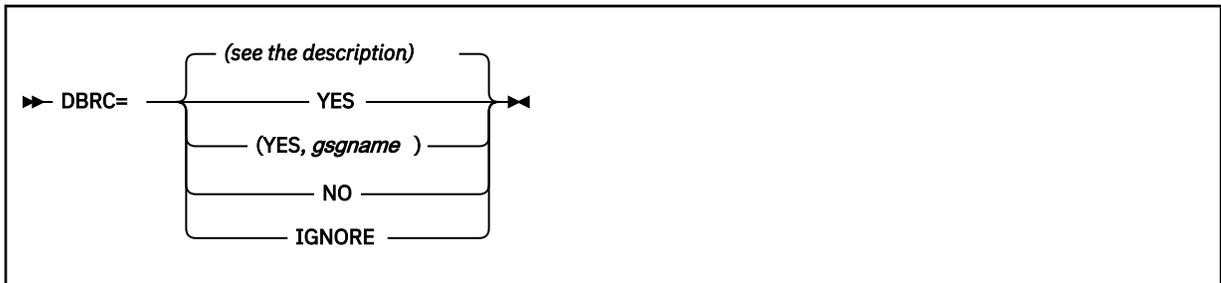
If you specify ICOUT DD, DBRC=YES, and ICNOTIFY=YES, FPA issues the NOTIFY.IC command to DBRC. This option is valid for the following commands:

- ANALYZE
- CHANGE
- REORG
- RELOAD
- UNLOAD

DBRC is an optional keyword for the GLOBAL command.

Related keywords:

"LIUGOPT keyword" on page 493

Format:**YES**

Request the DBRC process of the database.

gsgname

If FPA is running in a Remote Site Recovery (RSR) environment, specify a Global Service Group (GSG) name.

NO

Bypass the DBRC process of the database. If DBRC=FORCE is set for IMS, specifying DBRC=NO does not bypass DBRC.

IGNORE

Bypass the DBRC process of the database even if DBRC=FORCE is set for IMS.

Default:

DBRC runs under IMS control and is specified for IMS in the member DFSVC000 or DFSIDEF0 of the SDFSRESL library. The default value for the DBRC keyword is the parameter specified for IMS.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

DBREC_DISCARD_MSGLVL keyword

Use the DBREC_DISCARD_MSGLVL keyword to specify the severity level of the message that is issued when a randomized database record is discarded due to a USR file not being specified.

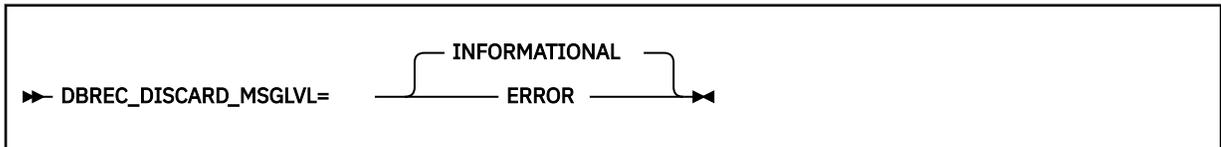
DBREC_DISCARD_MSGLVL is an optional keyword for the following commands:

- CHANGE
- UNLOAD

Related keywords:

None.

Format:



INFORMATIONAL

Issues informational message HFPJ0003I.

ERROR

Issues error message HFPJ0009E and sets the return code to 8.

When HFPJ0009E messages are issued, the following reports are not printed:

- Reloaded reports of the Change function.
- Output File reports of the Unload function.
- Analysis reports of the Change function when the job is run with the REPORT subcommand and the PTRCHKLVL keyword.
- The POINTER ERROR DETAILS section in Unloaded reports. The section describes pointer errors when PTRCHK=DETAIL is specified.

Default:

DBREC_DISCARD_MSGLVL=INFORMATIONAL

Site default support:

Supported.

Keyword abbreviation:

Supported.

DBRLDIST keyword

Use the DBRLDIST keyword to specify whether to generate the DB Record Length Distribution report.

DBRLDIST is an optional keyword for the REPORT subcommand.

Related keywords:

None.

Format:



YES

Print the DB Record Length Distribution report. The report is printed for the areas that are specified with the IAREA keyword. DBRLDIST=YES is effective only when PTRCHKLVL=FULL is specified.

NO

Do not print the DB Record Length Distribution report.

Default:

DBRLDIST=NO

Site default support:

Supported.

Keyword abbreviation:

Supported.

DDNAME keyword

Use the DDNAME keyword to specify the ddnames for the data sets to be processed by the subcommand.

If the JCL does not contain a DD statement with the ddnames that are referred to, an error message is issued. One copy of the output data set is written to each ddname that is specified.

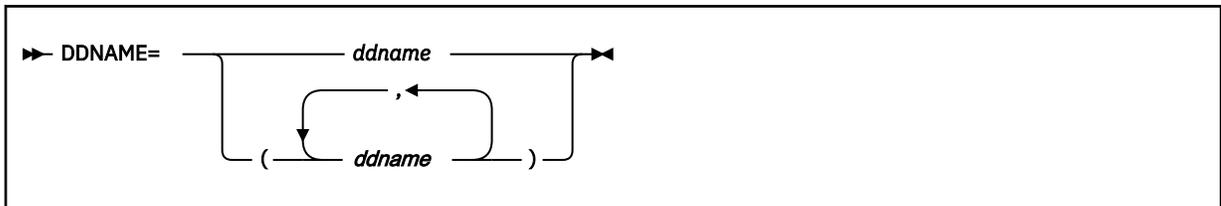
The subcommand must contain either the DSNAME or the DDNAME keyword. Do not specify both because they are mutually exclusive.

If you specify the INDEXBLD subcommand, you cannot specify the DDNAME keyword on the FILECTL subcommand.

DDNAME is an optional keyword for the FILECTL subcommand.

Related keywords:

[“DSNAME keyword” on page 456](#)

Format:**ddname**

Specify a 1- to 8-character data set name.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

DISP keyword

Use the DISP keyword to specify the status and disposition of the data set that is accessed by dynamic allocation.

For more information, see the *MVS JCL Reference*.

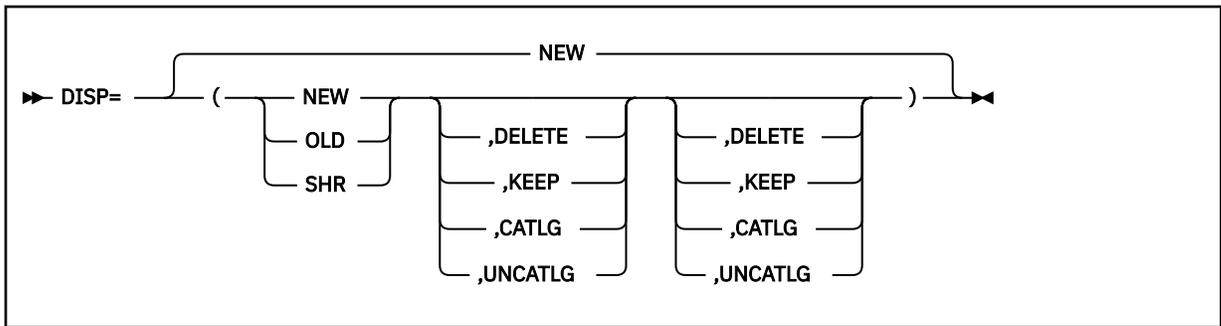
DISP is an optional keyword for the following subcommands:

- BLDMFILECTL
- FILECTL

Related keywords:

"DSNAME keyword" on page 456

Format:



The first parameter specifies the status of the data set:

NEW

Specifies to create a new data set.

OLD

Specifies to use an existing data set exclusively.

SHR

Specifies that the data set existed before this step and that it can be shared with other jobs so that it can be used by multiple jobs at the same time.

The second parameter specifies the action to take place when the data set is created successfully:

DELETE

Release the space allocated for the data set.

KEEP

Keep the data set on the volume.

CATLG

Place an entry pointing to the data set in the catalog.

UNCATLG

Remove the catalog entry to the data set, but retain the data set.

The third parameter specifies the action to occur if any error occurs while creating the data set:

DELETE

Release the space allocated for the data set.

KEEP

Keep the data set on the volume.

CATLG

Place an entry pointing to the data set in the catalog.

UNCATLG

Remove the catalog entry to the data set, but retain the data set.

Default:

`DISP=NEW`

Site default support:

Not supported.

Keyword abbreviation:

Supported.

DOVFFS_PCT keyword

Use the DOVFFS_PCT keyword to specify a threshold setting for the percentage of free space in DOVF. If the percentage of free space in DOVF is less than the value that you specified, a warning message is issued.

DOVFFS_PCT is an optional keyword for the THRESHOLD subcommand.

Related keywords:

None.

Format:

```
►► DOVFFS_PCT= value ◄◄
```

value

Specify a value in the range of 0 - 100.

Default:

None.

Site default support:

Not supported.

Alias:

DFSP

Keyword abbreviation:

Supported.

DOVFNULPTR_MSGLVL keyword

Use the DOVFNULPTR_MSGLVL keyword to specify that FPA issues a warning message if the first DOVF CI in any UOW contains a null pointer for the current overflow CI.

DOVFNULPTR_MSGLVL is an optional keyword for the following commands:

- ANALYZE
- UNLOAD

Related keywords:

None.

Format:

```
►► DOVFNULPTR_MSGLVL=  ◄◄
```

NONE

Does not issue an alert message when FPA detects a null pointer for the current overflow CI in the first DOVF CI of a UOW.

WARNING

Issues a warning message when FPA detects a null pointer for the current overflow CI in the first DOVF CI of a UOW. FPA prints the following messages, sets the return code to 4, and continues processing.

- A warning message in the Processing Messages report.
- An integrity verification message in the Resolve Pointers report. This message contains the RBA and the UOW number of the DOVF CI.

Default:

DOVFNULPTR_MSGLVL=NONE

Site default support:

Supported.

Keyword abbreviation:

Not supported.

DSNAME keyword

Use the DSNAME keyword to specify the data set name or the data set mask for the BLDMFILECTL, the FILECTL, or the IRENAME subcommand.

For the BLDMFILECTL or FILECTL subcommand

Use this keyword to specify the data set name or the data set mask that is used for dynamic allocation of the data set to be processed by this subcommand. The data set name produced by the mask (after variable substitutions are done) must conform to the standard naming rules of a data set.

When the data set name contains a reference to a partitioned data set (PDS) member name and it is used for dynamic allocation, DISP=SHR or OLD is required. When unloaded segment records files (USR files) are specified for two or more PDS data set names, each member name and each data set name must be unique. DSNAME may contain a reference to a generation data group (GDG), with or without a relative generation number such as "(+1)".

If DISP=OLD or DISP=SHR is specified, the data set name must refer to an existing cataloged data set. If DISP=NEW is specified, the data set name must not refer to an existing cataloged data set. If the data set name matches an existing GDG base name, the product will automatically append the appropriate generation information to the base name equivalent to a relative generation "(+1)". DISP=(NEW,CATLG) must be specified when a new generation data set is being created.

The subcommand must contain either the DSNAME or the DDNAME keyword. Do not specify both; they are mutually exclusive.

DSNAME is an optional keyword for the FILECTL subcommand.

For the IRENAME subcommand

Use this keyword to specify a mask for the data set name that is used by this subcommand to rename the data set. The data set name generated by the mask (after variable substitutions are done) must conform to the standard naming rules of a data set.

DSNAME is a required keyword for the IRENAME subcommand.

Related keywords:

For the FILECTL subcommand:

- [“DATACLAS keyword” on page 449](#)
- [“DDNAME keyword” on page 453](#)
- [“DISP keyword” on page 453](#)
- [“EXPDT keyword” on page 462](#)
- [“LIKE keyword” on page 492](#)
- [“MGMTCLAS keyword” on page 496](#)
- [“RETPD keyword” on page 519](#)
- [“SPACE keyword” on page 543](#)
- [“STORCLAS keyword” on page 546](#)
- [“UNIT keyword” on page 552](#)
- [“VOLCNT keyword” on page 556](#)
- [“VOLSER keyword” on page 556](#)

For the IRENAME subcommand:

- [“IAREA keyword” on page 469](#)

Format:

```
➤ DSNAME= 'dsname' ➤
```

dsname

Specify a data set name or data set mask with 1- to 64-characters. Enclose the *dsname* in single or double quotation marks. Use the following variables, as necessary.

&DBD

DBD name

&AREA

Area name

&AREA#

3-digit relative area number. (If you specify an area number that is greater than 999, FPA does not convert the mask variable to the relative area number and issues error message HFPU0099E.)

&AREAn#

n-digit relative area number. *n* is 1 - 5.

When the mask variable &AREA# is used for an area whose relative area number is greater than *n*-digit, for example, when &AREA2# is used for an area whose relative area number is greater than or equal to 100, FPA does not convert the mask variable to the relative area number and issues error message HFPU0099E.

&INDEX

Secondary index DBD name

&DATE

Current date ("Dyyddd")

&TIME

Current time ("Thhmmss")

&JOB

Job name

&STEP

Step name

&PROCSN

Procedure step name

&IMSID

IMS system ID (system ID from DFSVC000)

For example, if you specify DSNAME='HFP.TEST.DB.&AREA' and the area name is AREA123, the resulting data set will be 'HFP.TEST.DB.AREA123'. If you specify DSNAME='HFP.TEST.DB.HFP&AREA4#' and the relative area number is 1500, the resulting data set will be 'HFP.TEST.DB.HFP1500'.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

DUPKEYACT keyword

Use the DUPKEYACT keyword to save duplicate pointer segment records in duplicate pointer segment record data sets.

DUPKEYACT is an optional keyword for the INDEXBLD command.

Related keywords:

[“IDXPROC keyword” on page 478](#)

Format:**NONE**

Do not save duplicate pointer segment records.

SAVE

Save duplicate pointer segment records in duplicate pointer segment record data sets.

Default:

DUPKEYACT=NONE

Site default support:

Supported.

Keyword abbreviation:

Supported.

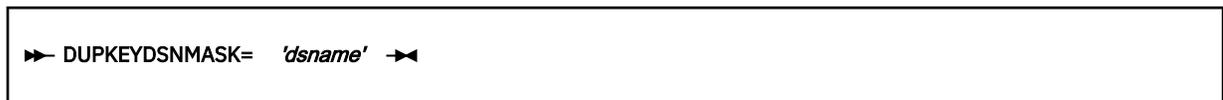
DUPKEYDSNMASK keyword

Use the DUPKEYDSNMASK keyword to specify the data set name or the mask to be used when you want the duplicate pointer segment record data set allocated dynamically.

DUPKEYDSNMASK is an optional keyword for the INDEXBLD command.

Related keywords:

- [“IDXPROC keyword” on page 478](#)
- [“DUPKEYACT keyword” on page 457](#)

Format:***dsname***

Specify a data set name or a data set mask. Enclose *dsname* in single quotation marks. Use the following variable to create the mask.

&INDEX

INDEX DD name.

For example, if you specify DUPKEYDSNMASK= 'HFP.TEST.DB.&INDEX' and the index name is IDX123, dynamic allocation of input data sets dynamically allocates 'HFP.TEST.DB.IDX123'.

Restriction:

DUPKEYMASK and FILECTL are mutually exclusive.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

DUPKEYMAX keyword

Use the DUPKEYMAX keyword to specify the threshold for the number of duplicate keys per secondary index database. When the threshold is reached, FPA stops processing that secondary index database.

Whether FPA continues processing the remaining secondary index databases or stops the job is controlled by the PROC_AFT_DUPKEY keyword.

If the number of duplicate keys reaches this value while processing a secondary index database, the Build Index job or the Resync job skips that secondary index database and continues processing the next secondary index database or stops processing. The action depends on the PROC_AFT_DUPKEY keyword parameter.

DUPKEYMAX is an optional keyword for the INDEXBLD command.

Related keywords:

[“PROC_AFT_DUPKEY keyword” on page 504](#)

Format:



value

Specify a value in the range of 0 - 999. A value of 0 means unlimited.

For example, if you do not want to process secondary index databases that contain 5 or more duplicate keys, specify DUPKEYMAX=5.

Default:

DUPKEYMAX=1

Site default support:

Supported.

Keyword abbreviation:

Supported.

EDITCOMP_SKIP keyword

Use the EDITCOMP_SKIP keyword to specify whether to skip the segment edit/compression exit routine process. This option is effective only for the Unload function and the Change function that is processed in non-REORG mode.

EDITCOMP_SKIP is an optional keyword for the following commands:

- CHANGE
- UNLOAD

Related keywords:

None.

Format:



NO

Calls the segment edit/compression exit routine and compresses or expands the segment records if a segment edit/compression exit routine was defined for that segment during DBDGEN.

YES

Does not call the segment edit/compression exit routine if the COMPRTN keyword parameters are the same between the following ACB libraries:

- For the Unload function, IMSACB and NEWACB
- For the Change function, OLDACB and IMSACB

Warning: If the segment edit/compression exit routines have the same name but different contents, running the job with the EDITCOMP_SKIP=YES option might destroy the database.

Default:

EDITCOMP_SKIP=NO

Site default support:

Supported.

Keyword abbreviation:

Supported.

Usage note:

Even if you specify EDITCOMP_SKIP=YES, the segment edit/compression exit routine is called when one or more of the following conditions apply:

- A user exit is specified.
- The segment definition has changed.
- COMPRESS=NO is specified, but a COMPRTN keyword exists in IMSACB.

EFORMAT keyword

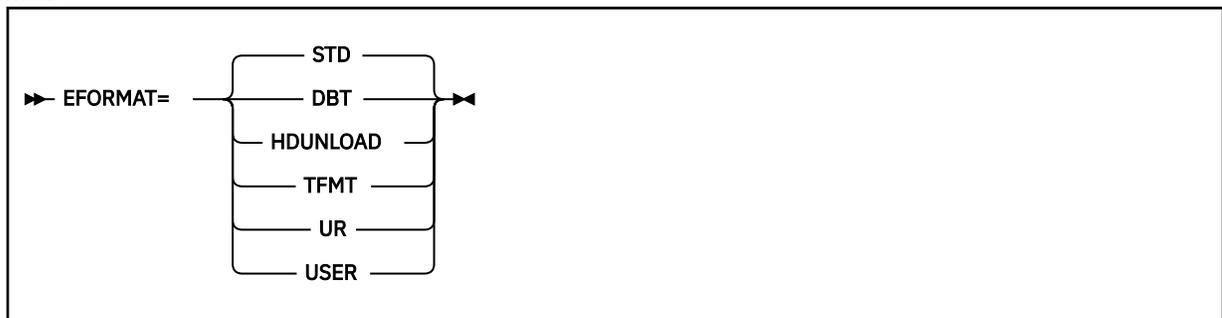
Use the EFORMAT keyword to specify the format of the output file for the EXTRACT command.

EFORMAT is an optional keyword for the EXTRACT command.

Related keywords:

None.

Format:



STD

Standard extract format is to be used. The format is the same as the standard format for the Online Data Extract (ODE) utility.

DBT

Format of the extracted segment records is the same as that of FPB Unload or Reload.

HDUNLOAD

Format of the extracted segment records is the same as that of HD Reorganization Unload utility.

TFMT

Extracted segment records are improved, which means that the prefix part of the record is generated on the basis of the maximum number of segment levels that are defined in a DBD.

UR

Format of the extracted segment records is the same as that of unload or reload of TRIMAR.

USER

Allows more granular control of the output file. This format should be used when an existing application program requires extracted data in a specific format.

Default:

EFORMAT=STD

Site default support:

Supported.

Keyword abbreviation:

Supported.

EVERY keyword

Use the EVERY keyword to select every Nth occurrence of a segment starting from the first segment.

If more than one of the EVERY, SKIP, or STOPAFT keyword is specified, SKIP is applied first, then EVERY, and then STOPAFT. For example, if the SKIP keyword is specified, the first segment is the next segment after the skipped number of segments.

EVERY is an optional keyword for the SELECT subcommand.

Related keywords:

- [“SKIP keyword” on page 540](#)
- [“STOPAFT keyword” on page 545](#)

Format:

```
▶▶ EVERY= value ▶▶
```

value

Specify a value in the range of 1 - 2147483647. To select all segments, specify 1.

Default:

If the EVERY keyword is not specified, all segments are selected.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

EXITRTN keyword

Use the EXITRTN keyword to specify the name of the user exit routine to be invoked prior to the write process of an Unload or a Reload function.

When this keyword is specified, you must either specify the EXITLIB DD statement or the library that contains the user exit routine in JOBLIB or STEPLIB DD.

For each segment that is processed, the exit routine specified by the EXITRTN=keyword parameter will be invoked as follows:

- UNLOAD will invoke the exit routine with an expand function.
- RELOAD will invoke the exit routine with the specified function.

If a segment is compressed by an IMS edit/compression routine, it will be expanded by the edit/compression routine before FPA calls a user exit routine. After returning from the user exit routine, the segment will be re-compressed by the edit/compression routine. Even if the segment is defined as fixed, the LL field of the segment is added and the record is passed to the user exit routine.

EXITRTN is an optional keyword for the following commands:

- RELOAD
- UNLOAD

Related keywords:

- [“IEXITRTN keyword” on page 479](#)
- [“OEXITRTN keyword” on page 502](#)

Format:

For the UNLOAD command:

```
➤➤ EXITRTN= exit_routine ➤➤
```

For the RELOAD command:

```
➤➤ EXITRTN=( exit_routine , { EXP  
                        { CMP  
                        { EXC } } ) ➤➤
```

exit_routine

Specify a 1- to 8-character name of the user exit routine.

For the RELOAD command, the second parameter specifies the function to be performed by the exit routine:

EXP

Expansion

CMP

Compression

EXC

Expansion then compression. This keyword invokes the exit routine twice.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

Requirement:

All exit routines should be re-entrant so that they can be invoked by multiple subtasks. If the routine is not re-entrant, processing of the user exit routines for multiple subtasks is serialized, and the performance might be degraded. The user exit routines must be capable of AMODE=31.

Related reference

Exit routine option and its interface

The programs for DEDB Unload and DEDB Reload of FPA and FPB, DEDB Change of FPA, the DEDB Reload Segment Data Set Create utility, and the DEDB Unloaded Segment Data Set Retrieve utility support a user exit routine.

EXPDT keyword

Use the EXPDT keyword to specify the expiration date of an output data set that was created by dynamic allocation.

Do not specify both the EXPDT keyword and the RETPD keyword because they are mutually exclusive.

For more information, see the *MVS JCL Reference*.

EXPDT is an optional keyword for the following subcommands:

- BLDMFILECTL
- FILECTL

Related keywords:

- [“RETPD keyword” on page 519](#)
- [“DSNAME keyword” on page 456](#)

Format:

Use either of the formats to specify the expiration date:

yyddd

yy is the two-digit year. *ddd* is the three-digit Julian day of the year.

yyyyddd

yyyy is the four-digit year. *ddd* is the three-digit Julian day of the year.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

FIELDS keyword

Use the FIELDS keyword to specify the content of the segment data records, user-defined header records, or user-defined trailer records that are written to the output file.

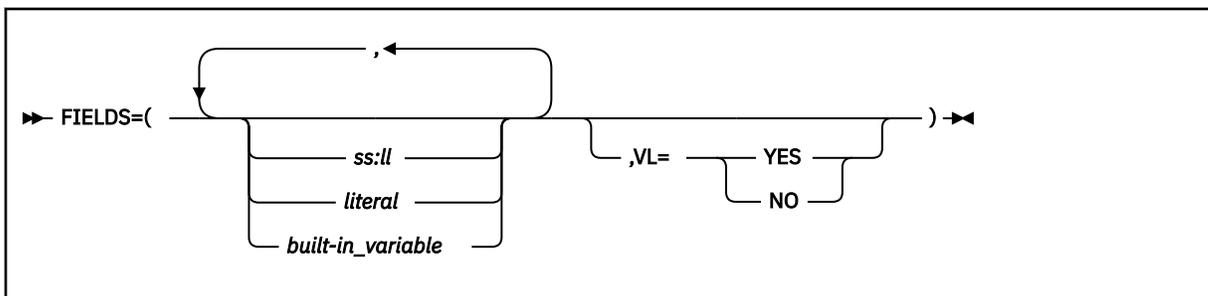
The extracted fields are in the output record in the order that is specified by the FIELDS parameter. When using the OUTPUT subcommand, and if you want a complete segment, do not use this keyword.

FIELDS is:

- An optional keyword for the OUTPUT subcommand.
- A required keyword for the USER_RECORD subcommand.

Related keywords:

None.

Format:**ss**

Integer specifying the position of the field that is to be compared relative to 1. The value of *ss* cannot be greater than the maximum segment length. This value can be specified only on the OUTPUT subcommand.

ll

Integer specifying the length of the field that is to be compared relative to 1. The value of *ll* cannot be greater than the maximum segment length. You can specify an asterisk (*) to mean (from *ss* to) the last position of the segment. This value can be specified only on the OUTPUT subcommand.

literal

Specify one or more alphabetic or numeric characters that are contained within apostrophes (') or within quotation marks ("). You can specify an optional data-type declaration as follows:

- C'xxxxxx' (character): Case-sensitive character data. You must specify a minimum of 1 and maximum of 28 characters (28 bytes).
- F'xxxxxx' (signed binary): Must contain only numeric digits. You must specify a minimum of 1 and maximum of 19 digits (4 bytes).
- P'xxxxxx' (packed decimal): Assumed positive. Must contain only numeric digits. You must specify a minimum of 1 and maximum of 31 digits (16 bytes).
- X'xxxxxx' (unsigned binary): Must be an even number of hexadecimal characters. You must specify a minimum of two hexadecimal characters (representing 1 byte) and a maximum of 56 hexadecimal characters (representing 28 bytes).

You can also declare the length, by specifying it before the data-type declaration.

Example:

```
8C 'CUSDB '  
2P '01 '
```

Built-in variable

The following built-in variables can be specified on the OUTPUT subcommand.

Built-in variable name	Data type	Contents
AREA_NAME	8C	Area name
AREA_NUMBER	1X	Area number (1 byte)
AREA_NUMBER2	2X	Area number (2 bytes)
DBD_AREAS	2F	Number of areas in DBD
DBD_LEVELS	2F	Maximum hierarchic level in DBD
DBD_NAME	8C	DBD name
DBD_SEGMENTS	2F	Number of segment types in DBD
KEY or SEGMENT_KEY	nC	Key for current segment
ROOT_KEY	nC	Key for root segment
ROOT_KEY_LENGTH	2F	Length of root segment key
RUNDATE	4X	Date when FPA started
RUNDATETIME	12X	Date and time when FPA started
RUNTIME	4X	Time when FPA started
SEGMENT_CKEY	nC	Concatenated key for current segment
SEGMENT_CKEY_LENGTH	2F	Length of concatenated key for current segment
SEGMENT_CODE	1X	Segment code
SEGMENT_KEY_LENGTH	2F	Length of current segment key
SEGMENT_LENGTH	2F	Length of segment
SEGMENT_LEVEL	1X	Segment level
SEGMENT_NAME	8C	Segment name
TODAY	4X	Date when FPA started

The following built-in variables can be specified on the USER_RECORD subcommand.

Built-in variable name	Data type	Contents
AREA_COUNT	2X	<ul style="list-style-type: none"> Number of areas that were attempted to be written to an extract file. (BREAK=(DATABASE, BEFORE)) Number of areas written to an extract file. (BREAK=(DATABASE, AFTER))
AREA_NAME	8C	Area name BREAK=(AREA,when) only
AREA_NUMBER	1X	Area number (1 byte) BREAK=(AREA,when) only
AREA_NUMBER2	2X	Area number (2 bytes) BREAK=(AREA,when) only
DBD_AREAS	2F	Number of areas in DBD
DBD_LEVELS	2F	Maximum hierarchic level in DBD
DBD_NAME	8C	DBD name
DBD_SEGMENTS	2F	Number of segment types in DBD
RUNDATE	4X	Date when FPA started
RUNDATETIME	12X	Date and time when FPA started
RUNTIME	4X	Time when FPA started
SEGMENT_COUNT_DETAIL	4X x 127	Number of extracted segments for each segment type
SEGMENT_COUNT_DETAIL_SEG	4X x n	Number of extracted segments for each segment type defined in DBD
SEGMENT_COUNT_TOTAL	4X	Number of extracted segments
TODAY	4X	Date when FPA started

VL=YES

Request a length prefix.

VL=NO

Suppress a length prefix.

Default:

VL=NO when the value of EFORMAT keyword is STD. VL=YES when the value of EFORMAT keyword is DBT, HDUNLOAD, TFMT, UR, or USER.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

Usage note:

If you specify the FIELDS keyword for a compressed segment, you must expand the segment with the COMPRESS=NO keyword.

FORMAT keyword

Use the FORMAT keyword to specify the format of the unloaded file.

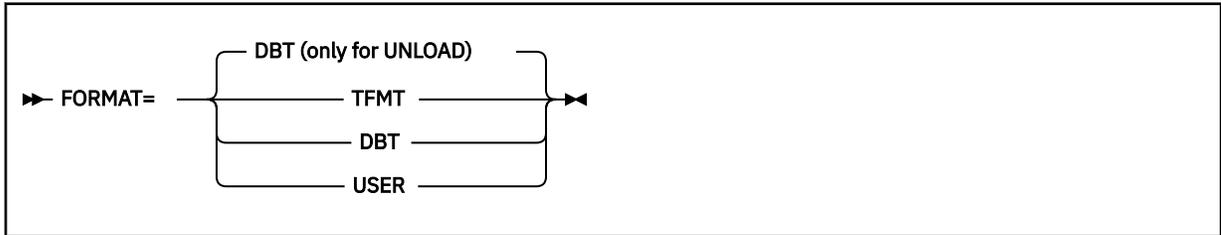
FORMAT is an optional keyword for the following commands:

- RELOAD
- UNLOAD

Related keywords:

None.

Format:



TFMT

Unloaded segment records are enhanced, which means that the prefix part of the record is generated based on the maximum number of segment levels that are defined in DBD. This value can be specified only on the UNLOAD command.

DBT

Format of the unloaded segment records is the same as that of FPB Unload or Reload. This value can be specified only on the UNLOAD command.

USER

Allows more granular control of the input file or the output file. This format should be used when an existing application program requires unloaded data in a specific format.

Default:

- For the UNLOAD command, FORMAT=DBT.
- For the RELOAD command, no default.

Site default support:

- For the RELOAD command, site default is not supported.
- For the UNLOAD command, site default is supported.

Keyword abbreviation:

Supported.

FS_DOVFIOVF keyword

Use the FS_DOVFIOVF keyword to specify a threshold setting for the free space in DOVF and in IOVF.

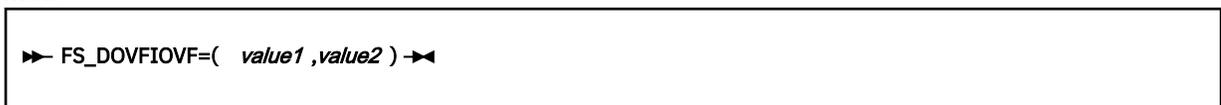
If the free space in DOVF is greater than the value that you specified, and if the free space in IOVF is less than the value that you specified, a warning message is issued.

FS_DOVFIOVF is an optional keyword for the THRESHOLD subcommand.

Related keywords:

None.

Format:



value1

value2

Specify a value in the range of 0 - 100 for each parameter. You must specify both values.

Default:

None.

Site default support:

Not supported.

Alias:

FSDI

Keyword abbreviation:

Supported.

FS_RAADOVF keyword

Use the FS_RAADOVF keyword to specify a threshold setting for the free space in RAA BASE and in DOVF.

If the free space in RAA BASE is greater than the value that you specified, and if the free space in DOVF is less than the value that you specified, a warning message is issued.

FS_RAADOVF is an optional keyword for the THRESHOLD subcommand. This keyword is mutually exclusive with the FS_RAADOVFIOVF= keyword.

Related keywords:

None.

Format:

```
►► FS_RAADOVF=( value1 ,value2 ) ◄◄
```

value1

value2

Specify a value in the range of 0 - 100 for each parameter. You must specify both values.

Default:

None.

Site default support:

Not supported.

Alias:

FSRD

Keyword abbreviation:

Supported.

FS_RAADOVFIOVF keyword

Use the FS_RAADOVFIOVF keyword to specify a threshold setting for the free space in RAA BASE, DOVF, and IOVF.

If the free space in RAA BASE is greater than the value that you specified, and if the free space in either DOVF or IOVF is less than the value that you specified, a warning message is issued.

FS_RAADOVFIOVF is an optional keyword for the THRESHOLD subcommand. This keyword is mutually exclusive with the FS_RAADOVF= and FS_RAATIOVF= keywords.

Related keywords:

None.

Format:

```
►► FS_RAADOVFIOVF=( value1 ,value2 ,value3 ) ◄◄
```

value1

value2

value3

Specify a value in the range of 0 - 100 for each parameter. You must specify all values.

Default:

None.

Site default support:

Not supported.

Alias:

FSRDI

Keyword abbreviation:

Supported.

FS_RAAIOVF keyword

Use the FS_RAAIOVF keyword to specify a threshold setting for the free space in RAA BASE and in IOVF.

If the free space in RAA BASE is greater than the value that you specified, and if the free space in IOVF is less than the value that you specified, a warning message is issued.

FS_RAAIOVF is an optional keyword for the THRESHOLD subcommand. This keyword is mutually exclusive with the FS_RAADOVFIOVF= keyword.

Related keywords:

None.

Format:

```
▶▶ FS_RAAIOVF=( value1 ,value2 ) -▶▶
```

value1**value2**

Specify a value in the range of 0 - 100 for each parameter. You must specify both values.

Default:

None.

Site default support:

Not supported.

Alias:

FSRI

Keyword abbreviation:

Supported.

FS_RECLAIM_REPORT keyword

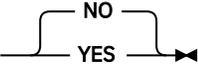
Use the FS_RECLAIM_REPORT keyword to specify whether to generate the Secondary Index Freespace Reclaim report and the Result of Access Method Services report.

FS_RECLAIM_REPORT is an optional keyword for the INDEXBLD command.

Related keywords:

None.

Format:

```
▶▶ FS_RECLAIM_REPORT=  -▶▶
```

YES

Generate the reports.

- CHANGE
- DMACPRT
- EXTRACT
- FILECTL
- INDEXBLD
- IRENAME
- RELOAD
- REORG
- TUNE
- UNLOAD

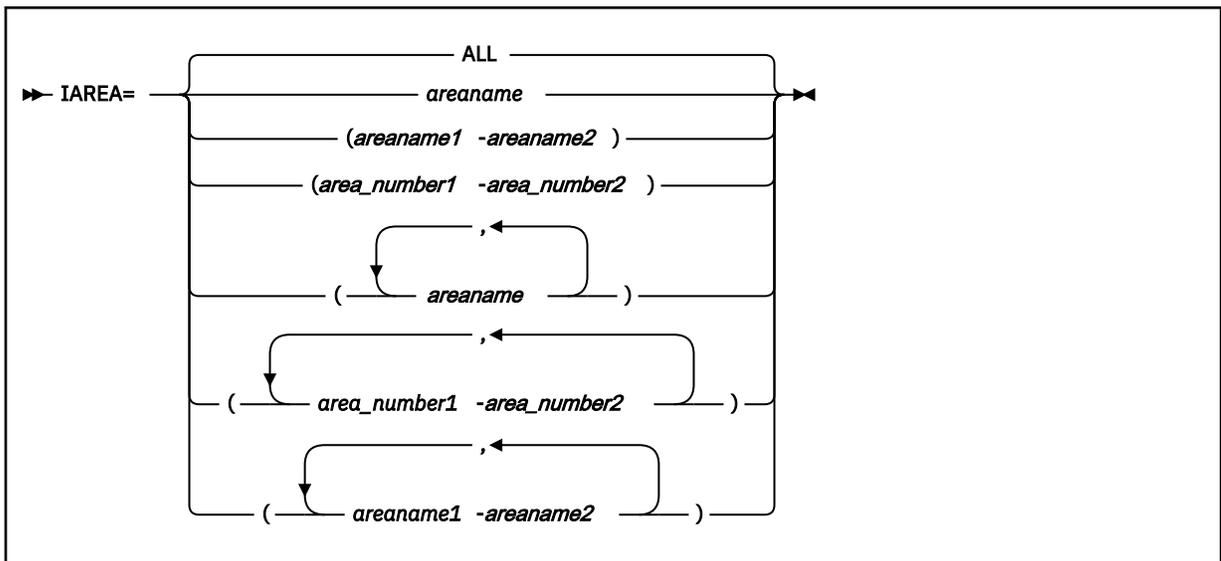
Restrictions:

- If you specify the IAREA keyword for one of the following commands, you must specify the same areas for the IAREA keyword and the OAREA keyword. Otherwise, FPA issues message HFPC0061E and ends with an error.
 - EXTRACT command
 - UNLOAD, CHANGE, or TUNE command in REORG mode
- You can specify the IAREA keyword for the FILECTL subcommand only when you specify the TUNE command.

Related keywords:

[“OAREA keyword” on page 497](#)

Format:



areaname

Specify a 1- to 8-character area name.

When you specify the HFPAUOWC DD and want to process multiple areas, specify the area names in ascending order of their area numbers.

You can also use certain wild card characters to specify areas. These wild card characters are the asterisk (*), which represents from 0 to 8 characters, and the percent sign (%), which represents any single character.

You can specify multiple areas in the following format: `IAREA=(area1,area2,area3)`

You can also use multiple lines to specify multiple areas:

```
IAREA=(area1,  
       area2,  
       area3),
```

However, you cannot specify as follows:

```
IAREA=(  
       area1,  
       area2,  
       area3  
       ),
```

area_number1-area_number2

Specify the range of areas by area numbers.

area_number1 and *area_number2* are area numbers, which are decimal numbers assigned sequentially to the areas within a database. The value for *area_number1* and *area_number2* can be in the range of 1 - 2048. *area_number2* must be equal to or greater than *area_number1*. At least one area must exist in the specified range.

areaname1-areaname2

Specify the range of areas by area names.

areaname1 and *areaname2* must be specified in alphabetical order. At least one area must exist in the specified range.

For example, the following areas are specified in the DBD:

Area number	Area name
1	AABBCC
2	CBA
3	ABCD
4	C
5	A
6	TEST1ALL
7	ALLTEST1
8	AAA
9	AA
10	AREA1234

- If you specify IAREA=%B*, areas CBA and ABCD are selected.
- If you specify IAREA=A-AAA, areas A, AA, and AAA are selected.
- If you specify IAREA=6-7, areas TEST1ALL and ALLTEST1 are selected.

Default:

IAREA=ALL

Site default support:

Not supported.

Keyword abbreviation:

Supported.

YES

Preload IOVF and SDEP parts into the data space when an ADS is used as the input data set with any FPA command except the ANALYZE command and the TUNE command.

If a VSAM ESDS is used as the input data set for the Analyze function or the Tune function, ICACHE parameter is forced to NO, even if ICACHE=YES is specified.

If an image copy is used as the input data set for the Analyze function, the Change function, the Extract function, the Tune function, or the Unload function, FPA preloads the root addressable part, the IOVF part, and the SDEP part into the data space.

NO

Do not preload IOVF and SDEP parts into the data space.

If an image copy is used as the input data set for the Analyze function, the Change function, the Extract function, the Tune function, or the Unload function, FPA does not preload the BASE section into the data space.

Default:

ICACHE=YES

Site default support:

Supported.

Keyword abbreviation:

Supported.

ICHECK keyword

Use the ICHECK keyword to control the input devices.

ICHECK is an optional keyword for the following commands:

- ANALYZE
- CHANGE
- EXTRACT
- INDEXBLD
- RELOAD
- TUNE
- UNLOAD

Related keywords:

None.

Format:**NODEFER**

Gets access to all devices that are associated with the input data sets at a time.

DEFER

Access to devices for obtaining input data sets are limited to the number that is specified by the ITASKCTL keyword.

Default:

ICHECK=NODEFER

Site default support:

Supported.

Keyword abbreviation:

Supported.

ICINCOMP keyword

Use the ICINCOMP keyword to specify, when an image copy is used as an input data set, whether the image copy records for the root addressable part are to be compressed by using the CSRCSRVS MVS macro when they are stored into a data space storage.

FPA stores all image copy records into a data space storage when an image copy is used as an input data set. Because the maximum size of one data space that is created by FPA is 2 GB, if the total size of the records for the BASE part, the IOVF part, or the SDEP part exceeds the limit, FPA obtains another data space storage and stores the records. When ICINCOMP=YES is specified, the records are compressed and the total size of the records is reduced.

Notes:

1. When ICINCOMP=YES is specified, FPA compresses the image copy records for the root addressable part by using the CSRCSRVS macro before storing them into a data space, and extracts them when they are processed.
2. The data compression and expansion processes increase the amount of CPU time.

ICINCOMP is an optional keyword for the following commands:

- ANALYZE
- CHANGE
- EXTRACT
- GLOBAL
- INDEXBLD
- TUNE
- UNLOAD

Related keywords:

None.

Format:**YES**

The image copy records for the root addressable part are stored into a data space storage after compression.

NO

The image copy records for the root addressable part are stored into a data space storage without compression.

Default:

ICINCOMP=NO

Site default support:

- For the ANALYZE, CHANGE, EXTRACT, and UNLOAD, site default is supported.
- For the GLOBAL command, site default is not supported.

Keyword abbreviation:

Supported.

ICNOTIFY keyword

Use the ICNOTIFY keyword to issue NOTIFY.IC command to DBRC.

If a pointer error is detected in an area, NOTIFY.IC command is not issued for any areas.

If you do not specify DBRC=YES and ICOUT=YES, this keyword is ignored.

ICNOTIFY is an optional keyword for the following commands:

- ANALYZE
- CHANGE
- GLOBAL
- RELOAD
- REORG
- UNLOAD

Related keywords:

- [“DBRC keyword” on page 451](#)
- [“ICOUT keyword” on page 476](#)

Format:



YES

When RESTART=NO is specified, the function issues the NOTIFY.IC commands if the return code of the preceding process is 4 or lower. If the return code of the preceding process is 8 or higher, the NOTIFY.IC command is not issued for any areas. When the primary image copy data set or the secondary image copy data set of an area has an error, the NOTIFY.IC command for that area is issued.

When RESTART=YES is specified, the function issues the NOTIFY.IC commands only for the areas that were processed without errors. When the primary image copy data set or the secondary image copy data set of an area has an error, the NOTIFY.IC command for that area is not issued.

NO

NOTIFY.IC commands are not issued.

COND

Conditionally issues the NOTIFY.IC commands.

When the preceding process ends with a return code of 4 due to either of the following causes, the NOTIFY.IC command is not issued to DBRC:

- Pointer errors detected
- Key sequence errors detected

Otherwise, the NOTIFY.IC commands are issued to DBRC.

When ICNOTIFY=COND is specified on the GLOBAL command for the RELOAD, CHANGE, ANALYZE, or REORG function, or ICNOTIFY=COND is specified on the RELOAD, CHANGE, ANALYZE, or REORG command, ICNOTIFY=COND is treated as ICNOTIFY=YES.

Default:

ICNOTIFY=YES

Site default support:

- For the ANALYZE, CHANGE, RELOAD, REORG, and UNLOAD commands, site default is supported.

- For the GLOBAL command, site default is not supported.

Keyword abbreviation:

Supported.

ICOUT keyword

Use the ICOUT keyword to generate one or more image copies.

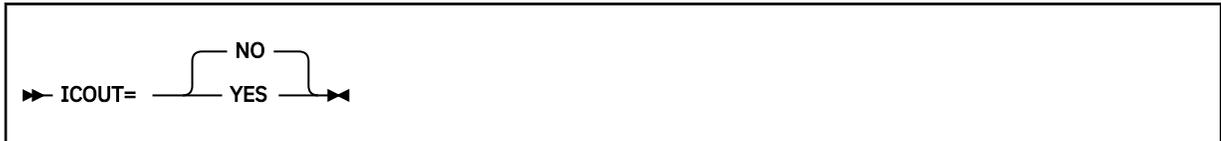
ICOUT is an optional keyword for the following commands:

- ANALYZE
- CHANGE
- GLOBAL
- RELOAD
- REORG
- UNLOAD

Related keywords:

[“ICNOTIFY keyword” on page 475](#)

Format:



YES

Generate one or more image copies.

NO

Do not generate any image copies.

Default:

ICOUT=NO

Site default support:

- For the ANALYZE, CHANGE, RELOAD, REORG, and UNLOAD commands, site default is supported.
- For the GLOBAL command, site default is not supported.

Keyword abbreviation:

Supported.

IDSNMASK keyword

Use the IDSNMASK keyword to specify the data set name or the mask for dynamically allocating cataloged pre-existing input data sets.

This keyword can be used with the ANALYZE, CHANGE, DMACPRT, EXTRACT, REORG, TUNE, or UNLOAD command to allocate the VSAM area data set or the image copy data set to be used as input. It can be used with the RELOAD command to allocate the input (UNLOAD) data set. You can make a relative reference to a GDG by specifying a relative generation number in the mask, a number enclosed in brackets such as in '*gdg-name*(-2)'. The number you specify in the brackets must be zero or smaller.

When the data set specified by IDSNMASK contains a reference to a partitioned data set (PDS) member name, and it is used for dynamic allocation, you must specify DISP=SHR or OLD.

IDSNMASK is an optional keyword for the following commands:

- ANALYZE
- CHANGE
- DMACPRT

- EXTRACT
- INDEXBLD
- RELOAD
- REORG
- TUNE
- UNLOAD

Related keywords:

None.

Format:

```
➤ IDSNMASK= 'dsname' ➤
```

dsname

Specify a data set name or a data set mask. Enclose the *dsname* in single quotation marks. Use the following variables, as necessary, to create the mask.

&DBD

DBD name

&AREA

Area name

&AREA#

3-digit relative area number. (If you specify an area number that is greater than 999, FPA does not convert the mask variable to the relative area number and issues error message HFPU0099E.)

&AREAn#

n-digit relative area number. *n* is 1 - 5.

When the mask variable &AREA# is used for an area whose relative area number is greater than *n*-digit, for example, when &AREA2# is used for an area whose relative area number is greater than or equal to 100, FPA does not convert the mask variable to the relative area number and issues error message HFPU0099E.

For example, if you specify IDSNMASK='HFP.TEST.DB.&AREA' and the area name is AREA123, dynamic allocation of input data sets will dynamically allocate 'HFP.TEST.DB.AREA123'. If you specify IDSNMASK='HFP.TEST.DB.HPFP&AREA4#' and the relative area number is 1500, the resulting data set name will be 'HFP.TEST.DB.HPFP1500'.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

IDX_DUMP keyword

Use the IDX_DUMP keyword to specify whether to generate the Pointer Segment Dump report when incorrect pointers are detected.

IDX_DUMP is an optional keyword for the ANALYZE command.

Related keywords:

None.

Format:



YES

Generate the Pointer Segment Dump report in the HFPRPTS data set.

NO

Do not generate the Pointer Segment Dump report.

Default:

IDXDUMP=NO

Site default support:

Supported.

Keyword abbreviation:

Supported.

IDXMAXERROR keyword

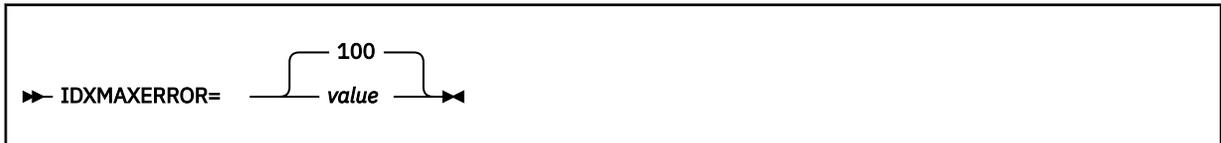
Use the IDXMAXERROR keyword to specify the maximum allowable number of pointer segment errors for the index database. When the maximum number is reached, the ANALYZE process ends abnormally.

IDXMAXERROR is an optional keyword for the ANALYZE command.

Related keywords:

None.

Format:



value

Specify a value in the range of 0 - 999. If 0 is specified, the maximum allowable number is not set.

Default:

IDXMAXERROR=100

Site default support:

Supported.

Keyword abbreviation:

Supported.

IDXPROC keyword

Use the IDXPROC keyword to specify the function that the Build Index process starts.

IDXPROC is an optional keyword for the INDEXBLD command.

Related keywords:

None.

Format:



BUILD

Starts the Build Index function and builds secondary index databases.

RESYNC

Starts the Resync function and synchronizes the primary DEDB database and its secondary index databases.

FS_RECLAIM

Starts the Freespace Reclaim function and reclaims CIs in secondary index data sets.

Default:

IDXPROC=BUILD

Site default support:

Supported.

Keyword abbreviation:

Supported.

IDXTASKCTL keyword

Use the IDXTASKCTL keyword to specify the maximum number of input secondary index databases that are to be processed concurrently.

IDXTASKCTL is an optional keyword for the ANALYZE command.

Related keywords:

None.

Format:**value**

Specify a value in the range of 1- 999. The value must be less than or equal to the number of input secondary index databases. If you specify a value that is greater than the number of input secondary index databases, FPA internally replaces the value with the number of input secondary index databases.

Default:

IDXTASKCTL=4

Site default support:

Supported.

Keyword abbreviation:

Supported.

IEXITRTN keyword

Use the IEXITRTN keyword to specify the name of the user exit routine that will be invoked after the area data set that is specified by the Change function is unloaded.

When this keyword is specified, you must either specify the EXITLIB DD statement or the library that contains the user exit routine in JOBLIB or STEPLIB DD.

For each segment that is processed, the exit routine that is specified by the IEXITRTN=keyword parameter will be invoked with an expand function. If a segment is compressed by an IMS edit/compression routine, it will be expanded by the edit/compression routine before FPA calls a user exit routine. Even if the segment is defined as fixed, the LL field of the segment is added and the record is passed to the user exit routine.

IEXITRTN is an optional keyword for the CHANGE command.

Related keywords:

- [“EXITRTN keyword” on page 461](#)
- [“OEXITRTN keyword” on page 502](#)

Format:

```
▶▶ IEXITRTN= exit_routine ▶▶
```

exit_routine

Specify a 1- to 8-character name of the user exit routine.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

Requirement:

All exit routines should be re-entrant so that they can be invoked by multiple subtasks. If the routine is not re-entrant, processing of the user exit routines for multiple subtasks is serialized, and the performance might be degraded. All exit routines must be capable of AMODE=31.

Related reference

[EXITRTN keyword](#)

Use the EXITRTN keyword to specify the name of the user exit routine to be invoked prior to the write process of an Unload or a Reload function.

IMSCATACB_INPUT keyword

Use the IMSCATACB_INPUT keyword to specify whether to retrieve the currently active ACB definition or the pending ACB definition from the IMS directory. IMSCATACB_INPUT keyword is effective only when the IMSCATHLQ=*bsdshlq* option is specified on the GLOBAL command.

IMSCATACB_INPUT is an optional keyword for the GLOBAL command.

Related keywords:

- [“IMSCATHLQ keyword” on page 481](#)
- [“IMSCATACB_OUTPUT keyword” on page 481](#)

Format:

```
▶▶ IMSCATACB_INPUT=  ▶▶
```

CURRENT

The currently active ACB member is retrieved from the IMS directory data sets.

PENDING

The pending ACB member is retrieved from the staging data set.

Default:

IMSCATACB_INPUT=CURRENT

Site default support:

Supported.

Keyword abbreviation:

Not supported.

IMSCATACB_OUTPUT keyword

Use the IMSCATACB_OUTPUT keyword to specify whether to retrieve the currently active ACB definition or the pending ACB definition from the IMS directory. This keyword specifies the new ACB definition, which is used when an ACB definition change is requested. IMSCATACB_OUTPUT keyword is effective only when the IMSCATHLQ=*bsdshlq* option is specified on the GLOBAL command.

IMSCATACB_OUTPUT is an optional keyword for the CHANGE command, the UNLOAD command, and the SIMULATE subcommand.

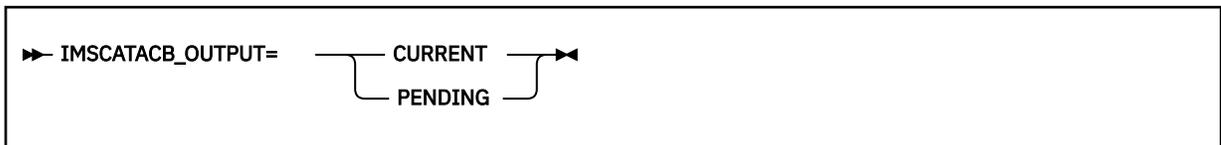
If you specify this keyword for the UNLOAD or the CHANGE command, the Unload or the Change function is processed in non-REORG mode. However, if the ACB member that the IMSCATACB_OUTPUT keyword specifies is the same as the ACB member that the IMSCATACB_INPUT keyword specifies and if the NEWRESLB DD statement for the UNLOAD command or the OLDRESLB DD statement for the CHANGE command is not specified in the JCL statement, the Unload or the Change function is processed in REORG mode.

If you specify this keyword for the SIMULATE subcommand, the Tune function retrieves the new ACB member, which is used when simulating changes in ACB definitions, from the IMS directory.

Related keywords:

- [“IMSCATHLQ keyword” on page 481](#)
- [“IMSCATACB_INPUT keyword” on page 480](#)

Format:



CURRENT

The currently active ACB member is retrieved from the IMS directory data sets and is used as the new ACB.

PENDING

The pending ACB member is retrieved from the staging data set and is used as the new ACB.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Not supported.

IMSCATHLQ keyword

Use the IMSCATHLQ keyword to specify the high-level qualifier of the bootstrap data set of the IMS directory, which is an extension of the IMS catalog.

IMSCATHLQ is an optional keyword for the GLOBAL command.

You must enable the IMS catalog and the IMS management of ACBs when you specify the high-level qualifier of the bootstrap data set of the IMS directory.

Related keywords:

- [“IMSCATACB_INPUT keyword” on page 480](#)
- [“IMSCATACB_OUTPUT keyword” on page 481](#)

Format:***bsdshlq***

Reads the ACB from the IMS directory instead of the ACB library by using IMS Tools Catalog Interface. *bsdshlq* specifies the high-level qualifier of the bootstrap data set of the IMS directory.

***NO**

Reads the ACB from the ACB library.

Default:

IMSCATHLQ=*NO

Site default support:

Supported.

Keyword abbreviation:

Not supported.

IMSID keyword

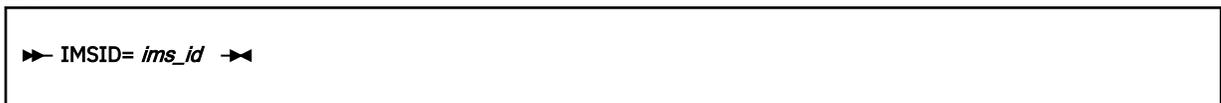
Use the IMSID keyword to specify the IMS system ID. The FPA process runs on the IMS system that is identified by this ID.

IMSID is a required keyword for the GLOBAL command when all of the following keywords are specified:

- IDXPROC=RESYNC
- TOICTL=NONE
- One of RESYNCMODE=VERIFYUPDATE, RESYNCMODE=UPDATE, and RESYNCMODE=DIRECTUPDATE

Related keywords:

None.

Format:***ims_id***

Specify a 1 - 4 byte IMS system ID.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

INDEXDBD keyword

Use the INDEXDBD keyword to specify the database description (DBD) for processing secondary index databases.

The DBD name must correspond to a member name in the IMSACB library.

INDEXDBD is an optional keyword for the following commands and subcommands:

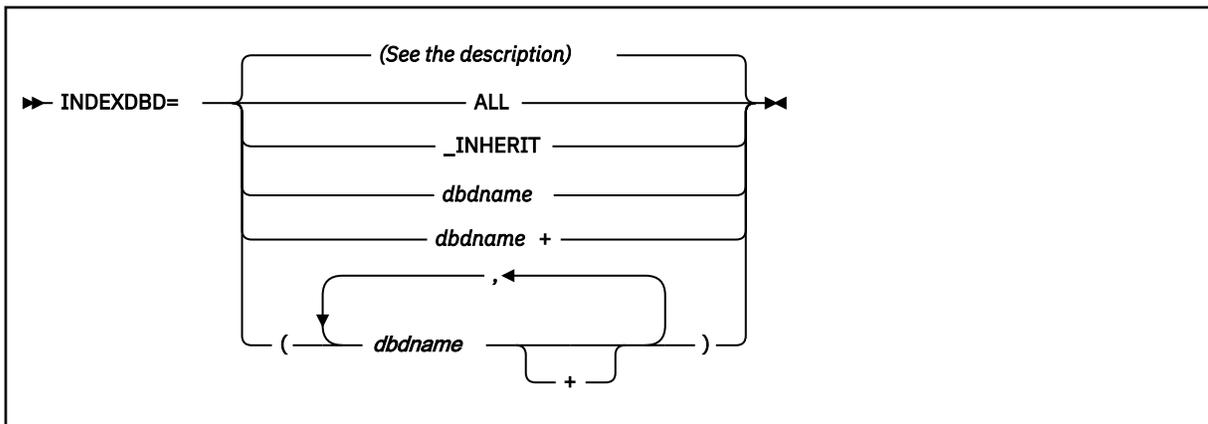
- ANALYZE
- INDEXBLD
- BLDMFILECTL

- FILECTL

Related keywords:

None.

Format:



ALL

Processes all the index databases that are defined in the DBD.

_INHERIT

Processes all the index databases that are specified by the INDEXDBD keyword of the INDEXBLD command. INDEXDBD=_INHERIT can be specified only for the FILECTL subcommand and the BLDMFILECTL subcommand.

dbdname, dbdname+, or (dbdname1,dbdname2,dbdname3,dbdname+,...)

Processes specified index databases in the DBD. *dbdname* must be 1 - 8 characters.

dbdname

Process the specified index database. The DBD can be a member of a partition group.

dbdname+

Process one or more index databases. For *dbdname*, specify the first member of a partition group. The plus sign (+) defines that all the members of the partition group are processed.

(dbdname1,dbdname2,dbdname3,dbdname+,...)

Process multiple index databases.

Restriction: You cannot specify a member multiple times. If the same *dbdname* is specified multiple times, the process ends with an error. If you specify a partition group and also specify a member that belongs to that group, the process ends with an error.

Examples:

For the purpose of these examples, a DBD is defined with two indexes as follows:

```
LCHILD NAME=(A1SEG1, (A1IDXDB1, A1IDXDB2, A1IDXDB3)), PTR=SYMB, . . . .)
LCHILD NAME=(B1SEG1, B1IDXDB1), PTR=SYMB, . . . .)
```

Based on this example DBD, you can specify the INDEXDBD command as shown in the following examples:

- To process only A1IDXDB2: INDEXDBD=A1IDXDB2
- To process A1IDXDB1 and B1IDXDB1: INDEXDBD=(A1IDXDB1, B1IDXDB1)
- To process A1IDXDB1, A1IDXDB2, A1IDXDB3, and B1IDXDB1: INDEXDBD=(A1IDXDB1+, B1IDXDB1)
- To process A1IDXDB1, A1IDXDB2, and A1IDXDB3: INDEXDBD=A1IDXDB1+
- To process A1IDXDB1, A1IDXDB2, and A1IDXDB3: INDEXDBD=(A1IDXDB1+)

Default:

- For INDEXBLD, INDEXDBD=ALL.

- For ANALYZE, none.
- For FILECTL and BLDMFILECTL, INDEXDBD=_INHERIT.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

INSERT_LIMIT_COUNT keyword

Use the INSERT_LIMIT_COUNT keyword to specify the maximum number of *segname* segments that can be inserted close to their root segment.

The maximum value is 32767 and the minimum value is 0. The minimum value 0 means that all specified segments are stored in the place that is specified by the LOCATION keyword.

Specifying a low value for the INSERT_LIMIT_COUNT keyword can seriously impair the performance of subsequent IMS access to the database. Therefore, no more than 10% of the database records being processed should exceed the INSERT_LIMIT_COUNT value.

INSERT_LIMIT_COUNT is an optional keyword for the LOADCTL subcommand.

Related keywords:

- [“LOCATION keyword” on page 494](#)
- [“SEGMENT keyword” on page 530](#)

Format:



value

Specify the number of segments to insert close to their root segment. Valid values are 0 - 32767.

Default:

INSERT_LIMIT_COUNT=0

Site default support:

Supported.

Alias:

ILC

Keyword abbreviation:

Supported.

IOVFCI_RBA_REPORT keyword

Use the IOVFCI_RBA_REPORT keyword to specify whether to generate the RBA of Used IOVF CI report.

IOVFCI_RBA_REPORT is an optional keyword for the REPORT subcommand.

This keyword is effective only for the ANALYZE, RELOAD, CHANGE, and REORG command processing with PTRCHKLVL=FULL or PTRCHKLVL=QUICK keyword.

Related keywords:

None.

Format:**YES**

Generate the RBA of Used IOVF CI report.

NO

Do not generate the RBA of Used IOVF CI report.

Default:

IOVFCI_RBA_REPORT=NO

Site default support:

Supported.

Keyword abbreviation:

Not supported.

IOVFFS_PCT keyword

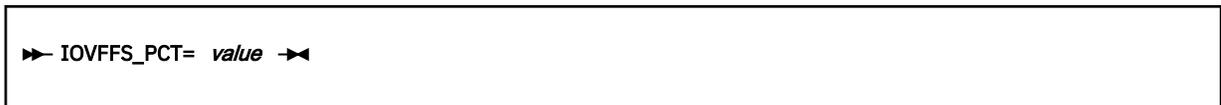
Use the IOVFFS_PCT keyword to specify a threshold setting for the percentage of free space in IOVF.

If the percentage is less than the user-specified value, a warning message is issued.

IOVFFS_PCT is an optional keyword for the THRESHOLD subcommand.

Related keywords:

None.

Format:***value***

Specify a value in the range of 0 - 100.

Default:

None.

Site default support:

Not supported.

Alias:

IFSP

Keyword abbreviation:

Supported.

IOVFUSED_PCT keyword

Use the IOVFUSED_PCT keyword to specify a threshold setting for the percentage of IOVF CIs that are used.

If the percentage is greater than the value that you specified, a warning message is issued.

IOVFUSED_PCT is an optional keyword for the THRESHOLD subcommand.

Related keywords:

None.

Format:

```
▶▶ IOVFUSED_PCT= value ◀◀
```

value

Specify a value in the range of 0 - 100.

Default:

None.

Site default support:

Not supported.

Alias:

IUP

Keyword abbreviation:

Supported.

ITASKCTL keyword

Use the ITASKCTL keyword to specify the maximum number of input areas to be processed concurrently.

ITASKCTL is an optional keyword for the following commands:

- CHANGE
- EXTRACT
- INDEXBLD
- RELOAD
- REORG
- TUNE
- UNLOAD

Related keywords:

[“OTASKCTL keyword” on page 503](#)

Format:

```
▶▶ ITASKCTL= { AREA } ◀◀
              { value }
```

AREA

The number of input areas.

If you specify AREA with the checkpoint restart option of the Unload function, AREA is internally replaced with the number of areas that are to be reprocessed.

value

Specify a value in the range of 1 - 2048. Specify a value less than or equal to the number of input areas. If you specify a value that is greater than the number of input areas, the value is replaced by the number of input areas.

For the CHANGE command in the REORG mode, the following value is set:

```
(ITASKCTL + OTASKCTL)/2 (the value is rounded off)
```

When this parameter is specified with the checkpoint restart option of the Unload function, and when *value* is larger than the number of areas to be reprocessed, *value* is internally replaced with the number of the areas that are to be reprocessed.

Default:

ITASKCTL=4

Site default support:

Supported.

Keyword abbreviation:

Supported.

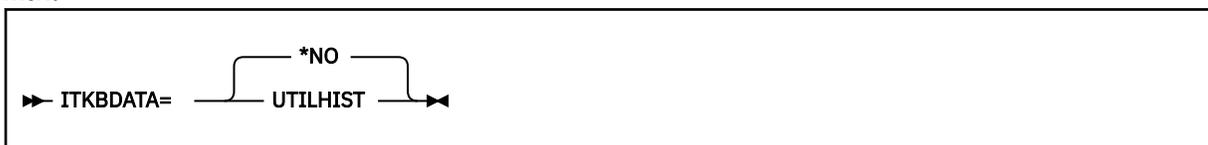
ITKBDATA keyword

Use the ITKBDATA keyword to specify the type of additional data to store in the IMS Tools Base IMS Tools Knowledge Base repository.

ITKBDATA is an optional keyword for the GLOBAL command. This keyword is effective only for the RELOAD, CHANGE, and REORG command processing.

Related keywords:

[“ITKBSRVR keyword” on page 488](#)

Format:**UTILHIST**

Store the utility history member for the FPA job in the IMS Tools KB Sensor Data repository. The stored utility history member contains the reorganization timestamp, which indicates the time when the DEDB areas were reorganized.

When you specify ITKBDATA=UTILHIST, you must also specify the name of the IMS Tools KB server XCF group with the ITKBSRVR keyword. Also, you must specify the IMS Tools Base load module library on the STEPLIB.

***NO**

Do not store the utility history member.

To learn more about utility history, see the topic "Utility history environment" in the *Tools Base IMS Tools Knowledge Base User's Guide*.

Default:

ITKBDATA=*NO

Site default support:

Supported.

Keyword abbreviation:

Supported.

ITKBLOAD keyword

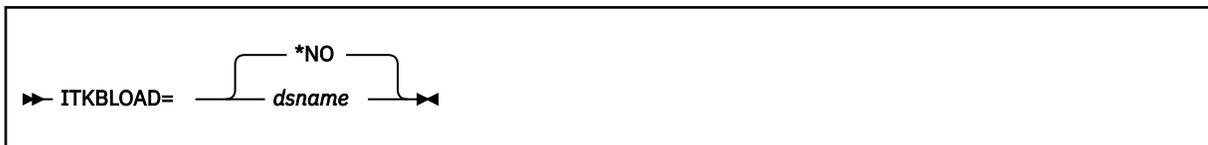
Use the ITKBLOAD keyword to specify the name of the load module library for IMS Tools KB.

ITKBLOAD is an optional keyword for the GLOBAL command.

It is used only when ITKBSRVR=*servername* is specified. If you do not explicitly specify this keyword, ITKBLOAD=*NO will be used and FPA attempts to load an IMS Tools KB module from the STEPLIB libraries.

Related keywords:

[“ITKBSRVR keyword” on page 488](#)

Format:***dsname***

Specifies the name of the load module library for IMS Tools KB.

ITKBLOAD=*dsname* is mutually exclusive with SENSOR=YES. When SENSOR=YES is specified, you must specify the IMS Tools Base load module library on the STEPLIB.

***NO**

The IMS Tools KB modules are loaded from the private library or the system library of the job.

Default:

ITKBLOAD=*NO

Site default support:

Supported.

Keyword abbreviation:

Supported.

Related reference

[Reports generated by each process of FPA](#)

With each process of FPA, several reports are generated.

ITKBSRVR keyword

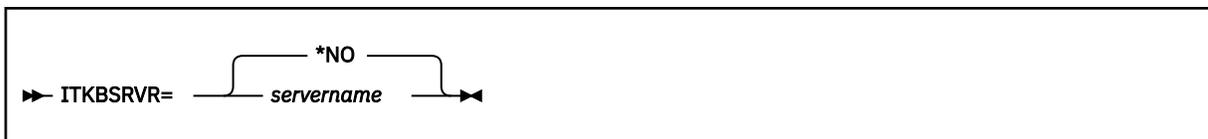
Use the ITKBSRVR keyword to specify the name of the IMS Tools KB server XCF group.

ITKBSRVR is an optional keyword for the GLOBAL command. You must specify the name of the IMS Tools KB server XCF group on this keyword when you specify SENSOR=YES or ITKBDATA=UTILHIST.

If you do not explicitly specify this keyword, ITKBSRVR=*NO is used.

Related keywords:

[“ITKBLOAD keyword” on page 487](#)

Format:***servename***

Store reports, sensor data, or both in the repositories of the specified IMS Tools KB server.

***NO**

Do not store reports or sensor data in the repositories of IMS Tools KB.

Default:

ITKBSRVR=*NO

Site default support:

Supported.

Keyword abbreviation:

Supported.

Related reference

[Reports generated by each process of FPA](#)

With each process of FPA, several reports are generated.

KEYCHGCHK keyword

Use the KEYCHGCHK keyword to specify whether the key sequence field was not changed by a user exit routine that is specified with EXITRTN, IEXITRTN, or OEXITRTN keyword.

KEYCHGCHK is an optional keyword for the following commands:

- CHANGE
- RELOAD
- UNLOAD

Related keywords:

[“KEYSEQCHK keyword” on page 490](#)

Format:



YES

Check altered key sequence field by a user exit routine.

NO

Do not check altered key sequence field by a user exit routine.

Default:

KEYCHGCHK=YES

Site default support:

Supported.

Keyword abbreviation:

Supported.

KEYLABEL keyword

Use the KEYLABEL keyword to specify the key label for encrypting a new VSAM cluster. When encryption is requested for the base cluster, any alternate indexes associated with the base cluster will also be encrypted and use the same key label as specified for the cluster.

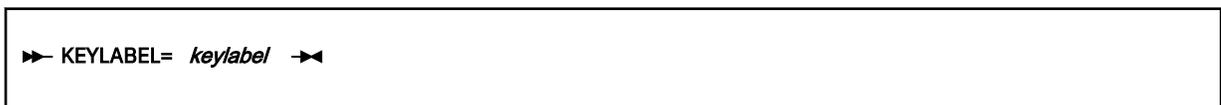
For more information about the KEYLABEL parameter, see *z/OS DFSMS Access Method Services Commands*.

KEYLABEL is an optional keyword for the ALLOCATE subcommand.

Related keywords:

[“DATACLAS keyword” on page 449](#)

Format:



keylabel

Specify a 1- to 64-character data key label.

Default:

None.

Site default support:

Not supported.

Alias

KLBL

Keyword abbreviation:

Not supported.

KEYSEQCHK keyword

Use the KEYSEQCHK keyword to specify what to do if a key sequence error is detected.

KEYSEQCHK is an optional keyword for the following commands:

- ANALYZE
- CHANGE
- EXTRACT
- RELOAD
- UNLOAD

Related keywords:

- [“KEYCHGCHK keyword” on page 489](#)
- [“PTRCHK keyword” on page 505](#)

Format:**NOCHECK**

Do not check any key sequence.

WARNING

When RESTART=NO is specified, the segment that has a key sequence error is processed and a warning message is issued. Processing continues.

KEYSEQCHK=WARNING cannot be specified for the ANALYZE command or when RESTART=YES.

BYPASS

When RESTART=NO is specified, the segment that has a key sequence error and its child or twin segments are bypassed and a warning message is issued. Processing resumes with the next valid segment in the hierarchical chain.

KEYSEQCHK=BYPASS cannot be specified for the ANALYZE command or when RESTART=YES.

ABEND

When RESTART=NO is specified, the job ends with an abend code of U3003 when any key sequence error is detected.

When RESTART=YES is specified, the processing for the area that has key sequence error ends with an error, but the remaining areas are processed. To reprocess the areas that have key sequence errors in the restart processing, specify KEYSEQCHK=ABEND.

KEYSEQCHK=ABEND cannot be specified for the ANALYZE command.

ERROR

The key sequence check is processed during the Analyze process. If a segment that has a key sequence error is detected, the Analyze process issues a resolve pointer report message and creates the Pointer Chain Reconstruction report and the CI map/CI dump report, which show details about the key sequence error.

KEYSEQCHK=ERROR is effective only for ANALYZE, RELOAD, and CHANGE commands.

When KEYSEQCHK=ERROR is specified, sequence error segment data records are not created in the HFPSEQER data set.

Default:

KEYSEQCHK=NOCHECK

Site default support:

Supported.

Keyword abbreviation:

Supported.

LARGEREC_COUNT keyword

Use the LARGEREC_COUNT keyword to specify the number of reporting entries in the Largest Database Records report and the Consolidated Largest Database Records report.

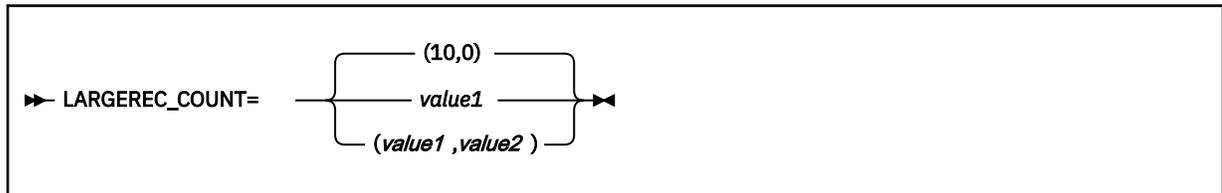
This keyword is effective only when PTRCHKLVL=FULL is specified.

LARGEREC_COUNT is an optional keyword for the REPORT subcommand.

Related keywords:

[“LARGEREC_REPORT keyword” on page 491](#)

Format:



value1

The number of reporting entries for the Largest Database Records report. The value is in the range of 0 - 9999. If 0 is specified, the Largest Database Records report is not generated.

value2

The number of reporting entries for the Consolidated Largest Database Records report. The value is in the range of 0 - 9999. If 0 is specified, the Consolidated Largest Database Records report is not generated.

Default:

LARGEREC_COUNT=(10,0)

If only the first parameter is specified, 0 is used for the second parameter.

Site default support:

Supported.

Keyword abbreviation:

Supported.

Usage note:

LARGEREC_COUNT=(0,0) is mutually exclusive with LARGEREC_REPORT=YES.

LARGEREC_REPORT keyword

Use the LARGEREC_REPORT keyword to specify whether to generate the Largest Database Records report and the Consolidated Largest Database Records report.

This keyword is effective only when PTRCHKLVL=FULL is specified.

LARGEREC_REPORT is an optional keyword for the REPORT subcommand.

Related keywords:

[“LARGEREC_COUNT keyword” on page 491](#)

Format:**YES**

Generate the Largest Database Records report and the Consolidated Largest Database Records report.

NO

Do not generate the Largest Database Records report and the Consolidated Largest Database Records report.

Default:

LARGEREC_REPORT=NO

Site default support:

Supported.

Keyword abbreviation:

Supported.

LIKE keyword

Use the LIKE keyword to specify the SMS allocation attributes of an output data set that was created by dynamic allocation.

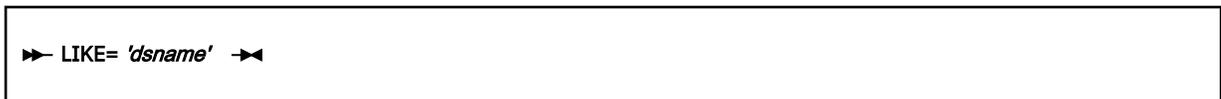
The model data set that you specify must be an existing SMS data set.

LIKE is an optional keyword for the following subcommands:

- BLDMFILECTL
- FILECTL

Related keywords:

[“DSNAME keyword” on page 456](#)

Format:***dsname***

Specify a data set name or a data set mask with 1- to 64-characters. Enclose the *dsname* in single or double quotation marks. Use the following variables, as necessary, to create the mask:

&DBD

DBD name

&AREA

Area name

&AREA#

3-digit relative area number. (If you specify an area number that is greater than 999, FPA does not convert the mask variable to the relative area number and issues error message HFPU0099E.)

&AREAn#

n-digit relative area number. *n* is 1 - 5.

When the mask variable &AREAn# is used for an area whose relative area number is greater than *n*-digit, for example, when &AREA2# is used for an area whose relative area number is greater than or equal to 100, FPA does not convert the mask variable to the relative area number and issues error message HFPU0099E.

&DATE

Current date ("Dyyddd")

&TIME

Current time ("Thhmmss")

&JOB

Job name

&STEP

Step name

&PROCSN

Procedure step name

For example, if you want to create an output file from area AREA123 and want the attributes of the new file to be the same as another pre-existing SMS file, specify the following values:

```
DSNAME='HFP.TEST.DB.&AREA',LIKE='HFP.TEST.DB.&DBD'
```

The resulting data set name will be 'HFP.TEST.DB.AREA123'. This data set will have the same SMS attributes as the data set specified by the LIKE keyword. (HFP.TEST.DB.MYDBD)

If you specify DSNAME='HFP.TEST.DB.AREA&AREA4#',LIKE='HFP.TEST.DB.&DBD' and the relative area number is 1500, the resulting data set will be 'HFP.TEST.DB.AREA1500'.

This data set will have the same SMS attributes as the data set named by the LIKE keyword ('HFP.TEST.DB.MYDBD').

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

LIUGOPT keyword

Use the LIUGOPT keyword to specify the LIU global option module that is to be used by IMS Library Integrity Utilities. The Integrity Checker utility of IMS Library Integrity Utilities requires that DBRC be active.

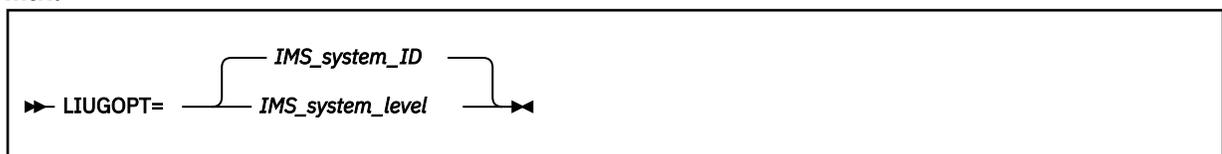
This keyword is effective only when all of the following conditions are met:

- IMS LIU load module library is concatenated to the JOBLIB/STEPLIB library.
- DBRC=YES is specified in HFPSYSIN.

LIUGOPT is an optional keyword for the GLOBAL command.

Related keywords:

["DBRC keyword" on page 451](#)

Format:**IMS_system_level**

Specify the IMS system ID with four characters. The LIU global option module named LIU@IMS_system_level will be used by IMS Library Integrity Utilities.

Default:

IMS system ID (system ID from DFSVC000)

Site default support:

Supported.

Keyword abbreviation:

Supported.

LOCATION keyword

Use the LOCATION keyword to specify the location to insert dependent segments that are specified by the SEGMENT keyword, which is either DOVF or IOVF.

LOCATION is an optional keyword for the LOADCTL subcommand.

Related keywords:

- [“INSERT_LIMIT_COUNT keyword” on page 484](#)
- [“SEGMENT keyword” on page 530](#)

Format:



DOVF

When the number of the segment insertions exceeds the value that is specified with INSERT_LIMIT_COUNT, all segments that should be far from their root segment will be placed in the available DOVF CI space. If the space is not enough in the DOVF CIs, they are placed in the IOVF part.

IOVF

When the number of segment insertions exceeds the value that is specified with INSERT_LIMIT_COUNT, all segments that should be far from their root will be placed in the available IOVF CI space.

Default:

LOCATION=IOVF

Site default support:

Supported.

Keyword abbreviation:

Supported.

LRECL keyword

Use the LRECL keyword to specify how to determine the LRECL of an unloaded segment records file when it is not specified in the DD control statement explicitly.

LRECL is an optional keyword for the UNLOAD command.

Related keywords:

- [“FORMAT keyword” on page 465](#)

Format:



SEGMENT

LRECL is determined on the basis of the maximum length of segments that are defined in DBD. This is the default.

SEGTFMT

LRECL is determined on the basis of the maximum length of segments and the maximum number of levels in DBD by specifying LRECL=SEGTFMT together with the FORMAT=TFMT keyword parameter.

Restriction:

Even though LRECL=SEGTFMT is specified, the number 3 is used as the maximum number of levels when the SDEP=PHYSICAL option is specified and the maximum number of segment levels is less than 3.

BLOCK

LRECL is determined as BLKSIZE - 4.

value

Value of LRECL. The value can be up to 32760. When the value is not appropriate, the system determined appropriate minimum value is overridden.

If FORMAT=USER is specified, these parameters are ignored except for LRECL=SEGMENT.

LRECL is determined on the basis of the following rules:

- The value of LRECL in JCL DD when it was specified. If the value is not appropriate, a valid one is determined and applied.
- The parameter of the LRECL keyword.
- The parameter of the LRECL keyword in site default.
- When DISP=NEW and no LRECL is specified, LRECL=SEGMENT is used.

Default:

LRECL=SEGMENT

Site default support:

Supported.

Keyword abbreviation:

Supported.

MAXERROR keyword

Use the MAXERROR keyword to specify the maximum number of pointer errors that might be encountered before the Analyzer process ends with an error.

MAXERROR is an optional keyword for the following commands:

- ANALYZE
- CHANGE
- RELOAD
- TUNE

Related keywords:

None.

Format:



value

Specify a value in the range of 0 - 999. If 0 is specified, there is no maximum number of errors.

Default:

MAXERROR=100

Site default support:

Supported.

Keyword abbreviation:

Supported.

MAXFAILS keyword

Use the MAXFAILS keyword to specify the maximum allowable number of UOWs that fail to be reorganized in an area, before the utility ends abnormally.

If the number of failures has exceeded the number specified in the MAXFAILS keyword, the reorganization processing is stopped and message HFPT0008E will be issued.

If MAXFAILS is specified as 0, then no limit is set and the reorganization continues the attempt to reorganize the next UOW until all UOW in the area has been processed or the reorganization limit, if it is specified, is reached.

MAXFAILS is an optional keyword for the REORG command.

Related keywords:

None.

Format:**value**

Specify a value in the range of 0 - 999.

Default:

MAXFAILS=5

Site default support:

Supported.

Keyword abbreviation:

Supported.

MGMTCLAS keyword

Use the MGMTCLAS keyword to specify the SMS management class of an output data set that was created by dynamic allocation.

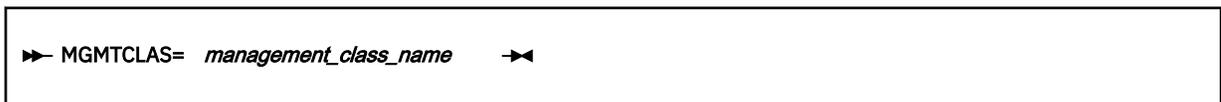
For more information about SMS management class, see the *MVS JCL Reference*.

MGMTCLAS is an optional keyword for the following subcommands:

- ALLOCATE
- BLDMPFILECTL
- FILECTL

Related keywords:

[“DSNAME keyword” on page 456](#)

Format:**management_class_name**

Specify a 1- to 8-character management class name.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

MSTIMESTAMP keyword

Use the MSTIMESTAMP keyword to specify whether to include a timestamp in the messages.

MSTIMESTAMP is an optional keyword for the GLOBAL command.

Related keywords:

None.

Format:**LOCAL**

Include a local timestamp in the messages.

NO

Do not include a timestamp in the messages.

Default:

MSTIMESTAMP=NO

Site default support:

Supported.

Keyword abbreviation:

Supported.

OAREA keyword

Use the OAREA keyword to specify one or more output areas for the process.

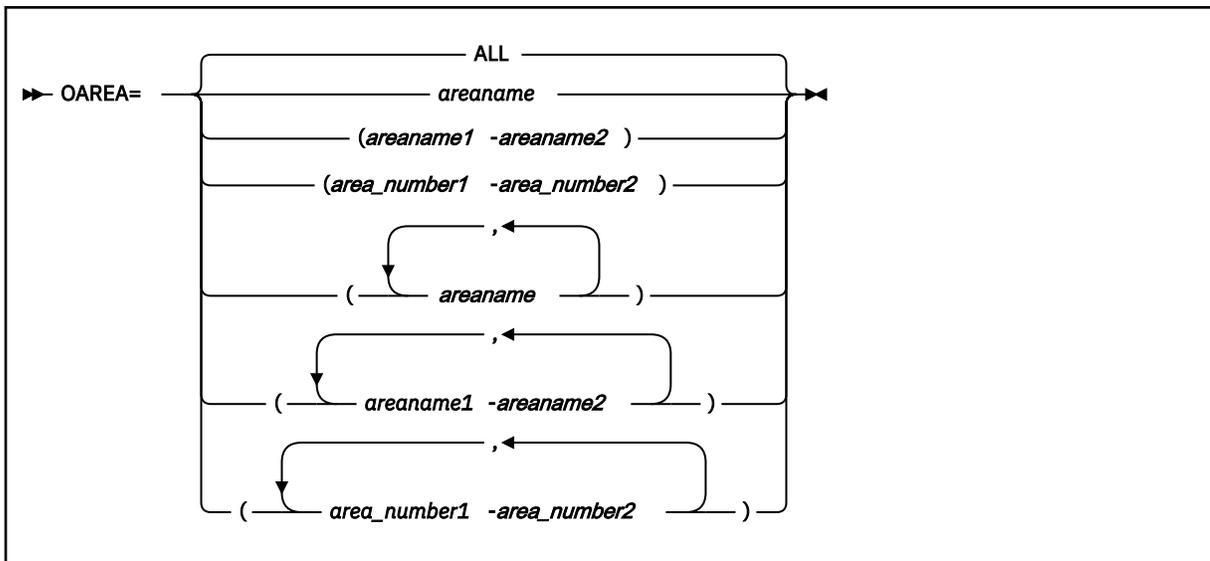
OAREA is a keyword for the following commands and subcommands:

- ALLOCATE
- AREADEF
- CHANGE
- FILECTL
- INITIALIZE
- RELOAD
- SIMULATE
- UNLOAD

Related keywords:

[“IAREA keyword” on page 469](#)

Format:



areaname

Specify a 1- to 8-character area name.

You can also use certain wild card characters to specify areas. These wild card characters are the asterisk (*), which represents from 0 to 8 characters, and the percent sign (%), which represents any single character.

You can specify multiple areas in the following format: OAREA=(*area1*,*area2*,*area3*)

You can also use multiple lines to specify multiple areas:

```
OAREA=(area1,  
       area2,  
       area3),
```

However, you cannot specify as follows:

```
OAREA=(  
       area1,  
       area2,  
       area3  
       ),
```

area_number1-area_number2

Specify the range of areas by area numbers.

area_number1 and *area_number2* are area numbers, which are decimal numbers assigned sequentially to the areas within a database. The value for *area_number1* and *area_number2* can be in the range of 1 - 2048. *area_number2* must be equal to or greater than *area_number1*. At least one area must exist in the specified range.

areaname1-areaname2

Specify the range of areas by area names.

areaname1 and *areaname2* must be specified in alphabetical order. At least one area must exist in the specified range.

For example, the following areas are specified in the DBD:

Area number	Area name
1	AABBCC
2	CBA
3	ABCD

Area number	Area name
4	C
5	A
6	TEST1ALL
7	ALLTEST1
8	AAA
9	AA
10	AREA1234

- If you specify OAREA=%B*, areas CBA and ABCD are selected.
- If you specify OAREA=A-AAA, areas A, AA, and AAA are selected.
- If you specify OAREA=6-7, areas TEST1ALL and ALLTEST1 are selected.

Default:

OAREA=ALL

Site default support:

Not supported.

Keyword abbreviation:

Supported.

Restrictions:

- If you specify the optional FILECTL subcommand to combine the targeted data that are in one or more output areas into a single output data set, you cannot specify the OAREA keyword on the UNLOAD command.
- If you specify the TUNE command or the INDEXBLD subcommand, you cannot specify the OAREA keyword on the FILECTL subcommand.
- The same area name cannot be specified on multiple OAREA keywords for the FILECTL subcommand.
- If you specify the OAREA keyword for one of the following commands or subcommands, you must specify the same areas for the IAREA keyword and the OAREA keyword. Otherwise, FPA issues message HFPC0061E and ends with an error.
 - FILECTL subcommand for the EXTRACT command
 - UNLOAD, CHANGE, or TUNE command in REORG mode

OCACHE keyword

Use the OCACHE keyword to specify the types of control intervals that are held in a data space storage until all processing has completed.

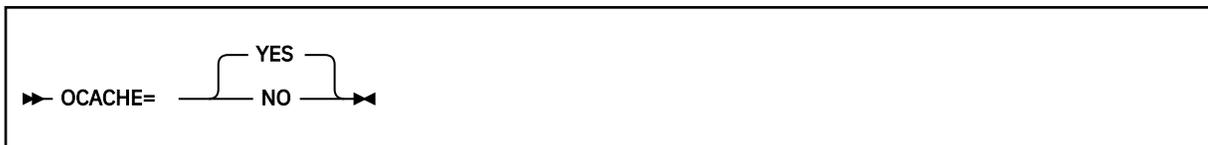
All control intervals held in a data space storage in this way are written to DASD when the data set is closed.

OCACHE is an optional keyword for the following commands:

- CHANGE
- RELOAD

Related keywords:

[“ICACHE keyword” on page 472](#)

Format:**YES**

Preload IOVF and SDEP parts into the data space.

NO

Do not preload IOVF and SDEP parts into the data space.

Default:

OCACHE=YES

Site default support:

Supported.

Keyword abbreviation:

Supported.

OCHECK keyword

Use the OCHECK keyword to control the concurrent number of output devices to access.

OCHECK is an optional keyword for the following commands:

- EXTRACT
- INITIALIZE
- RELOAD
- UNLOAD

Related keywords:

None.

Format:**NODEFER**

Gets access to all devices that are associated with the output data sets at a time.

DEFER

Access to devices for obtaining output data sets are limited to the number that is specified by the ITASKCTL keyword. This value is ignored in non-REORG mode (restructure).

Default:

OCHECK=NODEFER

Site default support:

Supported.

Keyword abbreviation:

Supported.

ODSNMASK keyword

Use the ODSNMASK keyword to specify the data set name or mask for dynamically allocating cataloged pre-existing output data sets.

When used with the INITIALIZE or the RELOAD command, ODSNMASK is used to allocate the output area being created. When used with the EXTRACT or UNLOAD command, ODSNMASK is used to allocate the

output file. You can make a relative reference to a GDG by specifying a relative generation number in the mask, a number enclosed in brackets such as in '*gdg-name*(-2)'

When the data set name contains a reference to a partitioned data set (PDS) member name, and it is used for dynamic allocation, you must specify DISP=SHR or OLD. When unloaded segment records files (USR files) are specified for two or more PDS data set member names, not only each member name but also each data set name must be unique. For an alternate method for specifying an output data set with EXTRACT or UNLOAD, see the FILECTL subcommand and its associated DSNAME keyword.

ODSNMASK is an optional keyword for the following commands:

- CHANGE
- EXTRACT
- INITIALIZE
- RELOAD
- UNLOAD

Related keywords:

None.

Format:

```
▶ ODSNMASK= 'dsname' ◀
```

dsname

Specify a data set name or a data set mask. Enclose *dsname* in single quotation marks. Use the following variables, as necessary, to create the mask.

&DBD

DBD name

&AREA

Area name

&AREA#

3-digit relative area number. (If you specify an area number that is greater than 999, FPA does not convert the mask variable to the relative area number and issues error message HFPU0099E.)

&AREAn#

n-digit relative area number. *n* is 1 - 5.

When the mask variable &AREA# is used for an area whose relative area number is greater than *n*-digit, for example, when &AREA2# is used for an area whose relative area number is greater than or equal to 100, FPA does not convert the mask variable to the relative area number and issues error message HFPU0099E.

For example, if you specify ODSNMASK='HFP.TEST.DB.&AREA' and the area name is AREA123, dynamic allocation of input data sets will dynamically allocate 'HFP.TEST.DB.AREA123'. If you specify ODSNMASK='HFP.TEST.DB.HFP&AREA4#' and the relative area number is 1500, the resulting data set name will be 'HFP.TEST.DB.HFP1500'.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

OEXITRTN keyword

Use the OEXITRTN keyword to specify the name of the user exit routine to be invoked prior to the write process of a Change function.

When this keyword is specified, you must either specify the EXITLIB DD statement or the library that contains the user exit routine in JOBLIB or STEPLIB DD.

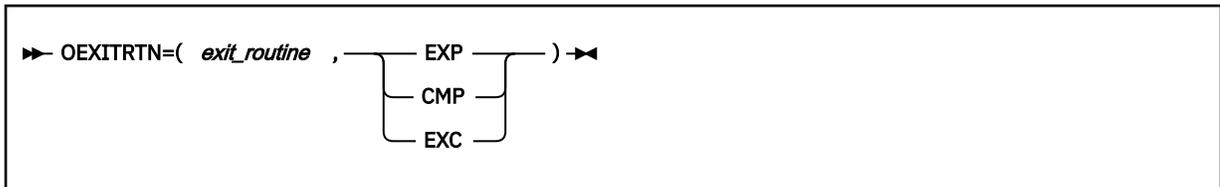
For each segment that is processed, the exit routine that is specified by the OEXITRTN=keyword parameter will be invoked with the specified function. If a segment is compressed by an IMS edit/compression routine, it will be expanded by the edit/compression routine before FPA calls a user exit routine. After returning from the user exit routine, the segment will be re-compressed by the edit/compression routine. Even if the segment is defined as fixed, the LL field of the segment is added and the record is passed to the user exit routine.

OEXITRTN is an optional keyword for the CHANGE command.

Related keywords:

- [“EXITRTN keyword” on page 461](#)
- [“IEXITRTN keyword” on page 479](#)

Format:



exit_routine

Specify a 1- to 8-character name of the user exit routine.

The second parameter specifies the function to be performed by the exit routine:

EXP

Expansion

CMP

Compression

EXC

Expansion then compression. This keyword value invokes the exit routine twice.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

Requirement:

All exit routines should be re-entrant so that they can be invoked by multiple subtasks. If the routine is not re-entrant, processing of the user exit routines for multiple subtasks is serialized, and the performance might be degraded. All exit routines must be capable of AMODE=31.

For the Interface of the exit routine, see the description of [“EXITRTN keyword” on page 461](#).

Related reference

[EXITRTN keyword](#)

Use the EXITRTN keyword to specify the name of the user exit routine to be invoked prior to the write process of an Unload or a Reload function.

OTASKCTL keyword

Use the OTASKCTL keyword to specify the number of resources to be processed concurrently.

- For the CHANGE command, the keyword specifies the maximum number of output areas to be processed concurrently.
- For the INDEXBLD command, the keyword specifies the maximum number of secondary index DBDs to be processed concurrently.
- For the TUNE command, the keyword specifies the maximum number of areas to be simulated concurrently.

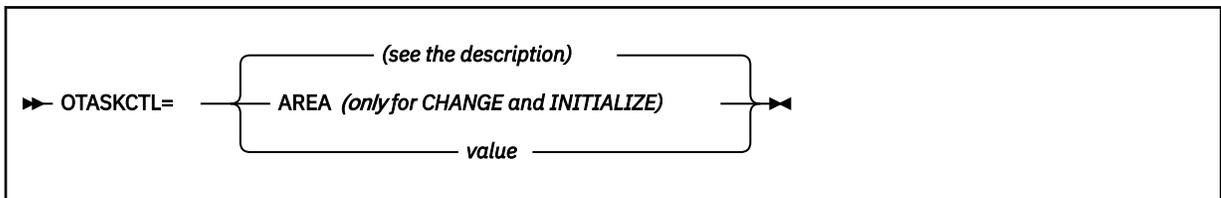
OTASKCTL is an optional keyword for the following commands:

- CHANGE
- INDEXBLD
- INITIALIZE
- TUNE

Related keywords:

[“ITASKCTL keyword” on page 486](#)

Format:



For the CHANGE or the INITIALIZE command:

AREA

The number of output areas.

value

Specify a value in the range of 1 - 2048. Specify a value less than or equal to the number of output areas. If you specify a value that is greater than the number of output areas, it will be replaced by the number of output areas. For the CHANGE command in the REORG mode, the following value is set:

$(ITASKCTL + OTASKCTL)/2$ (the value is rounded off)

For the INDEXBLD command:

value

Specify the number of secondary index databases to be processed concurrently.

The value can be in the range of 1 - 999. The value must be less than or equal to the number of output secondary index databases. If you specify a value that is greater than the number of output secondary index databases, FPA internally replaces this value with the number of secondary index databases.

For the Resync function, FPA processes the secondary index databases concurrently if either TOICTL=DBR or TOICTL=DBD is specified. If TOICTL=NONE is specified, FPA processes multiple secondary index databases concurrently when it reads the databases and generates resync pointer segment records. However, updates to the secondary index databases are done serially, one secondary index database at a time.

For the TUNE command:**value**

Specify the number of areas to simulate concurrently. The value can be in the range of 1 - 999. Specify a value that is less than or equal to the number of simulated areas. If you specify a value that is greater than the number of simulated areas, FPA internally replaces this value with the number of simulated areas.

Default:

- For the CHANGE, INDEXBLD, and INITIALIZE commands, OTASKCTL=4.
- For the TUNE command, the ITASKCTL value.

Site default support:

Supported.

Keyword abbreviation:

Supported.

PAD keyword

Use the PAD keyword to specify the padding character to be used when the segment minimum length is increased due to DBD definition change. The PAD keyword is effective for fixed-length segments and variable-length segments.

PAD is an optional keyword for the following commands:

- CHANGE
- UNLOAD

Related keywords:

None.

Format:**padding_character**

Code the value as two hexadecimal digits, for example X'nn'.

Default:

PAD=X'00'

Site default support:

Supported.

Keyword abbreviation:

Supported.

PROC_AFT_DUPKEY keyword

Use the PROC_AFT_DUPKEY keyword to specify whether to end the job step when the number of duplicate keys reaches the threshold defined by the DUPKEYMAX keyword.

PROC_AFT_DUPKEY is an optional keyword for the INDEXBLD command.

Related keywords:

["DUPKEYMAX keyword" on page 458](#)

Format:**CONT**

Continues processing the subsequent secondary index databases even when the number of duplicate keys reaches the threshold.

STOP

Stops the job when the number of duplicate keys reaches the threshold.

Default:

PROC_AFT_DUPKEY=CONT

Site default support:

Supported.

Keyword abbreviation:

Not supported.

PTRCHK keyword

Use the PTRCHK keyword to specify what to do when a pointer error is detected.

PTRCHK is an optional keyword for the following commands:

- CHANGE
- EXTRACT
- UNLOAD

Related keywords:

[“KEYSEQCHK keyword” on page 490](#)

Format:**ABEND**

When RESTART=NO is specified, the job ends with an abend code of U3003 when any pointer error is detected.

When RESTART=YES is specified, the processing for the area that has pointer errors ends with an error, but the remaining areas are processed. To reprocess the areas that have pointer errors in the restart processing, specify PTRCHK=ABEND or do not specify the PTRCHK keyword.

BYPASS

When RESTART=NO is specified, the segment that has a pointer error and its child or twin segments are bypassed. Processing resumes with the next valid segment in the hierarchical chain.

When RESTART=YES is specified, PTRCHK=BYPASS cannot be specified.

DETAIL

When RESTART=NO is specified, the segment that has a pointer error and its child or twin segments are bypassed, and the detailed information of pointer error is printed in the Unloaded report. Processing resumes with the next valid segment in the hierarchical chain.

When RESTART=YES is specified, PTRCHK=DETAIL cannot be specified.

This value can be specified only on the UNLOAD command.

Default:

PTRCHK=ABEND

Site default support:

Supported.

Keyword abbreviation:

Supported.

PTRCHKLVL keyword

Use the PTRCHKLVL keyword to select the method that is used to run integrity verification in RAA BASE, DOVF and IOVF.

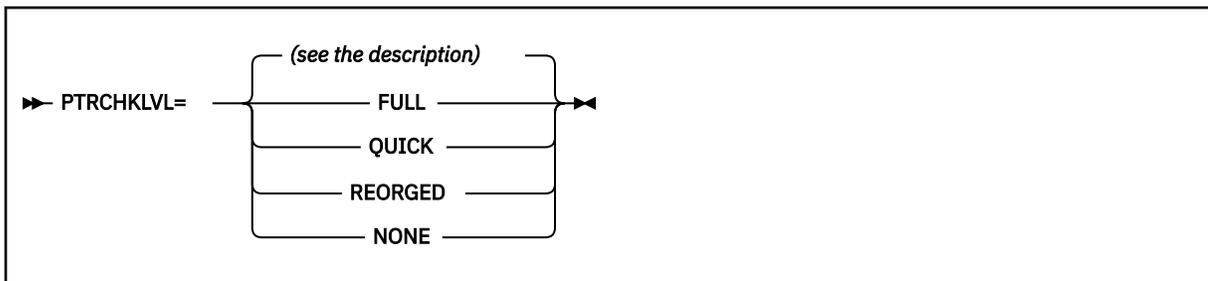
PTRCHKLVL is an optional keyword for the following commands:

- ANALYZE
- CHANGE
- GLOBAL
- RELOAD
- REORG

When you specify the INDEXDBD keyword for the ANALYZE command, you must specify PTRCHKLVL=FULL.

Related keywords:

[“SDEPCHKLVL keyword” on page 526](#)

Format:**FULL**

Do a cross-reference validation of pointers to the root and direct dependent segments.

QUICK

Do a checksum validation of pointers to the root and direct dependent segments.

REORGED

Do a checksum validation of pointers to the root and direct dependent segments for the UOWs that have been reorganized. This value can be specified only on the REORG command.

NONE

Do not run the Analyze function. You cannot specify this value on the ANALYZE command. Even if this value is specified on the GLOBAL command, it has no effect on the ANALYZE command.

Default:

- For the ANALYZE, CHANGE, and RELOAD commands, PTRCHKLVL=QUICK
- For the REORG command, PTRCHKLVL=REORGED
- For the GLOBAL command, the default parameter for the function unique command

Site default support:

- For the ANALYZE, CHANGE, RELOAD, and REORG commands, site default is supported.
- For the GLOBAL command, site default is not supported.

Keyword abbreviation:

Supported.

RAAFS_PCT keyword

Use the RAAFS_PCT keyword to specify a threshold setting for the percentage of free space in RAA BASE.

If the percentage is less than the value that you specified, a warning message is issued.

RAAFS_PCT is an optional keyword for the THRESHOLD subcommand.

Related keywords:

None.

Format:

```
▶▶ RAAFS_PCT= value ◀◀
```

value

Specify a value in the range of 0 - 100.

Default:

None.

Site default support:

Not supported.

Alias:

RFSP

Keyword abbreviation:

Supported.

RAP_LONGSYN_PCT keyword

Use the RAP_LONGSYN_PCT keyword to specify a threshold setting for the percentage of RAPs that have a long synonym chain among all the RAPs that are used. With this keyword, you specify a threshold value and the lower bound for synonym chain length.

FPA evaluates all the RAPs that are used, calculates the percentage of the RAPs that have a longer synonym chain than the lower bound, and compares the calculated value against the threshold. If the value exceeds the threshold, FPA issues a warning message.

RAP_LONGSYN_PCT is an optional keyword for the THRESHOLD subcommand, and can be used with the TUNE command.

Related keywords:

[“SYN_PCT_LEN keyword” on page 548](#)

Format:

```
▶▶ RAP_LONGSYN_PCT=( threshold ,lower_bound ) ◀◀
```

threshold

Specify the threshold value in the range of 0 - 100 for the percentage of RAPs that have a long synonym chain.

lower_bound

Specify the lower bound in the range of 1 - 99. FPA regards any RAP that has the number of synonyms equal to or greater than the lower bound as a *RAP with a long synonym chain*.

FPA evaluates the threshold by using the following expression:

$$PCT(lower_bound) = \frac{\sum_{n=lower_bound+1}^{\max} (\text{Number of RAPs that have } n \text{ root segments})}{\sum_{n=1}^{\max} (\text{Number of RAPs that have } n \text{ root segments})} * 100 > threshold$$

For example, assuming that the ROOTS PER RAP DISTRIBUTION GRAPH section in the DEDB Area Analysis reports displays the following graph:

```

ROOTS PER RAP DISTRIBUTION GRAPH
      1   2   3   4   5   6   7   8   9   10  11  12  % RAPS
      0   0   0   0   0   0   0   0   0   0   0   0
-----
# 0- |***** (48050)
  1- |***** (43996)
R 2- |***** (20056)
O 3- |*** (6201)
O 4- |* (1364)
T 5- |* (280)
S 6- |* (41)
  7- |* (9)
  8- |* (3)
  9- |
R 10- |
A 11- |
P 12- |

```

You specify RAP_LONGSYN_PCT=(10,2). Then, the percentage of RAPs that have two or more synonyms is calculated as follows:

$$PCT(lower_bound) = \frac{(6201 + 1364 + 280 + 41 + 9 + 3)}{(43996 + 20056 + 6201 + 1364 + 280 + 41 + 9 + 3)} * 100 = 10.98\%$$

Because the resulted value exceeds the value that you specified with the *threshold* parameter, which is 10, FPA issues a warning message.

FPA calculates the percentage of RAPs by using the Synonym Chain Analysis table. Therefore, if the value of the SYNTBL keyword that is specified in the HFPAPARM DD or the default value of the SYNTBL keyword is less than or equal to the *lower_bound* value of the RAP_LONGSYN_PCT keyword, FPA replaces the value of the SYNTBL keyword with the *lower_bound* value plus 1.

Default:

None.

Site default support:

Not supported.

Alias:

RLP

Keyword abbreviation:

Supported.

RAPOVFL_PCT keyword

Use the RAPOVFL_PCT keyword to specify a threshold setting for the percentage of RAA BASE CIs that use either DOVF or IOVF.

If the percentage is greater than the value that you specified, a warning message is issued.

RAPOVFL_PCT is an optional keyword for the THRESHOLD subcommand.

Related keywords:

None.

Format:

```
▶▶ RAPOVFL_PCT= value ◀◀
```

value

Specify a value in the range of 0 - 100.

Default:

None.

Site default support:

Not supported.

Alias:

ROP

Keyword abbreviation:

Supported.

RBASEFS keyword

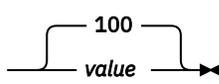
Use the RBASEFS keyword to specify a criterion to select UOWs to reorganize.

RBASEFS is an optional keyword for the REORG command.

Related keywords:

None.

Format:

```
▶▶ RBASEFS=  ◀◀
```

value

Specifies the percentage of free space in each RAP CI. If a RAP CI contains free space that is greater than the specified percentage, and the RAP CI also uses overflow CIs, the UOW is selected as a candidate for the reorganization. Specify an integer value in the range of 0 - 100. The maximum value 100 has a special meaning; if RBASEFS=100 is specified, the RBASEFS condition is not used to select an initial set of UOWs.

Default:

RBASEFS=100

Site default support:

Supported.

Keyword abbreviation:

Supported.

RDOVFFS keyword

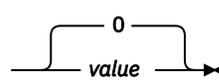
Use the RDOVFFS keyword to specify a criterion to select UOWs to reorganize.

RDOVFFS is an optional keyword for the REORG command.

Related keywords:

None.

Format:

```
▶▶ RDOVFFS=  ◀◀
```

value

Specifies the percentage of free space in the dependent overflow area in each UOW. If the free space percentage in the dependent overflow area in a UOW is greater than the specified percentage, and the UOW also uses IOVF CIs, the UOW is selected as a candidate for the reorganization. Specify an integer value in the range of 0 - 100. The maximum value 100 has a special meaning; if RDOVFFS=100 is specified, the RDOVFFS condition is not used to select an initial set of UOWs.

Default:

RDOVFFS=0

Site default support:

Supported.

Keyword abbreviation:

Supported.

RECIOVF_PCT keyword

Use the RECIOVF_PCT keyword to specify a threshold setting for the percentage of records that use IOVF CIs.

If the percentage is greater than the value that you specified, a warning message is issued.

RECIOVF_PCT is an optional keyword for the THRESHOLD subcommand.

Related keywords:

None.

Format:

```
▶▶ RECIOVF_PCT= value ▶▶
```

value

Specify a value in the range of 0 - 100.

Default:

None.

Site default support:

Not supported.

Alias:

RECIP

Keyword abbreviation:

Supported.

RECIO_AVG keyword

Use the RECIO_AVG keyword to specify a threshold setting for the average number of I/Os that can be done to access a database record.

If the number is greater than the value that you specified, a warning message is issued.

RECIO_AVG is an optional keyword for the THRESHOLD subcommand.

Related keywords:

None.

Format:

```
▶▶ RECIO_AVG= value ▶▶
```

value

Specify a value in the range of 0 - 32767.

Default:

None.

Site default support:

Not supported.

Alias:

RECIA

Keyword abbreviation:

Supported.

RECIO_MAX keyword

Use the RECIO_MAX keyword to specify to a threshold setting for the maximum number of I/Os that can be done to access a database record.

If the number is greater than the value that you specified, a warning message is issued.

RECIO_MAX is an optional keyword for the THRESHOLD subcommand.

Related keywords:

None.

Format:

```
►► RECIO_MAX= value ◄◄
```

value

Specify a value in the range of 0 - 32767.

Default:

None.

Site default support:

Not supported.

Alias:

RECIM

Keyword abbreviation:

Supported.

RECOVOPT keyword

Use the RECOVOPT keyword to specify the method for recovering an area when the process ends abnormally.

RECOVOPT is an optional keyword for the REORG command.

Related keywords:

None.

Format:

```
►► RECOVOPT=  ◄◄
```

NORMAL

The method for recovering an area depends on the first parameter value specified for the RFSSCAN keyword.

- RFSSCAN=(YES,*any*)

After processing the UOW that was being processed at the time the area ended abnormally, the REORG function ranks the set of UOWs that meet the RBASEFS and RDOVFFS criteria and continues to process the remaining UOWs in the order they are ranked.

Note: In the recovery process, the UOWs that were successfully reorganized in the previous job will not be reorganized because they will not meet the RBASEFS and RDOVFFS criteria.

- RFSSCAN=(NO,*any*)

Processes the UOW that was being processed at the time the area ended abnormally and continues to process the remaining UOWs in ascending order.

EMERGENCY

Process only the UOW that was being processed at the time the area ended abnormally. Other UOWs, even those that are not reorganized, will not be processed.

Default:

RECOVOPT=NORMAL

Site default support:

Supported.

Keyword abbreviation:

Supported.

RELOCATE_SDEP keyword

Use the RELOCATE_SDEP keyword to relocate the SDEP information.

SDEP relocation performs the following actions:

- Adjusts the SDEP Logical Begin to the first SDEP CI, and reduces the size of SDEP part.
- Dissolves the wraparound state of the SDEP parts of input ADS or IC, and reduces the size of SDEP part.

If you do not specify SDEP=PHYSICAL, this keyword is ignored.

RELOCATE_SDEP is an optional keyword for the UNLOAD and CHANGE commands.

Related keywords:

[“SDEP keyword” on page 524](#)

Format:



YES

Relocate the SDEP information.

NO

Do not relocate the SDEP information.

Default:

RELOCATE_SDEP=NO

Site default support:

Supported.

Keyword abbreviation:

Supported.

REORGLIM keyword

Use the REORGLIM keyword to specify the reorganization limit for the amount of UOWs that can be reorganized in the area.

The limit can be specified in the form of a maximum number of UOWs or a maximum percentage of the number of UOWs within the total number of UOWs defined in the area.

If a non-negative integer value is specified, this keyword specifies the maximum number of UOWs that can be reorganized. If the number exceeds the total number of UOWs that is defined in the area, the total number of UOWs is used as the reorganization limit. If a non-negative integer value followed by a % sign is specified, this keyword specifies the maximum percentage of the number of UOWs within the total number of UOWs that is defined in the area. The number must be less than or equal to 100.

If you just want to see the UOW Space Statistics report without reorganizing a UOW, specify REORGLIM=0, RFSSTAT=YES, and YES for the first parameter of RFSSCAN keyword.

If RFSSCAN=(NO,NO) or RFSSCAN=(NO,YES) is specified, the REORGLIM keyword cannot be specified.

REORGLIM is an optional keyword for the REORG command.

Related keywords:

- [“RFSSCAN keyword” on page 519](#)
- [“RFSSTAT keyword” on page 520](#)

Format:



value

You can specify this parameter in the following manner:

- Specify a non-negative integer value less than 32,767.
- Specify a non-negative integer value less than or equal to 100 that is followed by a % sign.

Default:

REORGLIM=100%

Site default support:

Supported.

Keyword abbreviation:

Supported.

REPORTDD keyword

Use the REPORTDD keyword to specify the destination *ddname* for specific reports.

REPORTDD is an optional keyword for the following subcommands:

- REPORT
- SIMULATE

For the REPORT subcommand, this keyword specifies the destination *ddname* for the following reports:

- DEDB Area Analysis reports
- UOW report
- UOW Exception report
- Largest Database Records report
- Consolidated Largest Database Records report

- Segment Length Distribution report
- DB Record Length Distribution report
- RBA of Used IOVF CI report

For the SIMULATE subcommand, this keyword specifies the destination *ddname* for the following reports:

- Area Tuning Summary report
- TUNED DEDB Area Analysis reports
- TUNED UOW report
- TUNED UOW Exception report
- TUNED Largest Database Records report
- TUNED Consolidated Largest Database Records report

Related keywords:

None.

Format:



ddname

Specify a 1- to 8-character ddname for the reports data set.

Default:

REPORTDD=HFP RPTS

Site default support:

Not supported.

Keyword abbreviation:

Supported.

RESTART keyword

Use the RESTART keyword to enable the checkpoint restart option for reprocessing failed areas.

RESTART is an optional keyword for the GLOBAL command.

This keyword is valid for the UNLOAD and RELOAD command processing.

Related keywords:

[“RESTART_COUNT keyword” on page 515](#)

Format:



YES

Enable the checkpoint restart option to reprocess only the failed areas.

When YES is specified, the HFPCHKPT DD must be specified.

For UNLOAD command processing, RESTART=YES is mutually exclusive with PTRCHK=BYPASS, PTRCHK=DETAIL, KEYSEQCHK=WARNING, or KEYSEQCHK=BYPASS.

NO

Do not enable the checkpoint restart option.

Default:

RESTART=NO

Site default support:

Supported.

Keyword abbreviation:

Supported.

Related concepts[Checkpoint restart option](#)

The Unload and the Reload functions of FPA support checkpoint restart capability. You can enable the checkpoint restart option to reprocess only the failed areas, without reprocessing previously successfully processed areas, by rerunning the same JCL stream that was used in the previous run.

RESTART_COUNT keyword

Use the RESTART_COUNT keyword to specify the maximum number of restart processing runs for the checkpoint restart option.

RESTART_COUNT is an optional keyword for the GLOBAL command.

This keyword is valid only when RESTART=YES is specified.

Related keywords:

[“RESTART keyword” on page 514](#)

Format:**value**

Specify a value in the range of 1 - 99.

When the number of restart processing runs exceeds the specified value, the contents of the checkpoint file are cleared. If the same JCL is submitted after that, the job is executed in checkpoint mode, which means that all the specified areas, whether they have been previously processed or not, are processed.

Default:

RESTART_COUNT=5

Site default support:

Supported.

Keyword abbreviation:

Supported.

Related concepts[Checkpoint restart option](#)

The Unload and the Reload functions of FPA support checkpoint restart capability. You can enable the checkpoint restart option to reprocess only the failed areas, without reprocessing previously successfully processed areas, by rerunning the same JCL stream that was used in the previous run.

RESYNCDSNMASK keyword

Use the RESYNCDSNMASK keyword to specify the data set mask for dynamically allocating cataloged pre-existing resync pointer segment record data sets or unverified resync pointer segment record data sets.

You can make a relative reference to a GDG by specifying a relative generation number in the mask, a number enclosed in brackets such as in '*gdg-name(-2)*'.

When the data set name contains a reference to a partitioned data set (PDS) member name and it is used for dynamic allocation, you must specify DISP=SHR or OLD.

You cannot use the FILECTL subcommand if you specify the RESYNCDSNMASK keyword.

RESYNCDSNMASK is an optional keyword for the INDEXBLD command.

Related keywords:

None.

Format:

```
➤ RESYNCDSNMASK= 'dsname' ➤
```

dsname

Specify a data set mask. Enclose the *dsname* in single quotation marks.

Symbolic variable &INDEX must be specified in the data set mask. &INDEX defines the secondary index DBD name. For example, if you specify RESYNCDSNMASK= 'HFP.&INDEX' and INDEXDBD=INDEXA, a data set named HFP.INDEXA is used for the resync pointer segment record data set or unverified resync pointer segment record data set.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

Related concepts

[Resync work records](#)

Resync pointer segment records and unverified resync pointer segment records are intermediate work records that the Resync function uses to pass data between Resync jobs.

Related reference

[Resync pointer segment records and duplicate pointer segment records layout](#)

Resync pointer segment records are created by the FPA Resync function and duplicate pointer segment records are created by the FPA Build Index function.

RESYNCDUMP keyword

Use the RESYNCDUMP keyword to specify whether to generate the Resync Pointer Segment Dump report. The index pointer segments that were or that will be updated to synchronize the secondary index databases with the primary DEDB database are written to the report.

RESYNCDUMP is an optional keyword for the INDEXBLD command.

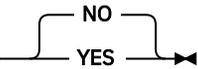
RESYNCDUMP keyword is effective only when both of the following keywords are specified:

- IDXPROC=RESYNC
- TOICTL=NONE

Related keywords:

[“RESYNCDUMPMAX keyword” on page 517](#)

Format:

```
➤ RESYNCDUMP=  ➤
```

YES

Generate the Resync Pointer Segment Dump report.

NO

Do not generate the Resync Pointer Segment Dump report.

Default:

RESYNCDUMP=NO

Site default support:

Supported.

Keyword abbreviation:

Supported.

Related concepts[Resync work records](#)

Resync pointer segment records and unverified resync pointer segment records are intermediate work records that the Resync function uses to pass data between Resync jobs.

Related reference[Resync pointer segment records and duplicate pointer segment records layout](#)

Resync pointer segment records are created by the FPA Resync function and duplicate pointer segment records are created by the FPA Build Index function.

RESYNCDUMPMAX keyword

Use the RESYNCDUMPMAX keyword to specify the maximum number of index pointer segments to be printed for each index database in the Resync Pointer Segment Dump report.

RESYNCDUMPMAX is an optional keyword for the INDEXBLD command.

Related keywords:

[“RESYNCDUMP keyword” on page 516](#)

Format:**value**

Print index pointer segments that were or that will be updated to synchronize the secondary index databases with the primary DEDB database, up to the number specified. Specify a value in the range of 1 - 32767.

Default:

RESYNCDUMPMAX=100

Site default support:

Supported.

Keyword abbreviation:

Supported.

RESYNCMODE keyword

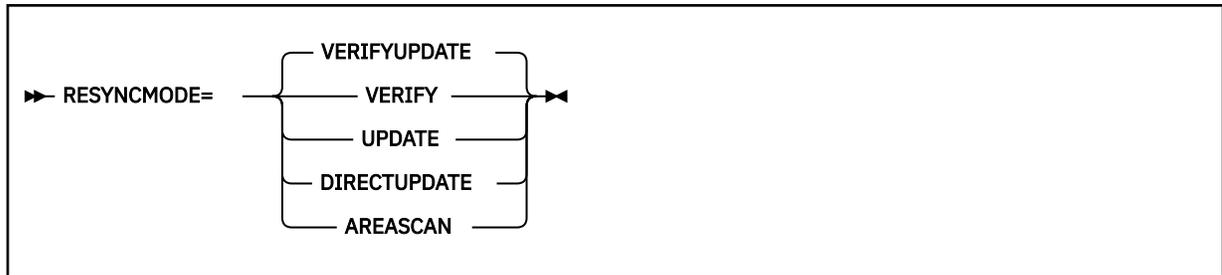
Use the RESYNCMODE keyword to specify the synchronization mode for synchronizing secondary index databases with their primary DEDB database.

RESYNCMODE is an optional keyword for the INDEXBLD command.

RESYNCMODE keyword is effective only when IDXPROC=RESYNC is specified.

Related keywords:

None.

Format:**VERIFYUPDATE**

The job runs in verify update mode. FPA scans the DEDB areas to read the source segments and verifies the index pointer segments against the source segments in the primary DEDB database. Then, it updates the index pointer segments in the secondary index databases.

If you specify the AREASCAN=NO option, FPA does not scan the DEDB areas. Instead, FPA reads the records in the unverified resync pointer segment records that were generated in area scan mode.

VERIFY

The job runs in verify mode. FPA verifies the index pointer segments against the source segments in the primary DEDB database. Then, it writes the index pointer segments that need to be updated to the resync pointer segment record data sets.

UPDATE

The job runs in update mode. FPA reads records in the resync pointer segment record data sets. Then, it updates the index pointer segments in the secondary index databases.

DIRECTUPDATE

The job runs in direct update mode. FPA scans the DEDB areas to read the source segments. Then, it inserts index pointer segments in the secondary index databases.

In direct update mode, index pointer segments in the secondary index databases are not read, therefore, pointers are not verified. If an index pointer segment that FPA attempts to insert already exists in the secondary index databases, the insert operation fails.

If you specify the AREASCAN=NO option, FPA does not scan the DEDB areas. Instead, FPA reads the records in the unverified resync pointer segment records that were generated in area scan mode.

AREASCAN

The job runs in area scan mode. FPA scans the DEDB areas, generates index pointer segments based on the source segments, and writes them to the unverified resync pointer segment record data sets.

Default:

RESYNCMODE=VERIFYUPDATE

Site default support:

Supported.

Keyword abbreviation:

Supported.

Related concepts

[Resync work records](#)

Resync pointer segment records and unverified resync pointer segment records are intermediate work records that the Resync function uses to pass data between Resync jobs.

Related reference

[Resync pointer segment records and duplicate pointer segment records layout](#)

Resync pointer segment records are created by the FPA Resync function and duplicate pointer segment records are created by the FPA Build Index function.

RETPD keyword

Use the RETPD keyword to specify the retention period of an output data set that was created by dynamic allocation.

Do not specify both the EXPDT keyword and the RETPD keyword because they are mutually exclusive.

For more information, see the *MVS JCL Reference*.

RETPD is an optional keyword for the following subcommands:

- BLDMFILECTL
- FILECTL

Related keywords:

- [“DSNAME keyword” on page 456](#)
- [“EXPDT keyword” on page 462](#)

Format:

```
▶▶ RETPD= value ◀◀
```

value

Specify the number of days to retain the data set. Valid values are 0 - 32767.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

RFSSCAN keyword

Use the RFSSCAN keyword to specify whether to gather the statistics before and after reorganization.

The RFSSCAN keyword takes two parameters that are enclosed by a pair of parenthesis and separated by a comma.

In order for REORGLIM keyword to be effective, the first parameter of RFSSCAN keyword must be YES.

In order for RFSSTAT=YES to be effective, at least one of the first and second parameters of the RFSSCAN keyword must be YES. If both the first and second parameters of RFSSCAN are NO, RFSSTAT=YES is ignored.

RFSSCAN is an optional keyword for the REORG command.

Related keywords:

- [“REORGLIM keyword” on page 513](#)
- [“RFSSTAT keyword” on page 520](#)

Format:

```
▶▶ RFSSCAN= ( YES , YES ) ◀◀
              |   |
              |   |
              NO  NO
```


Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

ROOT keyword

Use the ROOT keyword to simulate the areas with a new ROOT parameter. The ROOT parameter controls the number of UOWs in the area.

When this keyword is specified, the Tune function replaces the ROOT parameter that is defined in the DMB in the ACBLIB data set and simulates the area condition with the new ROOT parameter. Use this keyword to tune the number of UOWs.

ROOT is an optional keyword for the AREADEF subcommand.

Related keywords:

- [“CISIZE keyword” on page 448](#)
- [“UOW keyword” on page 553](#)

Format:

```
►► ROOT=( number ,overflow ) -◄◄
```

number

Specifies the total number of UOWs for the area. The value must be in the range of 2 - 32767.

overflow

Specifies the number of UOWs in the overflow section of the area.

The value must be equal to or greater than 1 and smaller than the total number of UOWs (*number*).

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

ROOTIO_AVG keyword

Use the ROOTIO_AVG keyword to specify a threshold setting for the average number of I/Os to access a root segment.

If the number is greater than the value that you specified, a warning message is issued.

ROOTIO_AVG is an optional keyword for the THRESHOLD subcommand.

Related keywords:

None.

Format:

```
►► ROOTIO_AVG= value -◄◄
```

value

Specify a value in the range of 0 - 32767.

Default:

None.

Site default support:

Not supported.

Alias:

ROOTIA

Keyword abbreviation:

Supported.

ROOTIO_MAX keyword

Use the ROOTIO_MAX keyword to specify a threshold setting for the maximum number of I/Os that can be done to access a root segment.

If the number is greater than the value that you specified, a warning message is issued.

ROOTIO_MAX is an optional keyword for the THRESHOLD subcommand.

Related keywords:

None.

Format:

```
▶▶ ROOTIO_MAX= value ◀◀
```

value

Specify a value in the range of 0 - 32767.

Default:

None.

Site default support:

Not supported.

Alias:

ROOTIM

Keyword abbreviation:

Supported.

SCAN keyword

Use the SCAN keyword to verify the syntax of command, keywords, and parameters in command sets without running FPA.

The Scan function also allocates some (but not all) internal resources for running the process function. Information concerning the command are routed to the data set that is referred to by the HFPRPTS DD statement.

SCAN is an optional keyword for the GLOBAL command.

Related keywords:

- [“SITE_DSNAME keyword” on page 539](#)
- [“SITE_MEMBER keyword” on page 540](#)

Format:



NO

Verify the syntax and parameters of the command and run FPA.

YES

Verify the syntax and parameters of the command but do not run FPA.

SITE_CREATE

Create a site default table. More than one command, their related subcommands, and their keywords can be specified to define site default values for the keywords. This parameter specifies that the command analysis is to be run, but does not get the access to DBRC.

SITE_REPORT

Report on the site default table. Commands and keywords are ignored. This parameter does not get access to DBRC.

Default:

SCAN=NO

Site default support:

Not supported.

Keyword abbreviation:

Supported.

SDEP keyword

Use the SDEP keyword to specify how SDEP segment data is to be processed.

If the DEDB that is being processed has an SDEP segment defined, SDEP is an optional keyword for the following commands:

- CHANGE
- UNLOAD

Related keywords:

None.

Format:



LOGICAL

Processing in logical order means that all valid SDEP segments are written in the new area or the unload file. The SDEP segments in each area of the new database area are physically reordered into the new area/RAP sequence of their parent root segment, but retain their logical order (entry sequence) within each database record. SORT=YES must be specified in Unload or Reload process.



Attention: If your application relies on the marker segment concept, do not use SDEP=LOGICAL.

PHYSICAL

Processing in physical order results in all segments between the SDEP logical beginning and logical end being extracted in physical sequence and inserted into the SDEP part of the new area. The SDEP segments remain in the same physical order (this mode fully supports the marker segment concept). The segment records of the SDEP CIs are unloaded as 512-byte blocks of CI data (they are not unloaded as discrete segments.)

Basically, the SDEP=PHYSICAL option has two purposes:

- Improve the performance of the unload process
- Keep the SDEP marker

If FORMAT=USER is specified for the UNLOAD command, this parameter is ignored.

Restrictions:

- When SDEP=PHYSICAL is specified, compressed segments will not be expanded even if you specify COMPRESS=YES.
- If you specify SDEP=PHYSICAL and RELOCATE_SDEP=NO, the job must be processed in REORG mode.
- If you specify SDEP=PHYSICAL and RELOCATE_SDEP=YES, the job can be processed in REORG mode or in non-REORG mode. However, the following restrictions apply:
 - You cannot change the compression routine.
 - If a DBD change is involved, you cannot change the CI size. Also, database records cannot be randomized to a different area.

NO

No SDEP process occurs when SDEP segments are not written in the new area or the unloaded file.

Default:

SDEP=NO

Site default support:

Not supported.

Keyword abbreviation:

Supported.

Related reference

[Determining how to process SDEP segments](#)

IMS HP Fast Path Utilities provides several utilities to process SDEP segments. You can use these utilities to analyze and maintain SDEP segments.

SDEP_BYPASS_MSGLVL keyword

Use the SDEP_BYPASS_MSGLVL keyword to specify the severity level of the message that FPA issues when the SDEP is defined but SDEP segment data is not unloaded because SDEP=NO is specified.

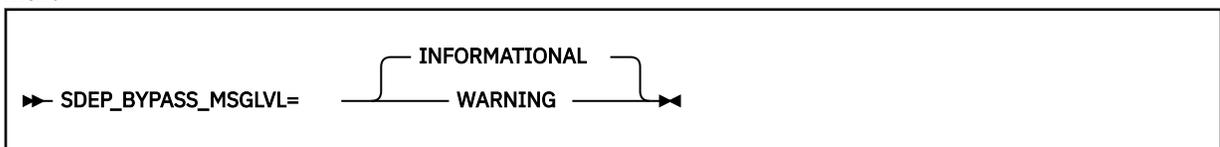
SDEP_BYPASS_MSGLVL is an optional keyword for the following commands:

- CHANGE
- UNLOAD

Related keywords:

[“SDEP keyword” on page 524](#)

Format:



INFORMATIONAL

Issues informational message HFPC0159I when the SDEP is defined but SDEP segment data is not unloaded.

WARNING

Issues warning message HFPC0160W when the SDEP is defined but SDEP segment data is not unloaded. FPA sets the return code to 4 and continues processing.

Default:

SDEP_BYPASS_MSGLVL=INFORMATIONAL

Site default support:

Supported.

Keyword abbreviation:

Not supported.

SDEPCHKLVL keyword

Use the SDEPCHKLVL keyword to select the method that is used to run an integrity verification in the SDEP part.

SDEPCHKLVL is an optional keyword for the following commands:

- ANALYZE
- CHANGE
- GLOBAL
- RELOAD
- REORG

Related keywords:

[“PTRCHKLVL keyword” on page 506](#)

Format:**FULL**

Do a cross-reference validation of pointers to SDEP segments. Either PTRCHKLVL=FULL or PTRCHKLVL=QUICK is required.

QUICK

Do a checksum validation of pointers to SDEP segments. Either PTRCHKLVL=FULL or PTRCHKLVL=QUICK is required.

NONE

Do not run the SDEP processing.

Default:

SDEPCHKLVL=QUICK

Site default support:

- For the ANALYZE, CHANGE, RELOAD, and REORG commands, site default is supported.
- For the GLOBAL command, site default is not supported.

Keyword abbreviation:

Supported.

Related reference

[Determining how to process SDEP segments](#)

IMS HP Fast Path Utilities provides several utilities to process SDEP segments. You can use these utilities to analyze and maintain SDEP segments.

SDEP_ORPHAN_MSGLVL keyword

Use the SDEP_ORPHAN_MSGLVL keyword to specify the error message level for an orphan SDEP segment. If you do not specify SDEPCHKLVL=FULL, this keyword is ignored.

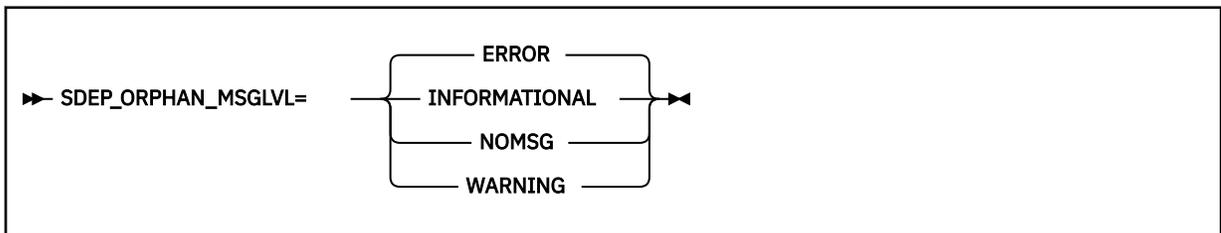
SDEP_ORPHAN_MSGLVL is an optional keyword for the following commands:

- ANALYZE
- CHANGE
- GLOBAL
- RELOAD
- REORG

Related keywords:

[“SDEPCHKLVL keyword” on page 526](#)

Format:



ERROR

Generate an error message on the Processing Messages report and an integrity verification message that contains the RBA of the orphan segment on the Resolve Pointer report. The return code is set to 8.

INFORMATIONAL

Generate an informational message on the Processing Messages report and an integrity verification message that contains the RBA of the orphan segment on the Resolve Pointer report.

NOMSG

Do not generate a message.

WARNING

Generate a warning message on the Processing Messages report and an integrity verification message that contains the RBA of the orphan segment on the Resolve Pointer report. The return code is set to 4.

Default:

SDEP_ORPHAN_MSGLVL=ERROR

Site default support:

- For the ANALYZE, CHANGE, RELOAD, and REORG commands, site default is supported.
- For the GLOBAL command, site default is not supported.

Keyword abbreviation:

Supported.

Related reference

[Determining how to process SDEP segments](#)

IMS HP Fast Path Utilities provides several utilities to process SDEP segments. You can use these utilities to analyze and maintain SDEP segments.

SDEP_PTRIOVF_MSGLVL keyword

Use the SDEP_PTRIOVF_MSGLVL keyword to specify the severity level of the message that is issued when detecting an SDEP pointer that points to the IOVF portion. If you specify SDEPCHKLVL=NONE, this keyword is ignored.

When you use the OAE utility to extend the IOVF portion of a DEDB area, the utility does not update the SDEP pointer value of each segment. Therefore, after the IOVF extension, some SDEP pointers might remain pointing to the address in the extended IOVF portion that was formerly in the SDEP portion. If you run the FPA Analyze function against such a DEDB and the function detects SDEP pointers pointing to the IOVF portion, FPA issues pointer error messages. However, you can ignore such pointer error messages because such pointers do not represent problems. By using the SDEP_PTRIOVF_MSGLVL keyword, you can suppress such pointer error messages or change them to informational messages (messages with the I suffix).

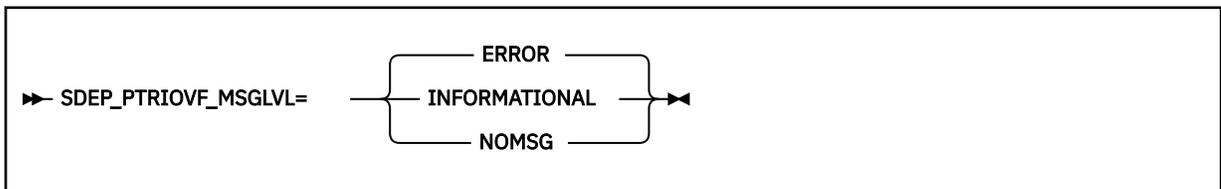
SDEP_PTRIOVF_MSGLVL is an optional keyword for the following commands:

- ANALYZE
- REORG

Related keywords:

[“SDEPCHKLVL keyword” on page 526](#)

Format:



ERROR

Set the return code to 8 and generate the following messages:

- Error message in the Processing Messages report
- Integrity verification message that identifies the RBA of the segment whose SPCF or SPTF pointer points to the IOVF portion in the Resolve Pointers report

INFORMATIONAL

Set the return code to 0 and generate the following messages:

- Informational message in the Processing Messages report
- Integrity verification message that identifies the RBA of the segment whose SPCF or SPTF pointer points to the IOVF portion in the Resolve Pointers report

NOMSG

Set the return code to 0 and generate no messages about the pointer.

Default:

SDEP_PTRIOVF_MSGLVL=ERROR

Site default support:

Supported.

Keyword abbreviation:

Not supported.

Related reference

[Determining how to process SDEP segments](#)

IMS HP Fast Path Utilities provides several utilities to process SDEP segments. You can use these utilities to analyze and maintain SDEP segments.

SEGLDIST keyword

Use the SEGLDIST keyword to specify whether to generate the Segment Length Distribution report. SEGLDIST is an optional keyword for the REPORT subcommand.

Related keywords:

None.

Format:



YES

Print the Segment Length Distribution report.

SEGLDIST=YES is effective only when PTRCHKLVL=FULL is specified.

NO

Do not print the Segment Length Distribution report.

Default:

SEGLDIST=NO

Site default support:

Supported.

Keyword abbreviation:

Supported.

SEGLSTAT keyword

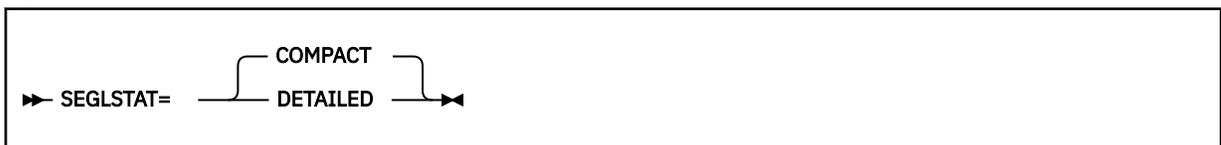
Use the SEGLSTAT keyword to specify the format of the DB Record Profile Analysis report, which is one of the DEDB Area Analysis reports.

SEGLSTAT is an optional keyword for the REPORT subcommand.

Related keywords:

None.

Format:



COMPACT

The DB Record Profile Analysis report provides analysis information of the database records. It does not include details such as prefix length and data flags.

DETAILED

The DB Record Profile Analysis report provides prefix length and data flags in addition to the default contents. This additional information is helpful in database tuning.

Default:

SEGLSTAT=COMPACT

Site default support:

Supported.

Keyword abbreviation:

Supported.

SEGMENT keyword

Use the SEGMENT keyword to specify the segment name for processing by its associated subcommand.

SEGMENT is a required keyword for the following subcommands:

- LOADCTL
- OUTPUT
- SELECT

Related keywords:

- [“INSERT_LIMIT_COUNT keyword” on page 484](#)
- [“LOCATION keyword” on page 494](#)

Format:

For the OUTPUT or the SELECT subcommand:

```
➤ SEGMENT= segment ➤
```

For the LOADCTL subcommand:

```
➤ SEGMENT=( segment , { ONLY  
DEPENDENTS  
BOTH } ) ➤
```

segment

Specify a 1- to 8-character alphanumeric name of the segment that is coded on the SEGM statement in your DBD. You cannot specify the root segment or the SDEP segment for the LOADCTL subcommand.

The following parameters are available only when the SEGMENT keyword is specified for the LOADCTL subcommand:

ONLY

The LOADCTL function applies only to the specified segment. You can also specify O instead of ONLY.

DEPENDENTS

The LOADCTL function applies to the dependent segments of the specified segment, and not to the segment itself. You can also specify D instead of DEPENDENTS.

BOTH

The LOADCTL function applies to both the specified segment and its dependent segments. You can also specify B instead of BOTH.

Default:

None. For the LOADCTL subcommand, if only the segment name is specified, ONLY is used for the second parameter.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

SEGREC_PREFIX keyword

Use the SEGREC_PREFIX keyword to specify the contents of the fields within the prefix portion of the segment records, which are to be written to an extract or an unload file.

This keyword is valid only when EFORMAT=USER is specified with the EXTRACT command, or FORMAT=USER is specified with the UNLOAD command or the RELOAD command.

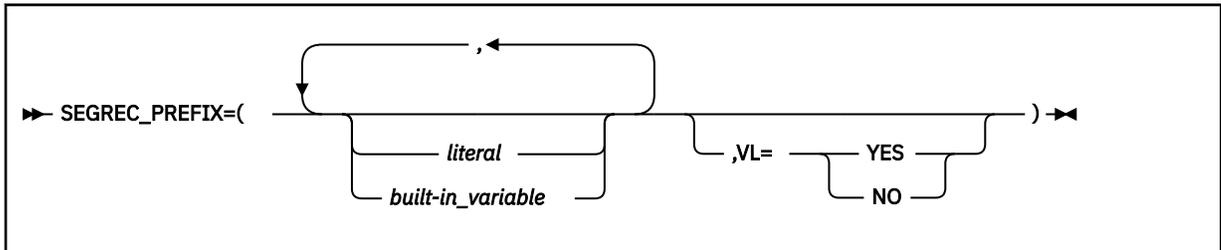
SEGREC_PREFIX is an optional keyword for the following commands:

- EXTRACT
- RELOAD
- UNLOAD

Related keywords:

[“SEGREC_SUFFIX keyword” on page 533](#)

Format:



literal

Specify one or more alphabetic or numeric characters that are contained within apostrophes (' ') or within quotation marks (" "). You can specify an optional data-type declaration as follows:

- C'xxxxxx' (character): Case-sensitive character data. You must specify a minimum of 1 and maximum of 28 characters (28 bytes).
- F'xxxxxx' (signed binary): Must contain only numeric digits. You must specify a minimum of 1 and maximum of 19 digits (4 bytes).
- P'xxxxxx' (packed decimal): Assumed positive. Must contain only numeric digits. You must specify a minimum of 1 and maximum of 31 digits (16 bytes).
- X'xxxxxx' (unsigned binary): Must be an even number of hexadecimal characters. You must specify a minimum of two hexadecimal characters (representing 1 byte) and a maximum of 56 hexadecimal characters (representing 28 bytes).

You can also declare the length, by specifying it before the data-type declaration.

Example:

```
8C 'CUSDB'
2P '01'
```

Built-in_variable

The following built-in variables can be specified.

Built-in variable name	Data type	Contents
AREA_NAME	8C	Area name
AREA_NUMBER	1X	Area number (1 byte)
AREA_NUMBER2	2X	Area number (2 bytes)
DBD_AREAS	2F	Number of areas in DBD
DBD_LEVELS	2F	Maximum hierarchic level in DBD
DBD_NAME	8C	DBD name

Built-in variable name	Data type	Contents
DBD_SEGMENTS	2F	Number of segment types in DBD
KEY or SEGMENT_KEY	nC	Key for current segment
ROOT_KEY	nC	Key for root segment
ROOT_KEY_LENGTH	2F	Length of root segment key
RUNDATE	4X	Date when FPA started
RUNDATETIME	12X	Date and time when FPA started
RUNTIME	4X	Time when FPA started
SEGMENT_ATTR	1X	Segment attribute (Only for the UNLOAD or the RELOAD command) X'40': SDEP segment X'04': Fixed-length segment X'02': Compressed segment
SEGMENT_CKEY	nC	Concatenated key for current segment
SEGMENT_CKEY_LENGTH	2F	Length of concatenated key for current segment
SEGMENT_CODE	1X	Segment code
SEGMENT_KEY_LENGTH	2F	Length of current segment key
SEGMENT_LENGTH	2F	Length of segment (When COMPRESS=YES is specified, the length of the compressed segment)
SEGMENT_LEVEL	1X	Segment level
SEGMENT_NAME	8C	Segment name
SEGMENT_OCCURRENCE	4X	Segment code and segment sequence number for the current hierarchical level (Only for the UNLOAD or the RELOAD command)
SEGMENT_SCSQ_DEF	(X 3X) x n	Segment code and segment sequence number for the defined hierarchical level (Only for the UNLOAD or the RELOAD command)
SEGMENT_SCSQ_MAX	(X 3X) x 14	Segment code and segment sequence number for all hierarchical levels (Only for the UNLOAD or the RELOAD command)
TODAY	4X	Date when FPA started

VL=YES

Request a length prefix.

VL=NO

Suppress a length prefix.

Default:

VL=YES

Site default support:

Not supported.

Alias:

PREFIX

Keyword abbreviation:

Supported.

SEGREC_SUFFIX keyword

Use the SEGREC_SUFFIX keyword to specify the contents of the fields within the suffix portion of the segment records, which are written to an extract or an unload file.

This keyword is valid only when EFORMAT=USER is specified with the EXTRACT command, or FORMAT=USER is specified with the UNLOAD command or the RELOAD command.

SEGREC_SUFFIX is an optional keyword for the following commands:

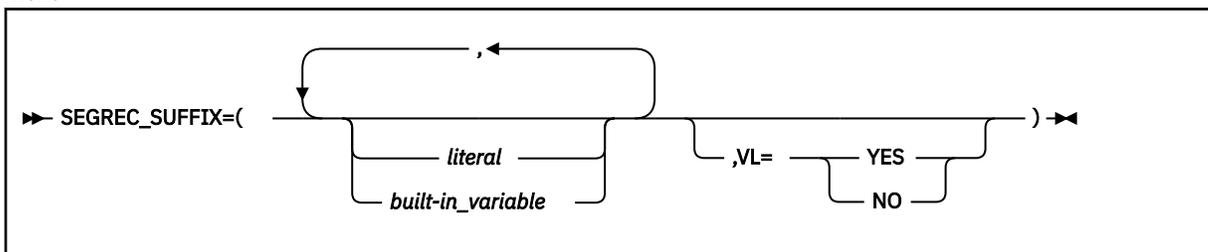
- EXTRACT
- RELOAD
- UNLOAD

Note: The specification of SEGREC_SUFFIX does not affect the RELOAD function.

Related keywords:

[“SEGREC_PREFIX keyword” on page 531](#)

Format:



literal

Specify one or more alphabetic or numeric characters that are contained within apostrophes (' ') or within quotation marks (" "). You can specify an optional data-type declaration as follows:

- C'xxxxxx' (character): Case-sensitive character data. You must specify a minimum of 1 and maximum of 28 characters (28 bytes).
- F'xxxxxx' (signed binary): Must contain only numeric digits. You must specify a minimum of 1 and maximum of 19 digits (4 bytes).
- P'xxxxxx' (packed decimal): Assumed positive. Must contain only numeric digits. You must specify a minimum of 1 and maximum of 31 digits (16 bytes).
- X'xxxxxx' (unsigned binary): Must be an even number of hexadecimal characters. You must specify a minimum of two hexadecimal characters (representing 1 byte) and a maximum of 56 hexadecimal characters (representing 28 bytes).

You can also declare the length, by specifying it before the data-type declaration.

Example:

```
8C 'CUSDB'  
2P '01'
```

Built-in_variable

The following built-in variables can be specified.

Built-in variable name	Data type	Contents
AREA_NAME	8C	Area name
AREA_NUMBER	1X	Area number (1 byte)
AREA_NUMBER2	2X	Area number (2 bytes)

Built-in variable name	Data type	Contents
DBD_AREAS	2F	Number of areas in DBD
DBD_LEVELS	2F	Maximum hierarchic level in DBD
DBD_NAME	8C	DBD name
DBD_SEGMENTS	2F	Number of segment types in DBD
KEY or SEGMENT_KEY	nC	Key for current segment
ROOT_KEY	nC	Key for root segment
ROOT_KEY_LENGTH	2F	Length of root segment key
RUNDATE	4X	Date when FPA started
RUNDATETIME	12X	Date and time when FPA started
RUNTIME	4X	Time when FPA started
SEGMENT_ATTR	1X	Segment attribute (Only for the UNLOAD or the RELOAD command) X'40': SDEP segment X'04': Fixed-length segment X'02': Compressed segment
SEGMENT_CKEY	nC	Concatenated key for current segment
SEGMENT_CKEY_LENGTH	2F	Length of concatenated key for current segment
SEGMENT_CODE	1X	Segment code
SEGMENT_KEY_LENGTH	2F	Length of current segment key
SEGMENT_LENGTH	2F	Length of segment (When COMPRESS=YES is specified, the length of the compressed segment)
SEGMENT_LEVEL	1X	Segment level
SEGMENT_NAME	8C	Segment name
SEGMENT_OCCURRENCE	4X	Segment code and segment sequence number for current hierarchical level (Only for the UNLOAD or the RELOAD command)
SEGMENT_SCSQ_DEF	(X 3X) x n	Segment code and segment sequence number for defined hierarchical level (Only for the UNLOAD or the RELOAD command)
SEGMENT_SCSQ_MAX	(X 3X) x 14	Segment code and segment sequence number for all hierarchical levels (Only for the UNLOAD or the RELOAD command)
TODAY	4X	Date when FPA started

VL=YES

Request a length prefix.

VL=NO

Suppress a length prefix.

Default:

VL=YES

Site default support:

Not supported.

Alias:

SUFFIX

Keyword abbreviation:

Supported.

SENSOR keyword

Use the SENSOR keyword to specify whether to store the sensor data in the Sensor Data repository of IMS Tools KB.

SENSOR is an optional keyword for the following commands:

- ANALYZE
- CHANGE
- RELOAD
- REORG

Related keywords:

- [“ADXCFGPR keyword” on page 439](#)
- [“ITKBLOAD keyword” on page 487](#)
- [“ITKBSRVR keyword” on page 488](#)
- [“SENSOR_DBREC keyword” on page 536](#)
- [“SENSOR_PTRERR keyword” on page 537](#)
- [“SENSOR_UOW keyword” on page 537](#)
- [“SENSOR_UOW_REPORT keyword” on page 538](#)

Format:**YES**

Store the sensor data in the Sensor Data repository of IMS Tools KB.

SENSOR=YES is effective only when PTRCHKLVL=FULL or PTRCHKLVL=QUICK is specified.

When SENSOR=YES is specified, you must specify the name of the IMS Tools KB server XCF group with the ITKBSRVR keyword. Also, you must specify the IMS Tools Base load module library on the STEPLIB.

Sensor data from the following areas is not stored:

- When SENSOR_PTRERR=YES is not specified, areas that have pointer errors
- When SDEP_ORPHAN_MSGLVL=ERROR is specified, areas that have orphan SDEP segments

NO

Do not store the sensor data.

Default:

SENSOR=NO

Site default support:

Supported.

Keyword abbreviation:

Supported.

For a list of the data elements that are collected by IMS HP Fast Path Utilities, see the topics about data elements in the *Tools Base Policy Services User's Guide*.

SENSOR_DBREC keyword

Use the SENSOR_DBREC keyword to specify whether to store the data elements that are related to database records and root segment distribution in the Sensor Data repository of IMS Tools KB.

The following data elements are related to database records and root segment distribution:

- DB_MAX_DBREC_LENGTH
- DB_MIN_DBREC_LENGTH
- DB_PCT_NUM_DBREC_IOVF
- DB_AVG_LEN_SYNONYM_CHAIN
- DB_MAX_LEN_SYNONYM_CHAIN
- DB_AVG_DBREC_IO
- DB_MAX_DBREC_IO
- DB_AVG_ROOT_IO
- DB_MAX_ROOT_IO

SENSOR_DBREC is an optional keyword for the following commands:

- ANALYZE
- CHANGE
- RELOAD
- REORG

Related keywords:

[“SENSOR keyword” on page 535](#)

Format:



YES

Store the sensor data for the data elements that are related to database records and root segment distribution in the Sensor Data repository of IMS Tools KB.

Requirement: When you specify SENSOR_DBREC=YES, you must also specify SENSOR=YES and PTRCHKLVL=FULL.

NO

Do not store the sensor data for the data elements that are related to database records and root segment distribution.

Considerations for SENSOR_DBREC and the policy evaluation process:

If SENSOR_DBREC=NO is specified, Policy Services and Autonomics Director cannot evaluate the data elements that are collected when SENSOR_DBREC=YES is specified, and they might not provide adequate information about the state of the area and the actions that you must take. Consider specifying SENSOR_DBREC=YES unless you have performance concerns.

When you specify SENSOR_DBREC=YES, the CPU time and the elapsed time increase compared to when SENSOR_DBREC=NO is specified.

Even if you specify SENSOR_DBREC=NO, rough estimations of the number of I/Os that are required to read database records and root segments are calculated and stored in the IMS Tools KB repository. These values are stored as DB_ESTIMATED_DB_REC_IO and DB_ESTIMATED_ROOT_IO data elements, and are collected regardless of the SENSOR_DBREC keyword specification.

Default:

SENSOR_DBREC=NO

Site default support:

Supported.

Keyword abbreviation:

Supported.

For a list of the data elements that are collected by IMS HP Fast Path Utilities, see the topics about data elements in the *Tools Base Policy Services User's Guide*.

SENSOR_PTRERR keyword

Use the SENSOR_PTRERR keyword to store the sensor data in the Sensor Data repository of IMS Tools KB even when pointer errors are detected in the area.

SENSOR_PTRERR is an optional keyword for the following commands:

- ANALYZE
- CHANGE
- RELOAD
- REORG

Related keywords:

[“SENSOR keyword” on page 535](#)

Format:**NO**

Do not store the sensor data in the Sensor Data repository of IMS Tools KB when pointer errors are detected in the area.

YES

Store the sensor data in the Sensor Data repository of IMS Tools KB even when pointer errors are detected in the area.

Default:

SENSOR_PTRERR=NO

Site default support:

Supported.

Keyword abbreviation:

Supported.

SENSOR_UOW keyword

Use the SENSOR_UOW keyword to specify whether to store the data elements that are related to UOW in the Sensor Data repository of IMS Tools KB.

The data elements that will be stored differ by the HFPAUOWC DD specification as follows:

- If you specify the HFPAUOWC DD statement, data elements are collected at the UOW group level (data elements that begin with DEBUG.)
- If you do not specify the HFPAUOWC DD statement, data elements are collected at the UOW level (data elements that begin with DBU.)

SENSOR_UOW is an optional keyword for the following commands:

- ANALYZE

- CHANGE
- RELOAD
- REORG

Related keywords:

- [“SENSOR keyword” on page 535](#)
- [“SENSOR_UOW_REPORT keyword” on page 538](#)

Format:



YES

Store the sensor data for the data elements that are related to UOW in the Sensor Data repository of IMS Tools KB.

Requirement: When you specify SENSOR_UOW=YES, you must also specify SENSOR=YES and SENSOR_DBREC=YES.

NO

Do not store the sensor data for the data elements that are related to UOW.

Considerations for SENSOR_UOW and the policy evaluation process

If you specify SENSOR_UOW=YES, data elements for each UOW or UOW group are collected. When you process many UOWs or UOW groups, be aware that the number of data elements that will be stored in the IMS Tools KB repository can be large. Processing a large number of data elements might increase elapsed time and require more storage space in the repository.

Default:

SENSOR_UOW=NO

Site default support:

Supported.

Keyword abbreviation:

Supported.

For a list of the data elements that are collected by IMS HP Fast Path Utilities, see the topics about data elements in the *Tools Base Policy Services User's Guide*.

SENSOR_UOW_REPORT keyword

Use the SENSOR_UOW_REPORT keyword to specify whether to print the data elements that are related to UOW in the Sensor Data Statistics report.

SENSOR_UOW_REPORT is an optional keyword for the following commands:

- ANALYZE
- CHANGE
- RELOAD
- REORG

Related keywords:

- [“SENSOR keyword” on page 535](#)
- [“SENSOR_UOW keyword” on page 537](#)

Format:**YES**

Print the data elements that are related to UOW in the Sensor Data Statistics report.

Requirement: When you specify `SENSOR_UOW_REPORT=YES`, you must also specify `SENSOR_UOW=YES`.

NO

Do not print the data elements that are related to UOW in the Sensor Data Statistics report.

Considerations for SENSOR_UOW_REPORT:

When you specify the HFPAUOWC DD statement, the `SENSOR_UOW=YES` keyword, and the `SENSOR_UOW_REPORT=YES` keyword, the data elements that are related to UOW are collected at the UOW group level and are printed in the Sensor Data Statistics report.

When you specify the `SENSOR_UOW=YES` keyword and the `SENSOR_UOW_REPORT=YES` keyword without an HFPAUOWC DD statement, the data elements that are related to UOW are collected at the UOW level and are printed in the Sensor Data Statistics report; however, the amount of data can be large and might require additional time to print the Sensor Data Statistics report.

Default:

`SENSOR_UOW_REPORT=NO`

Site default support:

Supported.

Keyword abbreviation:

Supported.

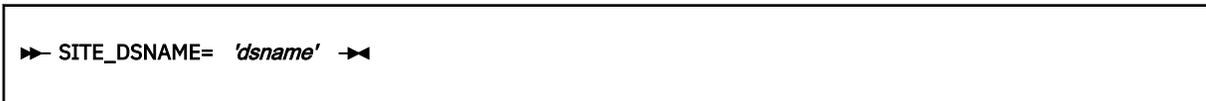
SITE_DSNAME keyword

Use the `SITE_DSNAME` keyword to specify the name of the data set that has the members of the site default tables.

`SITE_DSNAME` is an optional keyword for the GLOBAL command.

Related keywords:

- [“SCAN keyword” on page 523](#)
- [“SITE_MEMBER keyword” on page 540](#)

Format:***dsname***

Specify a data set name that is 1- to 44-characters. Enclose the *dsname* in single quotation marks. The HFPCSITE DD statement in the JCL is mutually exclusive with the `SITE_DSNAME` keyword in the HFPSYSIN DD statement. The data set attribute of HFPCSITE DD is PDS.

Default:

You can define the site default value for the keyword in the HFPCSITE module.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

Related tasks

[Specifying the site default table](#)This reference topic describes how to specify a site default table with the HFPCSITE module.

SITE_MEMBER keyword

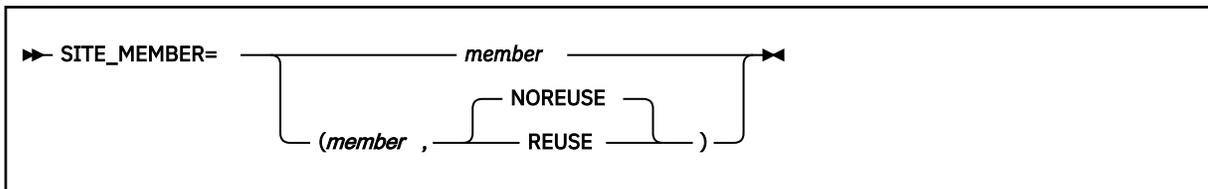
Use the SITE_MEMBER keyword to specify the member name of the site default table.

SITE_MEMBER is an optional keyword for the GLOBAL command.

Related keywords:

- [“SCAN keyword” on page 523](#)
- [“SITE_DSNAME keyword” on page 539](#)

Format:



member

Specify a member name of the site default table.

The second parameter is available only when SCAN=SITE_CREATE:

REUSE

When the member name exists, it is overridden. When the member name exists but there is no keyword to be added into the site table, the empty member is created.

NOREUSE

When the member name exists, the JOB ends with the error message HFPC0035E.

Default:

For *member*, you can define the site default value for the keyword in the HFPCSITE module. For the second parameter, default is NOREUSE.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

Related tasks

[Specifying the site default table](#)This reference topic describes how to specify a site default table with the HFPCSITE module.

SKIP keyword

Use the SKIP keyword to skip a number of segment occurrences under a parent.

If more than one of the EVERY, SKIP, or STOPAFT keyword is specified, SKIP is applied first, then EVERY, and then STOPAFT.

SKIP is an optional keyword for the SELECT subcommand.

Related keywords:

- [“EVERY keyword” on page 461](#)
- [“STOPAFT keyword” on page 545](#)

Format:

```
➤ SKIP= value ➤
```

value

Specify a value in the range of 0 - 2147483647. To select all segments, specify 0.

Default:

If the SKIP keyword is not specified, all segments are selected.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

SORT keyword

Use the SORT keyword to sort the records in unloaded segment record data sets or build pointer segment record data sets.

- For EXTRACT, UNLOAD, and RELOAD commands, the SORT keyword specifies to sort the unloaded segment records based on RAP (root anchor point).
- For the INDEXBLD command, the SORT keyword specifies to sort the build pointer segment records based on the record type and the key of the index pointer segment. The SORT keyword is effective only when the Build Index function runs in load mode (BUILDMODE=LOAD).

SORT is an optional keyword for the following command:

- EXTRACT
- INDEXBLD
- RELOAD
- UNLOAD

Related keywords:

None.

Format:

```
➤ SORT= { NO | YES | AUTO } ➤
```

NO

Do not sort the sequential file. If FORMAT=USER is specified for the RELOAD command, this parameter is changed to YES internally.

Important: If you specify SORT=NO for the INDEXBLD command, the Build Index function assumes that the records in the build pointer segment records are already sorted. If the records are not sorted, either specify SORT=YES or sort the records before you run the Build Index job.

YES

Sort the sequential file.

SORT=YES with the UNLOAD or RELOAD command is required only in one of the following scenarios:

- The SDEP=LOGICAL keyword with the UNLOAD command is used for the area that has SDEP segments.

- The FILECTL subcommand with the UNLOAD command that uses multiple UNLOAD tasks (not ITASKCTL=1) is used to group the target unload segment records that are in multiple areas into a single unloaded data set.
- The UNLOAD command is processed in the non-REORG mode.

For the UNLOAD command, this keyword is ignored when FORMAT=USER is specified or when all of the following conditions are met:

- A single area is unloaded into the file.
- The SDEP=LOGICAL keyword is not specified in HFPSYSIN.
- The Unload function is processed in the REORG mode.

For the EXTRACT command, this keyword is ignored when EFORMAT=HDUNLOAD or EFORMAT=USER is specified or when both of the following conditions are met:

- A single area is extracted into the file.
- The database that is specified in DBD does not have an SDEP segment.

AUTO

Enable the RELOAD command to sort the sequential file. This parameter is applicable only to the RELOAD command.

Default:

- For the EXTRACT, UNLOAD, and RELOAD commands, SORT=NO.
- For the INDEXBLD command, SORT=YES.

Site default support:

Supported.

Alias:

SOR

Keyword abbreviation:

Supported.

SORTOPT_FILSZ keyword

Use the SORTOPT_FILSZ keyword to specify that FPA adds the FILSZ parameter to the SORT control statement.

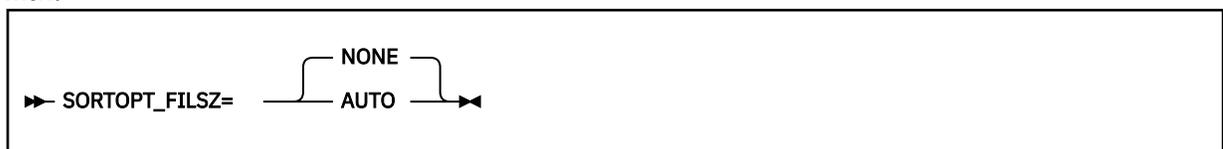
SORTOPT_FILSZ is an optional keyword for the following commands:

- ANALYZE
- CHANGE
- EXTRACT
- RELOAD
- REORG
- TUNE
- UNLOAD

Related keywords:

None.

Format:



AUTO

FPA uses the size of the areas and segment definitions to estimate the number of records to sort and adds the FILSZ parameter (FILSZ=Ennnnnnnn) to the SORT control statement.

Specify SORTOPT_FILSZ=AUTO if you received message ICE046A when DFSORT sorted the internal records of FPA.

NONE

FPA does not add the FILSZ parameter to the SORT control statement.

Even if you specify SORTOPT_FILSZ=NONE, if FPA can calculate the actual number of records to sort before the sort program is run, FPA adds the FILSZ parameter to the SORT control statement.

Default:

SORTOPT_FILSZ=NONE

Site default support:

Supported.

Keyword abbreviation:

Not supported.

SPACE keyword

Use the SPACE keyword to specify the space requirements of the output data set that is created by dynamic allocation.

For more information, see the *MVS JCL Reference*.

In an SMS environment, SPACE is an optional keyword for the following subcommands:

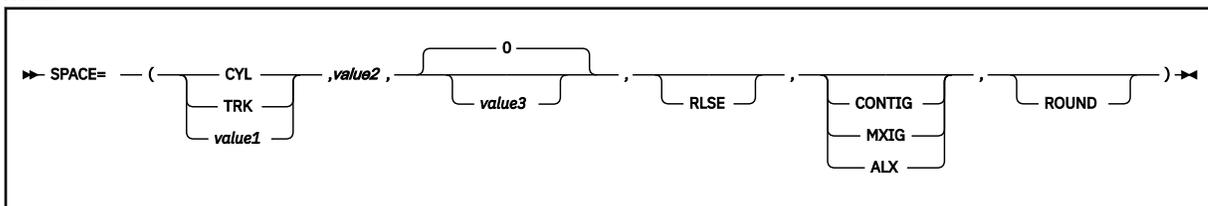
- ALLOCATE
- BLDMPFILECTL
- FILECTL

For non-SMS environment, SPACE is a required keyword for the ALLOCATE subcommand.

Related keywords:

- [“AVGREC keyword” on page 445](#)
- [“DSNAME keyword” on page 456](#)

Format:



The first parameter specifies the unit type for allocation. The following values are supported:

CYL

Allocation unit is in cylinders.

TRK

Allocation unit is in tracks.

value1

Allocation unit is either blocks or records, depending on the AVGREC keyword as follows:

- If the allocation unit is in blocks, the number specifies the average block size. Do not specify the AVGREC keyword if the unit type is in blocks.
- If the unit allocation is in records, the number specifies the average record size. Specify the AVGREC keyword to indicate that the allocation unit is in records.

Valid values are 512 - 32767.

The second parameter specifies the value for primary space for the allocated data set:

value2

Acceptable values are 1 - 16777215. The value represents the number of allocation units to be allocated.

You can optionally provide the following parameters:

value3

Specify a value for the space for allocating data sets when additional space is required. Acceptable values are 0 - 16777215. The value represents the number of allocation units to be allocated. The default value is 0.

RLSE

Specifies to release unused space.

CONTIG

Specifies that the space that is allocated for the data set must be contiguous on the volume. This subparameter applies only to primary space allocation.

MXIG

Specifies that the space that is allocated for the data set is the largest area of available contiguous space on the volume that is equal to or greater than the primary quantity.

ALX

Space that is allocated for the data set includes up to five separate contiguous areas on the volume, and each area must be equal to or greater than the primary quantity.

ROUND

Round the space to complete the number of allocated units.

For the ALLOCATE subcommand, you can specify up to three parameters. For the FILECTL subcommand, you can specify up to six parameters.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

SSP keyword

Use the SSP keyword to specify whether subset pointers should be retained or cleared.

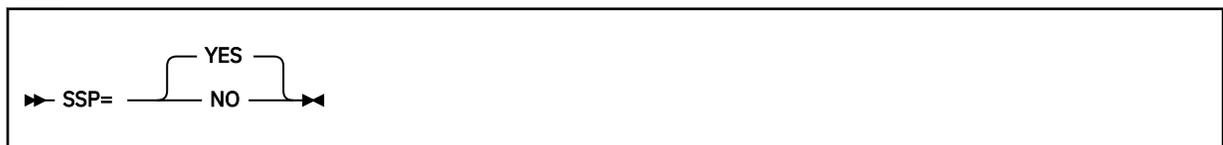
If subset pointers are defined in the DEDB to be processed, SSP is an optional keyword for the following commands:

- CHANGE
- RELOAD
- UNLOAD

Related keywords:

None.

Format:



YES

Retain subset pointers. If FORMAT=USER is specified for the UNLOAD command, this parameter is changed to NO internally.

NO

Do not retain subset pointers.

Default:

SSP=YES

Site default support:

Supported.

Keyword abbreviation:

Supported.

STARTUOW keyword

Use the STARTUOW keyword to specify the number that defines the first UOW to be reported in the UOW report and the UOW Exception report.

STARTUOW is an optional keyword for the REPORT subcommand.

Related keywords:

- [“STOPUOW keyword” on page 546](#)
- [“UOW_REPORT keyword” on page 555](#)

Format:**value**

Specify a value in the range of 0 - 32767. This value must be less than or equal to the value specified for the STOPUOW keyword.

Default:

STARTUOW=0

Site default support:

Supported.

Keyword abbreviation:

Supported.

STOPAFT keyword

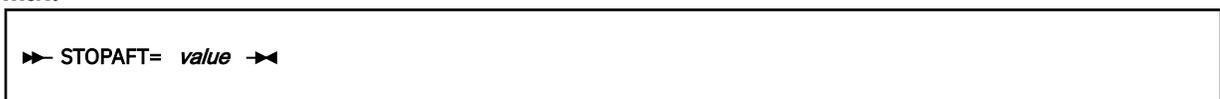
Use the STOPAFT keyword to specify the end of the selection of segments after a specific number of segment occurrences under a parent segment.

If more than one of the EVERY, SKIP, or STOPAFT keyword is specified, SKIP is applied first, then EVERY, and then STOPAFT.

STOPAFT is an optional keyword for the SELECT subcommand.

Related keywords:

- [“EVERY keyword” on page 461](#)
- [“SKIP keyword” on page 540](#)

Format:

value

Specify a value in the range of 0 - 2147483647. To set no limit, specify 0.

Default:

If the STOPAFT keyword is not specified, all segments are selected.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

STOPUOW keyword

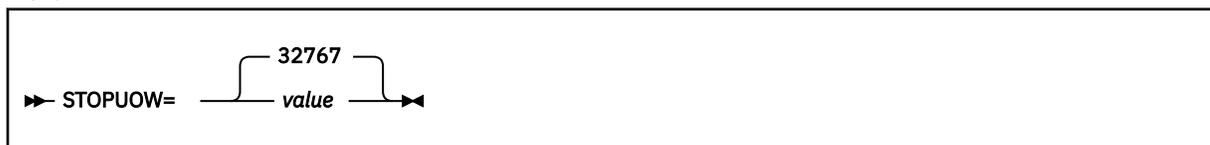
Use the STOPUOW keyword to specify the number that defines the last UOW to be reported in the UOW report and the UOW Exception report.

STOPUOW is an optional keyword for the REPORT subcommand.

Related keywords:

- [“STARTUOW keyword” on page 545](#)
- [“UOW_REPORT keyword” on page 555](#)

Format:



value

Specify a value in the range of 0 - 32767. This value must be greater than or equal to the value specified for the STARTUOW keyword.

Default:

STOPUOW=32767

Site default support:

Supported.

Keyword abbreviation:

Supported.

STORCLAS keyword

Use the STORCLAS keyword to specify the SMS storage class of the output data set that is created by dynamic allocation.

For more information about SMS storage class, see the *MVS JCL Reference*.

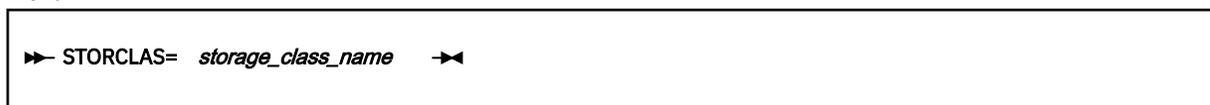
STORCLAS is an optional keyword for the following subcommands:

- ALLOCATE
- BLDMPFILECTL
- FILECTL

Related keywords:

[“DDNAME keyword” on page 453](#)

Format:



storage_class_name

Specify a 1- to 8-character storage class name.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

SYNC keyword

Use the SYNC keyword to specify the interval between the synchronization points for updating secondary index databases with DL/I calls.

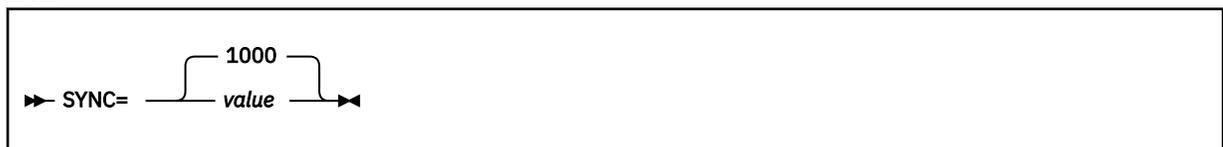
SYNC is an optional keyword for the GLOBAL command.

SYNC is effective only when all of the following keywords are specified:

- IDXPROC=RESYNC
- TOICTL=NONE
- One of RESYNCMODE=VERIFYUPDATE, RESYNCMODE=UPDATE, and RESYNCMODE=DIRECTUPDATE

Related keywords:

None.

Format:**value**

Specify the number of index pointer segments to insert or delete until a synchronization point is reached. Valid values are 1 - 9999999.

For example, if you specify SYNC=1000, a SYNC call is issued for each 1000 DL/I ISRT or DLET calls.

Default:

SYNC=1000

Site default support:

Supported.

Keyword abbreviation:

Supported.

SYN_LEN_AVG keyword

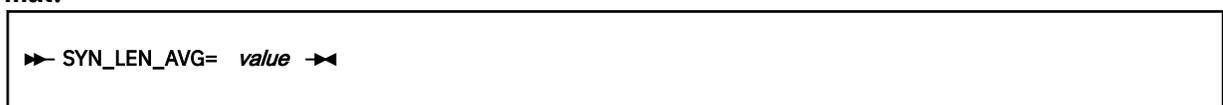
Use the SYN_LEN_AVG keyword to specify a threshold setting for the average synonym chain length.

If the length is greater than the value that you specified, a warning message is issued.

SYN_LEN_AVG is an optional keyword for the THRESHOLD subcommand.

Related keywords:

None.

Format:**value**

Specify a value in the range of 0 - 32767.

Default:

None.

Site default support:

Not supported.

Alias:

SCLA

Keyword abbreviation:

Supported.

SYN_MAX keyword

Use the SYN_MAX keyword to specify a threshold setting for the maximum synonym chain length.

If the length is greater than the value that you specified, a warning message is issued.

SYN_MAX is an optional keyword for the THRESHOLD subcommand.

Related keywords:

None.

Format:

```
▶▶ SYN_MAX= value ▶▶
```

value

Specify a value in the range of 0 - 32767.

Default:

None.

Site default support:

Not supported.

Alias:

SCM

Keyword abbreviation:

Supported.

SYN_PCT_LEN keyword

Use the SYN_PCT_LEN keyword to specify a threshold setting for the percentage of RAPs that have a long synonym chain among all the RAPs that have synonyms. With this keyword, you specify a threshold value and the lower bound for synonym chain length.

FPA evaluates all the RAPs that have synonyms, calculates the percentage of the RAPs that have a longer synonym chain than the lower bound, and compares the calculated value against the threshold. If the value exceeds the threshold, FPA issues a warning message.

SYN_PCT_LEN is an optional keyword for the THRESHOLD subcommand.

Related keywords:

[“RAP_LONGSYN_PCT keyword” on page 507](#)

Format:

```
▶▶ SYN_PCT_LEN=( threshold ,lower_bound ) ▶▶
```

threshold

Specify the threshold value in the range of 0 - 100 for the percentage of RAPs that have a long synonym chain.

lower_bound

Specify the lower bound. The value must be less than the value that is specified by SYNTBL in the HFPAPARM DD statement.

FPA regards any RAP that has the number of synonyms equal to or greater than the lower bound as a *RAP with a long synonym chain*.

FPA evaluates the threshold by using the following expression:

$$PCT(lower_bound) = \frac{\sum_{n=lower_bound+1}^{max} (\text{Number of RAPs that have } n \text{ root segments})}{\sum_{n=2}^{max} (\text{Number of RAPs that have } n \text{ root segments})} * 100 > threshold$$

Default:

None.

Site default support:

Not supported.

Alias:

SCP

Keyword abbreviation:

Supported.

TARDSNMASK keyword

Use the TARDSNMASK keyword to specify the data set name or the mask for dynamically allocating cataloged pre-existing tuning aid record data sets.

You can make a relative reference to a GDG by specifying a relative generation number in the mask, a number enclosed in brackets such as in '*gdg-name(-2)*'. The number you specify in the brackets must be zero or smaller.

You cannot use the FILECTL subcommand if you specify the TARDSNMASK keyword.

TARDSNMASK is an optional keyword for the TUNE command.

Related keywords:

- [“AREASCAN keyword” on page 442](#)
- [“TARECGEN keyword” on page 933](#)

Format:

```
►► TARDSNMASK= 'dsname' ◄◄
```

dsname

Specify a data set name or a data set mask. Enclose the *dsname* in single quotation marks. Use the following variables, as necessary, to create the mask.

&DBD

DBD name

&AREA

Area name

&AREA#

3-digit relative area number. (If you specify an area number that is greater than 999, FPA does not convert the mask variable to the relative area number and issues error message HFPU0099E.)

&AREAn#

n-digit relative area number. *n* is 1 - 5.

When the mask variable &AREA# is used for an area whose relative area number is greater than *n*-digit, for example, when &AREA2# is used for an area whose relative area number is greater than or equal to 100, FPA does not convert the mask variable to the relative area number and issues error message HFPU0099E.

For example, if you specify TARDSNMASK='HFP.TEST.TAR.&AREA' and the area name is AREA123, dynamic allocation of tuning aid record data sets will dynamically allocate 'HFP.TEST.TAR.AREA123'. If you specify TARDSNMASK='HFP.TEST.TAR.HPFP&AREA4#' and the relative area number is 1500, the resulting data set name will be 'HFP.TEST.TAR.HPFP1500'.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

Related concepts

Tuning aid records

Tuning aid records are the sorted and resolved data analysis records. A tuning aid record data set contains the analysis records for an area. The Tune function uses the tuning aid records to simulate new database attributes.

TOICTL keyword

Use the TOICTL keyword to temporarily stop the secondary index databases by using IMS Tools Online System Interface while the secondary index databases are being synchronized with the primary DEDB database.

This keyword is effective only for the Resync function (IDXPROC=RESYNC).

TOICTL is an optional keyword for the INDEXBLD command.

Related keywords:

None.

Format:



NONE

Do not stop the secondary index databases. Updates to the secondary index databases are done through DL/I ISRT and DLET calls.

DBR

Issue a /DBRECOVERY command to stop the secondary index databases. Updates to the secondary index databases are done through VSAM macro instructions.

DBD

Issue a /DBDUMP command to stop the secondary index databases. Updates to the secondary index databases are done through VSAM macro instructions.

Default:

TOICTL=NONE

Site default support:

Supported.

Keyword abbreviation:

Supported.

TOINUM keyword

Use the TOINUM keyword to specify the number of retries for issuing the command to obtain a response from the IMS Tools Online System Interface.

This keyword is effective only for offline synchronization of the Resync function (TOICTL=DBR or TOICTR=DBD).

TOINUM is an optional keyword for the GLOBAL command.

Related keywords:

[“TOITIME keyword” on page 551](#)

Format:



value

Specify the number of retries for issuing the command. Specify an integer value in the range of 1 - 100.

Default:

TOINUM=10

Site default support:

Supported.

Keyword abbreviation:

Supported.

TOITIME keyword

Use the TOITIME keyword to specify the time limit for obtaining a response from the IMS Tools Online System Interface.

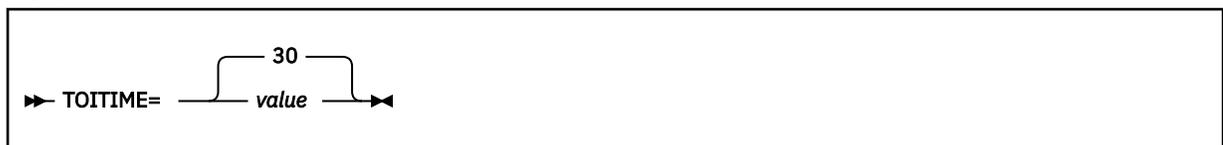
This keyword is effective only for offline synchronization of the Resync function (TOICTL=DBR or TOICTR=DBD).

TOITIME is an optional keyword for the GLOBAL command.

Related keywords:

[“TOINUM keyword” on page 551](#)

Format:



value

Specify, in seconds, the maximum allowable wait time for FPA to obtain a response from the IMS Tools Online System Interface that the database has stopped. Specify an integer value in the range of 1 -100.

If FPA does not receive a response from the IMS Tools Online System Interface within the specified time frame, FPA continues to reissue the command until the value specified on the TOINUM keyword is reached. If the maximum retry count is reached and FPA does not receive a response, an HFPX0103E message is issued.

Default:

TOITIME=30

Site default support:

Supported.

Keyword abbreviation:

Supported.

TOIXCFGRP keyword

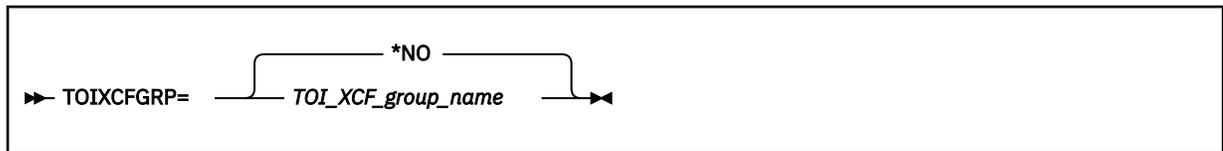
Use the TOIXCFGRP keyword to identify the XCF group that the IMS Tools Online System Interface uses to interface with FPA.

This keyword is effective only for offline synchronization of the Resync function (TOICTL=DBR or TOICTR=DBD).

TOIXCFGRP is an optional keyword for the GLOBAL command.

Related keywords:

None.

Format:***NO**

Does not use the IMS Tools Online System Interface.

TOI_XCF_group_name

Specify a 1- to 5-alphanumeric character to generate the XCF group name. The IMS Tools Online System Interface prefixes this string with characters "TOI" to generate the XCF group name.

Default:

TOIXCFGRP=*NO

Site default support:

Supported.

Keyword abbreviation:

Supported.

UNIT keyword

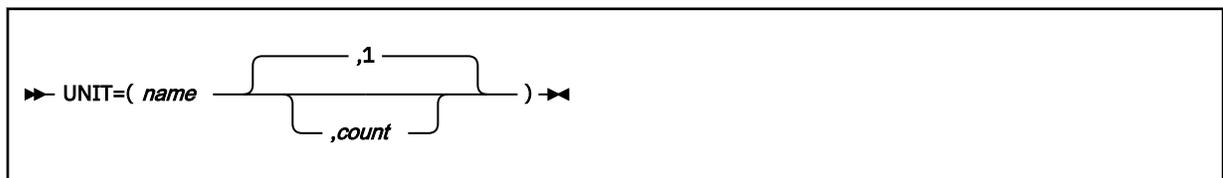
Use the UNIT keyword to specify the physical device type to be used for a data set that is accessed by dynamic allocation. Use this keyword also to specify the number of physical devices to be allocated.

UNIT is an optional keyword for the following subcommands:

- BLDMFILECTL
- FILECTL

Related keywords:

["DSNAME keyword" on page 456](#)

Format:***name***

Specify a 1- to 8-character alphanumeric name of the device type to be used for a data set that is accessed by dynamic allocation.

count

Specify the number of devices. Valid values are 1 - 59. The default value is 1.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

UOW keyword

Use the UOW keyword to simulate the areas with a new number of CIs for each unit of work (UOW).

When this keyword is specified, the Tune function replaces the number of CIs in each UOW, which is defined in the DMB in the ACBLIB data set, and simulates the area with the new condition. Use this keyword to tune the number of CIs in UOWs.

UOW is an optional keyword for the AREADEF subcommand.

Related keywords:

- [“CISIZE keyword” on page 448](#)
- [“ROOT keyword” on page 522](#)

Format:

```
▶▶ UOW=( number ,overflow ) ▶▶
```

number

Specifies the number of CIs in a UOW. The value must be in the range of 2 - 32767.

overflow

Specifies the number of CIs in a UOW for the overflow section of the area.

The value must be equal to or higher than 1 and smaller than the number of CIs in a UOW (*number*).

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

UOWDOVF_PCT keyword

Use the UOWDOVF_PCT keyword to specify a threshold setting for the percentage of UOWs that use DOVF.

If the percentage is greater than the value that you specified, a warning message is issued.

UOWDOVF_PCT is an optional keyword for the THRESHOLD subcommand.

Related keywords:

None.

Format:

```
▶▶ UOWDOVF_PCT= value ▶▶
```

value

Specify a value in the range of 0 - 100.

Default:

None.

Site default support:

Not supported.

Alias:

UDP

Keyword abbreviation:

Supported.

UOWIOVF_AVG keyword

Use the UOWIOVF_AVG keyword to specify a threshold setting for the average number of IOVF CIs that are used by a UOW.

If the number is greater than the value that you specified, a warning message is issued.

UOWIOVF_AVG is an optional keyword for the THRESHOLD subcommand.

Related keywords:

None.

Format:

```
▶▶ UOWIOVF_AVG= value ▶▶
```

value

Specify a value in the range of 0 - 32767.

Default:

None.

Site default support:

Not supported.

Alias:

UIA

Keyword abbreviation:

Supported.

UOWIOVF_MAX keyword

Use the UOWIOVF_MAX keyword to specify a threshold setting for the maximum number of IOVF CIs that are used by a UOW.

If the number is greater than the value that you specified, a warning message is issued.

UOWIOVF_MAX is an optional keyword for the THRESHOLD subcommand.

Related keywords:

None.

Format:

```
▶▶ UOWIOVF_MAX= value ▶▶
```

value

Specify a value in the range of 0 - 32767.

Default:

None.

Site default support:

Not supported.

Alias:

UIM

Keyword abbreviation:

Supported.

UOWIOVF_PCT keyword

Use the UOWIOVF_PCT keyword to specify a threshold setting for the percentage of UOWs that use IOVF.

If the percentage is greater than the value that you specified, a warning message is issued.

UOWIOVF_PCT is an optional keyword for the THRESHOLD subcommand.

Related keywords:

None.

Format:

```
▶▶ UOWIOVF_PCT= value ◀◀
```

value

Specify a value in the range of 0 - 100.

Default:

None.

Site default support:

Not supported.

Alias:

UIP

Keyword abbreviation:

Supported.

UOW_REPORT keyword

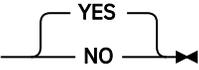
Use the UOW_REPORT keyword to specify whether to generate the UOW report.

UOW_REPORT is an optional keyword for the REPORT subcommand.

Related keywords:

- [“STARTUOW keyword” on page 545](#)
- [“STOPUOW keyword” on page 546](#)

Format:

```
▶▶ UOW_REPORT=  ◀◀
```

YES

Generate the UOW report.

NO

Do not generate the UOW report.

Default:

UOW_REPORT=YES

Site default support:

Supported.

Keyword abbreviation:

Supported.

VOLCNT keyword

Use the VOLCNT keyword to specify the maximum number of volumes that will be required by an output data set that is created by dynamic allocation that uses nonspecific volumes.

This keyword is required when the number of volumes required exceeds five (5), unless a list of specific volume serial numbers is supplied using the VOLSER keyword.

Do not specify both the VOLCNT and the VOLSER keywords; the keywords are mutually exclusive.

VOLCNT is an optional keyword for the following subcommands:

- BLDMPFILECTL
- FILECTL

Related keywords:

- [“DSNAME keyword” on page 456](#)
- [“VOLSER keyword” on page 556](#)

Format:



value

Specify a value in the range of 1 - 255.

Default:

None. If the keyword is not specified, the default is the MVS default.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

VOLSER keyword

Use the VOLSER keyword to specify one or more volume-serial identifiers for the data set that is accessed by dynamic allocation.

If this keyword is omitted for an output data set, non-specific (private) volumes are required.

Do not specify both the VOLSER and the VOLCNT keywords; the keywords are mutually exclusive.

In an SMS environment, VOLSER is an optional keyword for the following subcommands:

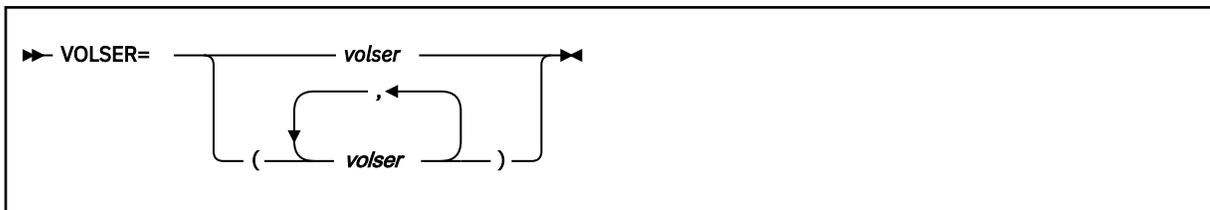
- ALLOCATE
- BLDMPFILECTL
- FILECTL

In non-SMS environment, VOLSER is a required keyword for the ALLOCATE subcommand.

Related keywords:

- [“DSNAME keyword” on page 456](#)
- [“VOLCNT keyword” on page 556](#)

Format:



volser

Specify 1- to 6-character volume serial identifier.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

WHERE keyword

Use the WHERE keyword to specify the criteria that is used to select segments for extraction.

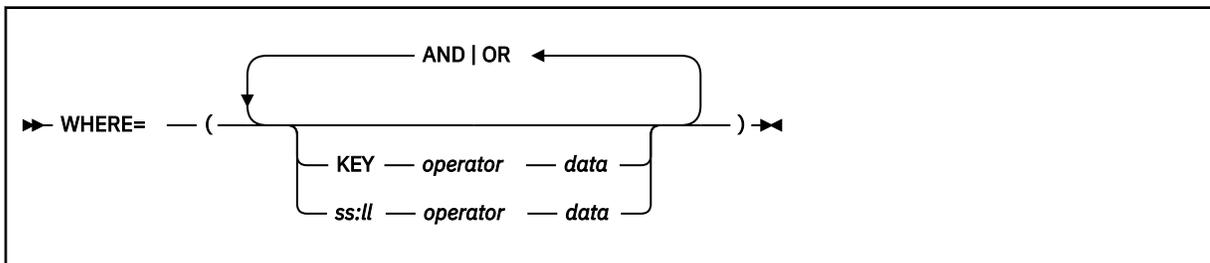
WHERE is an optional keyword for the following subcommands:

- OUTPUT
- SELECT

Related keywords:

None.

Format:



KEY

The segment key. When KEY is specified, the length of the data value (see the following descriptions) must be equal to or less than the key sequence field that is specified in the DBD.

ss

Integer that specifies the position of the field that is to be compared relative to 1. The value of ss cannot be greater than the maximum segment length.

ll

Integer that specifies the length of the field to be compared relative to 1. The value of ll cannot be greater than the maximum segment length.

operator

Relational operator. Specify one of the following operators.

Operator	Meaning
'EQ' or '=='	Equal to
'NE' or '<>'	Not equal to
'GT' or '>'	Greater than

Operator	Meaning
'LT' or '<'	Less than
'GE', '>=', or '=>'	Greater than or equal to
'LE', '<=', or '=<'	Less than or equal to

data

Value with which to match, in one of the following forms:

- C'xxxxxx' or 'xxxxxx' (character): Case-sensitive character data. You must specify a minimum of 1 and maximum of 28 characters (28 bytes).
- F'xxxxxx' (signed binary): It must contain only numeric digits. You must specify a minimum of 1 and maximum of 19 digits (4 bytes).
- P'xxxxxx' (packed decimal): It is assumed that it is a positive number. It must contain only numeric digits. You must specify a minimum of 1 and maximum of 31 digits (16 bytes).
- X'xxxxxx' (unsigned binary): It must be an even number of hexadecimal characters. You must specify a minimum of two hexadecimal characters (representing 1 byte) and a maximum of 56 hexadecimal characters (representing 28 bytes).

You can also declare the length, by specifying it before the data-type declaration.

Example:

```
8C 'CUSDB'  
2P '01'
```

If *data* is character data-type and its length is shorter than the length of KEY or *ll*, *data* is padded on the right with blanks to match the length. If *data* is a signed binary, a packed decimal, or an unsigned binary data-type, and its length is shorter than the length of KEY or *ll*, the precision of *data* is increased to match the length.

AND

Boolean AND operation.

OR

Boolean OR operation.

Each parameter must be separated by one or more blanks.

Default:

None.

Site default support:

Not supported.

Keyword abbreviation:

Supported.

Usage note:

If you specify the WHERE keyword for a compressed segment, you must expand the segment with the COMPRESS=NO keyword.

Chapter 17. Sample reports of FPA

These topics explain the reports that are generated by FPA.

Topics:

- [“Reports generated by each process of FPA” on page 560](#)
- [“Audit report” on page 564](#)
- [“Processing report” on page 565](#)
- [“DBD Definition report” on page 569](#)
- [“Unloaded report” on page 575](#)
- [“Reloaded report” on page 578](#)
- [“Output File report” on page 580](#)
- [“Input File report” on page 583](#)
- [“Resolve Pointers report” on page 585](#)
- [“Pointer Checksum report” on page 586](#)
- [“Pointer Chain Reconstruction report” on page 587](#)
- [“CI Map/CI Dump report” on page 588](#)
- [“Process Summary of Analysis report” on page 590](#)
- [“DEDB Area Analysis reports” on page 591](#)
- [“Largest Database Records report ” on page 603](#)
- [“Consolidated Largest Database Records report” on page 604](#)
- [“UOW report” on page 606](#)
- [“UOW Exception report” on page 607](#)
- [“Image Copy Output report” on page 608](#)
- [“Extracted report” on page 609](#)
- [“DMACPRT report” on page 612](#)
- [“Reorganization Processing Summary report” on page 615](#)
- [“UOW Space Statistics report” on page 616](#)
- [“Secondary Index Definition report” on page 617](#)
- [“Secondary Index Processing report” on page 620](#)
- [“Secondary Index Resync report” on page 623](#)
- [“Secondary Index Analysis report” on page 626](#)
- [“Secondary Index Duplicate Key report” on page 627](#)
- [“Resync Pointer Segment Dump report” on page 629](#)
- [“Secondary Index Freespace Reclaim report” on page 630](#)
- [“Result of Access Method Services report” on page 631](#)
- [“Initialization Processing Summary report” on page 632](#)
- [“Pointer Segment Dump report” on page 633](#)
- [“Site and System Default report” on page 634](#)
- [“Process Summary report ” on page 636](#)
- [“Sensor Data Statistics report” on page 637](#)
- [“Segment Length Distribution report” on page 645](#)
- [“DB Record Length Distribution report” on page 647](#)

- [“Area Tuning Summary report” on page 649](#)
- [“Tuning Aid Record Data Set report” on page 652](#)
- [“TUNED reports” on page 653](#)
- [“RBA of Used IOVF CI report” on page 656](#)

Reports generated by each process of FPA

With each process of FPA, several reports are generated.

The following table summarizes the reports that are generated with each process of FPA. The table also shows whether each report is stored in the IMS Tools Knowledge Base Output repository when the ITKBLOAD and the ITKBSRVR keywords are specified.

Table 74. Reports generated by each process of FPA

FPA process	Generated reports	Whether the report can be stored in the IMS Tools KB Output repository
Unload	Audit report	Yes
	Processing report	Yes
	DBD Definition report	Yes
	CI Map/CI Dump report	No
	Unloaded report	Yes
	Reloaded report	Yes
	Output File report	Yes
	Site and System Default report (if the site default module or the site default table is used)	Yes
	Process Summary report	Yes
	Image Copy Output report (ICOUT=YES)	Yes
Reload	Audit report	Yes
	Processing report	Yes
	DBD Definition report	Yes
	Reloaded report	Yes
	Input File report	Yes
	Reports related to the Analyze process (PTRCHKLVL=QUICK or FULL)	Yes (See Note)
	Site and System Default report (if the site default module or the site default table is used)	Yes
	Image Copy Output report (ICOUT=YES)	Yes
	Sensor Data Statistics report (SENSOR=YES)	Yes
	RBA of Used IOVF CI report (IOVFCI_RBA_REPORT=YES)	Yes

Table 74. Reports generated by each process of FPA (continued)

FPA process	Generated reports	Whether the report can be stored in the IMS Tools KB Output repository
Change	Audit report	Yes
	Processing report	Yes
	DBD Definition report	Yes
	Unloaded report	Yes
	Reloaded report	Yes
	Reports related to the Analyze process (PTRCHKLVL=QUICK or FULL)	Yes (See Note)
	Site and System Default report (if the site default module or the site default table is used)	Yes
	Image Copy Output report (ICOUT=YES)	Yes
	Sensor Data Statistics report (SENSOR=YES)	Yes
	RBA of Used IOVF CI report (IOVFCI_RBA_REPORT=YES)	Yes
Reorg	Audit report	Yes
	Processing report	Yes
	DBD Definition report	Yes
	Reorganization Processing Summary report	Yes
	UOW Space Statistics report (RFSSTAT=YES)	Yes
	Reports related to the Analyze process (PTRCHKLVL=QUICK or FULL)	Yes (See Note)
	Site and System Default report (if the site default module or the site default table is used)	Yes
	Image Copy Output report (ICOUT=YES)	Yes
	Sensor Data Statistics report (SENSOR=YES)	Yes
	RBA of Used IOVF CI report (IOVFCI_RBA_REPORT=YES)	Yes

Table 74. Reports generated by each process of FPA (continued)

FPA process	Generated reports	Whether the report can be stored in the IMS Tools KB Output repository
Analyze	Audit report	Yes
	Processing report	Yes
	DBD Definition report	Yes
	Resolve Pointers report	No
	Pointer Checksum report	No
	Pointer Chain Reconstruction report	No
	CI Map/CI Dump report	No
	Process Summary of Analysis report	Yes
	DEDB Area Analysis reports	Yes
	UOW report	Yes
	UOW Exception report	Yes
	Site and System Default report (if the site default module or the site default table is used)	Yes
	Image Copy Output report (ICOUT=YES)	Yes
	Largest Database Records report (LARGEREC_REPORT=YES)	Yes
	Consolidated Largest Database Records report (LARGEREC_REPORT=YES and LARGEREC_COUNT=(value1,value2))	Yes
	Secondary Index Definition report	Yes
	Secondary Index Analysis report	Yes
	Pointer Segment Dump report	No
	Sensor Data Statistics report (SENSOR=YES)	Yes
	Segment Length Distribution report (SEGLDIST=YES)	Yes
DB Record Length Distribution report (DBRLDIST=YES)	Yes	
RBA of Used IOVF CI report (IOVFCI_RBA_REPORT=YES)	Yes	
Extract	Audit report	No
	Processing report	No
	DBD Definition report	No
	Extracted report	No
	Output File report	No
	Site and System Default report (if the site default module or the site default table is used)	No

Table 74. Reports generated by each process of FPA (continued)

FPA process	Generated reports	Whether the report can be stored in the IMS Tools KB Output repository
Tune	Audit report	No
	Processing report	No
	DBD Definition report	No
	Site and System Default report (if the site default module or the site default table is used)	No
	Reports related to the Analyze process	No
	Area Tuning Summary report	No
	Tuning Aid Record Data Set report	No
	TUNED DEDB Area Analysis reports	No
	TUNED UOW report	No
	TUNED UOW Exception report	No
	TUNED Largest Database Records report (LARGEREC_REPORT=YES)	No
	TUNED Consolidated Largest Database Records report (LARGEREC_REPORT=YES and LARGEREC_COUNT=(value1,value2))	No
	TUNED Segment Length Distribution report (SEGLDIST=YES)	No
	TUNED DB Record Length Distribution report (DBRLDIST=YES)	No
DMAC Print	Audit report	Yes
	Processing report	Yes
	DBD Definition report	Yes
	DMACPRT report	Yes
Build Index	Audit report	Yes
	Processing report	Yes
	DBD Definition report	Yes
	Secondary Index Definition report	Yes
	Secondary Index Processing report	Yes
	Secondary Index Resync report	Yes
	Secondary Index Duplicate Key report	No
	Resync Pointer Segment Dump report	No
	Secondary Index Freespace Reclaim report	Yes
	Result of Access Method Services report	Yes

Table 74. Reports generated by each process of FPA (continued)

FPA process	Generated reports	Whether the report can be stored in the IMS Tools KB Output repository
Initialize	Audit report	Yes
	Processing report	Yes
	DBD Definition report	Yes
	Initialization Processing Summary report	Yes

Note: Resolve Pointers reports, Pointer Checksum reports, Pointer Chain Reconstruction reports, and CI Map/CI Dump reports cannot be stored in the IMS Tools KB Output repository.

Report retention settings are applied to all reports to control the growth of the report repository. The retention values for product reports are provided by the product Definition Table when the product is registered with IMS Tools KB. Because the Audit report, the Processing report, the DBD Definition report, and the Site and System Default report are generated by all processes of FPA, you can change their retention period to avoid being deleted earlier than other reports. For details, see the topic "Changing the retention values for individual reports" in the *Tools Base IMS Tools Knowledge Base User's Guide*.

Related reference

ITKBLOAD keyword

Use the ITKBLOAD keyword to specify the name of the load module library for IMS Tools KB.

ITKBSRVR keyword

Use the ITKBSRVR keyword to specify the name of the IMS Tools KB server XCF group.

Audit report

The Audit report provides information about the HFPSYSIN statements.

This report is generated by the following commands:

- ANALYZE
- CHANGE
- DMACPRT
- EXTRACT
- INDEXBLD
- INITIALIZE
- UNLOAD
- RELOAD
- REORG
- TUNE

The following figure is a sample of the Audit report.

```

HFPSYSIN
0.....1.....2.....3.....4.....5.....6.....7.....8
1234567890123456789012345678901234567890123456789012345678901234567890

- 1: GLOBAL
- 2:   DBRC=NO,
- 3:   SCAN=NO
- 4: UNLOAD DBD=DEDBJN21,
- 5:   IAREA=ALL,
- 6:   OAREA=ALL,
- 7:   SDEP=LOGICAL,
- 8:   SORT=YES,
- 9:   ITASKCTL=1

PROCESSING INFORMATION

EXEC
- DBRCGRP           :
- IMSPLEX           :

GLOBAL
- DBRC              : NO
- SCAN              : NO
    
```

Figure 193. Audit report

HFPSYSIN

Shows an echo of the control statements that you specified in the HFPSYSIN data set.

PROCESSING INFORMATION

Shows the parameters of the EXEC statement and the GLOBAL command in HFPSYSIN.

Processing report

The Processing report contains the values that the command process used.

The values shown in the report are determined by the value of HFPSYSIN, the site default options, and the system default.

This report is generated by the following commands:

- ANALYZE
- CHANGE
- DMACPRT
- EXTRACT
- INDEXBLD
- INITIALIZE
- UNLOAD
- RELOAD
- REORG
- TUNE

The following figure shows an example of the Processing report for the Unload function.

```

IMS HPFP UTILITIES - FPA UNLOAD                                "Processing report"
5698-FPP V2R1                                                PAGE: 1
                                                                2019-11-22 15:55:58

UNLOAD STARTED DATE: 2019-11-22 TIME: 15:55:58 ENDED DATE: 2019-11-22 TIME: 15:55:59

UNLOAD PROCESSING OPTION
- AREA_INFORMATION_RECORD      : YES
- COMPRESS                     : NO
- DBD                          : DEDBJN21
- ...
- ...

UNLOAD SUBCOMMAND PROCESSING OPTION
FILECTL
- AVGREC                       :
- DATACLAS                   :
- DDNAME                       : (ODB21AR0)
- ...
- ...
FILECTL
- AVGREC                       :
- DATACLAS                   :
- DDNAME                       : (OTHRAREA)
- ...
- ...

PROCESSING INFORMATION
- RANDOMIZER IS NOT CALLED FOR UNLOAD PROCESS.
- DBRC=NO IS SPECIFIED. - EEQE DETECTION IS NOT PERFORMED.

```

Figure 194. Processing report for the Unload function

The following figure shows an example of the Processing report for the Reload function.

```

IMS HPFP UTILITIES - FPA RELOAD                                "Processing report"
5698-FPP V2R1                                                PAGE: 1
                                                                2019-11-22 16:15:46

RELOAD STARTED DATE: 2019-11-22 TIME: 16:15:46 ENDED DATE: 2019-11-22 TIME: 16:15:47

RELOAD PROCESSING OPTION
- ATASKCTL                     : 0
- CISNAP                      : NO
- DBD                         : DEDBJN21
- ...
- ...

RELOAD SUBCOMMAND PROCESSING OPTION
ALLOCATE
- AMSOPT                      : 'SHAREOPTIONS(3 3) '
- ATTRIBUTE                   : (REUSE,EROPT=ABORT)
- AVGREC                      : B
- ...
- ...

```

Figure 195. Processing report for the Reload function

The following figure shows an example of the Processing report for the Change function.

```

IMS HPFP UTILITIES - FPA CHANGE                                "Processing report"                                PAGE: 1
5698-FPP V2R1                                                2019-11-22 16:27:55

CHANGE STARTED DATE: 2019-11-22 TIME: 16:27:55 ENDED DATE: 2019-11-22 TIME: 16:27:56

CHANGE PROCESSING OPTION
- ATASKCTL          : 0
- CISNAP            : NO
- DBD               : DEDBJN21
- ...
- ...

CHANGE SUBCOMMAND PROCESSING OPTION
ALLOCATE
- AMSOPT           : 'SHAREOPTIONS(3 3)'
- ATTRIBUTE        :
- AVGREC           : B
- ...
- ...
ALLOCATE
- AMSOPT           : 'SHAREOPTIONS(3 3)'
- ATTRIBUTE        :
- AVGREC           : B
- ...
- ...
ALLOCATE
- AMSOPT           : 'SHAREOPTIONS(3 3)'
- ATTRIBUTE        :
- AVGREC           : B
- ...
- ...

CHANGE SUBCOMMAND PROCESSING OPTION
IRENAME
- DSNAME           : 'HPFP.M&AREA.RENAME1'
- IAREA            : (DB21AR0,DB21AR3)
IRENAME
- DSNAME           : 'HPFP.M&AREA.RENAME2'
- IAREA            : (DB21AR1,DB21AR4)
IRENAME
- DSNAME           : 'HPFP.M&AREA.RENAME3'
- IAREA            : (DB21AR2)

PROCESSING INFORMATION
- RANDOMIZER IS NOT CALLED FOR UNLOAD PROCESS.
- DBRC=NO IS SPECIFIED. - EEQE DETECTION IS NOT PERFORMED.
- IOVF WAS NOT PRELOADED INTO A DATA SPACE FOR PROCESSING. AREA NO: 1, AREANAME: DB21AR1
- SDEP WAS NOT PRELOADED INTO A DATA SPACE FOR PROCESSING. AREA NO: 1, AREANAME: DB21AR1
- IOVF WAS NOT PRELOADED INTO A DATA SPACE FOR PROCESSING. AREA NO: 2, AREANAME: DB21AR2
- SDEP WAS NOT PRELOADED INTO A DATA SPACE FOR PROCESSING. AREA NO: 2, AREANAME: DB21AR2
- IOVF WAS NOT PRELOADED INTO A DATA SPACE FOR PROCESSING. AREA NO: 3, AREANAME: DB21AR3
- SDEP WAS NOT PRELOADED INTO A DATA SPACE FOR PROCESSING. AREA NO: 3, AREANAME: DB21AR3
- IOVF WAS NOT PRELOADED INTO A DATA SPACE FOR PROCESSING. AREA NO: 4, AREANAME: DB21AR4
- SDEP WAS NOT PRELOADED INTO A DATA SPACE FOR PROCESSING. AREA NO: 4, AREANAME: DB21AR4

```

Figure 196. Processing report for the Change function

The following figure shows an example of the Processing report for the Reorg function.

```

IMS HPFP UTILITIES - FPA REORG                                "Processing report"                                PAGE: 1
5698-FPP V2R1                                                2019-11-22 15:57:40

REORG STARTED DATE: 2019-11-22 TIME: 15:57:40 ENDED DATE: 2019-11-22 TIME: 15:57:42

REORG PROCESSING OPTION
- ATASKCTL          : 1
- CISNAP            : NO
- DBD               : DEDBJN23
- ...
- ...

```

Figure 197. Processing report for the Reorg function

The following figure shows an example of the Processing report for the Analyze function.

```

IMS HPFP UTILITIES - FPA ANALYZE                             "Processing report"                                PAGE: 1
5698-FPP V2R1                                                2019-11-22 18:30:14

ANALYZE STARTED DATE: 2019-11-22 TIME: 18:30:14 ENDED DATE: 2019-11-22 TIME: 18:30:16

ANALYZE PROCESSING OPTION
- ATASKCTL          : 1
- CISNAP            : NO
- DBD               : DEDBJN23
- ...
- ...

PROCESSING INFORMATION
- DBRC=NO IS SPECIFIED. - EEQE DETECTION IS NOT PERFORMED.
- IOVF WAS NOT PRELOADED INTO A DATA SPACE FOR PROCESSING. AREA NO: 1, AREANAME: DB21AR1
- SDEP WAS NOT PRELOADED INTO A DATA SPACE FOR PROCESSING. AREA NO: 1, AREANAME: DB21AR1

```

Figure 198. Processing report for the Analyze function

The following figure shows an example of the Processing report for the Extract function.

```

IMS HPFP UTILITIES - FPA EXTRACT                                "Processing report"                                PAGE: 1
5698-FPP V2R1                                                2019-11-22 17:00:06

EXTRACT STARTED DATE: 2019-11-22  TIME: 17:00:06    ENDED DATE: 2019-11-22  TIME: 17:00:07

EXTRACT PROCESSING OPTION
- COMPRESS                : NO
- DBD                     : DEDBJN21
- EFORMAT                 : STD
- ...
- ...

EXTRACT SUBCOMMAND PROCESSING OPTION
OUTPUT
- FIELDS                  : (SEGMENT_NAME,C'*KEY:',KEY)
- SEGMENT                 : ROOTSEG1
- ...

EXTRACT SUBCOMMAND PROCESSING OPTION
SELECT
- EVERY                   :
- SEGMENT                 : ROOTSEG1
- SKIP                    :
- ...
- ...

PROCESSING INFORMATION
- DBRC=NO IS SPECIFIED. - EEQE DETECTION IS NOT PERFORMED.

```

Figure 199. Processing report for the Extract function

The following figure shows an example of the Processing report for the Tune function.

```

IMS HPFP UTILITIES - FPA TUNE                                  "Processing report"                                PAGE: 1
5698-FPP V2R1                                                2019-11-22 16:40:53

TUNE STARTED DATE: 2019-11-22  TIME: 16:40:54    ENDED DATE: 2019-11-22  TIME: 16:40:54

TUNE PROCESSING OPTION
- AREASCAN                : YES
- CISNAP                  : NO
- DBD                     : DEDBJN30
- ...
- ...

TUNE SUBCOMMAND PROCESSING OPTION
SIMULATE
- ACBLIB                  : IMSVS.TEST.ACBLIB
- OAREA                   : (DB30AR0)
- REPORTDD                : HFPRPTS
- ...
- ...

TUNE SUBCOMMAND PROCESSING OPTION
AREADEF
- CISIZE                  :
- OAREA                   : (DB30AR0)
- ROOT                    :
- ...

```

Figure 200. Processing report for the Tune function

The following figure shows an example of the Processing report for the DMAC Print function.

```

IMS HPFP UTILITIES - FPA DMACPRT                              "Processing report"                                PAGE: 1
5698-FPP V2R1                                                2019-11-22 16:47:18

DMACPRT STARTED DATE: 2019-11-22  TIME: 16:47:18    ENDED DATE: 2019-11-22  TIME: 16:47:19

DMACPRT PROCESSING OPTION
- DBD                     : DEDBJN21
- IAREA                   : ALL
- IDSNMASK                :
- ...

```

Figure 201. Processing report for the DMAC Print function

The following figure shows an example of the Processing report for the Build Index function.

```

IMS HPFP UTILITIES - FPA INDEXBLD                             "Processing report"                                PAGE: 1
5698-FPP V2R1                                                2019-06-02 11:37:30

INDEXBLD STARTED DATE: 2019-06-02  TIME: 11:37:30    ENDED DATE: 2019-06-02  TIME: 11:37:30

INDEXBLD PROCESSING OPTION
- BUILDDSNMASK            : 'HPFP.BUILD.&INDEX'
- BUILDMODE               : SCAN
- DBD                     : DEDBJN21
- ...
- ...

PROCESSING INFORMATION
- DBRC=NO IS SPECIFIED. - EEQE DETECTION IS NOT PERFORMED.
- IOVF WAS NOT PRELOADED INTO A DATA SPACE FOR PROCESSING. AREA NO: 1, AREANAME: DB21AR1

```

Figure 202. Processing report for the Build Index function

The following figure shows an example of the Processing report for the Resync function.

```

IMS HPFP UTILITIES - FPA INDEXBLD                                "Processing report"                                PAGE: 1
5698-FPP V2R1                                                    2019-08-19 3:06:14
INDEXBLD GLOBAL SCAN=YES OPTION PROCESS
INDEXBLD PROCESSING OPTION
- DBD : DEDBJN21
- DUPKEYACT : NONE
- DUPKEYDSNMASK :
- ...
- ...
PROCESSING INFORMATION
- DBRC=NO IS SPECIFIED. - EEQE DETECTION IS NOT PERFORMED.

```

Figure 203. Processing report for the Resync function

The following figure shows an example of the Processing report for the Freespace Reclaim function.

```

IMS HPFP UTILITIES - FPA INDEXBLD                                "Processing report"                                PAGE: 1
5698-FPP V2R1                                                    2019-07-07 0:20:14
INDEXBLD STARTED DATE: 2019-07-07 TIME: 0:20:14 ENDED DATE: 2019-07-07 TIME: 0:20:16
INDEXBLD PROCESSING OPTION
- DBD : DEDBJN21
- FS_RECLAIM_REPORT : YES
- ...
- ...
PROCESSING INFORMATION
- DBRC=NO IS SPECIFIED. - EEQE DETECTION IS NOT PERFORMED.

```

Figure 204. Processing report for the Freespace Reclaim function

PROCESSING OPTION

Shows the values that the command process used.

IMSACB, NEWACB, and OLDACB each shows the name of the data set that is assigned to the DD statement. If you specify IMSCATHLQ=*bsdshlq*, either IMS CATALOG (CURRENT) or IMS CATALOG (PENDING) is printed instead of the data set name.

SUBCOMMAND PROCESSING OPTION

Shows the values that the subcommand process used.

PROCESSING INFORMATION

Shows the information messages that were issued during the command processing.

Message '- IOVF/SDEP WAS NOT PRELOADED INTO A DATA SPACE FOR PROCESSING. AREA NO: *nnnn*, AREANAME: *areaname*' is printed when the FPA Analyze or the FPA Tune function is used because these functions do not require data space. The FPA Reload, Change, and Reorg functions with the PTRCHKLVL=FULL option print the same message because this option internally calls the Analyze process.

Restrictions for concatenated data sets:

When data sets IMSACB, NEWACB, OLDACB, or all three are specified as concatenated, only the name of the first data set of all that are concatenated is printed in the report.

DBD Definition report

The DBD Definition report contains information about ACBLIB that the command process used.

This report is generated by the following commands:

- ANALYZE
- CHANGE
- DMACPRT
- EXTRACT
- INDEXBLD
- INITIALIZE
- UNLOAD
- RELOAD
- REORG
- TUNE

This report is not printed if you make one of the following specifications:

- GLOBAL SCAN=SITE_CREATE
- GLOBAL SCAN=SITE_REPORT

The following figure is a sample of the DBD Definition report for Unload when only IMSACB is specified.

```

IMS HPFP UTILITIES - FPA UNLOAD                "DBD definition report"                PAGE: 1
5698-FPP V2R1                                2019-11-22 17:00:06

- ACBLIB      : IMSACB
- ACB_DSNAME  : IMSVS.ACBLIB
- DBD_NAME    : DEDBJN23
- RANDOMIZER  : RMOD4
- IMS LEVEL   : VERSION 14 RELEASE 1

DB LARGEST INFO: CI-SIZE  UOW-1  NO(RAP'S/UOW)  UOW-2  SEG-LEN
                  -----  -----  -----  -----  -----
                  28,672   15         14         8       900

AREA:

AREA NO.  AREA NAME  CI-SIZE  UOW=  ROOT=  TOTAL CI'S  SDEP START
          (15,1)  (3,1)  (15,1)  (5,1)  (5,1)  (14,6)  (3,1)  (2,1)  (2,1)
          -----  -----  -----  -----  -----  -----  -----  -----  -----
          BASE  DOVF  IOVF  TOTAL  BLOCK#  RBA(HEX)
1 DB23AR0  1,024  (15,1)  (6,2)  56     4     29    89    -     -
2 DB23AR1  2,048  (3,1)  (2,1)  2      1     2     5    -     -
3 DB23AR2  2,048  (15,1)  (5,1)  56     4     14    74    -     -
4 DB23AR3  2,048  (15,1)  (5,1)  56     4     14    74    -     -
5 DB23AR4  1,024  (12,8)  (14,6)  32    64     71   167   -     -
6 DB23AR5  8,192  (10,4)  (3,1)  12     8     9     29   -     -
7 DB23AR6  28,672 (2,1)  (2,1)  1      1     1     3    -     -
8 DB23AR7  2,048  (3,1)  (2,1)  2      1     2     5    -     -

SEGMENT:

+-----+
| LEGEND FOR SEGMENT INFORMATION |
+-----+
| R: ROOT SEGMENT  D: DDEP SEGMENT  S: SDEP SEGMENT |
| F: FIXED LENGTH  V: VARIABLE LENGTH  I: COMP INIT  Y: PCL DEFINED |
+-----+

SEG. CODE  SEG. NAME  HIER LVL  PARENT S. CODE  TYPE  FIX|VAR  PARENT PCL SSP  LENGTH  KEY  COMP-RTN  SEC.
          (1)  (2)  (3)  (4)  (5)  (6)  (7)  (8)  (9)  (10)  (11)  (12)  (13)
          -----  -----  -----  -----  -----  -----  -----  -----  -----  -----  -----  -----  -----
1  ROOTSEG1  1  -  R  V  -  -  900  20  2  8  -  -  6
2  DD1  2  1  D  V  -  -  900  20  2  10  -  -  13
3  DD2  2  1  D  V  -  -  900  20  2  10  -  -  -
4  DD3  2  1  D  V  -  -  900  20  2  10  -  -  -
5  DD4  2  1  D  V  -  -  900  20  2  10  -  -  -
6  DD43  3  5  D  V  -  -  900  20  2  10  -  -  -
7  DD44  4  6  D  V  -  -  900  20  2  10  -  -  -
8  DD45  5  7  D  V  -  -  900  20  2  10  -  -  -
9  DD46  6  8  D  V  -  -  900  20  2  10  -  -  -
10 DD47  7  9  D  V  -  -  900  20  2  10  -  -  -
11 DD48  8  10 D  V  -  -  900  20  2  10  -  -  -
12 DD49  9  11 D  V  -  -  900  20  2  10  -  -  -
13 DD410 10 12 D  V  -  -  900  20  2  10  -  -  8
14 DD411 11 13 D  V  -  -  900  20  2  10  -  -  -
15 DD412 12 14 D  V  -  -  900  20  2  10  -  -  -

```

Figure 205. DBD Definition report for Unload with IMSACB DD only

The following figure is a sample of the DBD Definition report for Unload when IMSACB is specified together with the IMSACB DD and the NEWACB DD statements.

- ACBLIB : IMSACB
- ACB DSNAME : IMSVS.ACBLIB
- DBD NAME : DEDBJN22
- RANDOMIZER : RMOD5
- IMS LEVEL : VERSION 14 RELEASE 1

DEDB DEFINITION DIFFERENCE DETECTED. *: DIFFERENCE &: SEVERE DIFFERENCE

DB LARGEST INFO: CI-SIZE UOW-1 NO(RAP'S/UOW) UOW-2 SEG-LEN
2,048 13 10 3 900

AREA:

AREA NO.	AREA NAME	CI-SIZE	UOW=	ROOT=	TOTAL CI'S				SDEP START	
					BASE	DOVF	IOVF	TOTAL	BLOCK#	RBA(HEX)
1	DB22AR0	1,024	(5,1)	(5,1)	16	4	4	24	32	00008000
2	DB22AR1	2,048	(5,1)	(5,1)	16	4	4	24	32	00010000
3	DB22AR2	1,024	(13,3)	(15,3)	120	36	38	194	210	00034800
4	DB22AR3	1,024	(10,2)	(10,2)	64	16	19	99	112	0001C000
5	DB22AR4	2,048	(10,2)	(10,2)	64	16	19	99	112	00038000

SEGMENT:

```

+-----+
| LEGEND FOR SEGMENT INFORMATION |
+-----+
| R: ROOT SEGMENT  D: DDEP SEGMENT  S: SDEP SEGMENT |
| F: FIXED LENGTH  V: VARIABLE LENGTH I: COMP INIT   Y: PCL DEFINED |
+-----+

```

SEG. CODE	SEG. NAME	HIER LVL	PARENT S.CODE	TYPE	FIX VAR	PARENT		LENGTH		KEY		COMP-RTN		SEC. IDX
						PCL	SSP	MAX	MIN	OFF	LEN	NAME	INIT	
==> 1	ROOTSEG1	1	-	R	V	-	-	900	20	2	6	*	-	6
2	SDSEGMM1	2	1	S	V	-	-	900	20	-	-	-	-	-
3	DD1	2	1	D	V	-	-	900	20	2	7	-	-	13
4	DD2	2	1	D	V	-	-	900	20	2	7	-	-	1
5	DD3	2	1	D	V	-	-	900	20	-	-	-	-	-
6	DD4	2	1	D	V	-	-	900	20	2	7	-	-	-
7	DD43	3	6	D	V	-	-	900	20	2	7	-	-	-
8	DD44	4	7	D	V	-	-	900	20	2	7	-	-	-
9	DD45	5	8	D	V	-	-	900	20	2	7	-	-	-
10	DD46	6	9	D	V	-	-	900	20	2	7	-	-	-
11	DD47	7	10	D	V	-	-	900	20	2	7	-	-	-
12	DD48	8	11	D	V	-	-	900	20	2	7	-	-	-
13	DD49	9	12	D	V	-	-	900	20	2	7	-	-	-
14	DD410	10	13	D	V	-	-	900	20	2	7	-	-	8
15	DD411	11	14	D	V	-	-	900	20	2	7	-	-	-
16	DD412	12	15	D	V	-	-	900	20	2	7	-	-	-

Figure 206. DBD Definition report for Unload; IMSACB with DD statements IMSACB and NEWACB

The following figure is a sample of the DBD Definition report for Unload when NEWACB is specified together with the IMSACB DD and the NEWACB DD statements.

- ACBLIB : NEWACB
- ACB DSNAME : IMSVS.NEWACB
- DBD NAME : DEDBJN22
- RANDOMIZER : RMOD5
- IMS LEVEL : VERSION 14 RELEASE 1

DEDB DEFINITION DIFFERENCE DETECTED. *: DIFFERENCE &: SEVERE DIFFERENCE

DB LARGEST INFO: CI-SIZE UOW-1 NO(RAP'S/UOW) UOW-2 SEG-LEN
2,048 13 10 3 900

AREA:

AREA NO.	AREA NAME	CI-SIZE	UOW=	ROOT=	TOTAL CI'S				SDEP START	
					BASE	DOVF	IOVF	TOTAL	BLOCK#	RBA(HEX)
1	DB22AR0	1,024	(5,1)	(5,1)	16	4	4	24	32	00008000
2	DB22AR1	2,048	(5,1)	(5,1)	16	4	4	24	32	00010000
3	DB22AR2	1,024	(13,3)	(15,3)	120	36	38	194	210	00034800
4	DB22AR3	1,024	(10,2)	(10,2)	64	16	19	99	112	0001C000
5	DB22AR4	2,048	(10,2)	(10,2)	64	16	19	99	112	00038000

SEGMENT:

```

+-----+
| LEGEND FOR SEGMENT INFORMATION |
+-----+
| R: ROOT SEGMENT D: DDEP SEGMENT S: SDEP SEGMENT |
| F: FIXED LENGTH V: VARIABLE LENGTH I: COMP INIT |
| Y: PCL DEFINED |
+-----+

```

SEG. CODE	SEG. NAME	HIER LVL	PARENT S.CODE	TYPE	FIX VAR	PARENT		LENGTH		KEY		COMP-RTN		SEC. IDX
						PCL	SSP	MAX	MIN	OFF	LEN	NAME	INIT	
==> 1	ROOTSEG1	1	-	R	V	-	-	900	20	2	6	DFSCMPX0*	-	6
2	SDSEG1	2	1	S	V	-	-	900	20	-	-	-	-	-
3	DD1	2	1	D	V	-	-	900	20	2	7	-	-	13
4	DD2	2	1	D	V	-	-	900	20	2	7	-	-	1
5	DD3	2	1	D	V	-	-	900	20	-	-	-	-	-
6	DD4	2	1	D	V	-	-	900	20	2	7	-	-	-
7	DD43	3	6	D	V	-	-	900	20	2	7	-	-	-
8	DD44	4	7	D	V	-	-	900	20	2	7	-	-	-
9	DD45	5	8	D	V	-	-	900	20	2	7	-	-	-
10	DD46	6	9	D	V	-	-	900	20	2	7	-	-	-
11	DD47	7	10	D	V	-	-	900	20	2	7	-	-	-
12	DD48	8	11	D	V	-	-	900	20	2	7	-	-	-
13	DD49	9	12	D	V	-	-	900	20	2	7	-	-	-
14	DD410	10	13	D	V	-	-	900	20	2	7	-	-	8
15	DD411	11	14	D	V	-	-	900	20	2	7	-	-	-
16	DD412	12	15	D	V	-	-	900	20	2	7	-	-	-

Figure 207. DBD Definition report for Unload; NEWACB with DD statements IMSACB and NEWACB

The following figures show samples of the DBD Definition reports that are generated by the Reload function. These reports are generated when DBD information inconsistencies are detected between the library that is specified on the IMSACB DD statement and the USR files that were created during the Unload process.

When inconsistencies are detected, two DBD Definition reports are generated:

- DBD Definition report that contains DBD information in IMSACB
- DBD Definition report that contains DBD information in the USR file in which the first inconsistency was detected

The ampersand (&) character is shown after an item whenever a difference is found.

The following figure shows sample DBD Definition report that contains the DBD information in IMSACB.

```

IMS HPFP UTILITIES - FPA RELOAD                                "DBD definition report"                                PAGE: 1
5698-FPP V2R1                                                2019-11-22 18:08:02
- ACBLIB : IMSACB
- ACB DSNAME : FPTL02.DBTJIE1.IMS14.DB12.ACBLIB4
- DBD NAME : DEDBJN23
- RANDOMIZER : RMODTS2
- IMS LEVEL : VERSION 14 RELEASE 1
DEDB DEFINITION SEVERE DIFFERENCE DETECTED. *: DIFFERENCE &: SEVERE DIFFERENCE
DB LARGEST INFO: CI-SIZE UOW-1 NO(RAP'S/UOW) UOW-2 SEG-LEN
-----
                28,672      5              3          2          900
AREA:

```

AREA NO.	AREA NAME	CI-SIZE	UOW=	ROOT=	TOTAL CI'S				SDEP START	
					BASE	DOVF	IOVF	TOTAL	BLOCK#	RBA (HEX)
1	DB23AR1	2,048	(5,2)	(14,4)	30	20	19	69	77	00026800
2	DB23AR2	1,024	(2,1)	(16,4)	12	12	7	31	36	00009000
==> 3	DB23AR8&	4,096	(4,1)	(8,4)	12	4	15	31	38	00026000
4	DB23AR4	8,192	(2,1)	(2,1)	1	1	1	3	8	00010000
5	DB23AR5	28,672	(2,1)	(2,1)	1	1	1	3	8	00038000

```

SEGMENT:
+-----+
| LEGEND FOR SEGMENT INFORMATION |
+-----+
| R: ROOT SEGMENT  D: DDEP SEGMENT  S: SDEP SEGMENT |
| F: FIXED LENGTH  V: VARIABLE LENGTH I: COMP INIT   Y: PCL DEFINED |
+-----+

```

SEG. CODE	SEG. NAME	HIER LVL	PARENT S.CODE	TYPE	FIX VAR	PARENT		LENGTH		KEY		COMP-RTN	
						PCL	SSP	MAX	MIN	OFF	LEN	NAME	INIT
1	ROOTSEG1	1	-	R	V	-	-	900	100	2	10	-	-
2	SDEPSEG1	2	1	S	V	-	-	800	20	-	-	-	-
==> 3	DD1	2	1	D	V	-	-	800	80	-&	-	-	-
4	DD11	3	3	D	F	-	8	128	128	-	-	-	-
==> 5	DD2	2	1	D	V	Y	-	600	60	3&	5&	DFSCMPX0	I
6	DD21	3	5	D	V	Y	8	400	40	-	-	DFSCMPX0	I
==> 7	DD21A&	4	6	D	V	Y	-	200	40	-	-	-	-
8	DD212	4	6	D	V	Y	-	200	40	-	-	-	-
==> 9	DD2121	5	8	D	V	Y	-	200	40	2&	3	-	-

Figure 208. DBD Definition report for Reload (DBD information in IMSACB)

The following figure shows sample DBD Definition report that contains the DBD information in the USR file in which the first inconsistency was detected.

```

IMS HPFP UTILITIES - FPA RELOAD                                "DBD definition report"                                PAGE: 1
5698-FPP V2R1                                                2019-11-22 18:08:02
- ACBLIB : IAREA001
- ACB DSNAME : FPTL02.DBTJIE1.IMS14.DB12.ACBLIB
- DBD NAME : DEDBJN23
- RANDOMIZER : RMODTS2
- IMS LEVEL : VERSION 14 RELEASE 1
DEDB DEFINITION SEVERE DIFFERENCE DETECTED. *: DIFFERENCE &: SEVERE DIFFERENCE
DB LARGEST INFO: CI-SIZE UOW-1 NO(RAP'S/UOW) UOW-2 SEG-LEN
-----
                28,672      5              3          2          900
AREA:

```

AREA NO.	AREA NAME	CI-SIZE	UOW=	ROOT=	TOTAL CI'S				SDEP START	
					BASE	DOVF	IOVF	TOTAL	BLOCK#	RBA (HEX)
1	DB23AR1	2,048	(5,2)	(14,4)	30	20	19	69	77	00026800
2	DB23AR2	1,024	(2,1)	(16,4)	12	12	7	31	36	00009000
==> 3	DB23AR3&	4,096	(4,1)	(8,4)	12	4	15	31	38	00026000
4	DB23AR4	8,192	(2,1)	(2,1)	1	1	1	3	8	00010000
5	DB23AR5	28,672	(2,1)	(2,1)	1	1	1	3	8	00038000

```

SEGMENT:
+-----+
| LEGEND FOR SEGMENT INFORMATION |
+-----+
| R: ROOT SEGMENT  D: DDEP SEGMENT  S: SDEP SEGMENT |
| F: FIXED LENGTH  V: VARIABLE LENGTH I: COMP INIT   Y: PCL DEFINED |
+-----+

```

SEG. CODE	SEG. NAME	HIER LVL	PARENT S.CODE	TYPE	FIX VAR	PARENT		LENGTH		KEY		COMP-RTN	
						PCL	SSP	MAX	MIN	OFF	LEN	NAME	INIT
1	ROOTSEG1	1	-	R	V	-	-	900	100	2	10	-	-
2	SDEPSEG1	2	1	S	V	-	-	800	20	-	-	-	-
==> 3	DD1	2	1	D	V	-	-	800	80	2&	3	-	-
4	DD11	3	3	D	F	-	8	128	128	-	-	-	-
==> 5	DD2	2	1	D	V	Y	-	600	60	2&	3&	DFSCMPX0	I
6	DD21	3	5	D	V	Y	8	400	40	-	-	DFSCMPX0	I
==> 7	DD211&	4	6	D	V	Y	-	200	40	-	-	-	-
8	DD212	4	6	D	V	Y	-	200	40	-	-	-	-
==> 9	DD2121	5	8	D	V	Y	-	200	40	-&	-	-	-

Figure 209. DBD Definition report for Reload (DBD information in the USR file)

ACBLIB

The name of the ACBLIB DD statement.

In the DBD Definition reports that indicate the inconsistencies in the USR files, the DD name for the USR file is printed in this field.

When `IMSCATHLQ=bsdshlq` and `IMSCATACB_INPUT=CURRENT` is specified, `IMS CATALOG (CURRENT)` is printed in this field. When `IMSCATHLQ=bsdshlq` and `IMSCATACB_INPUT=PENDING` is specified, `IMS CATALOG (PENDING)` is printed in this field.

ACB DSNAME

The name of the ACBLIB data set specified by the `acblib` DD statement. The data set has the DBD definition. This field is printed when `IMSCATHLQ=*NO` is specified.

In the DBD Definition reports that indicate the inconsistencies in the USR files, the name of the ACBLIB data set that is stored in the corresponding USR file is printed in this field.

IMSCATHLQ

The high-level qualifier of the bootstrap data set of the IMS directory. This field is printed when `IMSCATHLQ=bsdshlq` is specified.

DBD NAME

The name of the DBD to be reported.

RANDOMIZER

The name of the randomizing module.

IMS LEVEL

The IMS level of the source data set from which the DBD definition information was reported.

DB LARGEST INFO

The largest value of the areas in the DBD as follows:

- `CI-SIZE` is the largest CI size of areas in the DBD.
- `UOW-1` is the largest number of CIs per UOW of areas in the DBD.
- `NO(RAP'S/UOW)` is the largest number of RAP CIs per UOW of areas in the DBD.
- `UOW-2` is the largest number of DOVF CIs per UOW of areas in the DBD.
- `SEG-LEN` is the largest length of segments of areas in the DBD.

AREA NO

The area number.

AREA NAME

The area name.

CI-SIZE

The CI size of the area.

UOW=

The `UOW=` parameter value for the area that is defined in the DBD.

ROOT=

The `ROOT=` parameter value for the area that is defined in the DBD.

TOTAL CI'S BASE

The total number of BASE CIs for the area that is defined in the DBD.

TOTAL CI'S DOVF

The total number of DOVF CIs for the area that is defined in the DBD.

TOTAL CI'S IOVF

The total number of IOVF CIs for the area that is defined in the DBD. IOVF bitmap CIs are not included.

TOTAL CI'S TOTAL

The total number of CIs for the area that is defined in the DBD. The first, second, IOVF bitmap, work UOW, and SDEP part CIs are not included.

SDEP START BLOCK#

The first block number of CI in the SDEP part of the area.

SDEP START RBA (HEX)

The hexadecimal value of the first CI in the SDEP part of the area.

SEG.CODE

The segment code.

SEG.NAME

The segment name.

HIER LVL

The hierarchical level of the segment.

PARENT S.CODE

The parent segment code of the segment.

TYPE

The type of the segment as follows:

- R is the root segment.
- S is the sequential dependent segment.
- D is the direct dependent segment.

FIX|VAR

The attribute of the segment as follows:

- F is the fixed-length segment.
- V is the variable-length segment.

PARENT PCL

Indicates whether the parent of the segment has a PCL pointer. Y means that the parent of the segment has a PCL pointer.

PARENT SSP

Indicates the number of subset pointers that the parent of the segment has, if any are defined.

LENGTH MAX

The maximum length of the segment that is defined in the DBD.

LENGTH MIN

The minimum length of the segment that is defined in the DBD.

KEY OFF

The offset of the key field of the segment, if one is defined.

KEY LEN

The key length of the segment, if one is defined.

COMP-RTN NAME

The name of the segment edit/compression exit routine of the segment, if one is defined.

COMP-RTN INIT

Tells whether COMPRTN= INIT subparameter has been defined in the DBD.

I: The COMPRTN= INIT subparameter has been defined.

SEC. IDX

The number of secondary index databases in which the segment is defined as a target segment.

In this example, both IMSACB and NEWACB are specified, and the DBD members are different in Unload processing. Two DBD Definition reports are reported, which show the differences among them marked with '*'. When there is a severe difference, this report shows the difference with '&', and processing stops.

Unloaded report

The Unloaded report provides a count of the number of segments (by segment name and segment code) that were unloaded from each area of the database.

A database total is also provided. The segment counts for each area should match the statistics that are generated by FPB DEDB Pointer Checker when it is run against the same data set. Comparing the unload segment counts with those generated by FPB DEDB Pointer Checker provides a means of verifying that all segments were unloaded. If segment count for an area is zero, it means that the unloaded area is empty.

This report is generated by the following commands:

- CHANGE
- UNLOAD

This report is not printed if you make one of the following specifications:

- GLOBAL SCAN=YES
- GLOBAL SCAN=SITE_CREATE
- GLOBAL SCAN=SITE_REPORT

The following figure is an example of the Unloaded report.

```

IMS HPFP UTILITIES - FPA UNLOAD                                "Unloaded report"
5698-FPP V2R1                                                PAGE: 1
                                                                2019-11-22 15:25:54

SEGMENTS UNLOADED FROM DATABASE: DEDBJN21

- AREA NO:      1  AREANAME: DB21AR0  DDNAME: DB21AR0  DSNAME: HPFP.MDB21AR0.ADSFP
  (JCL DD ALLOCATION)
  STARTED DATE: 2019-11-22  TIME: 15:25:54  ENDED DATE: 2019-11-22  TIME: 15:25:54

IOVF
- TOTAL CI'S      :      19833
- UNUSED CI'S     :      19827
- UNUSED CI RATIO :      100 (%)

SDEP
- TOTAL CI'S      :      193978
- UNUSED CI'S     :      186730
- UNUSED CI RATIO :         96 (%)
- RANGE           : X'35B76000' TO X'65130000'
- LOGICAL BEGIN   : CYCLE COUNT X'00000001' RBA X'35B76000'
- LOGICAL END     : CYCLE COUNT X'00000001' RBA X'377C5004'
- LOGICAL BEGIN TS : X'D194CC50C010E341'

SUMMARY OF KEY SEQUENCE ERRORS
- NUMBER OF RELATED DB RECORDS          :      0
- NUMBER OF THE SEGMENTS DETECTED AS KEY SEQUENCE ERROR :      0
- ERROR SEGMENTS SUM TOTAL INCLUDING CHILD SEGMENTS     :      0

SEGMENT  SEGMENT  SEGMENT      SEGMENTS  SEGMENTS  SKIPPED BY  SKIPPED BY
CODE     NAME     HIERARCHY    READ      UNLOADED  KEY ERROR   USER EXIT
-----
   1  ROOTSEG1  1              250,000   250,000     0           0
   2  SDEPSEG1 2              500,000   500,000     0           0 (SDEP=LOGICAL)
   3  DD102     2              250,000   250,000     0           0
   4  DD202     2              750,000   750,000     0           0
   5  DD302     2              500,000   500,000     0           0
   6  DD303     3              500,000   500,000     0           0
-----
** TOTAL **                2,750,000  2,750,000     0           0

- DATABASE: DEDBJN21 TOTAL

SEGMENT  SEGMENT  SEGMENT      SEGMENTS  SEGMENTS  SKIPPED BY  SKIPPED BY
CODE     NAME     HIERARCHY    READ      UNLOADED  KEY ERROR   USER EXIT
-----
   1  ROOTSEG1  1              250,000   250,000     0           0
   2  SDEPSEG1 2              500,000   500,000     0           0 (SDEP=LOGICAL)
   3  DD102     2              250,000   250,000     0           0
   4  DD202     2              750,000   750,000     0           0
   5  DD302     2              500,000   500,000     0           0
   6  DD303     3              500,000   500,000     0           0
-----
** TOTAL **                2,750,000  2,750,000     0           0

```

Figure 210. Unloaded report for Unload

AREA NO

The area number.

AREANAME

The area name.

DDNAME

The DDNAME with the following information:

- (DYNAMIC ALLOCATION): DDNAME is allocated dynamically.
- (JCL DD ALLOCATION): DDNAME is specified in the JCL statement.

DSNAME

The DSNAME of the area name DDNAME.

INPUT DSNAME WAS RENAMED TO DSNAME

The input area data set is renamed for the processing of the CHANGE command by specifying the IRENAME subcommand.

TOTAL CI'S

The total number of CI in each IOVF and SDEP part.

UNUSED CI'S

The number of unused CI in each IOVF and SDEP part.

UNUSED CI RATIO

The ratio of unused CI in each IOVF and SDEP part.

RANGE

The SDEP logical begin and logical end RBAs.

LOGICAL BEGIN

The cycle count and RBA of the SDEP logical begin.

LOGICAL END

The cycle count and RBA of the SDEP logical end.

LOGICAL BEGIN TS

The timestamp of the SDEP logical begin.

SUMMARY OF KEY SEQUENCE ERRORS

The summary of number of records when a sequence error in a key sequence field is detected.

NUMBER OF RELATED DB RECORDS

The number of the DB records that have segments that were detected as key sequence errors in unloading processes.

NUMBER OF THE SEGMENTS DETECTED AS KEY SEQUENCE ERROR

The number of the segments that were detected as key sequence errors in unloading processes.

ERROR SEGMENTS SUM TOTAL INCLUDING CHILD SEGMENTS

The number of the segments that were detected as key sequence errors in unloading processes and its child segments.

ERROR SEGMENTS WRITTEN TO HFPSEQER DATA SET

The number of the segments that were written to the HFPSEQER data set in unloading processes. This information is provided only when the HFPSEQER DD statement is specified together with KEYSEQCHK=BYPASS or KEYSEQCHK=WARNING.

NUMBER OF DB RECORDS WHOSE AREANAME IS CHANGED BY RANDOMIZER

The number of DB records whose area name is changed by the randomizer. This text is printed only when the randomizer is called.

NUMBER OF DB RECORDS THAT ARE NOT WRITTEN TO USR

The number of DB records that were not written to USR files because the records were randomized into new area that is not specified with the OAREA keyword.

SEGMENT CODE

The segment code.

SEGMENT NAME

The segment name.

SEGMENT HIERARCHY

The segment hierarchy.

SEGMENTS READ

The number of segments read.

SEGMENTS UNLOADED

The number of segments unloaded.

SKIPPED BY KEY ERROR

The number of segments skipped by the key sequence error.

SKIPPED BY USER EXIT

The number of segments skipped by the user exit.

SDEP PROCESS

- (SDEP=PHYSICAL) is printed after the number (that is, the number of 512-byte blocks of CI data) when SDEP=PHYSICAL is specified.
- (SDEP=LOGICAL) is printed after the number when SDEP=LOGICAL is specified.
- (SDEP=NO) is printed after the number when SDEP=NO is specified.

POINTER ERROR DETAILS

If PTRCHK=DETAIL is specified and the Unload process detects any pointer errors, the same messages as for the Resolve Pointers report are issued.

DATABASE TOTAL

The total number of segments unloaded from the database.

THE AREA WAS NOT UNLOADED DUE TO SOME PROBLEMS

The area was not processed due to processing errors.

Related reference

[Resolve Pointers report](#)

The Resolve Pointers report contains the processing information of pointer validation.

Reloaded report

The Reloaded report provides a count of the number of segments that are to be reloaded to each area of the new database and to the database total.

If the Unload or Reload is a simple reorganization of one or more areas, the segment counts and area totals for the Unloaded report and the Reloaded report will be the same. The database totals should always be the same. If segment count for an area is zero, it means that the reloaded area is empty.

This report is generated by the following commands:

- CHANGE
- UNLOAD
- RELOAD

This report is not printed if you make one of the following specifications:

- GLOBAL SCAN=YES
- GLOBAL SCAN=SITE_CREATE
- GLOBAL SCAN=SITE_REPORT

The Reloaded report for the CHANGE command is in the same format as the Reloaded report for the RELOAD command.

The following figure is a sample of the Reloaded report for Unload.

SEGMENTS RELOADED FROM DATABASE: DEDBJN21

- AREA NO: 1 AREANAME: DB21AR0

SEGMENT CODE	SEGMENT NAME	SEGMENT HIERARCHY	SEGMENTS RELOADED
1	ROOTSEG1	1	250,000
2	SDEPSEG1	2	500,000
3	DD102	2	250,000
4	DD202	2	750,000
5	DD302	2	500,000
6	DD303	3	500,000

(SDEP=LOGICAL)

** TOTAL ** 2,750,000

- DATABASE: DEDBJN21 TOTAL

SEGMENT CODE	SEGMENT NAME	SEGMENT HIERARCHY	SEGMENTS RELOADED
1	ROOTSEG1	1	250,000
2	SDEPSEG1	2	500,000
3	DD102	2	250,000
4	DD202	2	750,000
5	DD302	2	500,000
6	DD303	3	500,000

(SDEP=LOGICAL)

** TOTAL ** 2,750,000

Figure 211. Reloaded report for Unload

The following figure is a sample of the Reloaded report for Reload.

SEGMENTS RELOADED FROM DATABASE: DEDBJN21

- AREA NO: 1 AREANAME: DB21AR0 DDNAME: DB21AR0 DSNAME: HPFP.MDB21AR0.ADSFP
(JCL DD ALLOCATION)

STARTED DATE: 2019-11-22 TIME: 17:26:52 ENDED DATE: 2019-11-22 TIME: 17:26:53

IOVF

- TOTAL CI'S : 19833
- UNUSED CI'S : 19827
- UNUSED CI RATIO : 100 (%)

SDEP

- TOTAL CI'S : 193978
- UNUSED CI'S : 186834
- UNUSED CI RATIO : 96 (%)
- RANGE : X'35B76000' TO X'65130000'
- LOGICAL BEGIN : CYCLE COUNT X'00000001' RBA X'35B76000'
- LOGICAL END : CYCLE COUNT X'00000001' RBA X'3775D000'
- LOGICAL BEGIN TS : X'D194CC50C010E341'

SUMMARY OF KEY SEQUENCE ERRORS

- NUMBER OF RELATED DB RECORDS : 0
- NUMBER OF THE SEGMENTS DETECTED AS KEY SEQUENCE ERROR : 0
- ERROR SEGMENTS SUM TOTAL INCLUDING CHILD SEGMENTS : 0

SEGMENT CODE	SEGMENT NAME	SEGMENT HIERARCHY	SEGMENTS RELOADED
1	ROOTSEG1	1	250,000
2	SDEPSEG1	2	500,000
3	DD102	2	250,000
4	DD202	2	750,000
5	DD302	2	500,000
6	DD303	3	500,000

(SDEP=LOGICAL)

** TOTAL ** 2,750,000

- DATABASE: DEDBJN21 TOTAL

SEGMENT CODE	SEGMENT NAME	SEGMENT HIERARCHY	SEGMENTS RELOADED
1	ROOTSEG1	1	250,000
2	SDEPSEG1	2	500,000
3	DD102	2	250,000
4	DD202	2	750,000
5	DD302	2	500,000
6	DD303	3	500,000

(SDEP=LOGICAL)

** TOTAL ** 2,750,000

Figure 212. Reloaded report for Reload

AREA NO

The area number.

AREANAME

The area name.

SEGMENT CODE

The segment code.

SEGMENT NAME

The segment name.

SEGMENT HIERARCHY

The segment hierarchy.

SEGMENTS RELOADED

The number of segments that were reloaded. When SDEP segments are processed in physical order (SDEP=PHYSICAL), this field shows the value calculated by the following formula:

$$\text{The number of SDEP CIs to be processed} \times \frac{\text{CI size}}{512}$$

SDEP PROCESS

- (SDEP=PHYSICAL) is printed after the number when SDEP=PHYSICAL is specified.
- (SDEP=LOGICAL) is printed after the number when SDEP=LOGICAL is specified.
- (SDEP=NO) is printed after the number when SDEP=NO is specified.

In addition, the following four lines of text might be printed in DATABASE TOTAL.

- (SDEP=LOGICAL,PHYSICAL,NO) is printed after the number when the USR file consists of files that are processed with SDEP=LOGICAL, SDEP=PHYSICAL, and SDEP=NO.
- (SDEP=LOGICAL,PHYSICAL) is printed after the number when the USR file consists of files that are processed with SDEP=LOGICAL and SDEP=PHYSICAL.
- (SDEP=LOGICAL,NO) is printed after the number when the USR file consists of files that are processed with SDEP=LOGICAL and SDEP=NO.
- (SDEP=PHYSICAL,NO) is printed after the number when the USR file consists of files that are processed SDEP=PHYSICAL and SDEP=NO.

DATABASE TOTAL

The total number of segments that were reloaded for the database.

THE AREA WAS NOT RELOADED DUE TO NO USR RECORDS IN THE USR FILE(S) OR SOME USR FILE PROBLEMS

The area reloaded was not found in the USR file.

Output File report

The Output File report provides segment counts and area totals by the output file ddname.

File totals of each area and a database file total are also provided.

For UNLOAD commands, the area totals should match the area totals in the Reloaded report. The database file total should match the two preceding database file totals in the Unloaded report and the Reloaded report.

If FILECTL statements are not used (that is, ddnames default to area numbers), there is only one area per file. Conversely, if FILECTL statements are used, a file might contain data for more than one area.

This report is generated by the following commands:

- EXTRACT
- UNLOAD

This report is not printed if you make one of the following specifications:

- GLOBAL SCAN=YES

- GLOBAL SCAN=SITE_CREATE
- GLOBAL SCAN=SITE_REPORT

The following figure is a sample of the Output File report for Unload.

```

IMS HPFP UTILITIES - FPA UNLOAD                                "Output file report"
5698-FPP V2R1                                                PAGE: 7
                                                            2019-11-22 17:26:45

SEGMENT TOTALS BY OUTPUT FILE:

- FILE DDNAME: OAREA001 DSNAME: HPFP.FFPUSR.DB21AR0
  (JCL DD ALLOCATION)
- AREA NO: 1 AREANAME: DB21AR0

  SEGMENT  SEGMENT  SEGMENT  SEGMENTS
  CODE     NAME     HIERARCHY WRITTEN
  -----
    1  ROOTSEG1  1             250,000
    2  SDEPSEG1  2             500,000 (SDEP=LOGICAL)
    3  DD102     2             250,000
    4  DD202     2             750,000
    5  DD302     2             500,000
    6  DD303     3             500,000
  -----
  ** TOTAL **                2,750,000

- FILE DDNAME: OAREA001 TOTAL

  SEGMENT  SEGMENT  SEGMENT  SEGMENTS
  CODE     NAME     HIERARCHY WRITTEN
  -----
    1  ROOTSEG1  1             250,000
    2  SDEPSEG1  2             500,000 (SDEP=LOGICAL)
    3  DD102     2             250,000
    4  DD202     2             750,000
    5  DD302     2             500,000
    6  DD303     3             500,000
  -----
  ** TOTAL **                2,750,000

  *** FILE TOTAL ***                2,750,000

```

Figure 213. Output File report for Unload

The following figure is a sample of the Output File report for Unload when ACBCHK=YES is specified.

```

IMS HPFP UTILITIES - FPA UNLOAD                                "Output file report"
5698-FPP V2R1                                                PAGE: 4
                                                            2019-11-22 18:07:59

SEGMENT TOTALS BY OUTPUT FILE:

- FILE DDNAME: OAREA001 DSNAME: FPTL02.DBTJIE1.PM23240.OAREA001
  (JCL DD ALLOCATION)
- AREA NO: 1 AREANAME: DB23AR1

  SEGMENT  SEGMENT  SEGMENT  SEGMENTS
  CODE     NAME     HIERARCHY WRITTEN
  -----
    1  ROOTSEG1  1             62
    2  SDEPSEG1  2             18 (SDEP=LOGICAL)
    3  DD1       2             64
    4  DD11      3             44
    5  DD2       2             60
    6  DD21      3             102
    7  DD211     4             106
    8  DD212     4             14
    9  DD2121    5             126
  -----
  ** TOTAL **                596

- FILE DDNAME: OAREA001 TOTAL

  DBD INFORMATION RECORDS: 4

  SEGMENT  SEGMENT  SEGMENT  SEGMENTS
  CODE     NAME     HIERARCHY WRITTEN
  -----
    1  ROOTSEG1  1             62
    2  SDEPSEG1  2             18 (SDEP=LOGICAL)
    3  DD1       2             64
    4  DD11      3             44
    5  DD2       2             60
    6  DD21      3             102
    7  DD211     4             106
    8  DD212     4             14
    9  DD2121    5             126
  -----
  ** TOTAL **                596

  *** FILE TOTAL ***                596

```

Figure 214. Output File report for Unload when ACBCHK=YES is specified

The following figure is a sample of the Output File report for Extract.

SEGMENT TOTALS BY OUTPUT FILE:

- FILE DDNAME: OAREA001 DSNAME: HPFP.FFPUSR.DB21AR0
(JCL DD ALLOCATION)
- AREA NO: 1 AREANAME: DB21AR0

SEGMENT CODE	SEGMENT NAME	SEGMENT HIERARCHY	SEGMENTS WRITTEN
1	ROOTSEG1	1	250,000
2	SDEPSEG1	2	500,000
3	DD102	2	250,000
4	DD202	2	750,000
5	DD302	2	500,000
6	DD303	3	500,000

** TOTAL ** 2,750,000

- FILE DDNAME: OAREA001 TOTAL

SEGMENT CODE	SEGMENT NAME	SEGMENT HIERARCHY	SEGMENTS WRITTEN
1	ROOTSEG1	1	250,000
2	SDEPSEG1	2	500,000
3	DD102	2	250,000
4	DD202	2	750,000
5	DD302	2	500,000
6	DD303	3	500,000

** TOTAL ** 2,750,000

*** FILE TOTAL *** 2,750,000

Figure 215. Output File report for Extract

FILE DDNAME

The output file DDNAME with the following information:

- (DYNAMIC ALLOCATION): DDNAME is allocated dynamically.
- (JCL DD ALLOCATION): DDNAME is specified in the JCL statement.

DSNAME

The DSNAME of the output file DDNAME.

AREA NO

The area number.

AREANAME

The area name.

SEGMENT CODE

The segment code.

SEGMENT NAME

The segment name.

SEGMENT HIERARCHY

The segment hierarchy.

SEGMENTS WRITTEN

The number of segments that were written into the output file.

SDEP PROCESS

The SDEP process option. This item is not printed for the EXTRACT command.

- (SDEP=PHYSICAL) is printed after the number when SDEP=PHYSICAL is specified.
- (SDEP=LOGICAL) is printed after the number when SDEP=LOGICAL is specified.
- (SDEP=NO) is printed after the number when SDEP=NO is specified.

FILE DDNAME TOTAL

The total number of segments that were written into the output file for the database.

DBD INFORMATION RECORDS

The number of DBD information records that were written into the USR file by the Unload process when ACBCHK=YES is specified.

Input File report

The Input File report provides segment counts and area totals by the input file ddname.

File totals of each area and a database file total are also provided. The area totals should match the area totals in the Reloaded report. The database file total should match the preceding database total in the Reloaded report.

In the Unload process, if FILECTL statements are not used (that is, ddnames default to area numbers), there is only one area per file. Conversely, if FILECTL statements are used, a file might contain data for more than one area.

This report is generated by the RELOAD command.

This report is not printed if you make one of the following specifications:

- GLOBAL SCAN=YES
- GLOBAL SCAN=SITE_CREATE
- GLOBAL SCAN=SITE_REPORT

The following figure is an example of the Input File report.

```
IMS HFPF UTILITIES - FPA RELOAD                                "Input file report"                                PAGE: 5
5698-FPP V2R1                                                2019-11-22 18:59:47
```

SEGMENT TOTALS BY INPUT FILE :

```
- FILE DDNAME: IAREA001 DSNAME: HFPF.HFPUSR.DB21AR0
(JCL DD ALLOCATION)
- AREA NO: 1 AREANAME: DB21AR0
```

SEGMENT CODE	SEGMENT NAME	SEGMENT HIERARCHY	SEGMENTS READ	SEGMENTS RELOADED	SKIPPED BY KEY ERROR
1	ROOTSEG1	1	250,000	250,000	0
2	SDEPSEG1	2	500,000	500,000	0 (SDEP=LOGICAL)
3	DD102	2	250,000	250,000	0
4	DD202	2	750,000	750,000	0
5	DD302	2	500,000	500,000	0
6	DD303	3	500,000	500,000	0
** TOTAL **			2,750,000	2,750,000	0

```
- FILE DDNAME: IAREA001 TOTAL
```

SEGMENT CODE	SEGMENT NAME	SEGMENT HIERARCHY	SEGMENTS READ	SEGMENTS RELOADED	SKIPPED BY KEY ERROR
1	ROOTSEG1	1	250,000	250,000	0
2	SDEPSEG1	2	500,000	500,000	0 (SDEP=LOGICAL)
3	DD102	2	250,000	250,000	0
4	DD202	2	750,000	750,000	0
5	DD302	2	500,000	500,000	0
6	DD303	3	500,000	500,000	0
** TOTAL **			2,750,000	2,750,000	0

```
*** FILE TOTAL ***
```

*** FILE TOTAL ***			2,750,000	2,750,000	0
--------------------	--	--	-----------	-----------	---

Figure 216. Input File report for Reload

FILE DDNAME

The input file DDNAME with the following information:

- (DYNAMIC ALLOCATION): DDNAME is allocated dynamically.
- (JCL DD ALLOCATION): DDNAME is specified on the JCL statement.

DSNAME

The DSNAME of the input file DDNAME.

When USR files are specified as concatenated in a Reload process, only the name of the first data set of all that are concatenated is printed in DSNAME.

AREA NO

The area number.

AREANAME

The area name.

SEGMENT CODE

The segment code.

SEGMENT NAME

The segment name.

SEGMENT HIERARCHY

The segment hierarchy.

SEGMENTS READ

The number of segments read.

SEGMENTS RELOADED

The number of segments reloaded.

SKIPPED BY KEY ERROR

The number of segments skipped by the key sequence error.

SDEP PROCESS

- (SDEP=PHYSICAL) is printed after the number when SDEP=PHYSICAL is specified.
- (SDEP=LOGICAL) is printed after the number when SDEP=LOGICAL is specified.
- (SDEP=NO) is printed after the number when SDEP=NO is specified.

In addition, the following four lines of text might be printed in FILE DDNAME TOTAL.

- (SDEP=LOGICAL,PHYSICAL,NO) is printed after the number when the USR file consists of files that are processed with SDEP=LOGICAL, SDEP=PHYSICAL, and SDEP=NO.
- (SDEP=LOGICAL,PHYSICAL) is printed after the number when the USR file consists of files that are processed with SDEP=LOGICAL and SDEP=PHYSICAL.
- (SDEP=LOGICAL,NO) is printed after the number when the USR file consists of files that are processed with SDEP=LOGICAL and SDEP=NO.
- (SDEP=PHYSICAL,NO) is printed after the number when the USR file consists of files that are processed SDEP=PHYSICAL and SDEP=NO.

FILE DDNAME TOTAL

The total number of segments that were read from the input file for the database.

NO RECORDS FOR RELOAD AREAS SPECIFIED BY OAREA= KEYWORD ARE FOUND

Records for the area that is specified by the OAREA keyword are not found in the USR file.

NO RECORDS FOR RELOAD AREAS ARE PROCESSED (THE SAME INPUT FILE WAS PROCESSED BY ANOTHER INPUT DD)

The USR file was specified and has been processed by another DD statement. No records for the reloaded areas are processed for the USR file.

RECORDS FOR SAME RELOAD AREA WERE ALREADY PROCESSED BY DIFFERENT UNLOADED SEGMENT RECORD FILE

The USR file was specified for reloading a specific area, but the area was already reloaded by different USR files.

NO AREA INFORMATION RECORD WAS FOUND IN THIS USR FILE

The first record of the USR file is not an area information record.

UNLOADED SEGMENT RECORDS WITH SDEP=PHYSICAL ACCOMPANIED BY DBD CHANGE DETECTED

The USR file was created with SDEP=PHYSICAL and in a different DBD structure.

Resolve Pointers report

The Resolve Pointers report contains the processing information of pointer validation.

If the Analyze process detects any pointer errors, the error messages are issued in this report.

If the DEDB that is being processed has an SDEP segment defined, and SDEPCHKLVL=FULL is specified, logical begin (LB) and logical end (LE) are printed as SDEP Processing information in this report.

This report is generated by the following commands:

- ANALYZE with the PTRCHKLVL=FULL keyword or the SDEPCHKLVL=FULL keyword
- CHANGE with the PTRCHKLVL=FULL keyword or the SDEPCHKLVL=FULL keyword
- RELOAD with the PTRCHKLVL=FULL keyword or the SDEPCHKLVL=FULL keyword
- REORG with the PTRCHKLVL=FULL keyword or the SDEPCHKLVL=FULL keyword
- TUNE

The following figure is an example of the Resolve Pointers report.

```
IMS HFPF UTILITIES - FPA ANALYZE                                "Resolve pointers report"
5698-FPP V2R1                                                  PAGE: 1
- AREA NO: 1 AREANAME: DB21AR0 DDNAME: DB21AR0 DSNAME: HFPF.MDB21AR0.ADSFP (JCL DD ALLOCATION)
  2019-05-31 19:17:22
- Processing information -
  SDEP PROCESSING INFORMATION - LB: 00000001 00008000
                                LE: 00000001 00009400
                                TS: X'CDB3FD2E9C2EAB80'
CROSS-REFERENCE VALIDATION COMPLETED FOR ROOT, DDEP, AND SDEP SEGMENTS
- Error details -
0304: SEGMENT AT RBA: 00002808 (SEGCODE: 01) POINTED TO BY MULTIPLE SEGMENTS
      - POINTER IN SEGMENT AT RBA: 00001008 (SEGCODE: 01)
      - POINTER IN SEGMENT AT RBA: 00001000 (SEGCODE: 00)
0306: SEGMENT AT RBA: 00001008 IS AN ORPHAN
0308: SEGMENT AT RBA: 00001222 (SEGCODE: 03) NOT IN RAP CHAIN - POINTED TO BY SEGMENT AT RBA: 00001008
0308: SEGMENT AT RBA: 000012CC (SEGCODE: 04) NOT IN RAP CHAIN - POINTED TO BY SEGMENT AT RBA: 00001222
0308: SEGMENT AT RBA: 00001354 (SEGCODE: 04) NOT IN RAP CHAIN - POINTED TO BY SEGMENT AT RBA: 000012CC
0308: SEGMENT AT RBA: 000013DC (SEGCODE: 04) NOT IN RAP CHAIN - POINTED TO BY SEGMENT AT RBA: 00001354
0308: SEGMENT AT RBA: 00001464 (SEGCODE: 05) NOT IN RAP CHAIN - POINTED TO BY SEGMENT AT RBA: 00001008
0308: SEGMENT AT RBA: 00001512 (SEGCODE: 06) NOT IN RAP CHAIN - POINTED TO BY SEGMENT AT RBA: 00001464
0308: SEGMENT AT RBA: 000015A0 (SEGCODE: 06) NOT IN RAP CHAIN - POINTED TO BY SEGMENT AT RBA: 00001512
0308: SEGMENT AT RBA: 0000162E (SEGCODE: 06) NOT IN RAP CHAIN - POINTED TO BY SEGMENT AT RBA: 000015A0
0308: SEGMENT AT RBA: 0000168C (SEGCODE: 07) NOT IN RAP CHAIN - POINTED TO BY SEGMENT AT RBA: 0000162E
0308: SEGMENT AT RBA: 00001742 (SEGCODE: 05) NOT IN RAP CHAIN - POINTED TO BY SEGMENT AT RBA: 00001464
0308: SEGMENT AT RBA: 0001C008 (SEGCODE: 03) NOT IN RAP CHAIN - POINTED TO BY SEGMENT AT RBA: 00001222
0308: SEGMENT AT RBA: 0001C172 (SEGCODE: 04) NOT IN RAP CHAIN - POINTED TO BY SEGMENT AT RBA: 000013DC
0308: SEGMENT AT RBA: 0001C1FA (SEGCODE: 06) NOT IN RAP CHAIN - POINTED TO BY SEGMENT AT RBA: 00001742
0308: SEGMENT AT RBA: 0001C398 (SEGCODE: 06) NOT IN RAP CHAIN - POINTED TO BY SEGMENT AT RBA: 0001C1FA
0308: SEGMENT AT RBA: 0001C536 (SEGCODE: 06) NOT IN RAP CHAIN - POINTED TO BY SEGMENT AT RBA: 0001C398
0308: SEGMENT AT RBA: 0001C604 (SEGCODE: 06) NOT IN RAP CHAIN - POINTED TO BY SEGMENT AT RBA: 0001C536
18 POINTER ERRORS DETECTED.
```

Figure 217. Resolve Pointers report

AREA NO

The area number.

AREANAME

The area name.

DDNAME

The ddname of the area data set.

DSNAME

The data set name of the area that the DDNAME specifies.

(DYNAMIC ALLOCATION)

The data set is allocated dynamically.

(JCL DD ALLOCATION)

The data set is specified in the JCL statement.

SDEP PROCESSING INFORMATION

LB

The cycle count and RBA of the SDEP logical begin.

LE

The cycle count and RBA of the SDEP logical end.

TS

The time stamp of the SDEP logical begin.

CROSS-REFERENCE VALIDATION COMPLETED FOR [ROOT, DDEP, AND SDEP | ROOT AND DDEP | ROOT AND SDEP | ROOT | SDEP] SEGMENTS

This message indicates which segments were checked for cross-reference validation.

Error details

This section contains messages that describe the pointer errors that the Analyze process detected. For explanations of these messages, see [“Resolve Pointers report messages” on page 1416](#).

NO POINTER ERRORS DETECTED.

This message indicates that the Analyze process detected no pointer errors.

Related reference

[Resolve Pointers report messages](#)

The following information is about messages and codes that are generated in the Resolve Pointers report or the Pointer Checksum report.

Pointer Checksum report

The Pointer Checksum report contains the processing information of pointer checksum validation.

If the Analyze process detects any pointer errors, the error messages are issued in this report.

If the DEDB that is being processed has an SDEP segment defined, and SDEPCHKLVL=QUICK is specified, logical begin (LB) and logical end (LE) are printed as SDEP Processing information in this report.

This report is generated by the following commands specified with the PTRCHKLVL=QUICK keyword or the SDEPCHKLVL=QUICK keyword:

- ANALYZE
- CHANGE
- RELOAD
- REORG

The following figure is an example of the Pointer Checksum report.

```

IMS HPFP UTILITIES - FPA ANALYZE                "Pointer checksum report"
5698-FPP V2R1                                     PAGE: 1
                                                2019-05-31 19:17:22

- AREA NO:      1  AREANAME: DB21AR0  DDNAME: DB21AR0  DSNAME: HPFP.MDB21AR0.ADSFP
                                                (JCL DD ALLOCATION)
- Processing information -
SDEP PROCESSING INFORMATION - LB: 00000001 00008000
                               LE: 00000001 00009400
                               TS: X'CDB3FD2E9C2EAB80'
CHECKSUM VALIDATION COMPLETED FOR ROOT, DDEP, AND SDEP SEGMENTS
- Error details -
0101: INCORRECT BLK TYPE ID IN CI AT RBA: 00000800 - CI BYPASSED (DATA VALUE: 03 OFFSET: 0002)
CHECKSUM POINTER (PTF/PCF) ERRORS DETECTED.
CHECKSUM POINTER (SDEP) ERRORS DETECTED.

```

Figure 218. Pointer Checksum report

AREA NO

The area number.

AREANAME

The area name.

DDNAME

The ddname of the area data set.

DSNAME

The data set name of the area that the DDNAME specifies.

(DYNAMIC ALLOCATION)

The data set is allocated dynamically.

(JCL DD ALLOCATION)

The data set is specified in the JCL statement.

SDEP PROCESSING INFORMATION**LB**

The cycle count and RBA of the SDEP logical begin.

LE

The cycle count and RBA of the SDEP logical end.

TS

The time stamp of the SDEP logical begin.

CHECKSUM VALIDATION COMPLETED FOR [ROOT, DDEP, AND SDEP | ROOT AND DDEP | ROOT AND SDEP | ROOT | SDEP] SEGMENTS

This message indicates which segments were checked for checksum validation.

Error details

This section contains messages that describe the pointer errors that the Analyze process detected. For explanations of these messages, see [“Resolve Pointers report messages”](#) on page 1416.

NO CHECKSUM POINTER ERRORS DETECTED.

This message indicates that the Analyze process detected no checksum pointer errors.

CHECKSUM POINTER [(PTF/PCF) | (PCL)] ERRORS DETECTED.

This message indicates that the Analyze process detected checksum pointer errors in the root and direct dependent segments.

CHECKSUM POINTER (SDEP) ERRORS DETECTED.

This message indicates that the Analyze process detected checksum pointer errors in the SDEP segments.

Related reference

[Resolve Pointers report messages](#)

The following information is about messages and codes that are generated in the Resolve Pointers report or the Pointer Checksum report.

Pointer Chain Reconstruction report

The Pointer Chain Reconstruction report contains the pointer information and the CI dump number.

The Pointer Chain Reconstruction report shows the following information:

- A list of all pointers to user-specified targets that were detected during the analysis runs
- CI dump number of the CI dump that contains the user-specified target

This report is generated by the CISNAP=YES keyword.

The following figure is an example of the Pointer Chain Reconstruction report.

AREA NAME	AREA NO	DUMP NO	<-- TARGET RBA	--> SC	<---- SOURCE RBA	----> SC	PTR TYPE	----- MESSAGES ----->	
DB21AR0	1	0001	00002E2A	7				RECORD OF TARGET RBA HAS PROBLEM(S) WITH MESSAGE ID 0304	
			00002E2A	7				RECORD OF TARGET RBA HAS PROBLEM(S) WITH MESSAGE ID 0303	
			00002E2A	7				RECORD OF TARGET RBA HAS PROBLEM(S) WITH MESSAGE ID 0307	
			00002E2A	7				RECORD OF TARGET RBA HAS PROBLEM(S) WITH MESSAGE ID 0311	
			00002E2A	7				SEGMENT FOUND IN THE RBA	
			00002E2A	7	00001742	5	PCF		SOURCE SEGMENT POINTS TO THE RBA
					00002DDC	6	PCF		SOURCE SEGMENT POINTS TO THE RBA
					00002DDC	6	PCL		SOURCE SEGMENT POINTS TO THE RBA
			0002	0001C1FA	6				RECORD OF TARGET RBA HAS PROBLEM(S) WITH MESSAGE ID 0306
				0001C1FA	6				SEGMENT FOUND IN THE RBA
				0001C1FA	6				NO SOURCE SEGMENT POINTS TO THE RBA
			0002	0001C398	6				RECORD OF TARGET RBA HAS PROBLEM(S) WITH MESSAGE ID 0308
				0001C398	6				SEGMENT FOUND IN THE RBA
				0001C398	6	0001C1FA	6	PTF	SOURCE SEGMENT POINTS TO THE RBA
			0002	0001C536	6				RECORD OF TARGET RBA HAS PROBLEM(S) WITH MESSAGE ID 0308
				0001C536	6				SEGMENT FOUND IN THE RBA
				0001C536	6	0001C398	6	PTF	SOURCE SEGMENT POINTS TO THE RBA
			0002	0001C6D4	6				RECORD OF TARGET RBA HAS PROBLEM(S) WITH MESSAGE ID 0308
				0001C6D4	6				SEGMENT FOUND IN THE RBA
				0001C6D4	6	0001C536	6	PTF	SOURCE SEGMENT POINTS TO THE RBA
			00001742	5	PCL	SOURCE SEGMENT POINTS TO THE RBA			

*** END OF POINTER CHAIN RECONSTRUCTION REPORT ***

Figure 219. Pointer Chain Reconstruction report

AREA NAME

Shows the area name that has the pointer error.

AREA NO.

Shows the area number that has the pointer error.

DUMP NO.

Shows the number that is shown in the CI Map/CI Dump report of the HFPASnap data set.

The following two fields describe the target of a pointer:

RBA

Shows the relative byte address (in hexadecimal) of the target of a segment.

SC

Shows the segment code (in hexadecimal) of the target of the pointer.

The following four fields describe the segment that contains the pointer (also called the source of the pointer):

RBA

Shows the relative byte address (in hexadecimal) of the segment that contains the pointer.

SC

Shows the segment code (in hexadecimal) of the segment that contains the pointer.

PTR TYPE

Shows the type of pointer.

DESCRIPTION

Shows the error message text.

CI Map/CI Dump report

The CI Map/CI Dump report is used to analyze DEDB area CIs to determine the best way to repair them.

The report provides the following information:

- CI map, which is a list of CI header, all the segments, free space elements, scrap bytes, unknown area, and CI trailer that the Analyze process found in the CI
- CI dump that is logically formatted based on the CI map

CI Map section

The following figure is an example of the CI Map section.

IMS HPFP UTILITIES 5698-FPP V2R1		"CI map / CI dump report"							PAGE: 1 2019-11-22 17:22:49	
DBNAME: DEDBJN21 AREA#: 00001		AREANAME: DB21AR0		DSNAME: HPFP.MDB21AR0.ADSFP				DUMP NO. 0001		
CI-NO: 77 CI-RBA: 26800										
STORAGE	TYPE (HEX)	RBA	ADDRESS	FLAGS	SC	COUNTERS	LENGTH	ROOT#	TYPE (CHAR)	
9562000	C8C5C1C4C5D94040	00026800	091547C0	000000	00	0000000000000000	000C	0000	(HEADER)	
956202C	E2C4C5D7E2C5C7F1	0002680C	091547CC	000000	02	0000000000000000	0112	0000	(SDEPSEG1)	
9562058	E2C4C5D7E2C5C7F1	0002691E	091548DE	000000	02	0000000000000000	0112	0000	(SDEPSEG1)	
9562084	E2C4C5D7E2C5C7F1	00026A30	091549F0	000000	02	0000000000000000	0112	0000	(SDEPSEG1)	
95620B0	E2C4C5D7E2C5C7F1	00026B42	09154B02	000000	02	0000000000000000	0112	0000	(SDEPSEG1)	
95620DC	E2C4C5D7E2C5C7F1	00026C54	09154C14	000000	02	0000000000000000	0112	0000	(SDEPSEG1)	
9562108	E2C4C5D7E2C5C7F1	00026D66	09154D26	000000	02	0000000000000000	0112	0000	(SDEPSEG1)	
9562134	E405D2D5D6E6D540	00026E78	09154E38	000000	00	0000000000000000	017D	0000	(UNKNOWN)	
9562160	E3D9C1C9D3C5D940	00026FF5	09154FB5	000000	00	0000000000000000	000B	0000	(TRAILER)	

Figure 220. CI Map section

STORAGE

Shows the memory address of the first byte in that line of the CI map.

TYPE (HEX)

Shows the hexadecimal value of TYPE (CHAR).

RBA

Shows the relative byte address of the area that is defined by this CI map entry.

ADDRESS

Shows the address in memory.

FLAGS

Always contains zero.

SC

Shows the segment code. If TYPE contains the segment name, this field is zero.

COUNTERS

Always contains zero.

LENGTH

Shows the length of the area that is defined by this CI map entry.

ROOT#

Always contains zero.

TYPE (CHAR)

Shows one of the following entries in the database CI:

segment-name

Shows the segment that includes prefix and data.

FSE

Shows the free space element.

SHORT-FS

Shows scraps (that is, free space less than 4 bytes).

UNKNOWN

Shows the area that the Analyze process could not identify as a segment, FSE, or scrap bytes area. If the CI is in the SDEP part, segments and unused area are treated as unknown areas.

HEADER

Is the first 4 or 8 bytes of the CI that contains the following items:

FSEAP 2 bytes:

Show the offsets to the first Free Space Element (FSE). If the CI is in the SDEP part because it does not contain FSEs, these two bytes are unused.

CI 2 bytes:

Show the use of this CI and the meaning of the next four bytes. DSECT DBFDBLK in the distributed IMS libraries provides a detailed mapping of these fields.

RAP 4 bytes:

If this CI belongs to the base section of the root addressable part, this field shows the Root Anchor Point (RAP). All root segments randomizing to this CI are chained off by this RAP in ascending key sequence. There is only one RAP per CI.

Note: In the dependent and independent overflow parts, these 4 bytes are used by Fast Path for control information. There is no RAP in SDEP CIs.

TRAILER

Is the last 13 bytes of the CI that contains the following items:

CUSN 2 bytes:

Show the CI update sequence number (CUSN) that is maintained in each CI except in the SDEP part.

RBA 4 bytes:

Show the relative byte address (RBA) of this CI.

RDF 3 bytes:

Show the record definition field (VSAM control information).

CIDF 4 bytes:

Show the CI definition field (VSAM control information).

CI Dump section

The following figure is an example of the CI Dump section.

```
IMS HFPF UTILITIES                                "CI map / CI dump report"                                PAGE: 2
5698-FPP V2R1                                    2019-11-22 17:22:49

DBNAME: DEDBJN21  AREA#: 00001  AREANAME: DB21AR0  DSNAME: HFPF.MDB21AR0.ADSFP  DUMP NO. 0001
CI-NO: 77  CI-RBA: 26800
RBA  OFFSET

26800 0000 06781005 C8C6D7C1 40404040
2680C 000C 02000000 00000000 00000108 5C5C5C40 E2C4C5D7 E2C5C7F1 407BF0F1 60F0F140 *...HFPA *
      002C 5C5C5C40 40404040 40404040 40404040 40404040 40404040 40404040 40404040 *.....SDEPSEG1 #01-01 *
      004C 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 **** *
      LINES 0002686C-000268EC SAME AS ABOVE *
      010C 40404040 40404040 4040BD38 EF4690C4 6104 *.....D/. *
2691E 011E 02000000 00010002 680C0108 5C5C5C40 E2C4C5D7 E2C5C7F1 407BF0F1 60F0F240 *.....SDEPSEG1 #01-02 *
      013E 5C5C5C40 40404040 40404040 40404040 40404040 40404040 40404040 40404040 **** *
      015E 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 * *
      LINES 0002697E-000269FE SAME AS ABOVE *
      021E 40404040 40404040 4040BD38 EF4690C4 6104 *.....D/. *
26A30 0230 02000000 00010002 691E0108 5C5C5C40 E2C4C5D7 E2C5C7F1 407BF0F1 60F0F340 *.....SDEPSEG1 #01-03 *
      0250 5C5C5C40 40404040 40404040 40404040 40404040 40404040 40404040 40404040 **** *
      0270 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 * *
      LINES 00026A90-00026B10 SAME AS ABOVE *
      0330 40404040 40404040 4040BD38 EF4690C4 6104 *.....D/. *
26B42 0342 02000000 00010002 6A300108 5C5C5C40 E2C4C5D7 E2C5C7F1 407BF0F1 60F0F440 *.....SDEPSEG1 #01-04 *
      0362 5C5C5C40 40404040 40404040 40404040 40404040 40404040 40404040 40404040 **** *
      0382 40404040 40404040 40404040 40404040 40404040 40404040 40404040 40404040 * *
      LINES 00026BA2-00026C22 SAME AS ABOVE *
      0442 40404040 40404040 4040BD38 EF4690C4 6104 *.....D/. *
```

Figure 221. CI Dump section

RBA

Shows the relative byte address of the first byte in that line of the printed CI dump.

OFFSET

Shows the offset from the first byte of the CI to the first byte in that line of the printed CI dump.

Process Summary of Analysis report

The Process Summary of Analysis report contains the processing information of the analysis report.

An echo of the control statement that you specified in the HFPAPARM and some processing messages are issued in this report.

This report is generated by the following commands:

- ANALYZE with the REPORT subcommand and the PTRCHKLVL=FULL keyword or the PTRCHKLVL=QUICK keyword

- CHANGE with the REPORT subcommand and the PTRCHKLVL=FULL keyword or the PTRCHKLVL=QUICK keyword
- RELOAD with the REPORT subcommand and the PTRCHKLVL=FULL keyword or the PTRCHKLVL=QUICK keyword
- REORG with the REPORT subcommand and the PTRCHKLVL=FULL keyword or the PTRCHKLVL=QUICK keyword
- TUNE

The following figure is an example of the Process Summary of Analysis report.

```

IMS HPFP UTILITIES                                "Process summary of analysis report"                                PAGE:      1
5698-FPP V2R1                                     2019-11-22 14:04:10

-HFPAPARM
SYNTBL=99

0500I PROCESSING STARTED FOR AREA:      1 (AREANAME: DB21AR0 )
0501I PROCESSING ENDED NORMALLY FOR AREA:  1 (AREANAME: DB21AR0 )
0500I PROCESSING STARTED FOR AREA:      2 (AREANAME: DB21AR1 )
0501I PROCESSING ENDED NORMALLY FOR AREA:  2 (AREANAME: DB21AR1 )
0500I PROCESSING STARTED FOR AREA:      3 (AREANAME: DB21AR2 )
0501I PROCESSING ENDED NORMALLY FOR AREA:  3 (AREANAME: DB21AR2 )

```

Figure 222. Process Summary of Analysis report

Related reference

[Process Summary of Analysis report messages](#)

The following information is about messages and codes that are generated in the Process Summary of Analysis report and the TUNED Process Summary of Analysis report.

DEDB Area Analysis reports

The DEDB Area Analysis reports include the comprehensive analysis information of an area.

The DEDB Area Analysis reports include the following reports:

- Freespace Analysis report
- DB Record Profile Analysis report
- DB Record Placement Analysis report
- Segment Placement Analysis report
- Segment I/O Analysis report
- Synonym Chain Analysis report

This report is generated by the following commands:

- ANALYZE with the REPORT subcommand and the PTRCHKLVL=FULL keyword
- CHANGE with the REPORT subcommand and the PTRCHKLVL=FULL keyword
- RELOAD with the REPORT subcommand and the PTRCHKLVL=FULL keyword
- REORG with the REPORT subcommand and the PTRCHKLVL=FULL keyword
- TUNE

The Freespace Analysis reports are also generated by the following commands:

- ANALYZE with the REPORT subcommand and the PTRCHKLVL=QUICK keyword
- CHANGE with the REPORT subcommand and the PTRCHKLVL=QUICK keyword
- RELOAD with the REPORT subcommand and the PTRCHKLVL=QUICK keyword
- REORG with the REPORT subcommand and the PTRCHKLVL=QUICK keyword

Notes:

- The header of each report is DEDB AREA ANALYZIZ REPORTS.

- In DEDB Area Analysis reports, when a value exceeds the number of digits available for a column, asterisks (*) are shown instead of the value.

Freespace Analysis report

The Freespace Analysis report provides the information about the free space in an area.

The Freespace Analysis report provides the following information:

- Summary and detailed information about the amount of free space in each section of the database (that is, RAA BASE, DOVF, and IOVF)
- Statistics about the number and size of the free space elements in each section of the database
- Statistics about the number of pointers to other CIs in the database
- Overflow usage analysis (DOVF and IOVF)

The Freespace Analysis report provides information that can be used to determine:

- When to reorganize or expand an area
- The optimal values for the UOW and ROOT parameters

This report also provides information for monitoring DOVF and IOVF usage.

Notes on dynamic allocation:

- The total usable space in a CI is CI size minus 21 bytes.
- The length of the free space element (FSE) is included in usable free space.
- Scraps (that is, free space less than 4 bytes in length) are also included in the free space calculation.
- DATE and TIME in the report header are the date and time that show the time of run. They are printed as *yyyy-dd-mm* and *hh:mm:ss*.

The following sections contain DEDB Area Analysis reports and their descriptions. Percentage calculations are truncated, which permits more accurate monitoring of overflow usage (that is, the percentage of DOVF or IOVF is less than 100%).

Summary section

The following figure is an example of the summary section of the Freespace Analysis report.

```

IMS HPFP UTILITIES - FPA ANALYZE                                "DEDB AREA ANALYSIS REPORTS"                                PAGE: 1
5698-FPP V2R1                                                                                                                                           2019-10-12 20:10:50

DBDNAME: DEDB0005  AREA NAME  : DB05AR4
                   AREA NUMBER: 4

                   AREA DESCRIPTION
                   RMNAME  : RMODTS2N
                   CI-SIZE : 512
                   UOW=(30,1)  ROOT=(150,100)

                   UOW'S IN RAA: 50
                   NO. OF RAPS : 1,450
                   CI'S IN DOVF: 50
                   CI'S IN IOVF: 3,000
                   NO. IOVF DIR: 25
                   CI'S IN SDEP: 2,818
                   RAA RBA RANGE : X'00000400' TO X'000BBC00'
                   IOVF RBA RANGE: X'000BBC00' TO X'00232C00'
                   REORG RBA RANGE: X'00232C00' TO X'00236800'
                   SDEP RANGE    : X'00236800' TO X'00396C00'
                   LOGICAL BEGIN : CYCLE COUNT X'00000001' RBA X'00236800'
                   LOGICAL END   : CYCLE COUNT X'00000001' RBA X'00239A00'
                   SDEP LOGICAL BEGIN TS : X'D322FDB0C685F1C9'
                   END OF AREA RBA : X'00396C00'

```

Figure 223. Summary section of Freespace Analysis report

Overview and RAA BASE sections

The following figure is an example of the overview and the RAA BASE sections for the Freespace Analysis report.

```

IMS HPFP UTILITIES - FPA ANALYZE                                "DEDB AREA ANALYSIS REPORTS"                                PAGE: 2
5698-FPP V2R1                                                  DDNAME: DB23AR1          DSNAME : IMSVS.DB23.DB23AR1.ADS1    2019-11-22 14:00:09
DBDNAME: DEDBJN23      AREA NAME : DB23AR1                    DSTYPE: ADS
                        AREA NUMBER: 1

FREESPACE ANALYSIS - *** OVERVIEW ***                          %FREESPACE - RAA BASE: 55% %USABLE FREESPACE: 55%
                                                                DOVF : 32% 31%
                                                                IOVF : 78% 78%
                                                                SDEP : 94% (LB: 00000001 00078800, LE: 00000001 0007EC00)

FREESPACE ANALYSIS - RAA BASE
%FS  NO. CI'S  %CI  ---NO. OF FSE'S---  ---SIZE OF FSE'S---  # USE  ---PTR. TO DOVF---  ---PTR. TO IOVF---
      AVG S/D  MAX  MIN  AVG S/D  MAX  MIN  OVFLOW  AVG S/D  MAX  MIN  AVG S/D  MAX  MIN
100      84  43
90-99      0
80-89      0
70-79     10  5  1.6  .5  2  1  470 304  753  53  9  1.3  .9  2  0  .5  .8  2  0
60-69      0
50-59      6  3  1.0  .0  1  1  502  0  502  502  6  2.0  1.4  3  0  1.0  1.4  3  0
40-49      3  2  1.0  .0  1  1  471  9  477  459  3  1.3  .5  2  1  1.3  .5  2  1
30-39     12  6  1.0  .0  1  1  352 13  367  341 12  2.3  1.6  4  0  1.2  1.4  4  0
20-29     16  8  1.1  .4  2  1  211  91  252  1  15  1.6  .7  2  0  .1  .5  2  0
10-19     25 13  1.0  .0  1  1  140  25  193 116 22  1.5  1.3  4  0  .6  1.1  4  0
0-9       39 20  .6  .5  1  0  13  20  58  0  24  1.6  2.2 12  0  .3  1.0  4  0
OVFL USAGE SUMMARY
CI'S USING OVERFLOW: NO.: 100  %CI'S: 83%

```

Figure 224. Overview and RAA BASE sections of the Freespace Analysis report

The following descriptions pertain to the overview section.

%FREESPACE - RAA BASE

Shows the total amount of unused space in the RAA BASE section, expressed as a percentage of total usable space.

%FREESPACE DOVF

Shows the total amount of unused space in the DOVF section, expressed as a percentage of total usable space.

%FREESPACE IOVF

Shows the total amount of unused space in the IOVF section, expressed as a percentage of total usable space.

Note: IOVF bit maps are excluded from the free space calculations.

%FREESPACE SDEP

Shows the total amount of usable space, unused space, LB, and LE information in the SDEP part at the time of the run. Data for this calculation is obtained from the DMAC.

The following descriptions pertain to the RAA BASE section.

%FS

Shows a free space percentage range that is used for classifying CIs for reporting purposes.

NO. CI'S

Shows the total number of CIs in a section of the database that contain free space within the bounds of the reporting range.

%CI

Shows the percentage of CIs in a database section that contain free space within the bounds of the reporting range.

NO. OF FSE'S

Shows statistics about the number of free space elements (FSEs) that were found in CIs within a free space reporting range.

AVG

Shows the average number of FSEs in a CI.

S/D

Shows the standard deviation for the average number of FSEs and provides a measure of variation.

MAX

Shows the maximum number of FSEs in any CI within a free space reporting range.

MIN

Shows the minimum number of FSEs in any CI within a free space reporting range.

SIZE OF FSE'S

Shows statistics about the size of free space elements (FSEs) that were found in CIs within a free space reporting range.

AVG

Shows the average size of the FSEs.

S/D

Shows the standard deviation for the average FSE size.

MAX

Shows the length of the longest FSE in any CI within a free space reporting range.

MIN

Shows the length of the shortest FSE in any CI within a free space reporting range.

USE OVFLOW

Shows the number of RAA BASE CIs (that is, RAPs) within the free space reporting range that use (reference) either dependent overflow (DOVF) or independent overflow (IOVF) CIs.

PTR. TO DOVF (and PTR. TO IOVF)

Shows statistics about the number of references from a CI to CIs in the dependent overflow section (and independent overflow section). A reference is a PTF, PCF, PCL, or SSP pointer that points to a segment that is not in the same CI. For RAA BASE, references also include a RAP that points to a root that is in a different CI (that is, a DOVF or IOVF CI).

Note: Each specific reference is tallied. Two references from a CI to the same external CI are tallied as 2.

AVG

Shows the average number of external references for all CIs within a free space reporting range.

S/D

Shows the standard deviation for the average number of external references.

MAX

Shows the maximum number of external references from any CI within a free space reporting range.

MIN

Shows the minimum number of external references from any CI within a free space reporting range.

OVFL USAGE SUMMARY

Shows the total number and percentage of used RAA BASE CIs (that is, RAPs) that use DOVF or IOVF.

DOVF and IOVF sections

The following figure is an example of the DOVF and IOVF sections of the Freespace Analysis report.

IMS HPFP UTILITIES - FPA ANALYZE		"DEDB AREA ANALYSIS REPORTS"										PAGE: 3															
5698-FPP V2R1												2019-11-22 14:00:09															
DBDNAME: DEDBJN23		AREA NAME : DB23AR1		DDNAME: DB23AR1		DSNAME : IMSVS.DB23.DB23AR1.ADS1																					
		AREA NUMBER: 1		DSTYPE: ADS																							
FREESPACE ANALYSIS - DOVF																											
%DOVF	#UOWS	%	-DOVF CI'S USED				-NO. OF FSE'S--				----SIZE OF FSE'S----			--PTR TO BASE--			--PTR TO DOVF--			--PTR TO IOVF--							
FREE	UOW		AVG	S/D	MAX	MIN	AVG	S/D	MAX	MIN	AVG	S/D	MAX	MIN	AVG	S/D	MAX	MIN	AVG	S/D	MAX	MIN					
100	10	15																									
90-99	0																										
80-89	2	3	1.0	.0	1	1	1.0	.0	1	1	752	1	753	752	0				0								
70-79	4	6	1.0	.0	1	1	1.2	.4	2	1	401	165	502	76	0				0								
60-69	1	2	1.0	.0	1	1	1.0	.0	1	1	251	0	251	251	0				0								
50-59	1	2	1.0	.0	1	1	1.0	.0	1	1	45	0	45	45	0				0								
40-49	1	2	2.0	.0	2	2	1.0	.0	1	1	403	402	805	2	0				1.0	1.0	2	0	0				
30-39	0																										
20-29	12	18	2.0	.0	2	2	.9	.4	2	0	258	157	502	0	0				.3	.6	2	0	.1	.3	1	0	
10-19	8	12	2.0	.0	2	2	.8	.4	1	0	186	108	334	0	0				.4	.7	2	0	.5	.8	2	0	
0-9	26	40	2.0	.0	2	2	.4	.5	1	0	50	40	124	0	0				.9	1.6	9	0	.4	.7	2	0	
DOVF USAGE SUMMARY																											
UOW'S USING DOVF:			NO.:		55		%UOW'S:		85%																		
NO. DOVF CI'S USED:			AVG:		1.85		SDEV:		.35		MAX:		2		MIN:		1										
FREESPACE ANALYSIS - IOVF																											
%DOVF	#UOWS	%	#USE	-IOVF CI'S USED				-NO. OF FSE'S--				----SIZE OF FSE'S----			--PTR TO BASE--			--PTR TO DOVF--			--PTR TO IOVF--						
FREE	UOW		IOVF	AVG	S/D	MAX	MIN	AVG	S/D	MAX	MIN	AVG	S/D	MAX	MIN	AVG	S/D	MAX	MIN	AVG	S/D	MAX	MIN				
100	10	15																									
90-99	0																										
80-89	2	3	0																								
70-79	4	6	0																								
60-69	1	2	0																								
50-59	1	2	0																								
40-49	1	2	0																								
30-39	0																										
20-29	12	18	4	.5	.9	2	1	.8	.4	1	0	287	277	887	0	0			0				.2	.7	2	0	
10-19	8	12	7	1.6	1.0	3	1	.9	.3	1	0	169	155	538	0	0			0				.4	.8	2	0	
0-9	26	40	20	1.0	.7	3	1	.7	.4	1	0	555	319	945	0	0			0				.1	.6	2	0	
IOVF USAGE SUMMARY																											
UOW'S USING IOVF:			NO.:		31		%UOW'S:		48%																		
NO. IOVF CI'S USED:			AVG:		1.48		SDEV:		.67		MAX:		3		MIN:		1										
AVAILABLE IOVF CI'S:			NO.:		102		%CI'S:		68%																		

Figure 225. DOVF and IOVF sections of the Freespace Analysis report

The following descriptions pertain to the DOVF section.

% DOVF FREE

Shows a free space percentage range that was used for classifying UOWs for reporting purposes. DOVF part free space for a UOW includes all of the FSEs that are in used DOVF CIs plus all empty DOVF CIs.

#UOWS

Shows the total number of UOWs in the RAA BASE section of the database that contain free space within the bounds of the reporting range.

%UOW

Shows the percentage of UOWs in the RAA BASE section that contain free space within the bounds of the reporting range.

DOVF CI'S USED

Shows statistics about the number of DOVF CIs that were used by each UOW within the free space reporting range.

Note: All other DOVF part free space calculations do not include empty DOVF CIs. For example, the number and size of FSEs do not include the FSE in a totally empty CI.

DOVF USAGE SUMMARY

Shows the total number and percentage of UOWs that use DOVF, and statistics about the number of DOVF CIs used.

AVG

Shows the average number of DOVF CIs that were used by a UOW. This average excludes UOWs that do not use DOVF.

S/D

Shows the standard deviation for the average number of DOVF CIs.

MAX

Shows the maximum number of DOVF CIs that were used by any UOW within the free space reporting range.

MIN

Shows the minimum number of DOVF CIs that were used by any UOW within the reporting range. (Because UOWs that do not use DOVF are excluded from the calculations, the minimum cannot be less than 1.)

The following descriptions pertain to the IOVF section.

% DOVF FREE

Shows a free space percentage range that is used for classifying UOWs for reporting purposes. DOVF part free space for a UOW includes all of the FSEs that are in used DOVF CIs plus all empty DOVF CIs.

#UOWS

Shows the total number of UOWs in the RAA BASE section of the database that contain free space within the bounds of the reporting range.

%UOW

Shows the percentage of UOWs in the RAA BASE section that contain free space within the bounds of the reporting range.

#USE IOVF

Shows the number of UOWs within the DOVF part free space reporting range that also use one or more IOVF CIs.

Note: Reorganization can be beneficial for those UOWs that have a high percentage of DOVF free space but also use IOVF.

IOVF CI'S USED

Shows statistics about the number of IOVF CIs that were used by each UOW within the free space reporting range.

Note: All other IOVF part free space calculations are based on the actual number of IOVF CIs used.

IOVF USAGE SUMMARY

Shows the total number and percentage of UOWs that use IOVF, and statistics about the number of IOVF CIs that are used and available. (Bit maps are excluded from all calculations.)

AVG

Shows the average number of IOVF CIs that were used by a UOW. This average excludes UOWs that do not use IOVF.

S/D

Shows the standard deviation for the average number of IOVF CIs.

MAX

Shows the maximum number of IOVF CIs that were used by any UOW within the free space reporting range.

MIN

Shows the minimum number of IOVF CIs that were used by any UOW within the reporting range. (Because UOWs that do not use IOVF are excluded from the calculations, the minimum cannot be less than 1.)

AVAILABLE IOVF CI'S

Shows the number and percentage of unallocated and available IOVF CIs (that is, the number of totally empty IOVF CIs). A warning message is generated if the calculated value differs from the DMAC (DMACOCNT).

DB Record Profile Analysis report

The DB Record Profile Analysis report contains the analysis information of the database records in an area.

The DB Record Profile Analysis report provides the following information:

- Statistics about the number and length of the database records
- Dependent segment frequencies and length statistics

When SEGLSTAT=DETAILED is specified, the report contains prefix length and data flags. This additional information is helpful in database tuning.

Note: The ACTUAL LENGTH field does not include the length of prefix.

The DB Record Profile Analysis report provides information about the characteristics of the database records. Also, the information is used to determine the following factors:

- When to reorganize an area
- The optimal values for the ROOT and UOW parameters in the DBD

The following figure is an example of the DB Record Profile Analysis report.

```

IMS HPFP UTILITIES - FPA ANALYZE                                "DEDB AREA ANALYSIS REPORTS"                                PAGE: 4
5698-FPP V2R1                                                  DDNAME: DB23AR1    DSNAME : IMSVS.DB23.DB23AR1.ADS1    2019-11-22 14:10:50
DBDNAME: DEDBJN23    AREA NAME : DB23AR1    DDNAME: DB23AR1    DSNAME : IMSVS.DB23.DB23AR1.ADS1
                      AREA NUMBER: 1        DSTYPE: ADS

DB RECORD PROFILE ANALYSIS
NO. DB RECORDS: 168
AVG REC LENGTH: 1,234
STD. DEV. : 879
MAX REC LENGTH: 4,012
MIN REC LENGTH: 218
SEGNAME  SEG  SEG  TOT #OCCS  -----SEGMENT FREQUENCY-----  DEFINED LENGTH  -----ACTUAL LENGTH-----
          CD  LVL                                     AVG.  SDEV  MAX  MIN  MAX  MIN  AVG.  SDEV  MAX  MIN
!ROOTSEG1 1  1      168      N/A    N/A    N/A  N/A    900  40    243  145  900  134
!DD1      3  2      280      1.67  1.34   6    0     900  30    292  206  900  102
!DD11     4  3      261      1.55  2.79  18    0     900  20    226  105  900  154
!DD12     5  3         0
!DD2      6  2         0      0.36  0.61   2    0     100  *     52   0    52   52
  
```

Figure 226. DB Record Profile Analysis report with SEGLSTAT=COMPACT

The following figure shows another example of the report when SEGLSTAT=DETAILED and SEGLDIST=YES are specified.

```

IMS HPFP UTILITIES - FPA ANALYZE                                "DEDB AREA ANALYSIS REPORTS"                                PAGE: 4
5698-FPP V2R1                                                  DDNAME: DB23AR1    DSNAME : IMSVS.DB23.DB23AR1.ADS1    2019-11-22 14:02:19
DBDNAME: DEDBJN23    AREA NAME : DB23AR1    DDNAME: DB23AR1    DSNAME : IMSVS.DB23.DB23AR1.ADS1
                      AREA NUMBER: 1        DSTYPE: ADS

DB RECORD PROFILE ANALYSIS
NO. DB RECORDS: 12
AVG REC LENGTH: 961
STD. DEV. : 1
MAX REC LENGTH: 961
MIN REC LENGTH: 959
SEGNAME  SEG  SEG  TOT #OCCS  -----SEGMENT FREQUENCY-----  -----DEFINED LENGTH-----  -----ACTUAL DATA LENGTH-----
          CD  LVL                                     AVG.  SDEV  MAX  MIN  PREFIX  D-MAX  D-MIN  FLAGS  AVG.  SDEV  MAX  MIN
!ROOTSEG1 1  1      12      N/A    N/A    N/A  N/A    210  300  20  V      300  0    300  300
!DD11     2  2         0
!DD12     3  2      12      1.00  0.00   1    1     6  300  12  V      184  5    200  181
!DD21     4  2         0
!DD22     5  2      12      1.00  0.00   1    1     6  300  *  VC     47  6    50   29
!DD31     6  2         0
!DD32     7  2      12      1.00  0.00   1    1     6  300  *  F      202  0    202  202
  
```

! : ADDITIONAL INFORMATION IS AVAILABLE FOR THE SEGMENT ON "Segment Length Distribution Report"

Figure 227. DB Record Profile Analysis report with SEGLSTAT=DETAILED and SEGLDIST=YES

The following figure shows another example of the report when DBRLDIST=YES is specified. When DBRLDIST=YES is specified, an at sign (@) is printed to indicate that additional information about database records is available in the DB Record Length Distribution report.

```

IMS HPFP UTILITIES - FPA ANALYZE                                "DEDB AREA ANALYSIS REPORTS"                                PAGE: 4
5698-FPP V2R1                                                DDNAME: DB23AR11  DSNAME : IMSVS.DB23.DB23AR1.ADS1  2019-04-15 14:45:46
DBDNAME: DEDB000F  AREA NAME : DB23AR1  DSTYPE: ADS
AREA NUMBER: 1
DB RECORD PROFILE ANALYSIS (@)
NO. DB RECORDS: 62
AVG REC LENGTH: 1,340
STD. DEV. : 1,391
MAX REC LENGTH: 9,376
MIN REC LENGTH: 282
-----ACRONYMS INFORMATION FOR DEFINED LENGTH FLAGS-----
| F: FIXED LENGTH V: VARIABLE LENGTH C: COMPRESSED
|
SEGNAME  SEG  SEG  TOT #OCCS  -----SEGMENT FREQUENCY-----  -----DEFINED LENGTH-----  -----ACTUAL DATA LENGTH-----
CD  LVL  AVG.  SDEV  MAX  MIN  PREFIX  D-MAX  D-MIN  FLAGS  AVG.  SDEV  MAX  MIN
ROOTSEG1  1  1  62  N/A  N/A  N/A  N/A  26  900  100  V  534  311  900  100
DD1  3  2  64  1.03  1.20  6  0  42  800  80  V  310  242  800  80
DD11  4  3  44  0.71  1.25  4  0  6  128  *  F C  34  1  37  33
DD2  5  2  60  0.97  1.03  3  0  46  600  60  VC  40  2  46  37
DD21  6  3  102  1.65  2.29  10  0  22  400  40  VC  34  2  38  33
DD211  7  4  106  1.71  6.56  35  0  6  200  40  V  45  20  128  40
DD212  8  4  14  0.23  1.76  14  0  14  200  40  V  40  0  40  40
DD2121  9  5  126  2.03  15.87  126  0  6  200  40  V  50  0  50  50

```

(@): ADDITIONAL INFORMATION IS AVAILABLE FOR DB RECORD ON "DB Record Length Distribution Report"

Figure 228. B Record Profile Analysis report with DBRLDIST=YES

NO. DB RECORDS

Shows the number of database records in the area.

AVG REC LENGTH

Shows the total length of the average database record (that is, the root and all dependent segments). This length includes the lengths of the data portions and the segment prefixes.

STD. DEV.

Shows the standard deviation for the average length and provides a measure of variability for the length.

MAX REC LENGTH

Shows the total length of the longest database record.

MIN REC LENGTH

Shows the length of the shortest database record in the area.

SEGNAME

Shows the name of the segment as defined in the DBD.

An exclamation point (!) is displayed before the segment name if SEGLDIST=YES is specified and either or both of the following conditions are met:

- Segment is defined as variable length and its occurrence is not zero.
- Segment is compressed and its occurrence is not zero.

SEG CD and SEG LVL

Shows the segment code and the hierarchical level of the segment.

TOT #OCCS

Shows the total number of occurrences of the segment in an area.

SEG FREQUENCY

Shows statistics about the number of occurrences of each segment type.

AVG

Shows the average number of occurrences of a dependent segment within a database record.

SDEV

Shows the standard deviation for the average number of occurrences.

MAX

Shows the maximum number of occurrences of a dependent segment for a database record.

MIN

Shows the minimum number of occurrences of a dependent segment for a database record.

DEFINED LENGTH

Shows the prefix length, maximum and minimum segment lengths, and data flag that are defined in the DBD. Data flag of the segment is as follows:

- F** Fixed-length segment
- V** Variable-length segment
- C** Compressed segment

Note: ACBGEN ignores the minimum length that is specified in the DBD, and substitutes key length plus key offset with minimum value of 4.

ACTUAL LENGTH

Shows the actual average, maximum, and minimum segment lengths that were encountered in the database area.

Note: This length is the data portion only.

DB Record Placement Analysis report

The DB Record Placement Analysis report includes the number of database records that are contained in each of the seven placement categories.

The DB Record Placement Analysis report provides the following information:

- The number of database records that are contained in each of the seven placement categories
- Statistics about the number of DOVF and IOVF CIs that are used by a record

The DB Record Placement Analysis report provides information that can be used to determine:

- When to reorganize a DEDB area
- When to expand CI size (if it is less than 4 K)
- The optimal value for the UOW parameter in the DBD

The following figure is an example of the DB Record Placement Analysis report.

DB RECORD PLACEMENT ANALYSIS			---DOVF CI'S USED---				---IOVF CI'S USED---				
RECORDS IN:		NO. RECS	P/C	AVG.	SDEV	MAX	MIN	AVG.	SDEV	MAX	MIN
BASE ONLY		190,955	76.4								
BASE + DOVF		37,097	14.8	1.10	.30	2	1	1.00	.00	1	1
BASE + DOVF + IOVF		2	0.0	1.00	.00	1	1	1.00	.00	1	1
BASE + IOVF		1	0.0					1.00	.00	1	1
DOVF ONLY		21,935	8.8	1.17	.38	2	1	1.00	.00	1	1
DOVF + IOVF		1	0.0	1.00	.00	1	1	1.00	.00	1	1
IOVF ONLY		9	0.0					1.00	.00	1	1

Figure 229. DB Record Placement Analysis report

NO. RECS

Shows the total number of database records in the placement category.

P/C

Shows the percentage of the total number of database records in the placement category.

DOVF CI'S USED and IOVF CI'S USED

Show statistics about DOVF and IOVF CI usage.

AVG.

Shows the average number of DOVF or IOVF CIs that were used by records in the placement category.

SDEV

Shows the standard deviation for the average number of records in the placement category.

MAX

Shows the maximum number of DOVF or IOVF CIs that were used by any record in the placement category.

MIN

Shows the minimum number of DOVF or IOVF CIs that were used by any record in the placement category. (The minimum can never be less than 1.)

Recommendations:

- DOVF usage is preferable to IOVF usage.
- Less than 5 to 10% of the database records should use IOVF.
- Expand the DOVF part of the UOW to minimize IOVF usage.

Segment Placement Analysis report

The Segment Placement Analysis report provides an overview of where the segments reside in the database and can be used as introductory information for I/O analysis.

Note: SDEP information is displayed only if SDEP is defined. N/A is displayed when SDEPCHKLVL=NONE (FPA) or SDEP=NO (FPO) is specified.

The following figure is an example of the Segment Placement Analysis report.

```
IMS HPPF UTILITIES - FPA ANALYZE                                "DEDB AREA ANALYSIS REPORTS"                                PAGE: 5
5698-FPP V2R1                                                  DDNAME: DB23AR1    DSNAME : IMSVS.DB23.DB23AR1.ADS1    2019-11-22 14:10:50
DBDNAME: DEDBJN23    AREA NAME : DB23AR1    DDNAME: DB23AR1    DSNAME : IMSVS.DB23.DB23AR1.ADS1
AREA NUMBER: 1      DSTYPE: ADS

SEGMENT PLACEMENT ANALYSIS

-----IN RAA BASE-----      -----IN DOVF-----      -----IN IOVF-----
SEGNAME  SCD  LVL  TOT #OCCS  NO. OCCS  P/C      NO. OCCS  P/C      NO. OCCS  P/C
ROOTSEG1 1    1    168      132      78.6      28        16.7      8         4.8
SDEPSEG1 2    2     19       -         -         -         -         -         - (SDEP)
DD1       3    2    280      131      46.8      118       42.1      31        11.1
DD11      4    3    261      46       17.6      147       56.3      68        26.1
DD12      5    3     0        -         -         -         -         -         -
DD2       6    2    60       29       48.3      9         15.0      22        36.7
-----
TOTAL                    788      338      302      129
```

Figure 230. Segment Placement Analysis report

SEGNAME

Shows the name of the segment that is defined in the DBD.

SCD and LVL

Show the segment code and the hierarchical level of the segment, respectively.

TOT #OCCS

Shows the total number of occurrences of the segment in an area.

NO. OCCS

Shows the number of occurrences of a segment in a specific section of an area.

P/C

Shows the percentage of the total number of occurrences of a segment in a specific section of an area.

Segment I/O Analysis report

The Segment I/O Analysis report provides information for the average database record.

This report provides the following information:

- Statistics about the number of dependent segments in each section of the area
- Statistics about the number of physical I/O that is required to retrieve the dependent segments
- Statistics about the number of physical I/O that is required to retrieve the average database record
- Statistics about the number of physical I/O that is required to retrieve the average root segment

The Segment I/O Analysis report provides information for database performance tuning and for evaluating the I/O requirements for specific user transactions. It is also used to determine the optimal reorganization point and for validating the effects of UOW and ROOT value parameter changes.

The following figure is an example of the Segment I/O Analysis report.

SEGMENT I/O ANALYSIS - AVERAGE DB RECORD													
SEGNAME	SEG CD	SEG LVL	AVG. FREQ	---SEGMENT PLACEMENT STATISTICS---						---PHYSICAL I/O STATS---			
				AVG./SDEV	MAX/MIN	AVG./SDEV	MAX/MIN	AVG./SDEV	MAX/MIN	AVG.	SDEV	MAX	MIN
ROOTSEG1	1	1	1.00	0.91 0.28	N/A	0.09 0.28	N/A	0.00 0.01	N/A	1.10	0.34	4	1
DD102	3	2	1.00	0.91 0.28	1 0	0.09 0.28	1 0	0.00 0.01	1 0	0.00	0.02	1	0
DD202	4	2	3.00	2.59 0.88	3 0	0.41 0.88	3 0	0.00 0.02	3 0	0.15	0.36	1	0
DD302	5	2	2.00	1.53 0.85	2 0	0.47 0.85	2 0	0.00 0.01	2 0	0.03	0.17	1	0
DD303	6	3	2.00	1.53 0.85	2 0	0.47 0.85	2 0	0.00 0.01	2 0	0.00	0.02	1	0
** RECORD I/O **			AVG:	1.28	SDEV:	0.54	MAX:	5	MIN:	1			
*** ROOT I/O ***			AVG:	1.10	SDEV:	0.34	MAX:	4	MIN:	1			

Figure 231. Segment I/O Analysis report

SEGNAME

Shows the name of the segment that is defined in the DBD.

AVG. FREQ

Shows the average number of occurrences of a dependent segment within a database record.

SEGMENT PLACEMENT STATISTICS

Show statistics about the average, maximum, and minimum number of occurrences of a dependent segment within a section of an area.

PHYSICAL I/O STATS

Shows statistics about the average, maximum, and minimum number of incremental physical I/O that is required to retrieve the dependent segments in a hierarchical path.

For example, to retrieve all occurrences of segment code 4 (D11), which is at the third hierarchical level, the average number of physical I/O that is required is calculated as follows:

$$\text{average physical I/O} = 1.83 + .50 + .00$$

The maximum is six (which is maximum record I/O), and the minimum is one.

RECORD I/O

Shows statistics about the average, maximum, and minimum number of physical I/O that is required (that is, the number of CIs accessed) to retrieve an entire database record.

ROOT I/O

Shows statistics about the average, maximum, and minimum number of physical I/O that is required to retrieve a root segment.

All numeric fields have multiple formats, and the presentation format depends on the value of the numeric item.

Synonym Chain Analysis report

The Synonym Chain Analysis report provides detailed information about the frequency and length of synonym chains and statistics about root placement and physical I/O requirements.

The Synonym Chain Analysis report provides information for:

- Determining the effectiveness of a randomizing module. In theory, roots should be evenly distributed across all RAPs.
- Determining the optimal values for the ROOT and UOW parameters.
- Performance tuning the area.

The following figure is an example of the Synonym Chain Analysis report.

DBDNAME: DEDBJN21 AREA NAME : DB21AR0
AREA NUMBER: 1

ROOTS PER RAP DISTRIBUTION GRAPH

	1	2	3	4	5	6	7	8	9	10	% RAPS
#	0	1	2	3	4	5	6	7	8	9	10
R	0	1	2	3	4	5	6	7	8	9	10
O	0	1	2	3	4	5	6	7	8	9	10
T	0	1	2	3	4	5	6	7	8	9	10
S	0	1	2	3	4	5	6	7	8	9	10
/	0	1	2	3	4	5	6	7	8	9	10
R	0	1	2	3	4	5	6	7	8	9	10
A	0	1	2	3	4	5	6	7	8	9	10
P	0	1	2	3	4	5	6	7	8	9	10

SYNONYM CHAIN STATISTICS
 SYN CHAIN LONGEST: 10 RAP RBA: 1756A000
 SYN CHAIN MAX I/O: 4 RAP RBA: 00302000
 SYN CHAIN LENGTH: AVG: 2.79
 NO. ROOTS PER RAP: AVG: 2.15

SYNONYM CHAIN ANALYSIS

#ROOTS/ RAP	NO. RAPS	P/C	---ROOT PLACEMENT STATISTICS---				---PHYSICAL I/O STATS---				
			AVG./ SDEV	MAX/ MIN	AVG./ SDEV	MAX/ MIN	AVG.	SDEV	MAX	MIN	
0	23,565	16.8	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
1	41,915	29.9	1.00	1	0.00	N/A	0.00	1.00	0.00	1	1
2	37,420	26.7	2.00	2	0.00	0.00	0.00	1.00	0.00	1	1
3	22,117	15.8	3.00	3	0.00	0.00	0.00	1.00	0.00	1	1
4	9,911	7.1	3.00	3	1.00	1	0.00	2.00	0.00	2	2
5	3,617	2.6	3.00	3	0.01	0	0.01	2.29	0.46	3	2
6	1,100	0.8	3.00	3	2.00	2	0.00	2.89	0.31	3	2

Figure 232. Synonym Chain Analysis report

#ROOTS/RAP (vertical axis)

Shows the number of roots that were chained from a RAP (that is, the length of a synonym chain).

Note: The vertical axis increase in length as required, up to the number of synonym table entries that you specified. The default maximum is 40. All RAPs with synonym chains greater than the maximum that you specified (or default) are classified together.

Each graph line represents the percentage of RAPs that have the number of roots specified. Each number is the number of RAPs that have the specified number of roots in a synonym chain.

SYN CHAIN LONGEST and SYN CHAIN MAX I/O

Show the length of the longest synonym chain and the maximum number of physical I/O that is required for a synonym chain.

RAP RBA

Shows the RBA of the first CI that was encountered that met the condition.

SYN CHAIN LENGTH: AVG

Shows the average length of all synonym chains that have a length greater than 1 (that is, RAPs that have only one root are excluded from the calculation).

NO. ROOTS PER RAP: AVG

Shows the average number of roots per RAP excluding dead RAPs.

#ROOTS/RAP

Shows the number of roots that were chained from a RAP (that is, the length of a synonym chain). This table increases in length as required, up to the maximum entries that you specified (or default). All RAPs that have synonym chains of greater than the maximum that you specified (or default) are classified together.

NO. RAPS

Shows the number of RAPs (that is, CIs) that have synonym chains of the corresponding #Roots/RAP length.

P/C

Shows the percentage of RAPs that have synonym chains of a specified length.

ROOT PLACEMENT STATISTICS

Show statistics about the average, maximum, and minimum number of root segments that reside in the specified area section for a given synonym chain length.

PHYSICAL I/O STATS

Show statistics about the average, maximum, and minimum number of physical I/O that is required to retrieve the root segments from the specified area section for a given synonym chain length.

All numeric fields have multiple formats, and the presentation format depends on the value of the numeric item.

Largest Database Records report

The optional report for the largest database records analysis provides a description for each of the large database records, starting with the largest database record.

This report is generated by the following commands:

- ANALYZE with the LARGEREC_REPORT=YES keyword on the REPORT subcommand and the PTRCHKLVL=FULL keyword
- CHANGE with the LARGEREC_REPORT=YES keyword on the REPORT subcommand and the PTRCHKLVL=FULL keyword
- RELOAD with the LARGEREC_REPORT=YES keyword on the REPORT subcommand and the PTRCHKLVL=FULL keyword
- REORG with the LARGEREC_REPORT=YES keyword on the REPORT subcommand and the PTRCHKLVL=FULL keyword
- TUNE with the LARGEREC_REPORT=YES keyword on the REPORT subcommand

The following figure shows an example of the Largest Database Records report.

```
IMS HFPF UTILITIES - FPA ANALYZE                                "LARGEST DATABASE RECORDS REPORT"                                PAGE: 1
5698-FPP V2R1                                                    2019-11-22 15:32:33
DBDNAME: DEDBJN21  KEY LENGTH: 8  RECORD COUNT: 15
                   DDNAME: DB21AR0  DSNAME : HFPF.MDB21AR0.ADSFP
                   DSTYPE: ADS
```

RECORD SIZE	# OF SEG'S	AREANAME	AREA#	ROOT RBA	RAP#	RAP RBA	ROOT SEGMENT SEQUENCE FIELD (HEX)	(CHARACTER)
1,034	24	DB21AR0	1	00000408	0	00000400	10000000 00001F21	* * * * *
363	8	DB21AR0	1	00004282	0	00000400	10000000 00002F22	* * * * *
160	3	DB21AR0	1	00001208	6	00001200	10000011 00001F11	* * * * *
84	1	DB21AR0	1	00004408	0	00000400	10000000 00002F23	* * * * *
84	1	DB21AR0	1	0000445C	0	00000400	10000000 00002F24	* * * * *
84	1	DB21AR0	1	00004480	0	00000400	10000000 00002F25	* * * * *
84	1	DB21AR0	1	00004504	0	00000400	10000000 00002F26	* * * * *
84	1	DB21AR0	1	00004558	0	00000400	10000000 00002F27	* * * * *
84	1	DB21AR0	1	00004608	0	00000400	10000000 00002F28	* * * * *
84	1	DB21AR0	1	0000465C	0	00000400	10000000 00002F29	* * * * *
84	1	DB21AR0	1	000046B0	0	00000400	10000000 00002F30	* * * * *
84	1	DB21AR0	1	00004704	0	00000400	10000000 00002F31	* * * * *
84	1	DB21AR0	1	00004758	0	00000400	10000000 00002F32	* * * * *
84	1	DB21AR0	1	00001408	7	00001400	10000012 00001F11	* * * * *
84	1	DB21AR0	1	00001C08	10	00001C00	10000020 00001F11	* * * * *

Figure 233. Largest Database Records report

DBDNAME

The DBD name of the database.

KEY LENGTH

The length of the key in root segment.

RECORD COUNT

The number of reporting entries of the largest database records.

RECORD SIZE

The total length of the database record that includes prefix length of all segments in the record.

OF SEG'S

The total number of segments in the record.

AREANAME

The area name.

AREA#

The area number.

ROOT RBA

The value of root segment RBA.

RAP RBA

The value of RAP RBA.

ROOT SEGMENT SEQUENCE FIELD (HEX) and (CHARACTER)

The value of the key in root segment.

Consolidated Largest Database Records report

You can use this optional report to analyze the largest database records. The report provides information about each of the large database records, starting with the largest database record.

The difference between this report and the Largest Database Records report is that the Largest Database Records report provides information about the large database records for each area, whereas the Consolidated Largest Database Records report consolidates information about the large database records found in multiple areas into a single report.

This report is generated by the following commands when the second parameter of the LARGEREC_COUNT keyword, which specifies the number of reporting entries in the Consolidated Largest Database Record report, specifies a value that is equal to or larger than 1:

- ANALYZE with the LARGEREC_REPORT=YES keyword of the REPORT subcommand and the PTRCHKLVL=FULL keyword
- CHANGE with the LARGEREC_REPORT=YES keyword of the REPORT subcommand and the PTRCHKLVL=FULL keyword
- RELOAD with the LARGEREC_REPORT=YES keyword of the REPORT subcommand and the PTRCHKLVL=FULL keyword
- REORG with the LARGEREC_REPORT=YES keyword of the REPORT subcommand and the PTRCHKLVL=FULL keyword
- TUNE with the LARGEREC_REPORT=YES keyword of the REPORT subcommand

The following figure shows an example of the Consolidated Largest Database Records report.

DBDNAME: DEDB0003 KEY LENGTH: 10 RECORD COUNT: 30

TOTAL AREA#: 19

AREA: DB23AR1 DB23AR2 DB23AR3 DB23AR4 DB23AR5 DB230006 DB230007 DB230008 DB230009 DB230010 DB230011 DB230012 DB230013 DB230014
DB230015 DB230016 DB230017 DB230018 DB230019

RECORD SIZE	# OF SEG'S	AREANAME	AREA#	ROOT RBA	RAP#	RAP RBA	ROOT SEGMENT SEQUENCE FIELD (HEX)	(CHARACTER)
4,012	24	DB23AR1	1	00070808	9	00004400	F1F0F0F0 F0F0F0F1 F0F1	*1000000101 *
4,012	13	DB23AR1	1	00015408	48	00014800	F1F0F0F0 F0F0F0F4 F9F1	*1000000491 *
4,012	24	DB23AR4	4	00008744	9	00008000	F4F0F0F0 F0F0F0F1 F0F1	*4000000101 *
4,012	13	DB23AR4	4	0002FBA6	48	0002F800	F4F0F0F0 F0F0F0F4 F9F1	*4000000491 *
4,012	24	DB23AR5	5	00070808	9	00004400	F5F0F0F0 F0F0F0F1 F0F1	*5000000101 *
4,012	13	DB23AR5	5	00015408	48	00014800	F5F0F0F0 F0F0F0F4 F9F1	*5000000491 *
3,760	14	DB23AR1	1	00023AC4	84	00023800	F1F0F0F0 F0F0F0F8 F5F1	*1000000851 *
3,760	11	DB23AR1	1	000276C4	93	00027400	F1F0F0F0 F0F0F0F9 F4F1	*1000000941 *
3,760	14	DB23AR4	4	000532C4	84	00053000	F4F0F0F0 F0F0F0F8 F5F1	*4000000851 *
3,760	11	DB23AR4	4	0005CAC4	93	0005C800	F4F0F0F0 F0F0F0F9 F4F1	*4000000941 *
3,760	14	DB23AR5	5	00023AC4	84	00023800	F5F0F0F0 F0F0F0F8 F5F1	*5000000851 *
3,760	11	DB23AR5	5	000276C4	93	00027400	F5F0F0F0 F0F0F0F9 F4F1	*5000000941 *
3,510	19	DB23AR1	1	0000A008	21	00009400	F1F0F0F0 F0F0F0F2 F2F1	*1000000221 *
3,510	15	DB23AR1	1	0000DC08	30	0000D000	F1F0F0F0 F0F0F0F3 F1F1	*1000000311 *
3,510	19	DB23AR4	4	00015BA6	21	00015800	F4F0F0F0 F0F0F0F2 F2F1	*4000000221 *
3,510	15	DB23AR4	4	0001F3A6	30	0001F000	F4F0F0F0 F0F0F0F3 F1F1	*4000000311 *
3,510	19	DB23AR5	5	0000A008	21	00009400	F5F0F0F0 F0F0F0F2 F2F1	*5000000221 *
3,510	15	DB23AR5	5	0000DC08	30	0000D000	F5F0F0F0 F0F0F0F3 F1F1	*5000000311 *
3,265	5	DB23AR1	1	00028AC4	96	00028800	F1F0F0F0 F0F0F0F9 F7F1	*1000000971 *
3,265	5	DB23AR4	4	00060AC4	96	00060800	F4F0F0F0 F0F0F0F9 F7F1	*4000000971 *
3,265	5	DB23AR5	5	00028AC4	96	00028800	F5F0F0F0 F0F0F0F9 F7F1	*5000000971 *
3,259	12	DB23AR1	1	00036808	130	00036800	F1F0F0F0 F0F0F1F3 F1F1	*1000001311 *
3,259	9	DB23AR1	1	0003B808	142	0003B800	F1F0F0F0 F0F0F1F4 F3F1	*1000001431 *
3,259	7	DB23AR1	1	0003CC08	145	0003CC00	F1F0F0F0 F0F0F1F4 F6F1	*1000001461 *
3,259	12	DB23AR4	4	00083008	130	00083000	F4F0F0F0 F0F0F1F3 F1F1	*4000001311 *
3,259	9	DB23AR4	4	0008E008	142	0008E000	F4F0F0F0 F0F0F1F4 F3F1	*4000001431 *
3,259	7	DB23AR4	4	00092008	145	00092000	F4F0F0F0 F0F0F1F4 F6F1	*4000001461 *
3,259	12	DB23AR5	5	00036808	130	00036800	F5F0F0F0 F0F0F1F3 F1F1	*5000001311 *
3,259	9	DB23AR5	5	0003B808	142	0003B800	F5F0F0F0 F0F0F1F4 F3F1	*5000001431 *
3,259	7	DB23AR5	5	0003CC08	145	0003CC00	F5F0F0F0 F0F0F1F4 F6F1	*5000001461 *

Figure 234. Consolidated Largest Database Records report

DBDNAME

The DBD name of the database.

KEY LENGTH

The length of the key in the root segment.

RECORD COUNT

The number of reporting entries of the largest database records.

TOTAL AREA#

The number of areas that were analyzed.

AREA

The names of areas that were analyzed.

RECORD SIZE

The total length of the database record that includes prefix length of all segments in the record.

OF SEG'S

The total number of segments in the record.

AREANAME

The area name.

AREA#

The area number.

ROOT RBA

The value of root segment RBA.

RAP RBA

The value of RAP RBA.

ROOT SEGMENT SEQUENCE FIELD (HEX) and (CHARACTER)

The value of the key in the root segment.

UOW report

The optional report for UOW analysis provides detailed information about each UOW.

The UOW report can be used to:

- List detailed information about each UOW
- Identify which UOWs require reorganization
- Manage space utilization when the randomizer has been modified to group related data into contiguous UOWs
- List only those UOWs that violate the space utilization criteria that you specified

This report is generated by the following commands:

- ANALYZE with the REPORT subcommand and the PTRCHKLVL=FULL or the PTRCHKLVL=QUICK keyword
- CHANGE with the REPORT subcommand and the PTRCHKLVL=FULL or the PTRCHKLVL=QUICK keyword
- RELOAD with the REPORT subcommand and the PTRCHKLVL=FULL or the PTRCHKLVL=QUICK keyword
- REORG with the REPORT subcommand and the PTRCHKLVL=FULL or the PTRCHKLVL=QUICK keyword
- TUNE

The following figure is an example of the UOW report when PTRCHKLVL=FULL is specified or when the report is generated by the TUNE command.

```

IMS HFPF UTILITIES - FPA ANALYZE                                "UOW REPORT"
5698-FPP V2R1
DBDNAME: DE0BJN23      AREA NAME : DB23AR1      DDNAME: DB23AR1      DSNAME : IMSVS.DB23.DB23AR1.ADS1
                      AREA NUMBER: 1          DSTYPE: ADS
PAGE: 1
2019-11-22 14:10:50

```

UOW NO.	---NUMBER #RAPS	CI'S #DOVF	USED--- #IOVF	#RAPS OVFL	USE CI'S	---%RAP FS--- AVG MAX MIN	% DOVF PART FS	#DB RECORDS	----- RECORD AVERAGE	LENGTH MAXIMUM	STATS MINIMUM
0	0	0	0	0	0		100	0			
1	1	0	0	0	0	0 0 0	100	1	1,003	1,003	1,003
2	1	2	1	1	5 5 5	0	1	3,009	3,009	3,009	
3	1	2	2	1	5 5 5	0	1	4,012	4,012	4,012	
4	1	1	0	1	100 100 100	75	1	501	501	501	
5	1	0	0	0	0 0 0	100	1	1,002	1,002	1,002	
6	1	1	0	1	0 0 0	74	1	1,504	1,504	1,504	
7	1	2	1	1	13 13 13	0	1	3,510	3,510	3,510	
8	1	2	0	1	100 100 100	25	1	1,504	1,504	1,504	
9	1	2	1	1	47 47 47	1	1	3,008	3,008	3,008	
10	1	2	1	1	47 47 47	1	1	3,510	3,510	3,510	
11	3	2	0	3	100 100 100	25	3	501	926	289	
12	3	2	0	3	13 13 13	4	3	1,504	1,504	1,504	
13	3	2	1	3	13 13 13	4	3	1,838	2,507	1,504	
14	1	2	0	1	100 100 100	0	1	2,006	2,006	2,006	
15	1	2	0	1	0 0 0	0	1	3,009	3,009	3,009	
16	1	2	2	1	0 0 0	5	1	4,012	4,012	4,012	
17	3	2	0	3	100 100 100	0	3	668	754	501	
18	3	0	0	0	0 0 0	100	3	1,001	1,003	1,001	
19	3	2	0	3	22 25 16	16	3	1,337	1,422	1,170	
20	3	2	2	3	22 25 16	16	3	2,006	2,090	1,840	
21	1	0	0	0	75 75 75	100	1	250	250	250	
22	1	0	0	0	0 0 0	100	1	1,002	1,002	1,002	
23	1	1	0	1	0 0 0	87	1	1,253	1,253	1,253	
24	1	2	1	1	45 45 45	9	1	3,007	3,007	3,007	

Figure 235. UOW report

UOW NO.

The number of the unit of work.

#RAPS

The number of RAPS in the UOW that contain data.

#DOVF

The number of DOVF CIs that contain data in this UOW.

#IOVF

The number of IOVF CIs that are allocated to the UOW.

#RAPS USE OVFL CI'S

The number of RAP CIs that contain a pointer that references either DOVF or IOVF.

%RAP FS

Statistics about the average, maximum, and minimum percentage free space in used RAP CIs. Empty RAP CIs are excluded from the calculations.

% DOVF PART FS

The percentage of free space in the DOVF part of the UOW. DOVF part free space for a UOW includes all of the FSEs that are in used DOVF CIs plus all empty DOVF CIs.

#DB RECORDS

The number of database records in the UOW. When PTRCHKLVL=QUICK is specified, this field is not reported and is shown as '-'.

RECORD LENGTH STATS

Statistics about the average, maximum, and minimum record lengths. The record length includes both prefix and data lengths. When PTRCHKLVL=QUICK is specified, this field is not reported and is shown as '-'.

UOW Exception report

The UOW Exception report contains the analysis information of the UOW that satisfies the criteria specified by the HFPAPARM keywords.

Entries are written to this report when one or more of the reporting criteria are met. These reporting criteria are controlled by the keywords that you specify on the HFPAPARM DD statement.

- > *nn*% free space in a RAP CI and overflow CIs are used. Use the RBASEFS keyword to specify this parameter.
- > *nn*% free space in DOVF and the UOW uses IOVF CIs. Use the RDOVFFS keyword to specify this parameter.
- < *nn*% free space in the DOVF part of a UOW. Use the UDOVFFS keyword to specify this parameter.

The UOW Exception report can be used to:

- List detailed information about each UOW
- Identify which UOWs require reorganization
- Manage space utilization when the randomizer has been modified to group related data into contiguous UOWs
- List only those UOWs that violate the user-specified space utilization criteria

This report is generated by the following commands and by the HFPAPARM reporting criteria:

- ANALYZE with the REPORT subcommand and the PTRCHKLVL=FULL or the PTRCHKLVL=QUICK keyword
- CHANGE with the REPORT subcommand and the PTRCHKLVL=FULL or the PTRCHKLVL=QUICK keyword
- RELOAD with the REPORT subcommand and the PTRCHKLVL=FULL or the PTRCHKLVL=QUICK keyword
- REORG with the REPORT subcommand and the PTRCHKLVL=FULL or the PTRCHKLVL=QUICK keyword
- TUNE

For information about the HFPAPARM DD statement, see the description about the DD statement in the following topics:

- [“DD statements for the Analyze function” on page 203](#)
- [“DD statements for the Change function” on page 134](#)
- [“DD statements for the Reload function” on page 94](#)
- [“DD statements for the Reorg function” on page 170](#)

The following figure is an example of the UOW Exception report when PTRCHKLVL=FULL and the reporting criteria are specified on the HFPAPARM DD statement.

DBDNAME: DEDBJN21 AREA NAME : DB21AR0
AREA NUMBER: 1

UOW NO.	---NUMBER CI'S USED---			#RAPS USE		---%RAP FS---			% DOVF	#DB RECORDS	----- RECORD LENGTH STATS -----		
	#RAPS	#DOVF	#IOVF	OVFL	CI'S	AVG	MAX	MIN	PART FS		AVERAGE	MAXIMUM	MINIMUM
4	12	2	0	2	41	60	1	80	21	1,590	1,590	1,590	
7	12	5	0	4	37	60	1	29	30	1,590	1,590	1,590	
8	13	3	0	4	30	60	1	61	29	1,590	1,590	1,590	
10	13	3	0	5	31	60	1	58	29	1,590	1,590	1,590	

Figure 236. UOW Exception report

UOW NO.

The number of the unit of work.

#RAPS

The number of RAPS in the UOW that contain data (that is, the remaining RAPS are "dead RAPS").

#DOVF

The number of DOVF CIs that contain data in this UOW.

#IOVF

The number of IOVF CIs that are allocated to the UOW.

#RAPS USE OVFL CI'S

The number of RAP CIs that contain a pointer that refers to either DOVF or IOVF.

%RAP FS

Statistics about the average, maximum, and minimum percentage free space in used RAP CIs. Empty RAP CIs are excluded from the calculations.

% DOVF PART FS

The percentage of free space in the DOVF part of the UOW. DOVF part free space for a UOW includes all of the FSEs that are in used DOVF CIs plus all empty DOVF CIs.

#DB RECORDS

The number of database records in the UOW. When PTRCHKLVL=QUICK is specified, this field is not reported and is shown as '-'.

RECORD LENGTH STATS

Statistics about the average, maximum, and minimum record lengths. The record length includes both prefix and data lengths. When PTRCHKLVL=QUICK is specified, this field is not reported and is shown as '-'.

Image Copy Output report

The Image Copy Output report contains the processing information of the image copy output process.

This report is generated by the following commands specified with the ICOUT=YES keyword:

- ANALYZE
- CHANGE
- UNLOAD
- RELOAD
- REORG

This report is not printed if you make one of the following specifications:

- GLOBAL SCAN=SITE_CREATE
- GLOBAL SCAN=SITE_REPORT

The following figure shows an example of the Image Copy Output report.

PROCESSING INFORMATION
- NOTIFY.IC COMMAND NOT ISSUED, BECAUSE OF COMMAND PERFORMED BY DBRC=NO.

NOTIFY.IC COMMAND LIST
NOTIFY.IC DBD(DEDBJN21) AREA(DB21AR0) USID(0000000000)-
RUNTIME('2019.326 19:51:13.6 +09:00') -
BATCH -
ICDSN(HFPF.MDB21AR0.IC001) -
FILESEQ(0001) UNIT(3390) -
VOLLIST(HFPF01) -
RECDCT(0000002475)

Figure 237. Image Copy Output report

Extracted report

The Extracted report provides a count of the number of segments (by segment name and segment code) that were extracted from each area of the database.

A database total is also provided. If segment count for an area is zero, it means that the extracted area is empty.

This report is generated by the EXTRACT command.

If you make one of the following specifications, this report is not printed:

- GLOBAL SCAN=YES
- GLOBAL SCAN=SITE_CREATE
- GLOBAL SCAN=SITE_REPORT

The following figure shows an example of the Extracted report.

SEGMENTS EXTRACTED FROM DATABASE: DEDBJN21

- AREA NO: 1 AREANAME: DB21AR0 DDNAME: DB21AR0 DSNAME: HPFP.MDB21AR0.ADSFP
(JCL DD ALLOCATION)
STARTED DATE: 2019-11-22 TIME: 11:25:42 ENDED DATE: 2019-11-22 TIME: 11:25:42

IOVF

- TOTAL CI'S : 9
- UNUSED CI'S : 6
- UNUSED CI RATIO : 67 (%)

SDEP

- TOTAL CI'S : 77
- UNUSED CI'S : 52
- UNUSED CI RATIO : 68 (%)
- RANGE : X'00005800' TO X'00018C00'
- LOGICAL BEGIN : CYCLE COUNT X'00000001' RBA X'00005800'
- LOGICAL END : CYCLE COUNT X'00000001' RBA X'0000B800'
- LOGICAL BEGIN TS : X'D194CC50C010E341'

SUMMARY OF KEY SEQUENCE ERRORS

- NUMBER OF RELATED DB RECORDS : 0
- NUMBER OF THE SEGMENTS DETECTED AS KEY SEQUENCE ERROR : 0
- ERROR SEGMENTS SUM TOTAL INCLUDING CHILD SEGMENTS : 0

SEGMENT CODE	SEGMENT NAME	SEGMENT HIERARCHY	SEGMENTS READ	SEGMENTS EXTRACTED	SKIPPED BY KEY ERROR	SKIPPED BY CRITERIA (SEL)	SKIPPED BY CRITERIA (OUT)
1	ROOTSEG1	1	20	0	0	0	20
2	SDEPSEG1	2	18	0	0	18	0
3	DD101	2	9	0	0	0	9
4	DD201	3	9	8	0	1	0
5	DD301A	4	12	0	0	6	6
6	DD401	5	5	0	0	5	0
7	DD501	6	1	0	0	1	0
8	DD601	7	1	0	0	1	0
9	DD701	8	1	0	0	1	0
10	DD801	9	1	0	0	1	0
11	DD901	10	1	0	0	1	0
12	DDA01	11	1	0	0	1	0
13	ddb01	12	1	0	0	1	0
14	DDC01	13	1	0	0	1	0
15	DDD01	14	1	0	0	1	0
16	DDE01	15	1	0	0	1	0
17	DD301B	4	4	0	0	4	0
18	DD102	2	3	0	0	3	0
19	DD202	3	0	0	0	0	0
20	DD302	4	0	0	0	0	0
21	DD103	2	2	0	0	2	0
22	DD104	2	1	0	0	1	0

** TOTAL ** 93 8 0 50 35

- DATABASE: DEDBJN21 TOTAL

SEGMENT CODE	SEGMENT NAME	SEGMENT HIERARCHY	SEGMENTS READ	SEGMENTS EXTRACTED	SKIPPED BY KEY ERROR	SKIPPED BY CRITERIA (SEL)	SKIPPED BY CRITERIA (OUT)
1	ROOTSEG1	1	20	0	0	0	20
2	SDEPSEG1	2	18	0	0	18	0
3	DD101	2	9	0	0	0	9
4	DD201	3	9	8	0	1	0
5	DD301A	4	12	0	0	6	6
6	DD401	5	5	0	0	5	0
7	DD501	6	1	0	0	1	0
8	DD601	7	1	0	0	1	0
9	DD701	8	1	0	0	1	0
10	DD801	9	1	0	0	1	0
11	DD901	10	1	0	0	1	0
12	DDA01	11	1	0	0	1	0
13	ddb01	12	1	0	0	1	0
14	DDC01	13	1	0	0	1	0
15	DDD01	14	1	0	0	1	0
16	DDE01	15	1	0	0	1	0
17	DD301B	4	4	0	0	4	0
18	DD102	2	3	0	0	3	0
19	DD202	3	0	0	0	0	0
20	DD302	4	0	0	0	0	0
21	DD103	2	2	0	0	2	0
22	DD104	2	1	0	0	1	0

** TOTAL ** 93 8 0 50 35

Figure 238. Extracted report

AREA NO
The area number.

AREANAME
The area name.

DDNAME

The DDNAME with the following information.

(DYNAMIC ALLOCATION):

DDNAME is allocated dynamically.

(JCL DD ALLOCATION):

DDNAME is specified in the JCL statement.

DSNAME

The DSNAME of the area name DDNAME.

INPUT DSNAME WAS RENAMED TO DSNAME

The input area data set is renamed for the processing of the CHANGE command by specifying the IRENAME subcommand.

TOTAL CI'S

The total number of CI in each IOVF and SDEP part.

UNUSED CI'S

The number of unused CI in each IOVF and SDEP part.

UNUSED CI RATIO

The ratio of unused CI in each IOVF and SDEP part.

RANGE

The SDEP logical begin and logical end RBAs.

LOGICAL BEGIN

The cycle count and RBA of SDEP logical begin.

LOGICAL END

The cycle count and RBA of SDEP logical end.

LOGICAL BEGIN TS

The timestamp of the SDEP logical begin.

SUMMARY OF KEY SEQUENCE ERRORS

The summary of number of records when a sequence error in a key sequence field is detected.

NUMBER OF RELATED DB RECORDS

The number of the DB records that have segments that were detected as key sequence errors in extracting processes.

NUMBER OF THE SEGMENTS DETECTED AS KEY SEQUENCE ERROR

The number of the segments that were detected as key sequence errors in extracting processes.

ERROR SEGMENTS SUM TOTAL INCLUDING CHILD SEGMENTS

The number of the segments that were detected as key sequence errors in extracting processes and its child segments.

SEGMENT CODE

The segment code.

SEGMENT NAME

The segment name.

SEGMENT HIERARCHY

The segment hierarchy.

SEGMENTS READ

The number of segments that are read.

SEGMENTS EXTRACTED

The number of segments that are extracted.

SKIPPED BY KEY ERROR

The number of segments that are skipped by the key sequence error.

SKIPPED BY CRITERIA(SEL)

The number of segments that are skipped by the following reasons:

- The segments did not meet the criteria that are specified in the SELECT subcommand.
- This segment type was not treated as the input segment.

SKIPPED BY CRITERIA(OUT)

The number of segments that are skipped by the following reasons:

- The segments did not meet the criteria that are specified in the OUTPUT subcommand.
- This segment type was not treated as the output segment.

DATABASE TOTAL

The total number of segments that are extracted for the database.

DMACPRT report

The DMACPRT report shows the statistics of the common area information, the DMAC information, and the DMAC dump.

The DMACPRT report includes the following sections:

- Area information
- DMAC mapping
- DMAC dump

This report is generated by the DMACPRT command.

Note: The header of each report is "DMACPRT report".

This report is not printed if you make one of the following specifications:

- GLOBAL SCAN=SITE_CREATE
- GLOBAL SCAN=SITE_REPORT

Area information

This section shows the statistics of the common area information. The area information for the second CI and DMCB are reported and the differences between them are shown with '*'.

The following figure is an example of an area information.

DMAC MAPPING FROM DATABASE: DEDBJN21

- AREA NO: 1 AREANAME: DB21AR0 DDNAME: DB22AR0 DSNAME: HFPF.MDB21AR0.ADSFP
(JCL DD ALLOCATION)

- AREA INFORMATION

	2ND CI	DMCB	
			*: DIFFERENCE DETECTED
- IMS VERSION	141	141	
- DBDNAME	DEDBJN21	DEDBJN21	
- AREA NAME	DB21AR0	DB21AR0	
- CI SIZE	1024	1024	
- ROOT	(5,1)	(5,1)	
- UOW	(5,1)	(5,1)	
- BASE			
- NUMBER OF CI'S			
- TOTAL CI'S	16	16	
- DOVF			
- NUMBER OF CI'S			
- TOTAL CI'S	4	4	
- IOVF			
- NUMBER OF CI'S			
- TOTAL CI'S	4	4	
- SDEP			
- NUMBER OF CI'S			
- TOTAL CI'S	463	N/A	
- START BLOCK#	32	32	
- RANGE			
- FROM	X'00008000'	X'00008000'	
- TO	X'0007BC00'	N/A	
- LOGICAL BEGIN			
- CYCLE COUNT	X'00000001'	N/A	
- RBA	X'00008000'	N/A	
- LOGICAL END			
- CYCLE COUNT	X'00000001'	N/A	
- RBA	X'00009400'	N/A	
- LOGICAL BEGIN TS	X'D194CC50C010E341'	N/A	

Figure 239. Example of an area information

DMAC mapping

This section shows the DMAC information that is obtained by using the IMS mapping macro. The DMACPRT report shows both the global section and the local section in DMAC. In this report:

- OFFSET shows the offset from the second CI top address.
- LABEL shows the name of the IMS DBFDMAC macro.
- VALUE shows the value of the data.
- COMMENT shows what the data, which is quoted from IMS DBFDMAC macro, is.

The following figure is an example of the global section of DMAC mapping.

- AREA NO: 1 AREANAME: DB21AR0 DDNAME: DB22AR0 DSNAME: HPFP.MDB21AR0.ADSFP

- DMAC INFORMATION : DMAC GLOBAL SECTION

OFFSET	LABEL	VALUE	COMMENT
000000	DMACVNO	141	VERSION NUMBER
000003	DMACLKID	8	AREA LOCK ID
000004	DMACDBNM	DEDBJN22	* DMACLKCD C'8' : AREA LOCK ID FOR AREA
00000C	DMACNAME	DB22AR0	DATA BASE NAME
000014	DMACDBRC	X'00'	AREANAME FOR AREA
			DBRC USAGE INDICATOR
			DMACREGR X'80' : AREA REGISTERED WITH DBRC
			DMACWDUI X'40' : DBRC USAGE INDICATOR CHANGED
			DMACSDWRA.. X'08' : ALL LOCAL 2ND TIME.
			(DMACSDWRAPLCL2)
			DMACSDWRA.. X'04' : ALL LOCAL 1ST TIME.
			(DMACSDWRAPLCL1)
			DMACSDW2 X'02' : COMPAT WRAP 2ND TIME.
			DMACSDW1 X'01' : COMPAT WRAP 1ST TIME.
000015	DMACDUI	X'10306F0836128F'	DBRC USAGE INDICATOR
00001C	DMACYSI		IMS SUBSYSTEM ID
000024	DMACOCNT	2 X'00000002'	COUNT OF UNUSED INDEPENDENT OVERFLOW CI'S
000028	DMACCUSN	X'001C'	DMAC CUSN
00002A	DMACFLG4	X'88'	FLAG BITS...
			* DMACF4SH X'80' : AREA HAS A SHARABILITY
			DMACF4CC X'40' : DEDB CONVERSION COMPLETED
			DMACF4FC X'20' : FORCE CLOSE DUE TO IRLM FAIL
			DMACF4LR X'10' : LOCK REJECT AT COMMIT-ROLB AND RETRY THE LOCK AT CALL
			* DMACF4SD X'08' : AREA HAS SDEP SEGMENT
			DMACF4CS X'04' : AREA HAS A CACHED DEVICE WHICH RECEIVED AN I/O ERROR
			DMACCHG X'02' : OLC AREA LEVEL CHANGE
			DMACF4RC X'01' : MADS IOT RECOVERY STARTED

Figure 240. Example of a DMAC mapping: DMAC global section

The following figure is an example of the local section of DMAC mapping.

- DMAC INFORMATION : DMAC LOCAL SECTION

OFFSET	LABEL	VALUE	COMMENT
00009C	DMACFLG1	X'05'	FLAG BITS...
			DMACF1AF X'80' : SEQUENTIAL DEP. PART FULL
			DMACF1EP X'40' : I/O ERROR
			DMACF1ST X'20' : AREA STOP REQUEST
			DMACF1RE X'10' : AREA RESTART REQUEST
			DMACF1ER X'08' : ERROR RECOVERY NEEDED.
			* DMACF1CP X'04' : WAIT FOR W-M-C
			DMACF1BK X'02' : TEMPORARY BIT FOR BACKOUT.
			* DMACF1OP X'01' : AREA IS OPEN
00009D	DMACFLG3	X'80'	FLAG BITS...
			* DMACF3SP X'80' : AREA CLOSE IN PROCESS
			DMACNTFY X'40' : NOTIFY HAS BEEN SENT TO DBRC
			DMACF3ER X'20' : DMAC LOG EXIST DURING ERE
			DMACF3RN X'10' : AREA RECOVERY NEEDED REQUEST
			DMACF3CR X'08' : CREATE IS IN PROCESS SDEP
			DMACARCB X'04' : AVA. PHASE REACH BIT
			DMACANYC X'02' : AVA. PHASE NOT YET COMPLETED
			DMACF3AS X'01' : ADS STOP IN PROCESS
00009E	DMACFLG6	X'05'	FLAG BYTE 6
			DMACF6FU X'80' : FORCE DMAC SYNC 2ND CI UPD.
			DMACF6RV X'40' : CLOSE BY IRLM REVERIFICATION
			DMACF60P X'20' : AOCE REOPEN PENDING
			DMACF6WT X'10' : MFLG WAITING FOR WTCT
			DMACF6IP X'08' : DMAC INTERNAL STOP REQUEST
			* DMACF6CS X'04' : DMAC CTL BLK SYNC REQUEST
			DMACF6RQ X'02' : LB RECOVERY REQUEST
			* DMACF6UP X'01' : ACCESS INTENT IS UP/EX

Figure 241. Example of a DMAC mapping: DMAC local section

DMAC dump

The following figure shows the DMAC dump that has the DMACLEN length. The length depends on the version of the IMS system. When the report is for the available area data sets, EQE INFORMATION, CUSN, and RBA are also reported.

```

- AREA NO:      1  AREANAME: DB21AR0  DDNAME: DB22AR0  DSNAME: HPFP.MDB21AR0.ADSFP
- 2ND CI DUMP
- CI-RBA: 1024  (X'0400')
000000 F1F4F1F8 C4C5C4C2 D1D5F2F1 C4C2F2F1 C1D9F040 0010306F 0836128F 00000000 *1418DEDBJN21DB21AR0 *****
000020 00000000 00000002 001C8800 000001EF 00000000 00000001 00009400 00000001 * *****h*****
000040 00008000 00000000 00000001 00008000 00000001 00009000 00009000 00000001 * *****
000060 00000000 00000000 00000020 00000000 01C10004 00010005 00780002 00000400 * *****A*****
000080 00000016 0000001B 00000001 00000000 00000000 00018000 00000000 05800503 * *****;*****
0000A0 03A08002 00000001 00000000 00000002 00000000 00000000 40000000 D194CC50 * *****Wam*
0000C0 C010E341 00000000 00000000 00000000 00000000 00000000 00000000 40404040 * & *****
0000E0 0007BC00 000003C7 00000000 00000000 00000000 00008000 00000000 00000000 * *****G*****
000100 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
000120 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
000140 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
000160 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
000180 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
0001A0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
0001C0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
0001E0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
000200 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
000220 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
000240 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
000260 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
000280 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
0002A0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
0002C0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
0002E0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
000300 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
000320 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
000340 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
000360 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
000380 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
0003A0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
0003C0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 * *****
0003E0 00000000 00000000 00000000 00000000 00000000 1C000004 000003F9 03F90000 * *****g*****

- EQE INFORMATION (START RBA: X'03C7')
- EQE COUNTER : 0
0003C0 00 00000000 00000000 00000000 00000000 00000000 00000000 * *****
0003E0 00000000 00000000 00000000 00000000 000000 00000000 00000000 * *****

- CI SUFFIX INFORMATION
- CUSN : X'001C'
- RBA : X'00000400'
0003E0 00 1C000004 00

```

Figure 242. Example of a DMAC dump

Reorganization Processing Summary report

The Reorganization Processing Summary report shows the information of the area definition, the reorganization processing options, the result of UOW evaluation, and the result of reorganization.

This report is printed regardless of what is specified for the RFSSTAT parameter of the REORG command.

```

IMS HPFP UTILITIES - FPA REORG                               "Reorganization Processing Summary"
5698-FPP V2R1                                              PAGE: 1
DATABASE: DEDBJN23                                         2019-11-22 15:06:27
- AREA NO: 1 AREANAME: DB23AR0 DDNAME: DB23AR0 DSNAME: HPFP.MDB23AR0.ADSFP
  (JCL DD ALLOCATION)
  STARTED DATE: 2019-11-22 TIME: 15:06:27 ENDED DATE: 2019-11-22 TIME: 15:10:38
Area Information
Database name..... DEDBJN23
Area name..... DB23AR0
UOW parameter of DBD..... (20,10)
ROOT parameter of DBD..... (32767,1)
CI size..... 1024
Number of UOWs in Root Addressable Part..... 32766
Number of RAPs in a UOW..... 10
Number of CIs in DOVF of a UOW..... 10
Number of CIs in IOVF..... 20
Number of IOVF Space Map CIs..... 1
Processing Options
RBASEFS..... N/A
RDOVFFS..... N/A
Free Space Scan (RFSSCAN)
  before reorganization..... YES
  after reorganization..... YES
Reorganization Limit (REORGLIM)..... 75%
Number of failures allowed (MAXFAILS)..... 5
Result of UOW Evaluation
Number of UOWs that match RBASEFS..... 0
Number of UOWs that match RDOVFFS..... 0
Number of UOWs that match both RBASEFS and RDOVFFS.... 0
Number of UOWs that use at least one IOVF CI..... 18
Number of IOVF CIs used before Reorganization..... 18 ( 94% of total)
Result of Reorganization
Number of UOWs successfully reorganized..... 16383
Number of UOWs skipped..... 8191
Number of UOWs failed in reorganization..... 0
Number of IOVF CIs used after Reorganization..... 18 ( 94% of total)
Number of IOVF CIs reclaimed..... 0 ( 0% of total)

```

Figure 243. Reorganization Processing Summary report

UOW Space Statistics report

The UOW Space Statistics report contains the statistics about the free space in each UOW.

The UOW Space Statistics report is produced when RFSSTAT=YES is effective. The statistics are reported only for the UOWs that are selected for reorganization unless REORGLIM=0 or REORGLIM=0% is specified. If REORGLIM=0 or REORGLIM=0% is specified, the statistics for all UOWs are reported.

Figure 244 on page 617 shows the UOW Space Statistics report when RFSSCAN=(YES,YES) is specified.

The left-most column with the label 'UOW No.' shows the UOW number. The first row of the entry for each UOW shows the statistics before the reorganization; the second row shows the statistics after the reorganization.

The column with the label 'Number of CIs' is a collection of sub-columns, each of which show a number of CIs that satisfy a certain condition. The left three sub-columns show the number of CIs used in the RAA Base section of the UOW, in the DOVF section of the UOW, and in the IOVF part. The next sub-column with label 'OVERFLW' shows the number of RAP CIs that uses at least one overflow CI in the DOVF section or in the IOVF part. The sub-column with the label of 'RBASEFS' shows the number of RAP CIs that satisfy the condition specified by the RBASEFS parameter.

The next column with label 'PCT CI Free Space' shows the free space percentages of RAP CIs and the DOVF section. The sub-column named 'RAP CIs' consists of sub-columns 'OVERFLW' and 'All CIs'. The sub-column named 'OVERFLW' shows the percentage of the average, maximum, and minimum free space of RAP CIs that overflowed into DOVF or IOVF. The sub-column named 'All CIs' shows the percentage of the average, maximum, and minimum free space of all RAP CIs. Every RAP CI is included in the calculation of these values whether it is overflowed into DOVF or IOVF. The sub-column named 'DOVF CIs' shows the free space percentage, which is the average value of all DOVF CIs.

The next column 'Reorg Result Flags' shows the result of reorganization of each of the selected UOWs.

blank

The reorganization for the UOW completed successfully.

E

The UOW is empty.

F

The reorganization for the UOW failed.

S

The reorganization for the UOW skipped.

```

IMS HPFP UTILITIES - FPA REORG                                "UOW Space Statistics"
5698-FPP V2R1                                               PAGE: 2
                                                             2019-11-22 15:06:27
UOW Space Statistics before and after Reorganization:
Upper row for each UOW shows the statistics before reorganization
Lower row for each UOW shows the statistics after reorganization
<----- Number of CIs -----> <--- PCT CI Free Space ---->
<----- Used CIs -----> <-- RAP CIs --> <----- RAP CIs -----> DOVF Reorg
<----- IOVF OVERFLW RBASEFS <-OVERFLW-> <-All CIs-> CIs Result
UOW No.      RAP      DOVF      IOVF OVERFLW RBASEFS Avg Max Min Avg Max Min Avg CIs  Flags
-----
1001         10       10        1      10      0      23  23  23  23  23  23  10
           10       10        1      10      0      17  67  11  17  67  11  12
-----
2001         10       10        1      10      0      23  23  23  23  23  23  10
           10       10        1      10      0      17  67  11  17  67  11  12
-----
3001         10       10        1      10      0      23  23  23  23  23  23  10
           10       10        1      10      0      17  67  11  17  67  11  12
-----
3           10        6        0      10      0      23  23  23  23  23  23  44
           10        5        0      10      0      11  11  11  11  11  11  55
-----
5           10        6        0      10      0      23  23  23  23  23  23  44
           10        5        0      10      0      11  11  11  11  11  11  55
-----
7           10        6        0      10      0      23  23  23  23  23  23  44
           10        5        0      10      0      11  11  11  11  11  11  55
-----
9           10        6        0      10      0      23  23  23  23  23  23  44
           10        5        0      10      0      11  11  11  11  11  11  55
-----
2           10        0        0        0      0      -   -   -   67  67  67  100  S
           10        0        0        0      0      -   -   -   67  67  67  100
-----
4           10        0        0        0      0      -   -   -   67  67  67  100  S
           10        0        0        0      0      -   -   -   67  67  67  100
  
```

Figure 244. UOW Space Statistics report when RFSSCAN=(YES,YES)

The following figure shows the UOW Space Statistics report when RFSSCAN=(YES,NO) is specified.

```

IMS HPFP UTILITIES - FPA REORG                                "UOW Space Statistics"
5698-FPP V2R1                                               PAGE: 2
                                                             2019-11-22 15:06:27
UOW Space Statistics before Reorganization:
<----- Number of CIs -----> <--- PCT CI Free Space ---->
<----- Used CIs -----> <-- RAP CIs --> <----- RAP CIs -----> DOVF Reorg
<----- IOVF OVERFLW RBASEFS <-OVERFLW-> <-All CIs-> CIs Result
UOW No.      RAP      DOVF      IOVF OVERFLW RBASEFS Avg Max Min Avg Max Min Avg CIs  Flags
-----
1001         10       10        1      10      0      23  23  23  23  23  23  10
2001         10       10        1      10      0      23  23  23  23  23  23  10
3001         10       10        1      10      0      23  23  23  23  23  23  10
3           10        6        0      10      0      23  23  23  23  23  23  44
5           10        6        0      10      0      23  23  23  23  23  23  44
7           10        6        0      10      0      23  23  23  23  23  23  44
9           10        6        0      10      0      23  23  23  23  23  23  44
2           10        0        0        0      0      -   -   -   67  67  67  100  S
4           10        0        0        0      0      -   -   -   67  67  67  100  S
  
```

Figure 245. UOW Space Statistics report when RFSSCAN=(YES,NO)

Secondary Index Definition report

The Secondary Index Definition report shows information about secondary indexes that are defined.

This report is generated by the following commands:

- ANALYZE
- INDEXBLD

The following figure is an example of the Secondary Index Definition report.

```

IMS HFPF UTILITIES - FPA INDEXBLD                                "Secondary Index Definition Report"
5698-FPP V2R1                                                    PAGE: 1
- PRIMARY DEDB NAME      : DEDBJN21                             2019-11-22 16:52:55
- NUMBER OF SEC. INDEXES : 2

DEFINITION OF SEC. INDEX (XDFLD) : XNAME1 (TARGET SEG.: ROOTSEG1, SOURCE SEG.: ROOTSEG1)

ATTRIBUTES:
- DB ORGANIZATION      : HISAM
- POINTER SEG. NAME    : ROOTSEGX
- NULLVAL              : X'FF'
- MULTISEG            : NO
- INDEX MAINTENANCE EXIT ROUTINE : N/A
- PARTITION SELECTION EXIT ROUTINE : N/A
- PARTITION SELECTION OPTION : N/A

SEC. INDEX DBD/DDNAMES:
-----
DBDNAME  DDNAME  OVFL  DBDNAME  DDNAME  OVFL  DBDNAME  DDNAME  OVFL
-----
DEDBGS21 GS21KSDS GS21ESDS

RECORD LAYOUT:
-----
OFFSET  LENGTH  FIELD
-----
0        4     DUP. KEY POINTER
4        1     DELETE BYTE
5        10    SEARCH
15       5     SUBSEQ
20       0     DDATA
20       12   SYMBOLIC POINTER
32       13   USER DATA

IMS HFPF UTILITIES - FPA INDEXBLD                                "Secondary Index Definition Report"
5698-FPP V2R1                                                    PAGE: 2
DEFINITION OF SEC. INDEX (XDFLD) : XNAME7 (TARGET SEG.: ROOTSEG1, SOURCE SEG.: DD1 )
                                                                    2019-11-22 16:52:55

ATTRIBUTES:
- DB ORGANIZATION      : HISAM
- POINTER SEG. NAME    : DD1X
- NULLVAL              : N/A
- MULTISEG            : NO
- INDEX MAINTENANCE EXIT ROUTINE : N/A
- PARTITION SELECTION EXIT ROUTINE : EXITGS26
- PARTITION SELECTION OPTION : SNGL

SEC. INDEX DBD/DDNAMES:
-----
DBDNAME  DDNAME  OVFL  DBDNAME  DDNAME  OVFL  DBDNAME  DDNAME  OVFL
-----
DEDBGS25 GS25KSDS GS25ESDS  DEDBGS26 GS26KSDS N/A      DEDBGS27 GS27KSDS N/A
DEDBGS28 GS28KSDS GS28ESDS

RECORD LAYOUT:
-----
OFFSET  LENGTH  FIELD
-----
0        4     DUP. KEY POINTER
4        1     DELETE BYTE
5        11    SEARCH
16       0     SUBSEQ
16       0     DDATA
16       12   SYMBOLIC POINTER
28       3     USER DATA

*** END OF DATABASE DEFINITION REPORT ***

```

Figure 246. Secondary Index Definition report

PRIMARY DEDB NAME

The name of the primary DEDB.

NUMBER OF SEC. INDEXES

The number of the secondary indexes that are defined.

DEFINITION OF SEC. INDEX (XDFLD)

The XDFLD name that is defined.

TARGET SEG.

The target segment name that is defined.

SOURCE SEG.

The source segment name that is defined.

DB ORGANIZATION

The database structure: either HISAM or SHISAM.

POINTER SEG. NAME

The pointer segment name that is defined.

NULLVAL

The value that you specified for the NULLVAL parameter on the XDFLD statement (in hexadecimal format). If this value was not used, N/A is shown.

MULTISEG

Indicator that shows whether the secondary index segment is defined as a multiple secondary index segment.

INDEX MAINTENANCE EXIT ROUTINE

The index maintenance exit routine that is defined. When a routine is not defined, N/A is shown.

PARTITION SELECTION EXIT ROUTINE

The partition selection exit routine that is defined. When a routine is not defined, N/A is shown.

PARTITION SELECTION OPTION

The value that you specified for the XDFLD PSELOPT statement parameter when you generated the DBD. The value can be MULT, SNGL, or N/A. When a partition selection exit routine is not defined, this field shows N/A. For information about the PSELOPT parameter, see *IMS System Utilities*.

SEC. INDEX DBD/DDNAMES:

This field provides the following information:

DBDNAME

The name of the DBD.

DDNAME

The name of the KSDS data set.

OVFL DDNAME

The name of the ESDS data set. When an ESDS data set is not defined, N/A is shown.

RECORD LAYOUT:

The offset values, length, and the names of the fields in the index pointer segment. The information that is provided in this field is different based on the definition.

For a HISAM secondary index pointer, a secondary index pointer segment that consists of a prefix field and data fields is stored in a logical record. For a SHISAM secondary index pointer, a secondary index pointer segment that consists of only data fields is stored in a logical record.

DUP.KEY POINTER

The duplicate key pointer field is a 4-byte field that indicates that the pointer segment contains duplicate keys. This information is printed only for the HISAM secondary index pointer segments that have a non-unique key.

DELETE BYTE

The delete byte field is a 1-byte field. This information is printed only for the HISAM secondary index pointer segments. This field is contained in the prefix field.

SEARCH

The search field is a variable-length field that consists of up to five fields of the source segment. This field is contained in the data field.

SUBSEQ

The subsequence field is an optional variable-length field that consists of up to five fields of the source segment or IMS-generated values.

DDATA

The duplicate data field is an optional variable-length field that consists of up to five fields of the source segment.

SYMBOLIC POINTER

The symbolic pointer field is a variable-length field that contains the concatenated key of the target segment.

USER DATA

The user data field is an optional variable-length field that contains user data.

Secondary Index Processing report

The Secondary Index Processing report provides information about segments and data sets that are involved in building secondary index databases.

This report contains the following three sections:

Secondary Index Database Information

Segments to be loaded into the secondary index databases in load mode, and segments to be suppressed in load mode.

Build Pointer Segment Record Data Set Information

Build pointer segment record data sets that were allocated during the job.

Sort Control Statement Information

Sort control statement cards for each build pointer segment record data set. You can use these control statement cards to manually sort the records by using DFSORT.

This report is generated by the INDEXBLD command with IDXPROC=BUILD.

This report is not printed if you specify the GLOBAL SCAN=YES keyword.

Secondary Index Database Information section

The following figure is an example of the Secondary Index Database Information section of Secondary Index Processing report.

```

IMS HFPF UTILITIES - FPA INDEXBLD                                "Secondary Index Processing Report"                                PAGE: 1
5698-FPP V2R1                                                    2019-11-22 16:52:55
- PRIMARY DEDB NAME      : DEDBJN21
- NUMBER OF SEC. INDEX DATABASE: 5(PARTITION INCLUDED)

SEC. INDEX DATABASE INFORMATION:

PARTITION GROUP  DBDNAME  DDNAME  ALLOC TYPE  DSNAME  LOADED SEGMENTS  SUPPRESSED SEGMENTS  TOTAL SEGMENTS
-----
DEDBGS21         DEDBGS21  GS21KSDS  JCL  HFPF.GS21KSDS  0
                GS21ESDS  JCL  HFPF.GS21ESDS  0
                (TOTAL)  0  1,272  1,272
DEDBGS25         DEDBGS25  GS25KSDS  DYN  HFPF.GS25KSDS  15
                (TOTAL)  15
DEDBGS25         DEDBGS26  GS26KSDS  JCL  HFPF.GS26KSDS  3,000
                (TOTAL)  3,000
DEDBGS25         DEDBGS27  GS27KSDS  DYN  HFPF.GS27KSDS  0
                (TOTAL)  0
DEDBGS25         DEDBGS28  GS28KSDS  DYN  HFPF.GS28KSDS  1
                (TOTAL)  1
DEDBGS25         (GROUP TOTAL)  3,016  450  3,466
  
```

Figure 247. Secondary Index Processing report - Secondary Index Database Information section

The following figure shows another example of the Secondary Index Processing report.

- When duplicate segments are found in an index database, #: BUILD FAILED and \$: BUILD CANCELED are printed, and a number sign (#) is printed before the index DBD name.
- When PROC_AFT_DUPKEY=STOP is specified and the processing stops due to detecting duplicate segments, a dollar sign (\$) is printed before the index DBD name.

SEC. INDEX DATABASE INFORMATION:

#: BUILD FAILED \$: BUILD CANCELED

PARTITION GROUP	DBDNAME	DDNAME	ALLOC TYPE	DSNAME	LOADED SEGMENTS	SUPPRESSED SEGMENTS	TOTAL SEGMENTS
	#DEDBGS22	GS22KSDS	JCL	HPPF.GS22KSDS		1	
				(TOTAL)		1	1
	\$DEDBGS23	GS23KSDS	JCL	HPPF.GS23KSDS		0	
				(TOTAL)		0	0

Figure 248. Secondary Index Processing report - Secondary Index Database Information section (when duplicate segments are found)

PRIMARY DEDB NAME

The name of the primary DEDB.

NUMBER OF SEC. INDEXES

The number of secondary indexes that were specified. When one or more partitioned indexes were processed, (PARTITION INCLUDED) is printed after the number.

PARTITION GROUP

The first DBD name in the partition group.

DBDNAME

The name of the DBD.

DDNAME

The ddname of the secondary index database data set.

ALLOC TYPE

The method that was used to allocate the data set.

- JCL indicates that the DDNAME was specified in the JCL statement.
- DYN indicates that the DDNAME was allocated dynamically by FPA.
- N/A indicates that the DDNAME was not allocated by FPA. When the Build Index function runs in scan mode, the data sets for the secondary index databases are not allocated.

DSNAME

The name of the data set. When the Build Index function runs in scan mode, DATA SET IS NOT ALLOCATED is printed.

LOADED SEGMENTS | GENERATED SEGMENTS

LOADED SEGMENTS

The number of segments that were loaded. Printed only when the job runs in scan load mode or in load mode.

GENERATED SEGMENTS

The number of segments that were generated in the build pointer segment record data sets. Printed only when the job runs in scan mode.

SUPPRESSED SEGMENTS

The number of segments that were suppressed by the NULLVAL operand or by the index maintenance exit routine.

TOTAL SEGMENTS

The total number of segments. The value is a sum of loaded segments and suppressed segments.

Build Pointer Segment Record Data Set Information section

The following figure is an example of the Build Pointer Segment Record Data Set Information section of Secondary Index Processing report.

BUILD POINTER SEGMENT RECORD DATA SET INFORMATION:

INDEX DBDNAME	DDNAME	DSNAME	ALLOCATION
DEDBGS21	S0000010	HPFP.BUILD.DEDBGS21	DYNALLOC
DEDBGS25	S0000020	HPFP.BUILD.DEDBGS25	DYNALLOC
DEDBGS26	S0000030	HPFP.BUILD.DEDBGS26	DYNALLOC
DEDBGS27	S0000040	HPFP.BUILD.DEDBGS27	DYNALLOC
DEDBGS28	S0000050	HPFP.BUILD.DEDBGS28	DYNALLOC

Figure 249. Secondary Index Processing report – Build Pointer Segment Record Data Set Information section

INDEX DBDNAME

The name of the secondary index DBD.

DDNAME

The ddname of the build pointer segment record data set.

DSNAME

The name of the build pointer segment record data set.

ALLOCATION

One of the following values is printed:

JCL DD

The build pointer segment record data set was specified by the S0nnnnn0 DD statement in the JCL.

DYNALLOC

The build pointer segment record data set was allocated dynamically by the BUILDDSNMASK keyword or the BLDMFILECTL subcommand.

Sort Control Statement Information section

The following figure is an example of the Sort Control Statement Information section of Secondary Index Processing report.

SORT CONTROL STATEMENT INFORMATION:

INDEX DBDNAME: DEDBGS21 DSNAME: HPFP.BUILD.DEDBGS21

SORT CONTROL STATEMENT:

SORT FIELDS=(1,00019,CH,A)
RECORD TYPE=F,LENGTH=00044
OPTION FILSZ=E00000000001

PARTITION GROUP: DEDBGS25 INDEX DBDNAME: DEDBGS25 DSNAME: HPFP.BUILD.DEDBGS25
DEDBGS26 HPFP.BUILD.DEDBGS26
DEDBGS27 HPFP.BUILD.DEDBGS27
DEDBGS28 HPFP.BUILD.DEDBGS28

SORT CONTROL STATEMENT:

SORT FIELDS=(1,00015,CH,A)
RECORD TYPE=F,LENGTH=00030
OPTION FILSZ=E00000003001

Figure 250. Secondary Index Processing report – Sort Control Statement Information section

PARTITION GROUP

The name of the first DBD in the partition group. This field is printed only when partitioned indexes were processed.

INDEX DBDNAME

The name of the secondary index DBD. When multiple partitioned indexes were processed, all DBD names in the partition group are printed.

DSNAME

The name of the build pointer segment record data set. When multiple partitioned indexes were processed, names of all data sets that belong to the partition group are printed.

SORT CONTROL STATEMENT

Auto-generated SORT control statements that you can copy and paste in a DFSORT job to sort the records in the build pointer segment record data sets.

If you plan to use the build pointer segment records in load mode with the SORT=NO option, ensure that you run DFSORT or an equivalent program to sort the records in the build pointer segment record data sets in advance.

Secondary Index Resync report

The Secondary Index Resync report provides information about the segments that were inserted to or deleted from the secondary index databases.

This report is generated by the INDEXBLD command with IDXPROC=RESYNC. It is not generated if you specify the GLOBAL SCAN=YES keyword.

The following figure is an example of the Secondary Index Resync report that is generated during a Resync job in verify mode for online DEDB areas (TOICTL=NONE).

```

IMS HPFP UTILITIES - FPA INDEXBLD                                "Secondary Index Resync Report"
5698-FPP V2R1                                                    PAGE: 1
                                                                2019-05-02 10:29:17
- PRIMARY DEDB NAME      : DEDBJN23
- NUMBER OF SEC. INDEX DATABASE: 3(PARTITION INCLUDED)
- RESYNC MODE           : VERIFY
- TOICTL                : NONE

SEC. INDEX DATABASE INFORMATION:
* : RESYNC PERFORMED   #: RESYNC FAILED   $: RESYNC CANCELED

PARTITION GROUP          DBDNAME  DDNAME  DSNAME
-----
*DEDBGS22  GS22KSDS  HPFP.GS22KSDS
(TOTAL) 1,500,000  2,000,000  500,000  0
DEDBGS25 *DEDBGS25  GS25KSDS  HPFP.GS25KSDS
(TOTAL) 1,000,000  900,000  200,000  300,000
DEDBGS25  DEDBGS26  GS26KSDS  HPFP.GS26KSDS
(TOTAL) 400,000  400,000  0  0
DEDBGS25 (GROUP TOTAL) 1,400,000  1,300,000  200,000  300,000

RESYNC POINTER SEGMENT RECORD DATA SET INFORMATION:
INDEX
DBDNAME  DDNAME  DSNAME          ALLOCATION
-----
DEDBGS22 S000001R HPFP.RSNC.DEDBGS22 DYNALLOC
DEDBGS25 S000002R HPFP.RSNC.DEDBGS25 DYNALLOC
DEDBGS26 S000003R HPFP.RSNC.DEDBGS26 DYNALLOC

```

Figure 251. Secondary Index Resync report from Resync (Verify mode)

The following figure is an example of the Secondary Index Resync report that is generated during a Resync job in area scan mode for online DEDB areas (TOICTL=NONE).

- PRIMARY DEDB NAME : DEDBJN23
- NUMBER OF SEC. INDEX DATABASE: 3(PARTITION INCLUDED)
- RESYNC MODE : AREASCAN
- TOICTL : NONE

SEC. INDEX DATABASE INFORMATION:

*: RESYNC PERFORMED #: RESYNC FAILED \$: RESYNC CANCELED

PARTITION GROUP	DBDNAME	DDNAME	DSNAME	POINTER SEGMENT	
				GENERATED	
	DEDBGS22	GS22KSDS	DATA SET IS NOT ALLOCATED	(TOTAL)	2,000,000
DEDBGS25	DEDBGS25	GS25KSDS	DATA SET IS NOT ALLOCATED	(TOTAL)	900,000
DEDBGS25	DEDBGS26	GS26KSDS	DATA SET IS NOT ALLOCATED	(TOTAL)	400,000
DEDBGS25			(GROUP TOTAL)		1,300,000

UNVERIFIED RESYNC POINTER SEGMENT RECORD DATA SET INFORMATION:

INDEX DBDNAME	DDNAME	DSNAME	ALLOCATION
DEDBGS22	S0000010	HPFP.RSNC.DEDBGS22	DYNALLOC
DEDBGS25	S0000020	HPFP.RSNC.DEDBGS25	DYNALLOC
DEDBGS26	S0000030	HPFP.RSNC.DEDBGS26	DYNALLOC

SORT CONTROL STATEMENT INFORMATION:

INDEX DBDNAME: DEDBGS22 DSNAME: HPFP.RSNC.DEDBGS22
SORT CONTROL STATEMENT:
SORT FIELDS=(1,00035,CH,A)
RECORD TYPE=F,LENGTH=00104
OPTION FILSZ=E0002000000

PARTITION GROUP: DEDBGS25 INDEX DBDNAME: DEDBGS25 DSNAME: HPFP.RSNC.DEDBGS25
DEDBGS26 HPFP.RSNC.DEDBGS26

SORT CONTROL STATEMENT:
SORT FIELDS=(1,00012,CH,A)
RECORD TYPE=F,LENGTH=00064
OPTION FILSZ=E0000900000

Figure 252. Secondary Index Resync report from Resync (Area scan mode)

The following figure is an example of the Secondary Index Resync report after an offline synchronization (TOICTL=DBR or TOICTL=DBD).

- PRIMARY DEDB NAME : DEDBJN23
- NUMBER OF SEC. INDEX DATABASE: 3(PARTITION INCLUDED)
- RESYNC MODE : VERIFYUPDATE
- TOICTL : DBR

SEC. INDEX DATABASE INFORMATION:

*: RESYNC PERFORMED

PARTITION GROUP	DBDNAME	DDNAME	DSNAME	POINTER SEGMENT			
				BEFORE RESYNC	AFTER RESYNC	INSERTED	DELETED
	*DEDBGS21	GS21KSDS	HPFP.GS21KSDS	300,000	500,000	200,000	0
		GS21ESDS	HPFP.GS21ESDS	1,000,000	1,500,000	500,000	0
			(TOTAL)	1,300,000	2,000,000	700,000	0
DEDBGS25	*DEDBGS25	GS25KSDS	HPFP.GS25KSDS	1,000,000	900,000	200,000	300,000
			(TOTAL)	1,000,000	900,000	200,000	300,000
DEDBGS25	DEDBGS26	GS26KSDS	HPFP.GS26KSDS	400,000	400,000	0	0
			(TOTAL)	400,000	400,000	0	0
DEDBGS25			(GROUP TOTAL)	1,400,000	1,300,000	200,000	300,000

Figure 253. Secondary Index Resync report from offline synchronization

PRIMARY DEDB NAME

The name of the primary DEDB database.

NUMBER OF SEC. INDEXES

The number of secondary indexes that were specified. When one or more partitioned indexes were processed, (PARTITION INCLUDED) is printed after the number.

RESYNC MODE

Synchronization mode, which is specified by the RESYNCMODE keyword.

TOICTL

The IMS command issued by IMS Tools Online System Interface, which is specified by the TOICTL keyword. For online synchronization, NONE is printed.

SEC. INDEX DATABASE INFORMATION**PARTITION GROUP**

The first DBD name in the partition group.

DBDNAME

The name of the DBD.

DDNAME

The ddname of the secondary index database data set.

DSNAME

The name of the data set.

BEFORE RESYNC

The number of segments before synchronization. This field is not printed in area scan mode.

AFTER RESYNC

The number of segments after synchronization. This field is not printed in area scan mode.

INSERTED

The number of segments that were inserted during synchronization. This field is not printed in area scan mode.

DELETED

The number of segments that were deleted during synchronization. This field is not printed in area scan mode.

GENERATED

The number of segments that were generated from the index source segments in DEDB areas. This field is printed only in area scan mode.

The numbers of segments that are reported in this report are the numbers of the segments that point to the input areas.

RESYNC POINTER SEGMENT RECORD DATA SET INFORMATION

This information is printed only in verify mode or update mode.

INDEX DBDNAME

The name of the secondary index DBD.

DDNAME

The DD name of the resync pointer segment record data set.

DSNAME

The name of the resync pointer segment record data set.

ALLOCATION

One of the following values is printed:

JCL DD

The resync pointer segment record data set was specified by the S0nnnnnR DD statement in the JCL.

DYNALLOC

The resync pointer segment record data set was allocated dynamically by the RESYNCDSNMASK keyword or the FILECTL subcommand.

UNVERIFIED RESYNC POINTER SEGMENT RECORD DATA SET INFORMATION

This information is printed only in verify update mode or direct update mode with the AREASCAN=NO option.

INDEX DBDNAME

The name of the secondary index DBD.

DDNAME

The DD name of the unverified resync pointer segment record data set.

DSNAME

The name of the unverified resync pointer segment record data set.

ALLOCATION

One of the following values is printed.

JCL DD

The unverified resync pointer segment record data set was specified by the *S0nnnnn0* DD statement in the JCL.

DYNALLOC

The unverified resync pointer segment record data set was allocated dynamically by the RESYNCDSNMASK keyword or the FILECTL subcommand.

SORT CONTROL STATEMENT INFORMATION

This information is printed only in area scan mode.

PARTITION GROUP

The first DBD name in the partition group. This field is printed only when partitioned indexes were processed.

INDEX DBDNAME

The name of the secondary index DBD. When multiple partitioned indexes were processed, all DBD names in the partition group are printed.

DSNAME

The name of the unverified resync pointer segment record data set. When multiple partitioned indexes were processed, names of all data sets that belong to the partition group are printed.

SORT CONTROL STATEMENT

Auto-generated SORT control statements that you can copy and paste in a DFSORT job to sort the records in the unverified resync pointer segment record data sets.

If you want to use the unverified resync pointer segment records in verify update mode with the AREASCAN=NO option, you must use DFSORT or an equivalent program and sort the records in the unverified resync pointer segment record data sets before you run the job in verify update mode.

Secondary Index Analysis report

The Secondary Index Analysis report provides information about the pointer segments and the source segments of the processed secondary index databases.

This report is generated by the ANALYZE command with the INDEXDBD keyword.

This report is not printed if you specified the GLOBAL SCAN=YES keyword.

The following figure is an example of the Secondary Index Analysis report.

- PRIMARY DEDB NAME : DEDBJN21
- NUMBER OF SEC. INDEX DATABASE : 5 (PARTITION INCLUDED)

SEC. INDEX DATABASE INFORMATION:

*:ERROR DETECTED

PARTITION GROUP	DBDNAME	DDNAME	ALLOC TYPE	DSNAME	POINTER SEGMENTS	SOURCE SEGMENTS
	DEDBGS21	GS21KSDS	JCL	HPFP.GS21KSDS	0	
		GS21ESDS	JCL	HPFP.GS21ESDS	0	
				(TOTAL)	0	0
DEDBGS25	DEDBGS25	GS25KSDS	DYN	HPFP.GS25KSDS	15	
				(TOTAL)	15	15
DEDBGS25	DEDBGS26	GS26KSDS	JCL	HPFP.GS26KSDS	3,000	
				(TOTAL)	3,000	3,000
DEDBGS25	DEDBGS27	GS27KSDS	DYN	HPFP.GS27KSDS	0	
				(TOTAL)	0	0
DEDBGS25	*DEDBGS28	GS28KSDS	DYN	HPFP.GS28KSDS	1	
				(TOTAL)	1	27
DEDBGS25				(GROUP TOTAL)	3,016	3,042

Figure 254. Secondary Index Analysis report

PRIMARY DEDB NAME

The name of the primary DEDB.

NUMBER OF SEC. INDEXES

The number of secondary indexes that were specified. When one or more partitioned indexes were processed, (PARTITION INCLUDED) is printed after the number.

PARTITION GROUP

The first DBD name in the partition group.

DBDNAME

The name of the DBD. An asterisk (*) at the beginning of DBDNAME indicates that the corresponding index database has an error.

DDNAME

The ddname of the secondary index database data set.

ALLOC TYPE

The method that was used to allocate the data set.

- JCL indicates that the DDNAME was specified in the JCL statement.
- DYN indicates that the DDNAME was allocated dynamically by FPA.

DSNAME

The name of the data set.

POINTER SEGMENTS

The number of pointer segments in the corresponded index database that is located at targeted area.

SOURCE SEGMENTS

The number of source segment occurrences that correspond to the pointer segment.

Secondary Index Duplicate Key report

The Secondary Index Duplicate Key report shows information about secondary indexes that have duplicated keys.

This report is generated by the INDEXBLD command.

The following figure is an example of the Secondary Index Duplicate Key report.

DUPLICATE KEY VALUE INFORMATION:

PARTITION GROUP	DBDNAME	DDNAME	SEGMENT SEQUENCE FIELD (HEX)	(CHARACTER)	NUMBER OF DUPLICATES	# OF KEYS USED MORE THAN ONCE
	DBD00001	S000001R	C4C4F2F2 C9C4E7F1 C4C4F2F2 C9C4E7F2	*DD22IDX1 *DD22IDX2 (TOTAL)	* * 2	1 1 2
GROUP001	DBD00002	S000002R	F1F1F1F1 F1F1F1F1 F1F1F1F1 F1F1F1F1 F1F1F1F1 F1F1F1F1 F1F1F1F1 F1F1F1F1 F1F1F1F1 F1F1F1F1 F2F2F2F2 F2F2F2F2 F2F2F2F2 F2F2F2F2 F2F2F2F2 F2F2F2F2 F2F2F2F2 F2F2F2F2 F2F2F2F2 F2F2F2F2	*11111111111111111111111111111111* *11111111111111111111111111111111* *22222222222222222222222222222222* *22222222222222222222222222222222* (TOTAL)	* * * * 20	10 10 2
GROUP001	DBD00003	S000003R	F3F3F3F3 F3F3F3F3 F3F3F3F3 F3F3F3F3 F3F3F3F3 F3F3F3F3 F3F3F3F3 F3F3F3F3 F3F3F3F3 F3F3F3F3 F4F4F4F4 F4F4F4F4 F4F4F4F4 F4F4F4F4 F4F4F4F4 F4F4F4F4 F4F4F4F4 F4F4F4F4 F4F4F4F4 F4F4F4F4	*33333333333333333333333333333333* *33333333333333333333333333333333* *44444444444444444444444444444444* *44444444444444444444444444444444* (TOTAL)	* * * * 20	10 10 2
GROUP001				(GROUP TOTAL)		40

DUPLICATE POINTER SEGMENT DATA SET INFORMATION:

INDEX DBDNAME	DDNAME	DSNAME	ALLOCATION
DBD00001	S000001R	HPFP.S000001R.DUP	DYNALLOC
DBD00002	S000002R	HPFP.S000002R.DUP	DYNALLOC
DBD00003	S000003R	HPFP.S000003R.DUP	DYNALLOC

Figure 255. Secondary Index Duplicate Key report

PARTITION GROUP

The first DBD name in the partition group.

DBDNAME

The name of the DBD.

DDNAME

The ddname of the secondary index database data set.

SEGMENT SEQUENCE FIELD

The key value in hexadecimal and character format.

NUMBER OF DUPLICATES

The number of occurrences of this key value in this DBD. For example, 10 is printed if a key value is used 10 times.

OF KEYS USED MORE THAN ONCE

The number of duplicate keys. For example, 2 is printed if key values DD22IDX1 and DD22IDX2 are used more than once in this DBD.

DUPLICATE POINTER SEGMENT RECORD DATA SET INFORMATION

INDEX DBDNAME

The name of the secondary index DBD.

DDNAME

The DD name of the duplicate pointer segment record data set.

DSNAME

The name of the duplicate pointer segment record data set.

ALLOCATION

One of the following values is printed:

JCL DD

The duplicate pointer segment record data set was specified by a S0nnnnnR DD statement in the JCL.

DYNALLOC

The duplicate pointer segment record data set was allocated dynamically by the DUPKEYDSNMASK keyword or the FILECTL subcommand.

Resync Pointer Segment Dump report

The Resync Pointer Segment Dump report provides information about the segments that were or that will be inserted or deleted to synchronize the secondary index database with the primary DEDB database.

This report is generated by the INDEXBLD command with RESYNCDUMP=YES. It is not generated if you specify the GLOBAL SCAN=YES keyword.

The maximum number of pointer segments printed in the report is controlled by the RESYNCDUMPMAX keyword.

The report provides the following information:

- The index pointer segments that are inserted to or deleted from the secondary index databases.
- The index pointer segments that FPA failed to insert or delete.

The following figure is an example of the Resync Pointer Segment Dump report.

IMS HPPF UTILITIES - FPA INDEXBLD 5698-FPP V2R1		"Resync Pointer Segment Dump report"		PAGE: 1
				2019-05-02 10:29:17
DBDNAME: DEDBGS22	DDNAME: GS22KSDS	DSNAME: HPPF.GS22KSDS		
DELETED (THE RELATED AREA IS AREA NO: 1 AREANAME: DB23AR0)				
SEARCH	0000	C1F1F2	*A12	*
SUBSEQ	0003	F0F0F0F0 F1F2F1F1	*00001211	*
DDATA	000B	D1F1F2	*J12	*
SYMBOLIC PTR	000E	F1F0F0F0 F0F0F0F0 F2F1F0F0 F0F0F0F1 F2F1	*100000002100000121	*

INSERTED (THE RELATED AREA IS AREA NO: 1 AREANAME: DB23AR0)				
SEARCH	0000	C1F1F3	*A13	*
SUBSEQ	0003	F0F0F0F0 F1F2F1F1	*00001211	*
DDATA	000B	D1F1F3	*J13	*
SYMBOLIC PTR	000E	F1F0F0F0 F0F0F0F0 F2F1F0F0 F0F0F0F1 F2F1	*100000002100000121	*

NOT INSERTED BECAUSE PRIMARY DEDB WAS UPDATED (THE RELATED AREA IS AREA NO: 2 AREANAME: DB23AR1)				
SEARCH	0000	C1F2F1	*A21	*
SUBSEQ	0003	F0F0F0F0 F2F1F1F1	*00002111	*
DDATA	000B	D1F2F1	*J21	*
SYMBOLIC PTR	000E	F2F0F0F0 F0F0F0F0 F1F1F0F0 F0F0F0F2 F1F1	*200000001100000211	*

NOT DELETED BECAUSE PRIMARY DEDB WAS UPDATED (THE RELATED AREA IS AREA NO: 2 AREANAME: DB23AR1)				
SEARCH	0000	C1F2F2	*A22	*
SUBSEQ	0003	F0F0F0F0 F2F2F1F1	*00002211	*
DDATA	000B	D1F2F2	*J22	*
SYMBOLIC PTR	000E	F2F0F0F0 F0F0F0F0 F2F1F0F0 F0F0F0F2 F2F1	*200000002100000221	*

INSERTED (THE RELATED AREA IS AREA NO: 3 AREANAME: DB23AR2)				
SEARCH	0000	C1F3F1	*A31	*
SUBSEQ	0003	F0F0F0F0 F3F1F1F1	*00003111	*
DDATA	000B	D1F3F1	*J31	*
SYMBOLIC PTR	000E	F3F0F0F0 F0F0F0F0 F1F1F0F0 F0F0F0F3 F1F1	*300000001100000311	*

Figure 256. Resync Pointer Segment Dump report

DBDNAME

The name of the DBD.

DDNAME

The ddname of the secondary index database data set.

DSNAME

The name of the data set.

Each segment is reported in table format. The first line on the table shows the result of the operation that FPA performed on the segment:

INSERTED (THE RELATED AREA IS AREA NO: *nnnn* AREANAME: *areaname*)

The pointer segment was inserted to the secondary index database.

DELETED (THE RELATED AREA IS AREA NO: *nnnn* AREANAME: *areaname*)

The pointer segment was deleted from the secondary index database.

NOT INSERTED BECAUSE PRIMARY DEDB WAS UPDATED (THE RELATED AREA IS AREA NO: *nnnn* AREANAME: *areaname*)

The pointer segment was not inserted because the primary DEDB was updated after it was scanned.

NOT DELETED BECAUSE PRIMARY DEDB WAS UPDATED (THE RELATED AREA IS AREA NO: *nnnn* AREANAME: *areaname*)

The pointer segment was not deleted because the primary DEDB was updated after it was scanned.

NOT INSERTED BECAUSE DL/I CALL FAILED (THE RELATED AREA IS AREA NO: *nnnn* AREANAME: *areaname*)

The pointer segment was not inserted because the DL/I call to insert the pointer segment (ISRT) failed.

NOT DELETED BECAUSE DL/I CALL FAILED (THE RELATED AREA IS AREA NO: *nnnn* AREANAME: *areaname*)

The pointer segment was not deleted because the DL/I call to delete the pointer segment (DLET) failed.

NOT INSERTED BECAUSE PREVIOUS DL/I CALL FAILED (THE RELATED AREA IS AREA NO: *nnnn* AREANAME: *areaname*)

The DL/I call to insert this pointer segment was not issued because the DL/I call for a preceding pointer segment failed.

NOT DELETED BECAUSE PREVIOUS DL/I CALL FAILED (THE RELATED AREA IS AREA NO: *nnnn* AREANAME: *areaname*)

The DL/I call to delete this pointer segment was not issued because the DL/I call for a preceding pointer segment failed.

Here, AREA NO shows the area number that contains the segment that is pointed to by the pointer segment. AREANAME shows the area name that contains the segment that is pointed to by the pointer segment.

The first column of the table shows the name of the fields in the pointer segment.

SEARCH

The search field is a variable-length field that consists of up to five fields of the source segment. This field is contained in the data field.

SUBSEQ

The subsequence field is an optional variable-length field that consists of up to five fields of the source segment or IMSgenerated values.

DDATA

The duplicate data field is an optional variable-length field that consists of up to five fields of the source segment.

SYMBOLIC PTR

The symbolic pointer field is a variable-length field that contains the concatenated key of the target segment.

The second column of the table shows the offset from the first byte of the pointer segment excluding the prefix of the pointer segment.

The third column of the table shows the hexadecimal value of the field.

The fourth column of the table shows the character value of the field.

Secondary Index Freespace Reclaim report

The Secondary Index Freespace Reclaim report provides information about the result of Freespace Reclaim function.

This report is generated by the INDEXBLD command with FS_RECLAIM_REPORT=YES. It is not generated if you specify the GLOBAL_SCAN=YES keyword.

The following figure is an example of the Secondary Index Freespace Reclaim report that is generated during a Freespace Reclaim job for a DEDB.

PRIMARY DATABASE INFORMATION:
- PRIMARY DEDB NAME: DEDBJN10
- NUMBER OF SECONDARY INDEX DATABASES: 5 (PARTITIONS INCLUDED)

SECONDARY INDEX DATABASE INFORMATION:

PARTITION GROUP	DBDNAME	DDNAME	DSNAME	DATABASE TYPE	NUMBER OF USED CI		NBR OF POINTER SEGMENTS
					BEFORE RECLAIM	AFTER RECLAIM	
PINDX101	PINDX101	PX11KSDS	IMSDEV.ICF3.DEDBJN10.PX11KSDS	HISAM	4,004	2,256	12,943
PINDX101	PINDX102	PX12KSDS	IMSDEV.ICF3.DEDBJN10.PX12KSDS	HISAM	2,765	1,938	10,178
PINDX101	PINDX103	PX13KSDS	IMSDEV.ICF3.DEDBJN10.PX13KSDS	HISAM	2,836	2,004	11,130
PINDX101			(GROUP TOTAL)		9,605	6,198	34,251
PINDX201	PINDX201	PX21KSDS	IMSDEV.ICF3.DEDBJN10.PX21KSDS	SHISAM	N/A	N/A	N/A
PINDX201	PINDX202	PX22KSDS	IMSDEV.ICF3.DEDBJN10.PX22KSDS	SHISAM	N/A	N/A	N/A
PINDX201			(GROUP TOTAL)		N/A	N/A	N/A

Figure 257. Secondary Index Freespace Reclaim report

The PRIMARY DATABASE INFORMATION section contains the following fields:

PRIMARY DEDB NAME

The name of the primary DEDB database.

NUMBER OF SECONDARY INDEX DATABASES

The number of secondary indexes that were processed in the job. When one or more partitioned indexes were processed, (PARTITIONS INCLUDED) is printed after the number.

The SECONDARY INDEX DATABASE INFORMATION section contains the following fields:

PARTITION GROUP

The first DBD name in the partition group.

DBDNAME

The name of the DBD.

DDNAME

The ddname of the secondary index database data set.

DSNAME

The name of the data set.

DATABASE TYPE

The access method of the secondary index database.

NUMBER OF USED CI

BEFORE RECLAIM

The number of CIs that are used for pointer segments before freespace reclaim.

AFTER RECLAIM

The number of CIs that are used for pointer segments after freespace reclaim.

NBR OF POINTER SEGMENTS

The number of pointer segments in the secondary index database.

Result of Access Method Services report

The Result of Access Method Services report provides the output from Access Method Services (IDCAMS) which FPA invokes during the job.

This report is generated by the INDEXBLD command with FS_RECLAIM_REPORT=YES.

The following figure shows an example of the Result of Access Method Services report.

```

IMS HPFP UTILITIES - FPA INDEXBLD                "Result of Access Method Services"                PAGE: 1
5698-FPP V2R1                                2019-07-07 0:20:14
RESULT OF IDCAMS EXAMINE BEFORE FREESPACE RECLAIM:
EXAMINE -
  NAME(HPFP.GS22KSDS) -
  INDEXTST DATATEST ERRORLIMIT(100)
IDC01700I INDEXTST BEGINS
IDC11773I 1102 KEYS PROCESSED ON INDEX LEVEL 1, AVERAGE KEY LENGTH: 8.5
IDC11773I 23 KEYS PROCESSED ON INDEX LEVEL 2, AVERAGE KEY LENGTH: 9.9
IDC11774I CURRENT INDEX CISIZE IS 1024, RECOMMENDED MINIMUM INDEX CISIZE IS 1024
IDC01724I INDEXTST COMPLETE - NO ERRORS DETECTED
IDC01701I DATATEST BEGINS
IDC01709I DATATEST COMPLETE - NO ERRORS DETECTED
IDC01708I 1102 CONTROL INTERVALS ENCOUNTERED
IDC01710I DATA COMPONENT CONTAINS 3985 RECORDS
IDC01711I DATA COMPONENT CONTAINS 0 DELETED CONTROL INTERVALS
IDC01712I MAXIMUM LENGTH DATA RECORD CONTAINS 36 BYTES
IDC01722I 73 PERCENT FREE SPACE
IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 0

IDC0002I IDCAMS PROCESSING COMPLETE. MAXIMUM CONDITION CODE WAS 0

RESULT OF IDCAMS EXAMINE AFTER FREESPACE RECLAIM:
EXAMINE -
  NAME(HPFP.GS22KSDS) -
  INDEXTST DATATEST ERRORLIMIT(100)
IDC01700I INDEXTST BEGINS
IDC11773I 258 KEYS PROCESSED ON INDEX LEVEL 1, AVERAGE KEY LENGTH: 8.7
IDC11773I 23 KEYS PROCESSED ON INDEX LEVEL 2, AVERAGE KEY LENGTH: 9.9
IDC11774I CURRENT INDEX CISIZE IS 1024, RECOMMENDED MINIMUM INDEX CISIZE IS 1024
IDC01724I INDEXTST COMPLETE - NO ERRORS DETECTED
IDC01701I DATATEST BEGINS
IDC01728I FOUND 17 EMPTY CONTROL AREAS THAT HAVE NOT BEEN RECLAIMED
IDC01709I DATATEST COMPLETE - NO ERRORS DETECTED
IDC01708I 258 CONTROL INTERVALS ENCOUNTERED
IDC01710I DATA COMPONENT CONTAINS 3124 RECORDS
IDC01711I DATA COMPONENT CONTAINS 17 DELETED CONTROL INTERVALS
IDC01712I MAXIMUM LENGTH DATA RECORD CONTAINS 36 BYTES
IDC01722I 79 PERCENT FREE SPACE
IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 0

IDC0002I IDCAMS PROCESSING COMPLETE. MAXIMUM CONDITION CODE WAS 0

```

Figure 258. Result of Access Method Services report

Initialization Processing Summary report

The Initialization Processing Summary report provides information about the initialized areas.

This report is generated by the INITIALIZE command.

This report is not printed if you specify the GLOBAL SCAN=YES keyword.

The following figure is an example of the Initialization Processing Summary report.

```

IMS HPFP UTILITIES - FPA INITIALIZE                "Initialization Processing Summary"                PAGE: 1
5698-FPP V2R1                                2019-11-22 19:22:59

DATABASE: DEDBJN22
- AREA NO: 1 AREANAME: DB22AR0 DDNAME: DB22AR01 DSNAME: HPFP.DB22AR0.ADS1
(JCL DD ALLOCATION)
STARTED DATE: 2019-11-22 TIME: 19:22:59 ENDED DATE: 2019-11-22 TIME: 19:22:59
RAA PORTION:
- TOTAL BASE CI'S/UOW : 5
- TOTAL DOVF CI'S/UOW : 1
- TOTAL CI'S/UOW : 6
- TOTAL UOW'S : 4
IOVF PORTION:
- TOTAL CI'S : 11
- TOTAL MAP CI'S : 1
SDEP PORTION:
- TOTAL CI'S : 5
- RANGE : X'00005800' TO X'00006200'
- LOGICAL BEGIN : CYCLE COUNT X'00000001' RBA X'00005800'
- LOGICAL END : CYCLE COUNT X'00000001' RBA X'00005800'
- LOGICAL BEGIN TS : X'D194CC50C010E341'

```

Figure 259. Initialization Processing Summary report

AREA NO

The area number.

AREANAME

The area name.

DDNAME

The DDNAME with the following information:

(DYNAMIC ALLOCATION)

DDNAME is allocated dynamically.

(JCL DD ALLOCATION)

DDNAME is specified in the JCL statement.

DSNAME

The DSNAME of the area name DDNAME.

RAA PORTION**TOTAL BASE CI'S/UOW**

The total number of BASE CIs in each RAA part.

TOTAL DOVF CI'S/UOW

The total number of DOVF CIs in each RAA part.

TOTAL DOVF CI'S/UOW

The total number of CIs in each RAA part.

TOTAL UOW's

The total number of UOWs.

IOVF PORTION**TOTAL CI'S**

The total number of CIs in the IOVF part.

TOTAL MAP CI'S

The total number of MAP CIs in the IOVF part.

SDEP PORTION**TOTAL CI'S**

The total number of CIs in the SDEP part.

RANGE

The SDEP logical begin and logical end RBAs.

LOGICAL BEGIN

The cycle count and RBA of the SDEP logical begin.

LOGICAL END

The cycle count and RBA of the SDEP logical end.

LOGICAL BEGIN TS

The timestamp of the SDEP logical begin.

Pointer Segment Dump report

The Pointer Segment Dump report provides information about incorrect pointer segments.

This report is generated by the ANALYZE command when the INDEXDBD keyword and the IDXDUMP=YES option are specified, and when incorrect pointer segments are detected.

The report provides the following information:

- The pointer segments that are not found in the secondary index database.
- The pointer segments whose target segments are not found in the area.

The following figure is an example of the Pointer Segment Dump report.

DBDNAME: INDEX1 DDNAME: INDEXK1 DSNAME: HFPF.INDEXK1

THE RELATED AREA IS AREA NO: 1 AREANAME: DB22AR0
THE FOLLOWING POINTER SEGMENTS ARE NOT FOUND:

SEARCH	0000	4C6060C1	60E2F0F0	F1606060	60606E		*<--A-S001----->	*
SUBSEQ	000F	4C606060	60606060	E2E4C2E2	C5D86060	6060606E	*<-----SUBSEQ----->	*
DDATA	0023	4C606060	60C4C4C1	E3C16060	60606E		*<---DDATA----->	*
SYMBOLIC PTR	0032	10000020	00001F11				*.....*	*

THE TARGET SEGMENTS THAT ARE POINTED BY THE FOLLOWING POINTER SEGMENTS ARE NOT FOUND:

SEARCH	0000	4C6060C1	60E2F0F0	F1606060	60606E		*<--A-S001----->	*
SUBSEQ	000F	4C606060	60606060	E2E4C2E2	C5D86060	6060606E	*<-----SUBSEQ----->	*
DDATA	0023	4C406060	60C4C4C1	E3C16060	60606E		*<---DDATA----->	*
SYMBOLIC PTR	0032	10000020	00001F11				*.....*	*
USER DATA	003A	00000000	00000000	00000000			*.....*	*

Figure 260. Pointer Segment Dump report

First column

The name of the fields in the pointer segment. The following fields are displayed:

SEARCH

The search field is a variable-length field that consists of up to five fields of the source segment. This field is contained in the data field.

SUBSEQ

The subsequence field is an optional variable-length field that consists of up to five fields of the source segment or contains the value that is generated by IMS.

DDATA

The duplicate data field is an optional variable-length field that consists of up to five fields of the source segment.

SYMBOLIC PTR

The symbolic pointer field is a variable-length field that contains the concatenated key of the target segment.

USER DATA

The user data field is an optional variable-length field that contains user data.

Second column

The offset from the first byte of the pointer segment excluding the prefix of the pointer segment.

Third column

The hexadecimal value of the field.

Fourth column

The character value of the field.

Site and System Default report

The Site and System Default report shows the site default keyword parameter values that are registered either in the site default module HFPCZZ01 or in the site default table.

The column with the label SITE DEFAULT shows the site default keyword parameter values. The column with the label SYSTEM DEFAULT shows the system default values.

The following figure is an example of the Site and System Default report. The report shows the site default keyword parameter values that are registered in the site default module HFPCZZ01.

IMS HFPF UTILITIES		"Site and system default report"		PAGE: 1
5698-FPP V2R1				2019-11-22 15:32:28
SITE DEFAULT MODULE INFORMATION		:	HFPZZ01	
- RELEASE LEVEL		:	210	
- CREATE DATE		:	11/01/2019	
NOTE: '*' MEANS THAT THE VALUE SPECIFIED IS SAME AS SYSTEM DEFAULT VALUE. THE SPECIFICATION IGNORED.				
		SITE DEFAULT	-----	SYSTEM DEFAULT
ANALYZE	ATASKCTL	:		AREA
	CISNAP	:	YES	NO
	ICACHE	:	NO	YES
	ICHECK	:		NODEFER
	ICINCOMP	:		NO
	ICNOTIFY	:		YES
	ICOUT	:		NO
	IDXDUMP	:		NO
	IDXTASKCTL	:		4
	MAXERROR	:	*	100
	PTRCHKLVL	:	FULL	QUICK
	SDEP_ORPHAN_MSGLVL	:	WARNING	ERROR
	SDEPCHKLVL	:	FULL	QUICK
	SENSOR	:		NO
	SENSOR_DBREC	:		NO
	SENSOR_PTRERR	:	YES	NO
	SENSOR_UOW	:		NO
	SENSOR_UOW_REPORT	:		NO
CHANGE	CISNAP	:		NO
	ICACHE	:	NO	YES
	ICINCOMP	:		NO
	ICNOTIFY	:		YES
	ICOUT	:		NO
	ITASKCTL	:		AREA
	KEYCHGCHK	:	NO	YES
	KEYSEQCHK	:	ABEND	NOCHECK
	OCACHE	:	NO	YES
	OTASKCTL	:		AREA
	PAD	:	X'FF'	X'00'
	PTRCHK	:	BYPASS	ABEND
	PTRCHKLVL	:	FULL	QUICK
	RELOCATE_SDEP	:		NO
	SDEP_ORPHAN_MSGLVL	:	WARNING	ERROR
	SDEPCHKLVL	:	*	QUICK
	SENSOR	:		NO
	SENSOR_DBREC	:		NO
	SENSOR_PTRERR	:	YES	NO
	SENSOR_UOW	:		NO
	SENSOR_UOW_REPORT	:		NO
	SSP	:	NO	YES

Figure 261. Site and System Default report

The following figure is another example of the Site and System Default report. This report shows the site default keyword parameter values that are registered in the site default table.

IMS HFPF UTILITIES		"Site and system default report"		PAGE: 1
5698-FPP V2R1				2019-11-22 15:38:40
- SITE_DSNAME		:	HFPF.SITEDEF (HFPSYSIN)	
- SITE_MEMBER		:	SITEMEM (HFPSYSIN)	
SITE DEFAULT TABLE INFORMATION		:		
- RELEASE LEVEL		:	210	
- CREATE DATE		:	11/01/2019	
NOTE: '*' MEANS THAT THE VALUE SPECIFIED IS SAME AS SYSTEM DEFAULT VALUE. THE SPECIFICATION IGNORED.				
		SITE DEFAULT	-----	SYSTEM DEFAULT
GLOBAL	- KEYWORDS ALLOWED TO SPECIFY AS SITE DEFAULT ARE NOT SPECIFIED IN HFPSYSIN.			
ANALYZE	ATASKCTL	:		AREA
	CISNAP	:		NO
	ICACHE	:		YES
	ICHECK	:		NODEFER
	ICINCOMP	:		NO
	ICNOTIFY	:		YES
	ICOUT	:		NO
	IDXDUMP	:		NO
	IDXTASKCTL	:		4
	MAXERROR	:		100
	PTRCHKLVL	:	FULL	QUICK
	SDEP_ORPHAN_MSGLVL	:		ERROR
	SDEPCHKLVL	:		QUICK
	SENSOR	:	YES	NO
	SENSOR_DBREC	:	YES	NO
	SENSOR_PTRERR	:	YES	NO
	SENSOR_UOW	:		NO
	SENSOR_UOW_REPORT	:		NO

Figure 262. Site and System Default report

Related information

[Site default options support for FPA](#)

FPA provides the site default options to override the system default values. By using the site default options, you can omit specific values in the control statements in the HFPSYSIN DD data set.

Process Summary report

The Process Summary report summarizes the result of processing for each area.

This report is generated by the UNLOAD command and the RELOAD command.

The following figures are examples of the Process Summary report.

```

IMS HPFP UTILITIES - FPA UNLOAD                                "Process Summary Report"                                PAGE: 1
5698-FPP V2R1                                                2019-11-22 18:55:26

+-----+
| CHECKPOINT MODE                                           |
| # indicates the area that is to be reprocessed in the next restart processing |
+-----+
| UL: UNLOAD  IC: HPIC  IC REG: NOTIFY.IC                 |
+-----+

DBDNAME  AREANAME  AREA NO  UL  IC  IC  STATUS (If not blank, see messages in HFPPRINT/ICEPRINT.)
          #         RC   RC  REG
-----  -
DEDBJN22 DB22AR1    1    00  00  REG
DEDBJN22 DB22AR2    2    00  00  REG
DEDBJN22#DB22AR3    3    08                    Processing ended with errors.
DEDBJN22#DB22AR4    4    08                    Processing ended with errors.
DEDBJN22#DB22AR5    5    00  08                    Processing ended with errors.
DEDBJN22#DB22AR6    6    00  08                    Processing ended with errors.
DEDBJN22#DB22AR7    7    08                    Processing ended with errors.
DEDBJN22 DB22AR8    8    00  00  REG
DEDBJN22 DB22AR9    9    00  00  REG

```

Figure 263. Process Summary report generated by the UNLOAD command

```

IMS HPFP UTILITIES - FPA RELOAD                                "Process Summary Report"                                PAGE: 1
5698-FPP V2R1                                                2019-07-13 19:03:00

+-----+
| RESTART MODE ( 1ST RUN IN RESTART MODE)                 |
| # indicates the area that is to be reprocessed in the next restart processing |
+-----+
| RL: RELOAD  AN:ANALYZE  IC: HPIC  IC REG: NOTIFY.IC    |
+-----+

DBDNAME  AREANAME  AREA NO  RL  AN  IC  IC  STATUS (If not blank, see messages in HFPPRINT/ICEPRINT.)
          #         RC   RC  RC  REG
-----  -
DEDBJN23#DB23AR1    1    00  00  08                    Processing ended with errors.
DEDBJN23 DB23AR2    2    00  00  00  REG
DEDBJN23#DB23AR3    3    00  00  08                    Processing ended with errors.
DEDBJN23 DB23AR4    4    00  00  00  REG

```

Figure 264. Process Summary report generated by the RELOAD command

DBDNAME

Shows the name of the database.

AREANAME

Shows the name of the area. The area names are displayed in the order that they were defined.

- When RESTART=NO is specified, or RESTART=YES is specified and the job runs in checkpoint mode, all the areas that are specified for the IAREA keyword of the UNLOAD command or for the OAREA keyword of the RELOAD command are shown.
- When RESTART=YES is specified and the job runs in restart mode, only the areas that are reprocessed in the restart processing are shown.
- When RESTART=YES is specified, a number sign (#) at the beginning of AREANAME indicates the area that is to be reprocessed in the next restart processing.

AREA NO

Shows the area number.

UL RC

Shows the return code of the unload processing for the area. If the area was unloaded successfully in the previous run, NA is shown.

RL RC

Shows the return code of the reload processing for the area. If the area was reloaded successfully in the previous run, NA is shown.

AN RC

Shows the return code of the analyze processing for the area. If the area was analyzed successfully in the previous run (return code is 8 or lower), NA is shown.

IC RC

Shows the return code of the image copy processing for the area. If the image copies were taken successfully in the previous run, NA is shown.

IC REG

Shows whether the image copies for the area were successfully registered to DBRC RECON.

- REG: The image copies were successfully registered to DBRC.
- BYP: Registration to DBRC was bypassed because of errors.

STATUS

Shows the processing result for each area.

If FPA function ended successfully without errors, this column is blank. Otherwise, one of the following messages is displayed in this column:

Processing ended with warnings.

This message indicates that the processing ended with return code of 4.

Processing ended with errors.

This message indicates that the processing ended with return code of 8.

Processing did not complete.

This message indicates that the processing for the area did not complete because the processing ended abnormally.

Processing was skipped.

This message indicates that the processing for this area was skipped due to one of the following situations:

- A different area, which has been unloaded to the same unload data set, has an error.
- All input tasks ended abnormally. No more input tasks to process.
- The NOTIFY.IC commands for the precedent areas were issued, but ended with an error.
- Another image copy, which has been written to the same stack, has an error.

For details about these messages, see the HFPSYSIN data set, and when ICOUT=YES is specified, also see the ICEPRINT data set.

The following information is also displayed in this report when RESTART=YES is specified:

- Processing mode (whether the job was run in checkpoint mode or in restart mode). When the job was run in restart mode, the count of the restart processing is also displayed.
- The explanation for the number sign (#) indicator. The # character is shown at the beginning of AREANAME to indicate the area to be reprocessed in the next restart processing.

Sensor Data Statistics report

The Sensor Data Statistics report contains the names and the values of the data elements that are stored in the Sensor Data repository of IMS Tools KB.

This report is generated by the following commands:

- ANALYZE with the SENSOR=YES keyword
- CHANGE with the SENSOR=YES keyword and the PTRCHKLVL=FULL or PTRCHKLVL=QUICK keyword
- RELOAD with the SENSOR=YES keyword and the PTRCHKLVL=FULL or PTRCHKLVL=QUICK keyword
- REORG with the SENSOR=YES keyword and the PTRCHKLVL=FULL or PTRCHKLVL=QUICK keyword

For a list of the data elements that are collected by IMS HP Fast Path Utilities, see the topics about data elements in the *Tools Base Policy Services User's Guide*.

The following figures show an example of the Sensor Data Statistics report. The data elements that are underlined are printed only when SENSOR_DBREC=YES is specified.

```

IMS HPFP UTILITIES - DB SENSOR          "SENSOR DATA STATISTICS REPORT"          PAGE:      1
5698-FPP V2R1                          2019-03-22 17:11:31

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1
-----
AREA DEFINITION
-----
Data Element Name          Value
-----
DB_AREAEDEF_CISIZE         4,096
DB_AREAEDEF_UOW1           20
DB_AREAEDEF_UOW2           10
DB_AREAEDEF_ROOT1         1,500
DB_AREAEDEF_ROOT2         1,000
DB_AREAEDEF_NUM_SDEP_CIS   10,000
-----
FREE SPACE PERCENTAGE STATISTICS
-----
Data Element Name          Value
-----
DB_PCT_BYTES_FS_RAA        50%
DB_PCT_BYTES_FS_DOVF       50%
DB_PCT_BYTES_FS_IOVF       50%
DB_PCT_BYTES_FS_SDEP       50%
-----
OVERFLOW USAGE STATISTICS
-----
Data Element Name          Value
-----
DB_PCT_NUM_UOW_USE_DOVF    50%
DB_AVG_NUM_DOVFCI_BY_UOW   678.99
DB_MAX_NUM_DOVFCI_BY_UOW   9,999
DB_PCT_NUM_UOW_USE_IOVF    50%
DB_NUM_UOW_USE_IOVF        9,999
DB_AVG_NUM_IOVFCI_BY_UOW   678.99
DB_MAX_NUM_IOVFCI_BY_UOW   9,999
DB_MIN_NUM_IOVFCI_BY_UOW   9,999
DB_PCT_NUM_IOVFCI_USED     50%
DB_PCT_NUM_RAPCI_OVFL      50%
-----
SEGMENT OCCURRENCE STATISTICS
-----
Data Element Name          Value
-----
DB_NUM_SEG                  12,345,678

```

Figure 265. Sensor Data Statistics report (Part 1 of 2)

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1

```

=====
DATABASE RECORD STATISTICS
=====
Data Element Name          Value
-----
DB_NUM_ROOT                111,111
DB_AVG_DBREC_LENGTH        111.00
DB_MAX_DBREC_LENGTH        111,111
DB_MIN_DBREC_LENGTH        111,111
DB_PCT_NUM_DBREC_IOVF      50%
=====

RANDOMIZING STATISTICS
=====
Data Element Name          Value
-----
DB_AVG_LEN_SYNONYM_CHAIN    1.11
DB_MAX_LEN_SYNONYM_CHAIN    99,999
=====

PHYSICAL I/O STATISTICS
=====
Data Element Name          Value
-----
DB_ESTIMATED_DBREC_IO       1.25
DB_ESTIMATED_ROOT_IO        1.25
DB_AVG_DBREC_IO              678.99
DB_MAX_DBREC_IO              99,999
DB_AVG_ROOT_IO               678.99
DB_MAX_ROOT_IO               99,999
=====

UOW STATISTICS INFORMATION
=====
Data Element Name          Value
-----
DB_FLAG_UOW_DATA            N
DB_FLAG_UOW_GROUP_DATA      N
DB_THRESHOLD_RBASEFS        30%
DB_THRESHOLD_RDOVFFS        50%
DB_NUM_UOW_RFS_COND         50
DB_PCT_NUM_UOW_RFS_COND     10%
=====

```

Figure 266. Sensor Data Statistics report (Part 2 of 2)

The following figures show another example of the Sensor Data Statistics report. In this report, logical pages 3 and 4 contain data element values for the first UOW in the area that is specified by the IAREA keyword. This report is printed if both of the following conditions are met:

- SENSOR_UOW=YES is specified
- HFPAUOWC DD statement is not specified

In actual reports, data element values for other UOWs are printed in the subsequent pages.

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1

```

=====
AREA DEFINITION
=====
Data Element Name          Value
-----
DB_AREAEDEF_CISIZE        4,096
DB_AREAEDEF_UOW1          20
DB_AREAEDEF_UOW2          10
DB_AREAEDEF_ROOT1         1,500
DB_AREAEDEF_ROOT2         1,000
DB_AREAEDEF_NUM_SDEP_CIS  10,000
=====

FREE SPACE PERCENTAGE STATISTICS
=====
Data Element Name          Value
-----
DB_PCT_BYTES_FS_RAA        50%
DB_PCT_BYTES_FS_DOVF       50%
DB_PCT_BYTES_FS_IOVF       50%
DB_PCT_BYTES_FS_SDEP       52%
=====

OVERFLOW USAGE STATISTICS
=====
Data Element Name          Value
-----
DB_PCT_NUM_UOW_USE_DOVF    50%
DB_AVG_NUM_DOVFCI_BY_UOW   678.99
DB_MAX_NUM_DOVFCI_BY_UOW   9,999
DB_PCT_NUM_UOW_USE_IOVF    50%
DB_NUM_UOW_USE_IOVF        9,999
DB_AVG_NUM_IOVFCI_BY_UOW   678.99
DB_MAX_NUM_IOVFCI_BY_UOW   9,999
DB_MIN_NUM_IOVFCI_BY_UOW   9,999
DB_PCT_NUM_IOVFCI_USED     50%
DB_PCT_NUM_RAPCI_OVFL      50%
=====

SEGMENT OCCURRENCE STATISTICS
=====
Data Element Name          Value
-----
DB_NUM_SEG                  12,345,678
=====

```

Figure 267. Sensor Data Statistics report when SENSOR_UOW=YES (without HFPAUOWC DD) (Part 1 of 3)

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1

=====

DATABASE RECORD STATISTICS

=====

Data Element Name	Value
DB_NUM_ROOT	111,111
DB_AVG_DBREC_LENGTH	111.00
DB_MAX_DBREC_LENGTH	111,111
DB_MIN_DBREC_LENGTH	111,111
DB_PCT_NUM_DBREC_IOVF	50%

=====

RANDOMIZING STATISTICS

=====

Data Element Name	Value
DB_AVG_LEN_SYNONYM_CHAIN	1.11
DB_MAX_LEN_SYNONYM_CHAIN	99,999

=====

PHYSICAL I/O STATISTICS

=====

Data Element Name	Value
DB_ESTIMATED_DBREC_IO	1.25
DB_ESTIMATED_ROOT_IO	1.25
DB_AVG_DBREC_IO	678.99
DB_MAX_DBREC_IO	99,999
DB_AVG_ROOT_IO	678.99
DB_MAX_ROOT_IO	99,999

=====

UOW STATISTICS INFORMATION

=====

Data Element Name	Value
DB_FLAG_UOW_DATA	Y
DB_FLAG_UOW_GROUP_DATA	N
DB_THRESHOLD_RBASEFS	30%
DB_THRESHOLD_RDOVFFS	50%
DB_NUM_UOW_RFS_COND	50
DB_PCT_NUM_UOW_RFS_COND	10%

=====

REPOSITORY GROUP INFORMATION

=====

Data Element Name	Value
DB_SENSOR_DATA_GROUP_ID	IMS1

Figure 268. Sensor Data Statistics report when SENSOR_UOW=YES (without HFP AUOWC DD) (Part 2 of 3)

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1, UOW NUMBER: 0

=====

FREE SPACE PERCENTAGE STATISTICS

=====

Data Element Name	Value
DBU_PCT_BYTES_FS_RAA	80%
DBU_PCT_BYTES_FS_DOVF	80%
DBU_PCT_BYTES_FS_IOVF	95%
DBU_PCT_USABLE_RAAFS	80%
DBU_PCT_USABLE_DOVFFS	80%
DBU_PCT_USABLE_IOVFFS	95%
DBU_PCT_RAP_ROOTSZFS	0%
DBU_MAX_PCT_BYTES_RAPFS	0%
DBU_FLAG_UOW_USING_OVFL	Y
DBU_FLAG_UOW_USING_IOVF	Y

=====

OVERFLOW USAGE STATISTICS

=====

Data Element Name	Value
DBU_NUM_DOVFCI_BY_UOW	10
DBU_NUM_IOVFCI_BY_UOW	3
DBU_PCT_NUM_RAPCI_OVFL	100%

=====

DATABASE RECORD STATISTICS

=====

Data Element Name	Value
DBU_NUM_ROOT	120
DBU_AVG_DBREC_LENGTH	106.00
DBU_MAX_DBREC_LENGTH	582
DBU_MIN_DBREC_LENGTH	102
DBU_PCT_NUM_DBREC_IOVF	18%

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1, UOW NUMBER: 0

=====

RANDOMIZING STATISTICS

=====

Data Element Name	Value
DBU_AVG_LEN_SYN_CHAIN	120.00
DBU_MAX_LEN_SYN_CHAIN	120

=====

PHYSICAL I/O STATISTICS

=====

Data Element Name	Value
DBU_AVG_DBREC_IO	7.06
DBU_MAX_DBREC_IO	13
DBU_AVG_ROOT_IO	7.06
DBU_MAX_ROOT_IO	13

Figure 269. Sensor Data Statistics report when SENSOR_UOW=YES (without HFPAUOWC DD) (Part 3 of 3)

The following figures show another example of the Sensor Data Statistics report. In this report, logical pages 3 and 4 contain data element values for a UOW group that is specified by the HFPAUOWC DD statement. This report is printed if both of the following conditions are met:

- SENSOR_UOW=YES is specified
- HFPAUOWC DD statement is specified

In actual reports, if multiple UOW groups are specified, data element values for other UOW groups are printed in the subsequent pages.

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1

```

=====
AREA DEFINITION
=====
Data Element Name          Value
-----
DB_AREAEDEF_CISIZE        4,096
DB_AREAEDEF_UOW1          20
DB_AREAEDEF_UOW2          10
DB_AREAEDEF_ROOT1         1,500
DB_AREAEDEF_ROOT2         1,000
DB_AREAEDEF_NUM_SDEP_CIS  10,000
=====

FREE SPACE PERCENTAGE STATISTICS
=====
Data Element Name          Value
-----
DB_PCT_BYTES_FS_RAA        50%
DB_PCT_BYTES_FS_DOVF       50%
DB_PCT_BYTES_FS_IOVF       50%
DB_PCT_BYTES_FS_SDEP       52%
=====

OVERFLOW USAGE STATISTICS
=====
Data Element Name          Value
-----
DB_PCT_NUM_UOW_USE_DOVF    50%
DB_AVG_NUM_DOVFCI_BY_UOW   678.99
DB_MAX_NUM_DOVFCI_BY_UOW   9,999
DB_PCT_NUM_UOW_USE_IOVF    50%
DB_NUM_UOW_USE_IOVF        9,999
DB_AVG_NUM_IOVFCI_BY_UOW   678.99
DB_MAX_NUM_IOVFCI_BY_UOW   9,999
DB_MIN_NUM_IOVFCI_BY_UOW   9,999
DB_PCT_NUM_IOVFCI_USED     50%
DB_PCT_NUM_RAPCI_OVFL      50%
=====

SEGMENT OCCURRENCE STATISTICS
=====
Data Element Name          Value
-----
DB_NUM_SEG                  12,345,678
=====

```

Figure 270. Sensor Data Statistics report when SENSOR_UOW=YES (with HFPAUOWC DD) (Part 1 of 3)

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1

=====

DATABASE RECORD STATISTICS

=====

Data Element Name	Value
DB_NUM_ROOT	111,111
DB_AVG_DBREC_LENGTH	111.00
DB_MAX_DBREC_LENGTH	111,111
DB_MIN_DBREC_LENGTH	111,111
DB_PCT_NUM_DBREC_IOVF	50%

=====

RANDOMIZING STATISTICS

=====

Data Element Name	Value
DB_AVG_LEN_SYNONYM_CHAIN	1.11
DB_MAX_LEN_SYNONYM_CHAIN	99,999

=====

PHYSICAL I/O STATISTICS

=====

Data Element Name	Value
DB_ESTIMATED_DBREC_IO	1.25
DB_ESTIMATED_ROOT_IO	1.25
DB_AVG_DBREC_IO	678.99
DB_MAX_DBREC_IO	99,999
DB_AVG_ROOT_IO	678.99
DB_MAX_ROOT_IO	99,999

=====

UOW STATISTICS INFORMATION

=====

Data Element Name	Value
DB_FLAG_UOW_DATA	N
DB_FLAG_UOW_GROUP_DATA	Y
DB_NUM_UOW_GROUPS	2
DB_THRESHOLD_RBASEFS	30%
DB_THRESHOLD_RDOVFFS	50%
DB_NUM_UOW_RFS_COND	50
DB_PCT_NUM_UOW_RFS_COND	10%

=====

REPOSITORY GROUP INFORMATION

=====

Data Element Name	Value
DB_SENSOR_DATA_GROUP_ID	IMS1

Figure 271. Sensor Data Statistics report when SENSOR_UOW=YES (with HFP AUOWC DD) (Part 2 of 3)

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1, UOW GROUP (STARTUOW: 10001, STOPUOW: 20000)

=====

FREE SPACE PERCENTAGE STATISTICS

=====

Data Element Name	Value
DEBUG_PCT_BYTES_FS_RAA	80%
DEBUG_PCT_BYTES_FS_DOVF	80%
DEBUG_PCT_BYTES_FS_IOVF	95%
DEBUG_PCT_USABLE_RAAFS	80%
DEBUG_PCT_USABLE_DOVFFS	80%
DEBUG_PCT_USABLE_IOVFFS	95%
DEBUG_PCT_RAP_ROOTSZFS	0%

=====

OVERFLOW USAGE STATISTICS

=====

Data Element Name	Value
DEBUG_PCT_NUM_UOW_DOVF	80%
DEBUG_AVG_NUM_DOVFCCI	5
DEBUG_MAX_NUM_DOVFCCI	10
DEBUG_PCT_NUM_UOW_IOVF	80%
DEBUG_AVG_NUM_IOVFCCI	10
DEBUG_MAX_NUM_IOVFCCI	50
DEBUG_PCT_NUM_RAPCI_OVFL	100%

=====

DATABASE RECORD USAGE STATISTICS

=====

Data Element Name	Value
DEBUG_NUM_ROOT	120
DEBUG_AVG_DBREC_LENGTH	106.00
DEBUG_MAX_DBREC_LENGTH	582
DEBUG_MIN_DBREC_LENGTH	102
DEBUG_PCT_NUM_DBREC_IOVF	18%

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1, UOW GROUP (STARTUOW: 10001, STOPUOW: 20000)

=====

RANDOMIZING STATISTICS

=====

Data Element Name	Value
DEBUG_AVG_LEN_SYN_CHAIN	120.00
DEBUG_MAX_LEN_SYN_CHAIN	120

=====

PHYSICAL I/O STATISTICS

=====

Data Element Name	Value
DEBUG_AVG_DBREC_IO	7.06
DEBUG_MAX_DBREC_IO	13
DEBUG_AVG_ROOT_IO	7.06
DEBUG_MAX_ROOT_IO	13

Figure 272. Sensor Data Statistics report when SENSOR_UOW=YES (with HFPAUOWC DD) (Part 3 of 3)

Segment Length Distribution report

The Segment Length Distribution report provides detailed information about each segment. You can use this report to analyze segments and identify which segments require tuning.

This report is generated by the following commands:

- ANALYZE with the SEGLDIST=YES keyword on the REPORT subcommand and the PTRCHKLVL=FULL keyword
- CHANGE with the SEGLDIST=YES keyword on the REPORT subcommand and the PTRCHKLVL=FULL keyword
- RELOAD with the SEGLDIST=YES keyword on the REPORT subcommand and the PTRCHKLVL=FULL keyword
- REORG with the SEGLDIST=YES keyword on the REPORT subcommand and the PTRCHKLVL=FULL keyword

- TUNE with the SEGLDIST=YES keyword of the REPORT subcommand

The following figure shows an example of the Segment Length Distribution report.

```

IMS HPFP UTILITIES - FPA ANALYZE                                "Segment Length Distribution Report"                                PAGE: 2
5698-FPP V2R1                                                DBDNAME: DEDBJN23 AREA NAME : DB23AR1 DDNAME: DB23AR1 DSNAME : IMSVS.DB23.DB23AR1.ADS1
                                                                AREA NUMBER: 1 DSTYPE: ADS
BASIC FIGURES
-----
CI SIZE: 1,024
SEGMENT LENGTH: AVG: 184 MAX: 200 MIN: 181
-----
SC NAME SIZE RANGE OCCURRENCES PCT OF NUMBER OF SEGMENTS CUM. PCT FROM SHORTEST SIZE RANGES THAT
                                                                AND 1ST/3RD QUANTILES EXISTED
-----
3 DD12 181- 181 1 8.3 % 8.3 % MIN
      182- 182 5 41.7 % 50.0 % 1ST MED
      183- 183 5 41.7 % 91.7 % 3RD
      184- 184 0 0.0 % 91.7 % AVG
      185- 185 0 0.0 % 91.7 %
      186- 186 0 0.0 % 91.7 %
      187- 187 0 0.0 % 91.7 %
      188- 188 0 0.0 % 91.7 %
      189- 189 0 0.0 % 91.7 %
      190- 190 0 0.0 % 91.7 %
      191- 191 0 0.0 % 91.7 %
      192- 192 0 0.0 % 91.7 %
      193- 193 0 0.0 % 91.7 %
      194- 194 0 0.0 % 91.7 %
      195- 195 0 0.0 % 91.7 %
      196- 196 0 0.0 % 91.7 %
      197- 197 0 0.0 % 91.7 %
      198- 198 0 0.0 % 91.7 %
      199- 199 0 0.0 % 91.7 %
      200- 200 1 8.3 % 100.0 % MAX
-----
TOTAL 181- 200 12 100.0 %
-----
MORE THAN 90 PERCENT OF SEGMENT OCCURRENCES ARE INCLUDED IN THE RANGE: 181 - 183.

```

Figure 273. Segment Length Distribution report

CI SIZE

The CI size of the area.

SEGMENT LENGTH

AVG

The average length of the segments.

MAX

The length of the longest segment.

MIN

The length of the shortest segment.

SC

Segment code.

SEGMENT NAME

Segment name, as coded on the SEGM macro in the DBD.

SIZE RANGE

The size range (in bytes) of the part of the variable-length segment to be reported. For compressed segments, the length after compression is used for the size. The shortest and the longest sizes are actual values detected in the database, not extracted from the DBD.

This report divides the size range into 20. If the segment length is distributed within a certain narrow range, the size range might become less than 20 or the size range might not be fixed.

OCCURRENCES

The number of occurrences that are included in this range.

PCT OF NUMBER OF SEGMENTS

The percentage of occurrences of this range, and what percentage it makes of the total.

CUM. PCT FROM SHORTEST

The cumulative value of PERCENTAGE.

SIZE RANGES THAT AVG, MAX, MIN, MED, AND 1ST/3RD QUARTILES EXISTED

Data flag of segments is as follows:

AVG:

Size range in which the average segment length lies.

MAX:

Size range in which the maximum segment length lies.

MIX:

Size range in which the minimum segment length lies.

MED:

Size range in which the median of segment length lies.

1ST QUARTILE:

Size range in which the top quartile of segment lies.

3RD QUARTILE:

Size range in which the lower quartile of segment lies.

TOTALS

The minimum length, maximum length, and occurrences of the segment.

MORE THAN 90 PERCENT OF SEGMENT OCCURRENCES ARE INCLUDED IN THE RANGE xxx - xxx.

This field shows the range that contains more than 90% of segment occurrences. This field is based on a more specific range that includes the average segment length.

DB Record Length Distribution report

The optional report for database record analysis provides database record distribution broken down into 20 CI size ranges.

The reported database record length includes the length of the segment prefix and the length of the segment data as stored on DASD (for compressible segments, the length of the segment after compression). The table contains the number, the percentage, and the accumulated percentage, of database records for each size range.

From the asterisks (*) printed in the DISTRIBUTION OF DB RECORD LENGTH (PCT) column, you can learn how the database records are distributed.

This report is generated by the following commands:

- ANALYZE with the DBRLDIST=YES keyword on the REPORT subcommand and the PTRCHKLVL=FULL keyword
- CHANGE with the DBRLDIST=YES keyword on the REPORT subcommand and the PTRCHKLVL=FULL keyword
- RELOAD with the DBRLDIST=YES keyword on the REPORT subcommand and the PTRCHKLVL=FULL keyword
- REORG with the DBRLDIST=YES keyword on the REPORT subcommand and the PTRCHKLVL=FULL keyword
- TUNE with the DBRLDIST=YES keyword on the REPORT subcommand

The following figure shows an example of the DB Record Length Distribution report.

DBDNAME: DEDB000F AREA NAME : DB23AR1 DDNAME: DB23AR11 DSNAME : IMSVS.DB23.DB23AR1.ADS1
AREA NUMBER: 1 DSTYPE: ADS

BASIC FIGURES

CI SIZE: 2,048
DB RECORD LENGTH: AVG: 1,340 MAX: 9,376 MIN: 282

DB RECORD LENGTH (INCLUDING THE LENGTH OF SEGMENT PREFIX):

DB RECORD LENGTH RANGE (RANGE SIZE APPROX. 1/ 20 OF CI SIZE)	NUMBER OF DB RECORDS THAT FIT IN THE RANGE	PCT OF NBR OF DB RECS	CUM PCT FROM SHORTEST	DISTRIBUTION OF DB RECORD LENGTHS (PCT)										LENGTH RANGES IN WHICH AVG, MIN, MAX, MED, AND 1ST/3RD QUARTILES EXISTED	
				1	2	3	4	5	6	7	8	9	0		
1 - 102	0	0.0 %	0.0 %	0	0	0	0	0	0	0	0	0	0	0	
103 - 204	0	0.0 %	0.0 %												
205 - 307	4	6.5 %	6.5 %	***											MIN
308 - 409	8	12.9 %	19.4 %	*****											1ST
410 - 512	6	9.7 %	29.0 %	****											
513 - 614	0	0.0 %	29.0 %												
615 - 716	0	0.0 %	29.0 %												
717 - 819	0	0.0 %	29.0 %												
820 - 921	0	0.0 %	29.0 %												
922 - 1,024	16	25.8 %	54.8 %	*****											MED
1,025 - 1,126	2	3.2 %	58.1 %	*											
1,127 - 1,228	4	6.5 %	64.5 %	***											
1,229 - 1,331	2	3.2 %	67.7 %	*											
1,332 - 1,433	4	6.5 %	74.2 %	***											AVG
1,434 - 1,536	0	0.0 %	74.2 %												
1,537 - 1,638	1	1.6 %	75.8 %												3RD
1,639 - 1,740	2	3.2 %	79.0 %	*											
1,741 - 1,843	4	6.5 %	85.5 %	***											
1,844 - 1,945	2	3.2 %	88.7 %	*											
1,946 - 2,048	0	0.0 %	88.7 %												
MORE THAN 2,048	7	11.3 %	100.0 %	*****											MAX
TOTAL	62	100.0 %													

Figure 274. DB Record Length Distribution report

CI Size

The CI size of the area.

DB RECORD LENGTH

AVG

The average length of the database records.

MAX

The length of the longest database record.

MIN

The length of the shortest database record.

DB RECORD LENGTH RANGE (RANGE SIZE APPROX. 1/ 20 OF CI SIZE)

Database record distribution range, broken down into 20 ranges based on the CI size. The entry at that bottom (MORE THAN ...) is for database records that do not fit into these ranges.

NUMBER OF DB RECORDS THAT FIT IN THE RANGE

The number of database records in the range.

PCT OF NBR OF DB RECS

The percentage of database records in the range, and what percentage it makes of the total.

CUM PCT FROM SHORTEST

The cumulative value of PCT OF NBR OF DB RECS.

LENGTH RANGES IN WHICH AVG, MIN, MAX, MED, AND 1ST/3RD QUARTILES EXISTED

Data flag of database records is as follows:

AVG

Size range in which the average database record length lies.

MAX

Size range in which the maximum database record length lies.

MIX

Size range in which the minimum database record length lies.

MED

Size range in which the median of database record length lies.

1ST QUARTILE

Size range in which the top quartile of database record lies.

3RD QUARTILE

Size range in which the lower quartile of database record lies.

Area Tuning Summary report

The Area Tuning Summary report contains information about threshold exceptions, statistics, and DBD definitions of the current areas and the simulated areas.

This report is generated by the TUNE command, but it is not printed if one of the following options is specified:

- GLOBAL SCAN=YES
- GLOBAL SCAN=SITE_CREATE
- GLOBAL SCAN=SITE_REPORT

This report contains the following three sections:

Threshold Exceptions

This section lists the threshold exceptions that are detected in each area. The first field lists the exceptions that are detected in the current areas, and the second field lists the exceptions that are detected in the simulated areas.

The exceptions are detected based on the threshold values supplied through the THRESHOLD subcommand keywords. When no THRESHOLD subcommand keywords are specified, this section is not printed.

Area Statistics

This section summarizes the threshold value for each factor that is evaluated (supplied through the THRESHOLD subcommand keywords) and the statistics that correspond to the factor.

The threshold values are supplied through the THRESHOLD subcommand keywords. When no THRESHOLD subcommand keywords are specified, this section is not printed.

Database Definition and Database Record Distribution

This section shows the database definitions and the number of database records in each area. For each area, the first row provides information about the current area, and the second row provides information about the simulated area.

The following figure is an example of the Area Tuning Summary report.

DBD NAME: DEDBJN22
SIMULATION MODE: RESTRUCTURE (AREADEF subcommands are specified)
CURRENT RESOURCES:
- ACBLIB DDNAME: IMSACB
- RANDOMIZER : RMOD4
SIMULATED RESOURCES:
- ACBLIB DSNAME: IMSVS.ACBLIB1
- RANDOMIZER : RMOD5N (RMODDSN=IMSVS.TEST.RMODLIB)

Threshold Exceptions:

Exceptions from current areas

Area	Threshold	Description of Threshold Exception
DB22AR0	IOVFFS_PCT	The percentage of free space in IOVF fell below a threshold.
	RECIO_MAX	The maximum number of I/Os per DB record exceeded a threshold.

Exceptions from simulated areas

Area	Threshold	Description of Threshold Exception
*** No exception was detected ***		

Area Statistics:

*: Shows the point of threshold exception.

AREA NAME: DB22AR0

Keyword	Threshold Value	Current Value	Simulated Value	Difference
IOVFFS_PCT	90%	70%*	95%	+25
RECIO_MAX	6	8*	3	-5

AREA NAME: DB22AR01

Keyword	Threshold Value	Current Value	Simulated Value	Difference
IOVFFS_PCT	90%	-	100%	-
RECIO_MAX	6	-	2	-

AREA NAME: DB22AR1

Keyword	Threshold Value	Current Value	Simulated Value	Difference
IOVFFS_PCT	90%	100%	100%	0
RECIO_MAX	6	4	4	0

Database Definition and Database Record Distribution:

Upper row for each area shows the definition and number of database records of current areas
Lower row for each area shows the definition and number of database records of simulated areas

* shows the difference

AREA NAME	AREA NO.	CI-SIZE	UOW=	ROOT=	TOTAL CI'S				SDEP START		NUMBER OF DB RECORDS
					BASE	DOVF	IOVF	TOTAL	BLOCK#	RBA(HEX)	
DB22AR0	1	2,048	(24,4)	(160,40)	2,400	480	952	3,832	3,866	0078D000	4,122
DB22AR0	1	4,096*	(34,4)*	(160,40)	3,600	480	1,348	5,428	5,476	01564000	3,282
DB22AR01*	2*	1,024	(24,4)	(80,20)	1,200	240	476	1,916	1,946	001E6800	840
DB22AR1	2	2,048	(24,4)	(160,40)	2,400	480	952	3,832	3,866	0078D000	1,856
DB22AR1	3*	2,048	(24,4)	(160,40)	2,400	480	952	3,832	3,866	0078D000	1,856

Figure 275. Area Tuning Summary report

DBD NAME

The name of the DBD.

SIMULATION MODE

Simulation mode.

REORGANIZATION

Simulates a reorganization of the specified areas.

RESTRUCTURE

Simulates structure changes of specified areas.

If AREADEF subcommands are specified, AREADEF subcommands are specified is printed.

For more information about simulation modes, see "Simulating modes" on page 230.

CURRENT RESOURCES

The resources that are used for evaluating the current area status.

ACBLIB DDNAME

The name of the *acblib* DD statement.

When IMSCATHLQ=*bsdshlq* and IMSCATACB_INPUT=CURRENT is specified, IMS CATALOG (CURRENT) is printed in this field. When IMSCATHLQ=*bsdshlq* and IMSCATACB_INPUT=PENDING is specified, IMS CATALOG (PENDING) is printed in this field.

RANDOMIZER

The name of the randomizer.

SIMULATED RESOURCES

The resources that are used for evaluating the status of simulated area.

ACBLIB DSNAME

The name of the ACBLIB data set that is specified by the ACBLIB keyword or the *acblib* DD statement. The data set contains the modified DBD definition.

When IMSCATHLQ=*bsdshlq* and IMSCATACB_INPUT=CURRENT is specified, IMS CATALOG (CURRENT) is printed in this field. When IMSCATHLQ=*bsdshlq* and IMSCATACB_INPUT=PENDING is specified, IMS CATALOG (PENDING) is printed in this field.

RANDOMIZER

The name of the randomizer used in simulation.

RMODDSN

The name of the data set that contains the randomizer used in simulation. This information is printed only when the RMODLIB keyword is specified.

The SIMULATED RESOURCES field is not printed when the simulation mode is REORGANIZATION.

Threshold Exceptions section**Exceptions from current areas**

A list of the detected exceptions in the current areas. The exceptions are detected based on the threshold values supplied through the THRESHOLD subcommand keywords.

Exceptions from simulated areas

A list of the detected exceptions in the simulated areas. The exceptions are detected based on the threshold values supplied through the THRESHOLD subcommand keywords.

Area

The name of the area in which the threshold exceptions were detected.

Threshold

Keywords that are related to the threshold exceptions.

Description of Threshold Exception

Descriptions of the detected threshold exceptions.

For more information about exceptions, see [“Comparison of the exception detection methods”](#) on page 1120.

Area Statistics section**AREA NAME**

The area name.

Keyword

Keywords that are specified on the THRESHOLD subcommand as thresholds.

Threshold Value

Values specified by the THRESHOLD subcommand keywords as threshold values.

Current Value

Database statistics that are related to the threshold values obtained from the current areas. If a value exceeds the threshold, an asterisk (*) is printed next to the value.

Simulated Value

Database statistics that are related to the threshold values obtained from the simulated areas. If a value exceeds the threshold, an asterisk (*) is printed next to the value.

Difference

The difference between the values that are obtained from the current area and the simulated area.

Database Definition and Database Record Distribution section**AREA NAME**

The area name.

CI-SIZE

The CI size of the area.

UOW=

The UOW= parameter value for the area that is defined in the DBD.

ROOT=

The ROOT= parameter value for the area that is defined in the DBD.

TOTAL CI'S**BASE**

The total number of BASE CIs for the area that is defined in the DBD.

DOVF

The total number of DOVF CIs for the area that is defined in the DBD.

IOVF

The total number of IOVF CIs for the area that is defined in the DBD. IOVF bitmap CIs are not included.

TOTAL

The total number of CIs for the area that is defined in the DBD. The first, second, IOVF bitmap, work UOW, and SDEP part CIs are not included.

SDEP START**BLOCK#**

The first block number of CI in the SDEP part of the area.

RBA (HEX)

The hexadecimal value of the first CI in the SDEP part of the area.

NUMBER OF DB RECORDS

The number of database records in the area.

Tuning Aid Record Data Set report

The Tuning Aid Record Data Set report provides a list of the tuning aid record data sets.

This report is generated by the TUNE command, but it is not printed if one of the following options is specified:

- GLOBAL SCAN=YES
- GLOBAL SCAN=SITE_CREATE
- GLOBAL SCAN=SITE_REPORT

The following figure is an example of the Tuning Aid Record Data Set report when AREASCAN=YES is specified.

DBD NAME: DEDBJN30

OUTPUT FILES

AREANAME	DDNAME	DSNAME	ALLOCATION
DB30AR1	TAR00001	HPPF.TAR.DB30AR1	JCL DD
DB30AR2	TAR00002	HPPF.TAR.DB30AR2	JCL DD
DB30AR3	TAR00003	HPPF.TAR.DB30AR3	JCL DD

Figure 276. Tuning Aid Record Data Set report when AREASCAN=YES

The following figure is an example of the Tuning Aid Record Data Set report when AREASCAN=NO is specified.

DBD NAME: DEDBJN30

INPUT FILES

AREANAME	DDNAME	DSNAME	ALLOCATION
DB30AR1	TAR00001	HPPF.TAR.DB30AR1	DYNALLOC
DB30AR2	TAR00002	HPPF.TAR.DB30AR2	DYNALLOC
DB30AR3	TAR00003	HPPF.TAR.DB30AR3	DYNALLOC

Figure 277. Tuning Aid Record Data Set report when AREASCAN=NO

DBD NAME

The DBD name.

AREANAME

The area name.

DDNAME

The tuning aid record DD name for the area.

DSNAME

The name of the tuning aid record data set that is specified by the tuning aid record DD statement.

When AREASCAN=YES is specified and TARnnnnn DD, TARDSNMASK keyword, and FILECTL subcommand are not specified for an area, the tuning aid record is created in a temporary data set, which is deleted at the end of the job. In this case, "(TEMPORARY DATA SET)" is printed in the DSNAME field for the area.

ALLOCATION

One of the following values is printed:

JCL DD

The tuning aid record data set is specified by the TARnnnnn DD statement in the JCL.

DYNALLOC

The tuning aid record data set is allocated dynamically by the TARDSNMASK keyword or the FILECTL subcommand.

TUNED reports

The FPA Tune function generates reports that begin with TUNED in the title. The contents of the reports are the same as the reports generated by other functions of FPA except that these TUNED reports contain information that is obtained from simulated areas.

The following reports can be generated by the Tune function:

TUNED Consolidated Largest Database Records report

Use this report to analyze the largest database records in the simulated areas. The report provides information about each of the large database records, starting with the largest database record.

To generate this report, specify the TUNE command with the following keywords on the REPORT subcommand:

- LARGEREC_REPORT=YES
- LARGEREC_COUNT=(*parameter1,parameter2*)

For *parameter2*, specify a value that is equal to or larger than 1.

For an example and contents of the Consolidated Largest Database Records report, see [“Consolidated Largest Database Records report”](#) on page 604.

TUNED DB Record Length Distribution report

The optional report for database record analysis provides database record distribution broken down into 20 CI size ranges.

To generate this report, specify the TUNE command with the DBRLDIST=YES keyword on the REPORT subcommand.

For an example and contents of the DB Record Length Distribution report, see [“DB Record Length Distribution report”](#) on page 647.

TUNED DEDB Area Analysis reports

The TUNED DEDB Area Analysis reports include the comprehensive analysis information obtained from the simulated area.

The TUNED DEDB Area Analysis reports include the following reports:

- Freespace Analysis report
- DB Record Profile Analysis report
- DB Record Placement Analysis report
- Segment Placement Analysis report
- Segment I/O Analysis report
- Synonym Chain Analysis report

These reports are always generated if the TUNE command is specified.

In the Freespace Analysis report of the TUNED DEDB Area Analysis reports (generated for the simulated areas), fields PTR. TO DOVF and PTR. TO IOVF are different from the fields in the Freespace Analysis report of the DEDB Area Analysis reports (generated for the current areas). In the report that is generated for the simulated areas, the fields display the number of PTF pointers and PCF pointers that point to the DOVF and IOVF sections, whereas the fields in the report that is generated for the current areas display the number of PTR pointers, PCF pointers, PCL pointers, and SSP pointers that point to the DOVF and IOVF sections.

For examples and contents of these reports, see [“DEDB Area Analysis reports”](#) on page 591.

TUNED Largest Database Records report

Use this report to analyze the largest database records. This report provides a description for each of the large database records, starting with the largest database record, obtained from the simulated area.

To generate this report, specify the TUNE command with the LARGEREC_REPORT=YES keyword on the REPORT subcommand.

For an example and contents of the Largest Database Records report, see [“Largest Database Records report”](#) on page 603.

TUNED Process Summary of Analysis report

The Process Summary of Analysis report contains the processing information of the analysis report. An echo of the control statement that you specified in the HFPAPARM and some processing messages are issued in this report.

This report is always generated if the TUNE command is specified.

For an example and contents of the Process Summary of Analysis report, see [“Process Summary report” on page 636](#).

TUNED Segment Length Distribution report

This report provides detailed information about each segment in the simulated area. You can use this report to analyze the segments. The Segment Length Distribution report can be used to:

- List detailed information about each segment
- Identify which segment requires tuning

To generate this report, specify the TUNE command with the SEGLDIST=YES keyword on the REPORT subcommand.

For an example and contents of the Segment Length Distribution report, see [“Segment Length Distribution report” on page 645](#).

TUNED UOW report

This report provides detailed information about each UOW in the simulated area.

You can use this report to:

- List detailed information about each UOW
- Identify which UOWs require reorganization
- Manage space utilization when the randomizer has been modified to group related data into contiguous UOWs
- List only those UOWs that violate the space utilization criteria that you specified

This report is always generated for the TUNE command unless the UOW_REPORT=NO keyword is specified on the REPORT subcommand.

For an example and contents of the UOW report, see [“UOW report” on page 606](#).

TUNED UOW Exception report

This report contains the analysis information of the UOW that exceeds the user-specified reporting parameters in the simulated area.

You can use this report to:

- List detailed information about each UOW
- Identify which UOWs require reorganization
- Manage space utilization when the randomizer has been modified to group related data into contiguous UOWs
- List only those UOWs that violate the user-specified space utilization criteria

This report is generated by the TUNE command and by the reporting parameters that you specify on the HFPAPARM DD statement.

For an example and contents of the UOW Exception report, see [“UOW Exception report” on page 607](#).

RBA of Used IOVF CI report

This optional report for IOVF analysis provides a list of RBAs of used IOVF CIs for each UOW.

The RBA of Used IOVF CI report can be used to:

- Identify UOWs that use IOVF (caused by data overflow in RAP CIs and DOVF)
- Identify the RBAs of used IOVF CIs

This report is generated by the following commands if the IOVF CI_RBA_REPORT=YES keyword is specified for the REPORT subcommand and either the PTRCHKLVL=FULL or the PTRCHKLVL=QUICK keyword is specified:

- ANALYZE
- CHANGE
- RELOAD
- REORG

The following figure is an example of the RBA of Used IOVF CI report.

IMS HFPF UTILITIES - FPA ANALYZE		"RBA OF USED IOVF CI REPORT"								PAGE: 1	
5698-FPP V2R1										2019-10-12 20:10:50	
DBDNAME: DEDB0005		AREA NAME : DB05AR3	DDNAME: DB05AR3	DSNAME : IMSDEV.FPT.TS5759.IMS14.DB05AR3.ADS1							
		AREA NUMBER: 3	DSTYPE: ADS								
UOW NO.	RBA OF IOVF CIs										
0	0002AE00	0002B000	0002B200	0002B400	0002B600	0004E800	0004EA00	0004EC00	0004EE00		
1	0000F600	0000F800	0000FA00	0000FC00	0000FE00	00048E00	00049000	00049200	00049400	00049600	
2	0001BE00	0001C000	0001C200	0001C400	0001C600	0001C800	0001CA00	0001CC00	0001CE00	0001D000	
	0001D200	0001D400	0001D600	0001D800	0001DA00	0001DC00	0001DE00	0001E000	0001E200	0001E400	
	0001E600	0001E800	0001EA00	0001EC00	0001EE00	0001F000	0001F200	0001F400	0001F600	0001F800	
	0001FA00	0001FC00	0001FE00	00020000	00020200	00020400	00020600	00020800	00020A00	00020C00	
	00020E00	00021000	00021200	00021400	00021600	00021800	00021A00	00021C00	00021E00	00022000	
	00022200	00022400	00022600	00022800	00022A00	00022C00	00022E00	00023000	00023200	00023400	
	0002B800	0002BA00	0002BC00	0002BE00	0002C000	0002C200	0002C400	0002C600	0002C800	0002CA00	
	0002CC00	0002CE00	0002D000	0002D200	0002D400	0002D600	0002D800	0002DA00	0002DC00	0002DE00	
	0004F000	0004F200	0004F400	0004F600	0004F800	0004FA00	0004FC00	0004FE00	00050000	00050200	
	00050400	00050600	00050800	00050A00	00050C00	00050E00	00051000	00051200	00051400	00051600	
3	0002E000	0002E200	0002E400	0002E600	0002E800	0002EA00	0002EC00	0002EE00	0002F000	0002F200	
	0002F400	0002F600	0002F800	0002FA00	0002FC00	0002FE00	00030000	00030200	00030400	00030600	
	00030800	00030A00	00030C00	00030E00	00031000	00031200	00031400	00031600	00031800	00031A00	
	00031C00	00031E00	00032000	00032200	00032400	00032600	00032800	00032A00	00032C00	00032E00	
	00033000	00033200	00033400	00033600	00033800	00033A00	00033C00	00033E00	00034000	00034200	
	00034400	00034600	00034800	00034A00	00034C00	00034E00	00035000	00035200	00035400	00035600	
	00035800	00035A00	00035C00	00035E00	00036000	00036200	00036400	00036600	00036800	00036A00	
	00036C00	00036E00	00037000	00037200	00037400	00037600	00037800	00037A00	00037C00	00037E00	
	00039400	00039600	00039800								
4	0000CE00	0000D000	0000D200	0000D400	0000D600	0000D800	0000DA00	0000DC00	0000DE00	0000E000	
	0000E200	0000E400	0000E600	0000E800	0000EA00	0000EC00	0000EE00	0000F000	0000F200	0000F400	
	00038000	00038200	00038400	00038600	00038800	00038A00	00038C00	00038E00	00039000	00039200	
	00039E00	0003A000	0003A200	0003A400	0003A600	0003A800	0003AA00	0003AC00	0003AE00	0003B000	
	0003B200	0003B400	0003B600	0003B800	0003BA00	0003BC00	0003BE00	0003C000	0003C200	0003C400	
	00049800	00049A00	00049C00	00049E00	0004A000	0004A200	0004A400	0004A600	0004A800	0004AA00	
	0004AC00	0004AE00	0004B000	0004B200	0004B400	0004B600	0004B800	0004BA00	0004BC00	0004BE00	
	0004C000	0004C200	0004C400	0004C600	0004C800	0004CA00	0004CC00	0004CE00	0004D000	0004D200	
	0004D400	0004D600	0004D800	0004DA00	0004DC00	0004DE00	0004E000	0004E200	0004E400	0004E600	

Figure 278. RBA OF USED OF Used IOVF CI report

UOW NO.

The number of the unit of work (UOW).

RBA OF IOVF CIs

The RBAs of used IOVF CIs.

Chapter 18. Site default options support for FPA

FPA provides the site default options to override the system default values. By using the site default options, you can omit specific values in the control statements in the HFPSYSIN DD data set.

Topics:

- [“Supported keywords for the site default option” on page 657](#)
- [“Enabling site default values” on page 658](#)

Supported keywords for the site default option

You can specify various keywords for your site default.

The following table shows the keywords that can be specified in the site default module or the site default table.

Table 75. Keywords for site default option

Command/ Subcommand	Keywords
ANALYZE	ATASKCTL, CISNAP, DOVFNULPTR_MSGLVL, ICACHE, ICHECK, ICINCOMP, ICNOTIFY, ICOUT, IDX_DUMP, IDX_MAXERROR, IDXTASKCTL, KEYSEQCHK, MAXERROR, PTRCHKLVL, SDEPCHKLVL, SDEP_ORPHAN_MSGLVL, SDEP_PTRIOVF_MSGLVL, SENSOR, SENSOR_DBREC, SENSOR_PTRERR, SENSOR_UOW, SENSOR_UOW_REPORT, SORTOPT_FILSZ
CHANGE	CISNAP, DBREC_DISCARD_MSGLVL, EDITCOMP_SKIP, ICACHE, ICHECK, ICINCOMP, ICNOTIFY, ICOUT, ITASKCTL, KEYCHGCHK, KEYSEQCHK, MAXERROR, OCACHE, OTASKCTL, PAD, PTRCHK, PTRCHKLVL, RELOCATE_SDEP, SDEP_BYPASS_MSGLVL, SDEPCHKLVL, SDEP_ORPHAN_MSGLVL, SENSOR, SENSOR_DBREC, SENSOR_PTRERR, SENSOR_UOW, SENSOR_UOW_REPORT, SORTOPT_FILSZ, SSP
EXTRACT	COMPRESS, EFORMAT, ICACHE, ICHECK, ICINCOMP, ITASKCTL, KEYSEQCHK, OCHECK, PTRCHK, SORT, SORTOPT_FILSZ
GLOBAL	ADXCFGRP, AREADSN, IMSCATACB_INPUT, IMSCATHLQ, ITKBDATA, ITKBLOAD, ITKBSRVR, MSTIMESTAMP, LIUGOPT, RESTART, RESTART_COUNT, SYNC, TOINUM, TOITIME, TOIXCFGRP
INDEXBLD	AREASCAN, BUILDMODE, DUPKEYACT, DUPKEYMAX, FS_RECLAIM_REPORT, ICACHE, ICHECK, ITASKCTL, ICINCOMP, IDXPROC, ITASKCTL, OTASKCTL, PROC_AFT_DUPKEY, RESYNCDUMP, RESYNCDUMP_MAX, RESYNCMODE, SORT, TOICTL
INITIALIZE	ALLOCONLY, OCHECK, OTASKCTL
RELOAD	ACBCHK, CISNAP, ICHECK, ICNOTIFY, ICOUT, ITASKCTL, KEYCHGCHK, KEYSEQCHK, MAXERROR, OCACHE, OCHECK, PTRCHKLVL, SDEPCHKLVL, SDEP_ORPHAN_MSGLVL, SENSOR, SENSOR_DBREC, SENSOR_PTRERR, SENSOR_UOW, SENSOR_UOW_REPORT, SORT, SORTOPT_FILSZ, SSP
REORG	CISNAP, ICACHE, ICNOTIFY, ICOUT, ITASKCTL, MAXFAILS, PTRCHKLVL, RBASEFS, RDOVFFS, RECOVPT, REORGLIM, RFSSCAN, RFSSTAT, SDEP_ORPHAN_MSGLVL, SDEP_PTRIOVF_MSGLVL, SDEPCHKLVL, SENSOR, SENSOR_DBREC, SENSOR_PTRERR, SENSOR_UOW, SENSOR_UOW_REPORT, SORTOPT_FILSZ
TUNE	AREASCAN, ICACHE, ICHECK, ICINCOMP, ITASKCTL, MAXERROR, OTASKCTL, SORTOPT_FILSZ

Table 75. Keywords for site default option (continued)

Command/ Subcommand	Keywords
UNLOAD	ACBCHK, AREA_INFORMATION_RECORD, CISNAP, COMPRESS, DBREC_DISCARD_MSGlvl, DOVFNuLLPTR_MSGlvl, EDITCOMP_SKIP, FORMAT, ICACHE, ICHECK, ICINCOMP, ICNOTIFY, ICOUT, ITASKCTL, KEYCHGCHK, KEYSEQCHK, LRECL, OCHECK, PAD, PTRCHK, RELOCATE_SDEP, SDEP_BYPASS_MSGlvl, SORT, SORTOPT_FILSZ, SSP
ALLOCATE	AVGREC
BLDMFILECTL	AVGREC
FILECTL	AVGREC
LOADCTL	INSERT_LIMIT_COUNT, LOCATION
REPORT	DBRLDIST, IOVFCI_RBA_REPORT, LARGEREC_COUNT, LARGEREC_REPORT, SEGLDIST, SEGLSTAT, STARTUOW, STOPUOW, UOW_REPORT

Enabling site default values

You can register the site default keyword parameter values either in a site default module (HFPCZZ01) or in a site default table.

Registering and using site defaults in a site default module (HFPCZZ01)

You can generate and use the site default module to specify your site default keyword parameter values.

About this task

The site default keyword parameter values are registered in a site default module as a PDS member. The sample JCL (HFPCZZMJ), which generates the site default module, is provided in the SHFPSAMP data set.

Notes:

- If you use the site default module (HFPCZZ01), the library that includes the site default module must be concatenated to IMS Fast Path Advanced Tool execution library (STEPLIB or JOBLIB).
- When the site default module is used, the site default table and the HFPCZZ00 module (migration module) are ignored.

You can register the site defaults keyword parameter values in the site default module by doing the following steps:

Procedure

1. Copy member HFPCZZMJ in your SHFPSAMP to your JCL library.
2. Edit member HFPCZZMJ in your JCL library according to your environment. Multiple commands except GLOBAL command can be specified in the HFPSYSIN DD.
3. Submit job HFPCZZMJ to generate the site default module (HFPCZZ01) in your library.

Example

The following figure shows example JCL statement to create a site default module.

```

//SITEMOD  PROC
//*
//HFPCRE  EXEC  PGM=HFPMAIN0
//STEPLIB DD  DISP=SHR,DSN=HPFP.SHFPLMD0
//HFPPRINT DD  SYSOUT=*
//HFPRPTS DD  SYSOUT=*
//SYSPUNCH DD  DISP=(NEW,PASS,DELETE),
//          SPACE=(TRK,(1,1)),UNIT=SYSDA,DSN=*&&SOURCE
//*
//ASM     EXEC  PGM=ASMA90,PARM='OBJECT,NODECK',COND=(4,LT,HFPCRE)
//SYSLIB  DD  DISP=SHR,DSN=SYS1.MACLIB
//SYSLIN  DD  DSN=*&&OBJLIB,DISP=(NEW,PASS,DELETE),
//          UNIT=SYSDA,SPACE=(TRK,(1,1))
//SYSPRINT DD  SYSOUT=*
//SYSUT1  DD  UNIT=SYSDA,SPACE=(TRK,(1,1))
//SYSIN   DD  DISP=(OLD,DELETE,DELETE),DSN=*&&SOURCE
//*
//LKED    EXEC  PGM=IEWL,COND=(4,LE,ASM)
//SYSPRINT DD  SYSOUT=*
//SYSLIN  DD  DSN=*&&OBJLIB,DISP=(OLD,DELETE,DELETE)
//SYSUT1  DD  UNIT=SYSDA,SPACE=(TRK,(1,1))
//SYSLMOD DD  DISP=OLD,DSN=HPFP.USERLIB(HFPCZZ01)
//*
//          PEND
//*

```

Figure 279. Creating a site default module (Part 1 of 2)

```

//STEP01 EXEC SITEMOD
//HFPCRE.HFPSYSIN DD *
ANALYZE
  CISNAP=YES,
  ICACHE=NO,
  PTRCHKLVL=FULL,
  SDEP_ORPHAN_MSGLVL=WARNING,
  SDEPCHKLVL=FULL
CHANGE
  CISNAP=NO,
  ICACHE=NO,
  KEYCHGCHK=NO,
  KEYSEQCHK=ABEND,
  OCACHE=NO,
  PAD=X'FF',
  PTRCHK=BYPASS,
  PTRCHKLVL=FULL,
  SDEP_ORPHAN_MSGLVL=WARNING,
  SDEPCHKLVL=NONE,
  SSP=NO
ALLOCATE
  VOLSER=(HFPVOL),
  SPACE=(TRK,15),
  AVGREC=K
LOADCTL
  INSERT_LIMIT_COUNT=5,
  LOCATION=DOVF
RELOAD
  CISNAP=YES,
  KEYCHGCHK=YES,
  KEYSEQCHK=ABEND,
  OCACHE=NO,
  PTRCHKLVL=FULL,
  SDEP_ORPHAN_MSGLVL=WARNING,
  SDEPCHKLVL=FULL,
  SORT=YES,
  SSP=YES
ALLOCATE
  VOLSER=(HFPVOL),
  SPACE=(TRK,15),
  AVGREC=B
LOADCTL
  INSERT_LIMIT_COUNT=0,
  LOCATION=DOVF
UNLOAD
  AREA_INFORMATION_RECORD=YES,
  COMPRESS=YES,
  FORMAT=TFMT,
  ICACHE=NO,
  KEYCHGCHK=NO,
  KEYSEQCHK=ABEND,
  PAD=X'FF',
  PTRCHK=BYPASS,
  SORT=NO,
  SSP=NO
FILECTL
  DDNAME=OAREA001,
  OAREA=ALL,
  AVGREC=K
/*
/**

```

Figure 280. Creating a site default module (Part 2 of 2)

Related concepts

[Compatibility between IMS Fast Path Advanced Tool and former releases of IMS Fast Path Advanced Tool](#)This topic describes how to prepare the HFPCZZ00 module (migration module).

Registering and using site defaults in a site default table

You can register the site default keyword parameter values in a site default table and use the site default table when running FPA.

About this task

The site default keyword parameter values are registered in a site default table as a PDS member. You can specify an appropriate site default table when running FPA.

The site default options provide the following functions regarding the site default tables:

- Creates a site default table
- Runs the JCL using the site default table
- Reports the site default table

You can also prepare an HFPCSITE module that specifies the default data set name and the default member name to determine the site default table. In this way, you do not need to modify the existing JCL.

Related concepts

[Compatibility between IMS Fast Path Advanced Tool and former releases of IMS Fast Path Advanced Tool](#)This topic describes how to prepare the HFPCZZ00 module (migration module).

Creating a site default table

Do the following steps to create a site default table.

Procedure

1. To create a site default table, specify the following keywords with the GLOBAL command, and other necessary commands, subcommands, and keywords to define the site default values for the keywords.
 - SCAN=SITE_CREATE
 - SITE_DSNAME=*dsname*
 - SITE_MEMBER=*member*

You can create a site default table only by adding these keywords to the active JCL to run. You can also specify multiple commands in the control statement. When multiple subcommands are specified, the last subcommand is used.

Notes:

- Multiple commands can be specified in the HFPSYSIN DD for SCAN=SITE_CREATE.
 - When the values are the same as the system default values, they are not used and '*' is shown for the keyword in the Site and System Default report.
2. Run the JCL.

The site default table is created into the specified partition data set.

Example

The following figure shows example JCL statement to create a site default table.

```

//*****
//** PREPARE: DEL/DEF SITE DEFAULT TABLE DATASET **
//*****
//DELDS1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
DELETE ( HPFP.SITEDEF ) NONVSAM PURGE
/*
//*
//ALLOC EXEC PGM=IEFBR14
//SITE DD UNIT=SYSDA,DISP=(NEW,CATLG,CATLG),
// SPACE=(TRK,(10,1,10)),
// DCB=(RECFM=FB,LRECL=4096,BLKSIZE=4096),
// DSN=HPFP.SITEDEF
//*
//*****
//** CREATE SITE DEFAULT TABLE **
//*****
//CREATE EXEC PGM=HFPMAIN0
//*
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
//IMSACB DD DISP=SHR,DSN=IMSVS.ACBLIB
//HFPPRINT DD SYSOUT=A
//HFPRPTS DD SYSOUT=A
//*
//HFPSYSIN DD *
GLOBAL
SCAN=SITE_CREATE,
SITE_DSNAME='HPFP.SITEDEF',
SITE_MEMBER=SITEAL
ANALYZE
CISNAP=YES,
ICACHE=NO,
PTRCHKLVL=FULL,
SDEP_ORPHAN_MSGLVL=WARNING,
SDEPCHKLVL=FULL,
DBD=IVPDB3
CHANGE
CISNAP=NO,
ICACHE=NO,
KEYCHGCHK=NO,
KEYSEQCHK=ABEND,
OCACHE=NO,
PAD=X'FF',
PTRCHK=BYPASS,
PTRCHKLVL=FULL,
SDEP_ORPHAN_MSGLVL=WARNING,
SDEPCHKLVL=NONE,
SSP=NO,
DBD=IVPDB3

```

Figure 281. Creating a site default table (Part 1 of 2)

```

ALLOCATE
  VOLSER=(EASY02),
  SPACE=(TRK,15),
  AVGREC=K
LOADCTL
  SEGMENT=DD1,
  INSERT_LIMIT_COUNT=5,
  LOCATION=DOVF
RELOAD
  CISNAP=YES,
  KEYCHGCHK=YES,
  KEYSEQCHK=ABEND,
  OCACHE=NO,
  PTRCHKLVL=FULL,
  SDEP_ORPHAN_MSGLVL=WARNING,
  SDEPCHKLVL=FULL,
  SORT=YES,
  SSP=YES,
  DBD=IVPDB3
ALLOCATE
  VOLSER=(EASY02),
  SPACE=(TRK,15),
  AVGREC=B
LOADCTL
  SEGMENT=DD1,
  INSERT_LIMIT_COUNT=0,
  LOCATION=DOVF
UNLOAD
  AREA_INFORMATION_RECORD=YES,
  COMPRESS=YES,
  FORMAT=TFMT,
  ICACHE=NO,
  KEYCHGCHK=NO,
  KEYSEQCHK=ABEND,
  PAD=X'FF',
  PTRCHK=BYPASS,
  SORT=NO,
  SSP=NO,
  DBD=IVPDB3
FILECTL
  DDNAME=0AREA001,
  OAREA=ALL,
  AVGREC=K
/*
/**

```

Figure 282. Creating a site default table (Part 2 of 2)

Running a job with a site default table

Do the following steps to run FPA processes by using a site default table.

Procedure

1. Specify the following keywords with the GLOBAL command.
 - SITE_DSNAME=*dsname*
 - SITE_MEMBER=*member*
2. Run the JCL.

Example

The following figure shows example JCL statement to use the site default table.

```

//*****
//** USE SITE DEFAULT TABLE      **
//*****
//USE      EXEC PGM=HFPMAIN0
//*
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//HFPPRINT DD SYSOUT=A
//HFPRPTS DD SYSOUT=A
//*
//HFPSYSIN DD *
GLOBAL
        DBRC=NO,
        SITE_DSNAME='HPFP.SITEDEF',
        SITE_MEMBER=SITEAL
CHANGE
        DBD=IVPDB3
LOADCTL
        SEGMENT=DD1
/*
//IAREA001 DD DISP=OLD,DSN=IMSVS.OLD.DFSIVD31
//IAREA002 DD DISP=OLD,DSN=IMSVS.OLD.DFSIVD33
//DFSIVD3A DD DISP=OLD,DSN=IMSVS.DFSIVD31
//DFSIVD3B DD DISP=OLD,DSN=IMSVS.DFSIVD33
//

```

Figure 283. Running a JCL job with a site default table

Reporting the site default table

Do the following steps to report contents of a site default table.

Procedure

1. Specify the following keywords with the GLOBAL command.
 - SCAN=SITE_REPORT
 - SITE_DSNAME='dsname'
 - SITE_MEMBER=member
2. Run the JCL.

Example

The following figure shows example JCL statement to report the site default table.

```

//*****
//** REPORT FOR SITE DEFAULT TABLE **
//*****
//REPORT   EXEC PGM=HFPMAIN0
//*
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//HFPPRINT DD SYSOUT=A
//HFPRPTS DD SYSOUT=A
//*
//HFPSYSIN DD *
GLOBAL
        SCAN=SITE_REPORT,
        SITE_DSNAME='HPFP.SITEDEF',
        SITE_MEMBER=SITEAL
/*
//

```

Figure 284. Reporting a site default table

Specifying the site default table

There are several methods to specify the site default table.

About this task

You can specify the site default table with one or more of the following methods. The data set name and member name have the following order of priority; that is, a value in a higher position in the list overrides the one in a lower position.

1. HFPSYSIN DD control card
 - Specify the SITE_DSNAME keyword for the data set name
 - Specify the SITE_MEMBER keyword for the member name
2. HFPCSITE on the JCL
 - Specify the data set name
3. HFPCSITE module
 - Include the default data set name
 - Include the default member name

Note: When the site default options are used, the HFPCZZ00 module (migration module) is ignored.

To specify with the HFPCSITE module:

FPA provides the HFPCSITE module to define the site default values for the SITE_DSNAME and the SITE_MEMBER keyword parameters. By these default values, the default site default table is produced. In this way, you can use the site default table without modifying the existing JCL.

The HFPCSITE module is generated by using the sample HFPCSITE JCL in the sample JCL distribution library (usually HPFP.SHFPSAMP).

Procedure

1. Specify the data set name and the member name in the HFPCSITE module.
2. Concatenate the library that includes the HFPCSITE module to the STEPLIB or the JOBLIB library.

Note: If the site default table that is specified in the HFPCSITE module is not found, message HFPC0027I is issued and the processing continues.

Example

The following figure shows example JCL statement to register the site default values of the SITE_DSNAME and the SITE_MEMBER parameters in the HFPCSITE member.

```
//*-----*
//* ASSEMBLE & LINK HFPCSITE MEMBER *
//*-----*
//STEP01 EXEC ASMACL
//C.SYSLIB DD DISP=SHR,DSN=HPFP.SHFPMA0
// DD DISP=SHR,DSN=SYS1.MACLIB
//C.SYSIN DD *
*****
* <<< SITE DEFAULT TABLE DEFINITION >>> *
*****
* HFPCSITE DSNAME=HPFP.SITEDEF, X
* MEMBER=SITEDEF
*
* END
/*
//L.SYSLMOD DD DISP=OLD,DSN=HPFP.SHFPLM0(HFPCSITE)
//
```

Figure 285. Registering site default values of the SITE_DSNAME and the SITE_MEMBER parameters

Part 3. IMS Fast Path Online Tools

The following topics introduce IMS Fast Path Online Tools and explain how to use the functions that are provided by the tools.

Topics:

- [Chapter 19, “Overview of IMS Fast Path Online Tools,” on page 669](#)
- [Chapter 20, “Analyzing a DEDB online with FPO,” on page 673](#)
- [Chapter 21, “Extracting a DEDB online with FPO,” on page 729](#)
- [Chapter 22, “Extending a DEDB online with FPO,” on page 747](#)
- [Chapter 23, “Reorganizing a DEDB online with FPO,” on page 779](#)
- [Chapter 24, “Printing DMAC blocks in a DEDB online with FPO,” on page 821](#)
- [Chapter 25, “Keyword reference for FPO,” on page 835](#)
- [Chapter 26, “FPO reports stored in IMS Tools KB,” on page 941](#)
- [Chapter 27, “Site default options support for FPO,” on page 943](#)
- [Chapter 28, “Return code user exit routine \(FPXRCEX0\),” on page 947](#)
- [Chapter 29, “RACF considerations for FPO,” on page 951](#)

Chapter 19. Overview of IMS Fast Path Online Tools

These topics provide you with an overview of IMS Fast Path Online Tools (also referred to as FPO).

Topics:

- [“Functions of FPO” on page 669](#)
- [“Benefits of FPO” on page 669](#)
- [“Program overview of FPO” on page 671](#)
- [“IMS Fast Path Online Tools compatibility” on page 672](#)

Functions of FPO

The online utilities of IMS Fast Path Online Tools are designed to assist in the tasks of monitoring and managing the databases.

IMS Fast Path DEDB Online Pointer Checker

The IMS Fast Path DEDB Online Pointer Checker utility (also referred to as Online Pointer Checker or OPC) is a DEDB integrity verification program. It gets access to online database areas, verifies pointers, and produces various analysis reports.

IMS Fast Path DEDB Online Data Extract

The IMS Fast Path DEDB Online Data Extract utility (also referred to as Online Data Extract or ODE) extracts segment data from an online DEDB area and writes the output to a sequential file.

IMS Fast Path DEDB Online Area Extender

The IMS Fast Path DEDB Online Area Extender utility (also referred to as Online Area Extender or OAE) extends the size of the DEDB area, without taking the DEDB area offline.

IMS Fast Path DEDB Online Expert Reorganization

The IMS Fast Path DEDB Online Expert Reorganization utility (also referred to as Online Expert Reorganization or OER) reorganizes a set of UOWs in a DEDB area on the basis of a ranked set of UOWs.

IMS Fast Path DEDB Online DMAC Print

The IMS Fast Path DEDB Online DMAC Print utility (also referred to as Online DMAC Print or ODM) provides maps for the storage that is used by the IMS control region by using the IMS mapping macro and the IMS storage dump of DMAC.

Benefits of FPO

IMS Fast Path Online Tools offers a number of benefits for those involved in database management, maintenance, and performance tuning.

Subsections:

- [“Benefits of Online Pointer Checker” on page 669](#)
- [“Benefits of Online Data Extract” on page 670](#)
- [“Benefits of Online Area Extender” on page 670](#)
- [“Benefits of Online Expert Reorganization” on page 671](#)
- [“Benefits of Online DMAC Print” on page 671](#)

Benefits of Online Pointer Checker

- Performs regular monitoring and pointer error diagnosis while the area is *online* and being read and updated.

- Verifies the complete integrity of all IMS pointer chains, FSE chains, VSAM control fields, and space utilization.
- Provides a special operating mode for detecting pointer integrity errors in a fast and efficient manner.
- Produces a comprehensive set of reports, which are organized by DEDB area, to give multiple views of space utilization, record and segment profiles and placement, and physical I/O requirements.
- Produces the optional CI SNAP Dump as a result of processing, which enables you to analyze the CIs containing errors and to determine the best way to repair them without performing another run to obtain such information.
- Provides optional automated threshold reporting, which causes warning messages to be generated, and a special return code to be set when conditions within a database do not meet user-specified criteria. This feature can be used to show when reorganization, expansion, or analysis is required, or when database performance is likely to fall below acceptable levels.
- Provides optional generation of a report identifying the UOWs that exceed a user-specified reorganization criteria. Identifying and reorganizing specific UOWs can significantly reduce the time and resources required for database maintenance.
- Provides optional generation of a history file containing the key space utilization and performance values for each area. This file lets you carry out statistical and trend analysis using packaged software or user-written programs.
- Creates an image copy of the database.

Benefits of Online Data Extract

- Provides an easy-to-use, flexible tool to extract data from a DEDB area while the area is online.
- Lets you specify which segments are to be extracted by comparing data at an offset into a segment against a specified value.
- Allows more than one comparison per segment type.
- Gives you control over the amount of data you get back by letting you:
 - Skip a number of segment occurrences.
 - Stop selection after a number of segment occurrences.
 - Select every Nth occurrence.
- Lets you extract SDEP segments in standard format, IMS DBT Unload/Reload format, and user-specified format.
- Lets you extract SDEP segment records immediately after the ROOT record and each record that contains the ROOT key.
- Lets you write a subset of a segment's data to the output file.
- Provides an option to write output data in multiple formats compatible with the Unload/Reload formats.
- Provides an option to write records in a format suitable for sort.
- Lets you invoke the utility multiple times, for different areas, in a single run.
- Ensures a consistent view of the data by using appropriate locking protocols.
- Uses standard IMS data access techniques, which ensure ODE functions correctly with all DEDB facilities, including data sharing, MADS, VSO, and shared VSO, EQE.

Benefits of Online Area Extender

- Provides an easy-to-use, flexible tool to extend a DEDB area while the area remains online.
- Lets you specify which portion of the DEDB you want to extend: IOVF, SDEP, or both.
- Lets you specify which target area data sets you want to extend.

- Uses standard IMS data access techniques, which ensure OAE functions correctly with all DEDB facilities, including data sharing, MADS, EQE, and IMS V7 EEQE support.
- Lets you restart the utility, if needed.
- Supports the following two types to extend the ADS:

Shadow expansion

Enables you to specify the target area data sets that you want to expand. Original area data set is made UNAVAIL status with DBRC at the end of OAE.

Original expansion

Expands the original area data set.

Benefits of Online Expert Reorganization

- Reorganizes a set of UOWs in a DEDB area on the basis of a ranked set of UOWs.
- Includes the UOW Evaluation phase where the reorganization free space score (RFS score) for a set of UOWs of the DEDB area is determined, and the UOW Ranking phase where the set of UOWs is ranked according to the RFS score.
- Reduces the possibility of having too many or too few UOWs reorganized by ranking the set of UOWs in the DEDB area, and by allowing the user to limit the size of the subset of UOWs that will be reorganized.
- Runs OPC after the subset of UOWs have been reorganized to produce reports that contain the space statistics of each UOW that has been reorganized.

Benefits of Online DMAC Print

- Provides maps for storage that is used by the IMS control region by using the IMS mapping macro, and the IMS storage dump of DMAC (Data Management Area Control block).
- Provides the maps by using the IMS mapping macro, the dump, and the EQE list of the second CI that are obtained from all the available area data sets (ADS).

Program overview of FPO

IMS Fast Path Online Tools runs as IMS IFP utilities, using standard JCL and control statements, plus one or more additional files unique to the utility.

FPO can be used for analyzing, reorganizing, extending, and extracting the online DEDBs. It also provides the following functions:

- Reorganizes an online DEDB area and analyzes the reorganized area in only one step.
- Analyzes an online DEDB area and generates the image copies in one step.

FPO offers the following features:

- By specifying control statements, a single program can run multiple utilities.
- The following input and output data sets can be allocated dynamically to minimize JCL DD statements:
 - Output data set for the messages
 - Output data set for the reports
 - Data sets for freespace and area information records, segment information records, and pointer information records, which are the output of OPC
 - SORT work data sets
- Site default overrides system default values and saves the effort of specifying keywords in FPXIN DD.

Note: FPO cannot process the DEDBs that were started with the /START DB ACCESS=RO command or the /START DB ACCESS=RD command because FPO jobs update the DMAC.

IMS Fast Path Online Tools compatibility

The following topics describe the compatibilities of the tools within IMS HP Fast Path Utilities.

Compatibility between former releases of IMS Fast Path Online Tools

The compatibility between IMS Fast Path Online Tools (FPO) and the former releases of FPO is described in this topic.

- The JCL streams and the input control statements that are used in the former release are compatible with this release of FPO.
- Site default options that are used in the former releases of the product can be used in this release of the product. However, if the control statements or the keywords that are introduced in this release are used as site default options, those site default options cannot be used in the former releases.
- Return code user exit routine is compatible.

Compatibility between IMS Fast Path Advanced Tool and IMS Fast Path Online Tools

There are compatibilities and incompatibilities between the IMS Fast Path Advanced Tool (FPA) functions and the IMS Fast Path Online Tools (FPO) utilities.

Product material

The following items are compatible:

- The Extracted records data set that is created by the FPA Extract function and the FPO ODE utility.
- The History and History 2 file records that are created by the FPA Analyze function and the FPO OPC utility.
- The sensor data that is collected from DEDB areas and that is stored in the IMS Tools KB Sensor Data repository by the FPA Analyze function and the FPO OPC utility.
- The tuning aid record data sets that are created by the FPA Tune function, FPO OPC utility, and the FPO OER utility.

DD statements

DD statements are not compatible.

Utility control statements

The utility control statements, keywords, and parameters are not compatible.

Chapter 20. Analyzing a DEDB online with FPO

You can analyze an online DEDB by using the Online Pointer Checker utility (also referred to as OPC) of FPO.

The three Pointer Checkers of IMS HP Fast Path Utilities

There are three distinct pointer checker products. Therefore, before getting started, it is important to read this topic in order to avoid confusion later on.

This topic describes the relationship among the following three Pointer Checkers:

- Analyze function of FPA
- DEDB Pointer Checker of FPB
- Online Pointer Checker of FPO

The Analyze function of FPA and DEDB Pointer Checker of FPB run in the offline environment and Online Pointer Checker of FPO runs in the online environment.

The Analyze function of FPA is part of the IMS HP Fast Path Utilities package. It has the function similar to that of DEDB Pointer Checker, which runs while the database area is offline. FPA verifies the integrity of all IMS pointer values and verifies the integrity of the pointer segments in secondary index databases. It also produces complete analysis reports in one step, whereas DEDB Pointer Checker runs the same functions in multiple steps.

DEDB Pointer Checker of FPB is also part of the IMS HP Fast Path Utilities package. It uses program FABADA1 to collect pointer verification data. This program must be run while the database area is offline. The collected data can then be passed to the pointer integrity checking programs and various report generating programs in this package.

Online Pointer Checker of FPO is also part of the IMS HP Fast Path Utilities package. It runs as a Fast Path utility to verify the integrity of all IMS pointer values and produce complete analysis reports in one step while the database area is online. However, you can use OPC to run a fast scan of database integrity and collect pointer verification data from an online DEDB area, then pass it to DEDB Pointer Checker of FPB to check cross-reference of pointers and produce complete analysis reports. This type reduces the online run time but must be run in multiple steps.

Topics:

- [“Functions of OPC” on page 674](#)
- [“Input and output for OPC” on page 677](#)
- [“Considerations for running OPC” on page 679](#)
- [“Running OPC” on page 682](#)
- [“Region size requirements for OPC” on page 685](#)
- [“DD statements for OPC” on page 685](#)
- [“Option control statement for OPC” on page 695](#)
- [“Utility control statements for OPC” on page 696](#)
- [“Output for OPC” on page 702](#)
- [“Examples for OPC” on page 720](#)

Related information

[Analyzing a DEDB offline with FPA](#)

You can analyze a DEDB by using the FPA Analyze function.

Functions of OPC

OPC provides a fast, efficient mechanism for verifying the integrity of all IMS pointer values, free space element chains, VSAM control fields, and the space utilization in IMS DEDBs. OPC uses as direct input an online DEDB database area. This means that the database area can remain fully accessible to users while pointer checking is being performed.

OPC provides the complete analysis reports to effectively manage the space utilization, performance characteristics, and physical attributes of IMS DEDBs. Together the reports are an invaluable aid for database administrators and technical support personnel in database management, maintenance, and performance tuning.

OPC can check the cross-reference of pointers and produce complete analysis reports in one step. OPC can also generate the files that are used as input for the report and analysis functions of the (batch) DEDB Pointer Checker utility (OPC replaces the FABADA1 program in DEDB Pointer Checker).

DEDB integrity verification

OPC provides extensive DEDB integrity verification.

Specifically, the following areas of integrity are addressed:

- All the following pointer values are verified to ensure that they refer to valid segments:
 - Root Anchor Point (RAP)
 - Physical Twin Forward (PTF)
 - Physical Child First (PCF)
 - Physical Child Last (PCL)
 - Subset (SSP)
 - Sequential Dependent (SDEP)

All orphan segments, invalid pointer values, and broken pointer chains are detected and reported. PCF/PCL/SSP pointer interdependency errors are also detected and reported.

- All free space element (FSE) chains are checked for validity.
- Space utilization is verified by totalizing the lengths of all segments, FSEs, and scraps within a CI, and ensuring that the calculated total equals the usable CI size (CI size minus VSAM and IMS overhead). This will detect FSEs within segments, or segments within FSEs.
- The VSAM CIDE and RDE fields are also verified to ensure that their values are correct. OPC will report the RBA of any invalid CIs encountered and continue processing.
- Scrap bytes are also validated.
- OPC provides the CI Dump report that can be used to determine the best way to repair incorrect CIs.

Methodology

OPC reads and serially deblocks each CI in the area. Serially deblocking a CI is carried out by starting at offset +8 and advancing through the CI based on the lengths of the items encountered. Valid items are segments, scraps, and FSEs. During the deblocking process, OPC accumulates the number and lengths of all segments, scraps, and FSEs. At end-of-CI, OPC chases the FSE chain and tallies the number and length of all FSEs. OPC then ensures that the number and length of FSEs encountered during the serial deblocking process matches the number found by following the FSE chain. This test finds segments within FSEs and FSEs within segments. OPC verifies CI integrity by ensuring that the sum of all segment, FSE, and scrap-lengths plus IMS and VSAM overhead equals CI size.

OPC also verifies that the VSAM control fields (CIDE, RDE) are correct. This provides a rapid way of identifying the exact location of corrupted values when the Image Copy utility ends abnormally because of incorrect VSAM control fields.

QUICK mode

OPC runs the integrity verification in the QUICK mode when TYPRUN=PTRSCAN is specified. The QUICK mode provides a fast scan of database integrity.

In addition to deblocking and analyzing CIs, and doing checksum analysis, QUICK mode attempts to navigate the database using the pointer chains. In most cases, this allows exact reporting of any detected errors, although there are still some cases, such as orphan segments, that are detected only by checksum logic and thus the exact error RBA might not be available.

The following errors are detected during navigation:

- pointer that is arithmetically incorrect (outside proper portion of area)
- pointer that does not point at correct segment code
- twin keys that are not ascending
- unmatched nonzero subset pointers
- PCL that does not point to end of twin chain
- Root -> SDEP pointer not pointing into SDEP part of area
- loop in twin chain
- pre version 6 format SDEP CI
- SDEP segment marked IN-DOUBT
- SDEP CI construction error - FSE that does not point after last segment in CI
- IOVF CI not marked owned by referencing RAA UOW
- CI space utilization discrepancies (that is, the total length of all segments, FSEs, and scraps plus IMS and VSAM overhead does not equal CI size)

The following errors are detected during PTRSCAN CI analysis:

- invalid VSAM CIDF/RDF
- invalid CI block type
- CI construction errors (FSE to segment overlap, etc.)

The following are detected during IOVF space map CI validation:

- SM CI block id not equal to IOVF space map
- CUSN bit not set in SM CI prefix
- Free chain next pointer too large or not divisible by 4
- Free chain element that does not start with X'80' (DBLKFREE)
- Loop in free chain
- Element with DBLKFREE set that is not in free chain (an orphan free element)
- Element that does not start with DBLKFREE or DBLKALLO (X'80' or X'40')
- Map marks CI as FREE but target CI prefix is not DBLKFREE
- Map marks CI as ALLO but target CI prefix is not DBLKALLO

OPC also does a checksum verification of the following pointers versus the appropriate segment RBAs:

- PCF/PTF versus appropriate segment RBAs
- PCL versus appropriate segment RBAs
- SDEP pointer values (between LB and LE)

The checksum verification process can detect the presence of invalid pointers but cannot report the exact RBA of the problem. Checksum verification cannot be done for subset pointers.

FULL mode

OPC runs the integrity verification in the FULL mode when TYPRUN=PTRALL is specified. The FULL mode provides in-depth pointer verification. Besides the QUICK mode integrity verification functions, OPC generates pointer and segment information entities for each pointer and segment to carry out in-depth pointer analysis through a matching process. This process reports the exact RBAs of the following pointer problems:

- Orphan segments (that is, segments that are not pointed to by any correct pointer)
- Pointers that reference nonexistent segments
- Pointers to wrong segment types (that is, OPC verifies that the pointer points to the correct segment type)
- Segments pointed to by multiple parents
- Segments that cannot be resolved back to a correct RAP RBA

Subset pointer validation can be done only in this verification.

DEDB analysis and report

The complete analysis reports are provided to effectively manage the space utilization, performance characteristics, and physical attributes of IMS DEDBs. It provides integrity verification reports, and summary and detailed analysis reports for database administrators and technical support personnel in database management, maintenance, and performance tuning.

The statistical information includes:

- Free space analysis, including statistics on the number and size of the free space elements and inter-database section references.
- Database record and segment occurrence profiles.
- Database record and segment placement analysis.
- Complete overflow usage analysis of both dependent overflow (DOVF) and independent overflow (IOVF).
- Physical I/O statistics for each segment type and for the typical database record.
- Root distribution and synonym chain analysis.
- Optional automated threshold reporting, which generates warning messages when conditions within the database do not meet user-specified criteria. This feature can be used to show when reorganization or expansion is required, or if database performance falls below acceptable standards.
- Optional generation of:
 - A report that describes the largest database records.
 - A report that identifies the units of work (UOWs) that exceed a user-specified reorganization criteria.
 - The control statements that are required to run the IMS High-Speed DEDB Direct Reorganization utility (DBFUMDR0).

Reorganizing specific UOWs can significantly reduce the time and resources that are required for database maintenance.

Root addressable area and IOVF processing

In root addressable area and IOVF processing, OPC deblocks each CI into its component parts (that is, segments, scraps, and FSEs).

It does this by starting at offset 8 and continuing until the end of CI, by incrementing the offset by the length of the item. During the deblocking process, the total lengths of all FSEs, scraps, and segments are accumulated.

After a CI is deblocked, the program chases the FSE chain and totals the number and length of FSEs. If this value differs from the value calculated during the deblocking process, a DEDB integrity problem has

been detected. This test finds FSEs within segments, or segments within FSEs. OPC also verifies that the total of segment lengths, plus total scrap and FSE lengths, plus VSAM and IMS overhead, equals CI size.

Three output files are created:

- Area and free space information
- Segment information for all segments that reside in the Root Addressable part of an area
- Segment information for segments in IOVF and SDEP, and pointer information

The record types generated are as follows:

Type

Purpose

1,0

Contains area definition information from the DMB (one record per area)

1,1

Contains segment definition information from the DMB (four segments per record)

2,F

Contains the length and location of each FSE or scrap encountered (one per FSE, no record generated for empty DOVF or IOVF CIs)

2,R

Contains a count of the number of references to other CIs in the area (one per CI that has external references)

2,T

Contains total free space in each of the component parts of the area (one record per area)

3,C

Contains the RBA of all nonzero pointers: subtype 0 = RAP pointer, 1 = PTF pointer, 2 = PCF pointer, 3 = PCL pointer, 4 = SSP pointer (one per nonzero pointer)

3,S

Contains the RBA and length of all segments (one per segment)

4,P

Contains the RBA of all nonzero SDEP pointers (one per nonzero SDEP pointer)

4,S

Contains the RBA of all SDEP segments (one per SDEP segment)

OPC ends abnormally if integrity errors or invalid VSAM CIDE and RDF fields are encountered in more CIs for the area being analyzed than the number specified for MAXERROR.

Input and output for OPC

OPC is composed of Online Pointer Checker, In-depth Pointer Checker, and Report Generator.

After the online pointer checking, it can call the in-depth pointer checking program and the report generating program that are part of the Analyze function of FPA to do the in-depth pointer analysis and resolution, and to generate complete analysis reports.

The following figure shows the input and output for OPC when FULLSTEP=YES is specified.

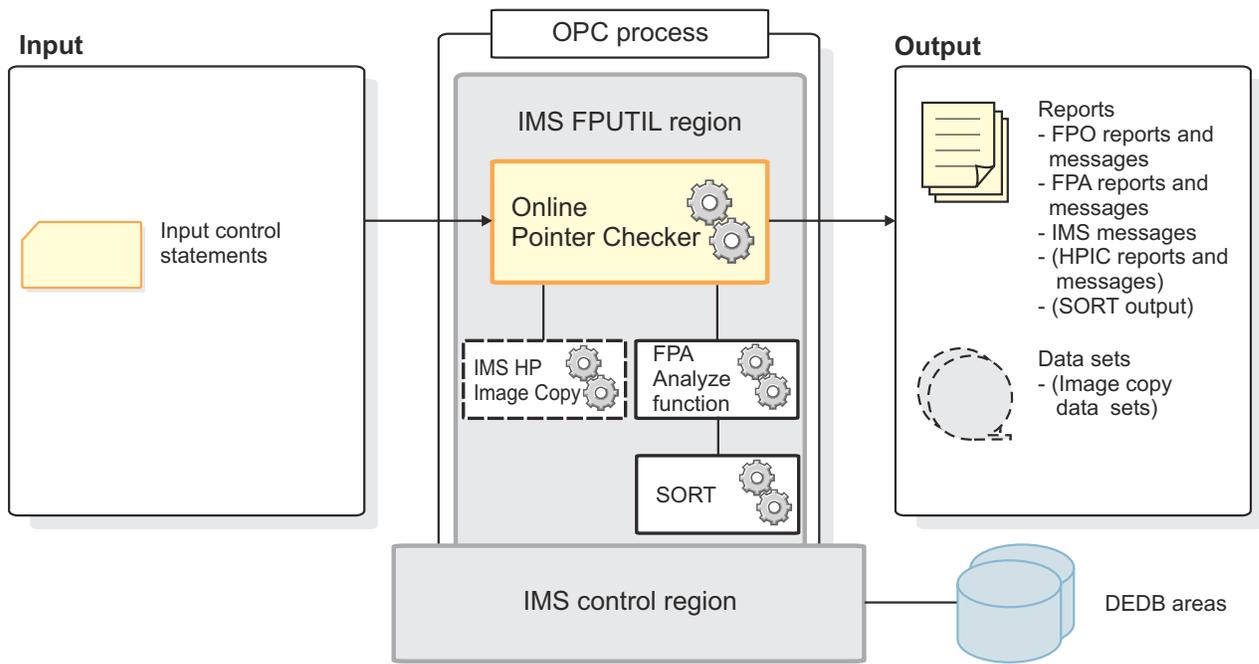


Figure 286. Input and output for OPC (FULLSTEP=YES)

The following figure shows the input and output for OPC when FULLSTEP=NO is specified. It shows how OPC provides a fast scan of database integrity and collects the pointer verification data from an online DEDB area. It carries out the same function supported by OPC accepting SYSIN and DASYSIN DD files as the input.

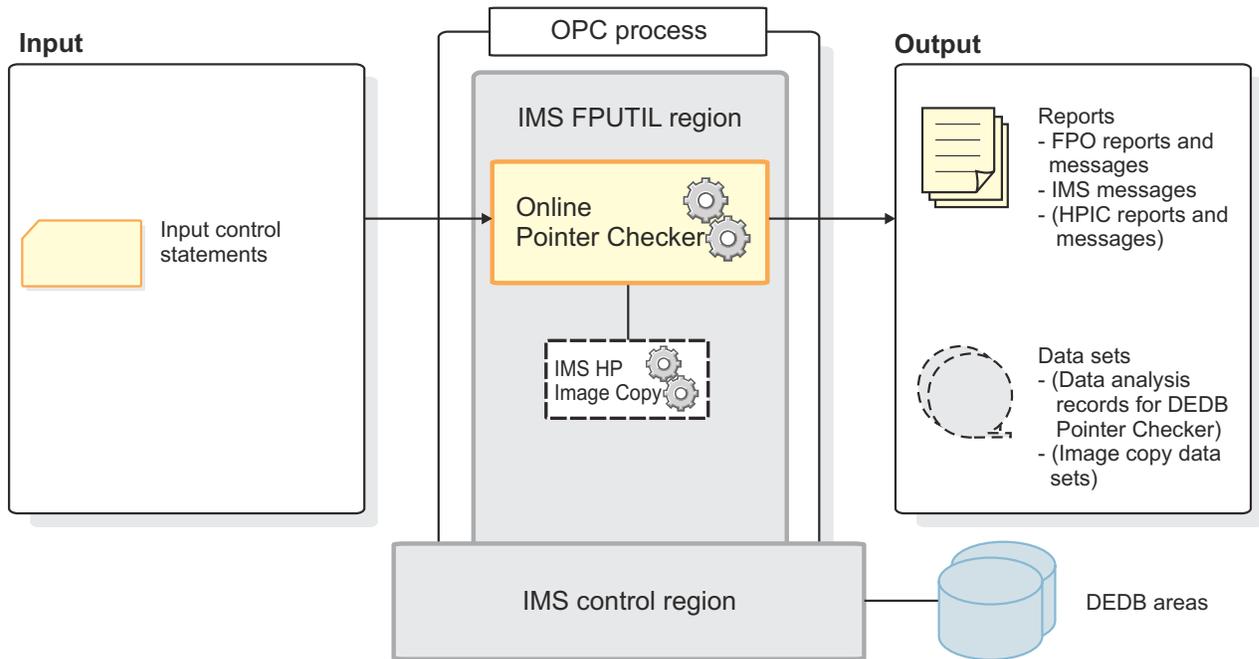


Figure 287. Input and output for OPC (FULLSTEP=NO)

Considerations for running OPC

The following topics describe what you need to consider before running OPC.

Selecting the operating modes

OPC runs in different operating modes according to the specification of the TYPRUN= value.

The following table summarizes the functions performed by OPC for each TYPRUN= value. Choose the TYPRUN= value that corresponds to the functions you want OPC to perform.

Table 76. TYPRUN= values and associated functions

TYPRUN=	Integrity verification	Analysis and report
PTRSCAN	<ul style="list-style-type: none"> Fast scan of database integrity Optional CI Map/CI Dump report 	-
FS	<ul style="list-style-type: none"> Fast scan of database integrity Optional CI Map/CI Dump report 	<p>When FULLSTEP=YES:</p> <ul style="list-style-type: none"> Freespace Analysis reports only Optional Reorg control statement generation Optional threshold analysis (freespace variables only) <p>When FULLSTEP=NO:</p> <ul style="list-style-type: none"> Freespace Analysis reports only Optional Reorg control statement generation Optional History file update Optional History2 file update Optional threshold analysis (freespace variables only)
MODEL	<ul style="list-style-type: none"> Fast scan of database integrity In-depth pointer analysis and resolution Optional CI Map/CI Dump report Optional Pointer Chain Reconstruction report 	<ul style="list-style-type: none"> Complete analysis reports Root key value appended to data records for tuning Optional Reorg control statement generation Optional History file updates Optional History2 file update Optional threshold analysis
PTRALL	<ul style="list-style-type: none"> Fast scan of database integrity In-depth pointer analysis and resolution Optional CI Map/CI Dump report Optional Pointer Chain Reconstruction report 	-

Table 76. TYPRUN= values and associated functions (continued)

TYPRUN=	Integrity verification	Analysis and report
RPT	<ul style="list-style-type: none"> • Fast scan of database integrity • In-depth pointer analysis and resolution • Optional CI Map/CI Dump report • Optional Pointer Chain Reconstruction report 	<ul style="list-style-type: none"> • Complete analysis reports • Optional Largest Database Records report • Optional Consolidated Largest Database Records report • Optional Reorg control statement generation • Optional History file update • Optional History2 file update • Optional threshold analysis

When FULLSTEP=YES is not specified, OPC provides a partial function to run fast scan of database integrity and collect the pointer verification data, similar to program FABADA1 of the (batch) DEDB Pointer Checker, with the added benefit that the database can remain fully accessible to users. OPC generates files that can be used as input for DEDB Pointer Checker to run in-depth pointer check and produce complete analysis reports. This type reduces the online run time but must be run in multiple steps.

Related reference

FULLSTEP keyword

The FULLSTEP keyword specifies whether OPC continues to process the pointer verification data collected from an online DEDB area.

TYPRUN keyword

The TYPRUN keyword specifies the operating mode.

Data sets required for each operating mode

Once you have decided which values to specify for your TYPRUN parameter and FULLSTEP parameter, you can determine the output data sets that you need to specify in JCL statements.

The following table shows TYPRUN=, FULLSTEP=, and the input and output data sets.

Table 77. Input and output data sets for each specification of TYPRUN= and FULLSTEP=

TYPRUN=	FULLSTEP=	Required input data sets	Required output data sets	Optional output data sets
PTRSCAN	N/A	SYSIN and DASYSIN	SYSPRINT	FPXMFIL, FPXRPTS, SNAPPIT
	NO	FPXIN	SYSPRINT	FPXMFIL, FPXRPTS, FPXSPRT, SNAPPIT
FS	N/A	SYSIN and DASYSIN	SYSPRINT, DAR120, SORT12	FPXMFIL, FPXRPTS, SNAPPIT
	NO	FPXIN	SYSPRINT, DAR120, SORT12	FPXMFIL, FPXRPTS, FPXSPRT, SNAPPIT
	YES	FPXIN	SYSPRINT	FPXMFIL, FPXRPTS, FPXSPRT, SNAPPIT, FPXARERG
MODEL	NO	FPXIN	SYSPRINT, DAR120, SORT12, DAR13SO, DAR3CSO, SORT3CS	FPXMFIL, FPXRPTS, SNAPPIT

Table 77. Input and output data sets for each specification of TYPRUN= and FULLSTEP= (continued)

TYPRUN=	FULLSTEP=	Required input data sets	Required output data sets	Optional output data sets
PTRALL	N/A	SYSIN and DASYSIN	SYSPRINT, DAR13SO, DAR3CSO, SORT13CS	FPXMFIL, FPXRPTS, SNAPPIT
	NO	FPXIN	SYSPRINT, DAR13SO, DAR3CSO, SORT13CS	FPXMFIL, FPXRPTS, FPXSPRT, SNAPPIT
	YES	FPXIN	SYSPRINT	FPXMFIL, FPXRPTS, FPXSPRT, SNAPPIT
RPT	N/A	SYSIN and DASYSIN	SYSPRINT, DAR120, SORT12, DAR13SO, DAR3CSO, SORT13CS	FPXMFIL, FPXRPTS, SNAPPIT
	NO	FPXIN	SYSPRINT, DAR120, SORT12, DAR13SO, DAR3CSO, SORT13CS	FPXMFIL, FPXRPTS, FPXSPRT, SNAPPIT
	YES	FPXIN	SYSPRINT	FPXMFIL, FPXRPTS, FPXSPRT, SNAPPIT, FPXAHST, FPXAHST2, FPXARERG

When FULLSTEP=YES is specified, DAR120 DD, DAR13SO DD, DAR3CSO DD, SORT12 DD, and SORT13CS DD data sets are not used.

Now that you know the output data sets to be used, you need to know how much space to allocate to them. Space requirements for the various work data sets created by OPC are summarized in the following table. Most output data sets do not require DCB information to be specified in your JCL.

Table 78. OPC data set sizes

DDNAME	Space requirements
DAR120	# CI/area x 1.5 x 20 bytes
DAR13SO	# segments x 40 bytes
DAR3CSO	# segments x 1.5 x 26 bytes
SORT12	< 1 track
SORT3CS	< 1 track

For the LRECL and BLKSIZE values to use, the following considerations apply:

- The output for DAR120, DAR13SO, and DAR3CSO is blocked to the block size of the output device. Because the blocking factor is determined at run time, standard labels must be used on all output data sets.
- Optionally, you can specify the block size on the DD statement. Do not specify RECFM and LRECL. If you do not specify LRECL on the DD statement, OPC calculates the appropriate size in accordance with the record. RECFM is VB. The DCB attributes are calculated by FPXOXIT0. When BatchPipes® is applied, OPC does nothing.

3380

The default block size is 23476 bytes. If LRECL is larger than 23476 bytes, this LRECL is used.

3390

The default block size is 27998 bytes. If LRECL is larger than 27998 bytes, this LRECL is used.

9345

The default block size is 22928 bytes. If LRECL is larger than 22928 bytes, this LRECL is used.

Other device

Defaults to the maximum block size of the output device.

Performance of OPC

OPC is intended to favor online system performance as much as possible, while retaining a consistent view of IMS data.

UOW locking is used.

SDEP pointer integrity will not be verified unless SDEP=YES is specified.

DBRC SCI registration exit routine

If you are using the RECON data set, in which an IMSplex name and a DBRC group are set, you must use the DBRC SCI registration exit routine (DSPSCIX0) to pass the IMSplex name and the DBRC group to DBRC.

You must specify DSPSCIX0 to use the parallel RECON access.

The SCI registration exit routine is called by DBRC before registering with the SCI. The exit routine supplies the IMSplex name and the DBRC group needed for SCI registration. For information about the DBRC SCI registration exit routine, see *IMS Exit Routines* for the version of IMS that you are using.

Running OPC

You can run OPC to perform complete DEDB integrity verification and generate the complete analysis reports. You can also use this function to correct integrity problems that were detected by a DEDB integrity verification process. The reports generated by OPC can be used to effectively manage IMS DEDBs.

About this task

There are four main steps in running OPC.

1. Allocate and initialize the permanent data set required by OPC. If you want to change the return codes of OPC, you need to create the return code user exit routines. This is a preprocess to run OPC, and is done only once.
2. Run OPC.
3. Correct integrity problems, if there are any.
4. If necessary, analyze the analysis reports and tune your environment.

Related reference

[Return code user exit routine \(FPXRCEX0\)](#)

IMS Fast Path Online Tools supports a return code user exit routine that can be used to change the return codes of OPC, ODE, or OER.

Preprocess for OPC: Initializing permanent data set

If you specify the TYPRUN=RPT keyword and the FULLSTEP=YES keyword for OPC, you can maintain the history of your OPC results by defining a Statistics History data set or a Statistics History2 data set. If you

want to maintain the history, you must allocate and initialize a Statistics History data set or a Statistics History2 data set.

About this task

Use of the Statistics History data set and the Statistics History2 data set is optional. If you do not intend to maintain a history of your OPC results, you do not need to allocate and initialize a Statistics History data set or a Statistics History2 data set; you can skip this preprocess.

Procedure

The Statistics History data set is used only when both the TYPRUN=RPT keyword and the FULLSTEP=YES keyword are specified, and either of the following specifications is present:

- The HISTORYDD keyword with a DD data set name as its parameter
- The FPXAHST DD statement

Unless you specify a DD data set name by using the HISTORYDD keyword, the Statistics History data set is automatically defined by the FPXAHST DD statement. When a Statistics History data set is defined for OPC by either of these specifications, the Statistics History data set is updated every time the utility is run. Each OPC process adds one record for each DEDB area that it processes, which contains the key space utilization and performance information for that area.

The Statistics History2 data set is automatically defined by the FPXAHST2 DD statement. When a Statistics History2 data set is defined for OPC, the Statistics History2 data set is updated every time an OPC process is run. Each OPC process adds one record for each UOW group in area that it processes, which contains the key space utilization and performance information for that UOW group. The UOW groups are specified by FPXAUOWC DD.

For the Statistics History data set and the Statistics History2 data set, you must use DISP=(MOD,KEEP,KEEP).

Example

The following figure shows the JCL that you can use to allocate and initialize a Statistics History data set and a Statistics History2 data set.

```
//*****  
//** ALLOCATE AND INITIALIZE HISTORY DATA SET  
//*****  
//IDCAMS EXEC PGM=IDCAMS  
//SYSPRINT DD SYSOUT=A  
//SYSIN DD *  
    DELETE ( HPFP.STAT.HIST ) NONVSAM  
    DELETE ( HPFP.STAT.HIST2 ) NONVSAM  
    SET MAXCC = 0  
/*  
//*****  
//** ALLOCATE STATISTICS HISTORY DATA SET  
//*****  
//IEFBR14 EXEC PGM=IEFBR14  
//DASDHIO DD DSN=HPFP.STAT.HIST,DISP=(NEW,CATLG,CATLG),  
//          DCB=(DSORG=PS,RECFM=FB,LRECL=100,BLKSIZE=800,  
//          UNIT=SYSDA,SPACE=(CYL,(2,2),RLSE)  
//DASDHIO2 DD DSN=HPFP.STAT.HIST2,DISP=(NEW,CATLG,CATLG),  
//          DCB=(DSORG=PS,RECFM=FB,LRECL=200,BLKSIZE=1600,  
//          UNIT=SYSDA,SPACE=(CYL,(4,4),RLSE)
```

Figure 288. Sample JCL to initialize permanent data set (History data sets)

Running the DEDB integrity verification process

OPC runs as an IMS IFP utility, using standard JCL and control statements. You need to specify an EXEC statement and DD statements that define the input and output data sets in your JCL.

Procedure

1. Specify the EXEC statement in one of the following formats:

```
//name EXEC procname
```

where *procname* is the IMS-supplied IFP procedure FPUTIL or any other IFP utility supporting procedure.

```
//name EXEC PGM=DFSRRRC00,PARM='IFP,dbdname,DBF#FPU0'
```

where *dbdname* is the DBDNAME associated with the DEDB.

Related reading:

- For information about the FPUTIL utility, see *IMS System Definition* for the version of IMS that you are using.
 - For information about the DEDB online utility commands, see *IMS Database Utilities* for the version of IMS that you are using.
2. Specify the DD statements to define input data sets and output data sets.
 3. Run the JCL.

Related reference

Examples for OPC

There are many ways to run the OPC utility. The examples provided in the following topics show some of the typical ways to run the OPC utility.

[Tips and best practices for the FPO Online Pointer Checker utility](#)

The FPO Online Pointer Checker utility (OPC) supports several keywords that improve the performance of OPC jobs and that further simplify management of DEDBs.

Correcting integrity problems in DEDB areas

The integrity problem messages always provide the exact RBAs (addresses) of the pointer or integrity discrepancies.

Procedure

To correct integrity problems, follow these steps:

1. Run OPC to create a CI SNAP dump. If desired, use the image copy option to create a backup.
2. Analyze the results from the reports. Verify the CI content and determine the appropriate corrections.
3. Repair the DEDB area. The IMS DBT VSAM Zapper utility provides a convenient tool for printing and changing the contents of a CI without modifying the Format 1 DSCB in the VTOC.
4. Rerun OPC to ensure that the problem is fixed. Optionally, you can use the image copy option to make a backup of the repaired DEDB area.
5. If desired, run the IDCAMS program to print the area for additional information.

Related reading: For information about the IDCAMS program, see *DFSMS Access Method Services for the Integrated Catalog Facility*.

Analyzing the complete analysis reports

The complete analysis reports can be used to effectively manage the space utilization, performance characteristics, and physical attributes of IMS DEDBs.

Related information

[DEDB monitoring and tuning guide](#)

You can use the utilities of IMS Fast Path Solution Pack to collect statistics data from DEDB areas, be notified of exceptional states of areas, and generate statistical reports to tune the areas.

Region size requirements for OPC

The region size for OPC depends on the size of a UOW and the keyword specifications.

Because OPC operates in a virtual storage region as an IMS IFP utility, the region size required to run OPC is larger than the following values:

- For the buffer sets portion for UOW processing, OPC requires the storage calculated by the following formula:

```
Buffer sets: UOW1 * CISZ * 5
```

If the value of UOW1 is smaller than 120, 120 is used instead of UOW1.

- For the IOVFPPOOL-dependent portion, OPC requires the storage calculated by the following formulas:

```
Work buffer: (UOW1 + IOVFPPOOL) * CISZ * 10
IOVF private buffer: IOVFPPOOL * CISZ * 5
```

- If the return code user exit routine is specified, the required region size of OPC is also affected by the MAXERROR keyword as follows:

```
Message and return code table: MAXERROR * 8
```

- If the FULLSTEP=YES keyword is specified, OPC internally calls the FPA Analyze function to perform a FULL mode pointer checking and generate complete analysis reports. For the additional region size required by the FPA Analyze function, see [“Region size requirements for the Analyze function”](#) on page 200.
- To generate the RBA of Used IOVF CI report (generated when FULLSTEP=YES, TYPRUN=RPT or FS, and IOVFCI_RBA_REPORT=YES are specified), OPC requires the storage calculated by the following formulas:

```
IOVF space map buffer: 4 * 119 * the number of space map CIs
Internal work buffer: 8 * maximum UOW + (8+64) * maximum number of IOVF CI
```

DD statements for OPC

DD statements for OPC determine the input and output data sets and specify how to run OPC.

The following table summarizes the DD statements for OPC. The optional output data sets that are used by OPC are determined by the combination of the TYPRUN keyword and the FULLSTEP keyword of the ANALYZE control statement.

Table 79. DD statements for OPC

DDNAME	Use	Format	Required or optional
JOBLIB or STEPLIB	Input	PDS	Required
FPXOPTS	Input	LRECL=80	Optional
FPXIN, or SYSIN and DASYSIN	Input	LRECL=80	Required

Table 79. DD statements for OPC (continued)

DDNAME	Use	Format	Required or optional
FPXMFILE	Output	LRECL=133	Optional
FPXRPTS	Output	LRECL=133	Optional
FPXSPRT	Output	LRECL=133	Optional
HFPPRINT	Output	LRECL=133	Optional
<i>reportdd</i>	Output	LRECL=133	Optional
SYSPRINT	Output	LRECL=121	Required
DFSUDUMP (<i>ic1outdd</i> or <i>ic2outdd</i>)	Output	IC1 or IC2	Optional
SNAPPIT	Output	LRECL=133	Optional
FPXAUOWC	Input	LRECL=80	Required when FPXAHST2 DD is specified
FPXAHST	Output	DISP=MOD	Optional
FPXAHST2	Output	LRECL=133	Required when FPXAUOWC DD is specified
FPXARERG	Output	LRECL=80	Optional
TAR <i>nnnnn</i>	Output	PS	Required when both of the following conditions are met: <ul style="list-style-type: none"> • TARECGEN=YES or TARECGEN=ONLY • TARSDNMASK keyword is not specified
DAR12O	Output		Required when FULLSTEP=NO
DAR13SO	Output		Required when FULLSTEP=NO
SORT12	Output	LRECL=80	Required when FULLSTEP=NO
SORT3CS	Output	LRECL=80	Required when FULLSTEP=NO
DAR3CSO	Output		Required when FULLSTEP=NO
IMSDALIB	Input	PDS	Optional
RECON <i>n</i>	Input/ Output		Required when all of the following conditions are met: <ul style="list-style-type: none"> • IC=YES and ICNOTIFY=YES are specified, or ITKBSRVR=<i>servername</i> and DBRC=YES are specified. • DFSMDA member for RECON<i>n</i> data sets are not used.
TOOLSNP1	Output	LRECL=125	Optional
TOOLSNP2	Output	LRECL=125	Optional
HFPABEND	Output	LRECL=125	Optional
SYSOUT	Output	LRECL=121	Optional
<i>i001WKnn</i>	Input/ Output		Optional
<i>i001MSG</i>	Output	LRECL=133	Optional

Table 79. DD statements for OPC (continued)

DDNAME	Use	Format	Required or optional
ICEIN	Input	LRECL=80	Optional
ICEPRINT	Output	LRECL=133	Optional
ICERPRT	Output	LRECL=133	Optional
<i>dataout1</i>	Output	PS	Optional
<i>dataout2</i>	Output	PS	Optional
D2Xnnnn	Input/ Output		Optional
D3Cnnnn	Input/ Output		Optional
D3SRnnnn	Input/ Output		Optional
D3SInnnn	Input/ Output		Optional
D4Pnnnn	Input/ Output		Optional
D4Snnnn	Input/ Output		Optional
WPRNnnn	Input/ Output		Optional
WSNAP	Input/ Output	LRECL=133	Optional

STEPLIB DD

Defines the library that contains the OPC utility.

- The load library of IMS HP Image Copy must be concatenated if you specify the *comprtn* parameter on the IC keyword or the IC=(YES,ICEIN) keyword.
- The IMS Tools Base library (SHKTLOAD) is required if you specify the ADXCFGRP keyword, the ITKBSRVR keyword, or the SENSOR=YES option.
- The library that contains DFSMDA dynamic allocation members for the RECON data sets.

The data set that contains the DBRC SCI registration exit routine (DSPSCIX0) must be concatenated when using the RECON data set in which an IMSplex name is set.

FPXOPTS DD

Defines the input data set that contains the option control statements for the process. FPXOPTS usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte fixed-length records. Block size, if coded, must be a multiple of 80. It can be specified as DUMMY, NULLFILE, or an empty file.

For details of the option control statements, see [“Option control statement for OPC”](#) on page 695.

FPXIN DD

Defines the control statements for OPC. FPXIN describes the area to be processed, and it can contain optional user requests that further define the operating mode. FPXIN DD and SYSIN DD are mutually exclusive. FPXIN usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte fixed-length records. Block size, if coded, must be a multiple of 80. It cannot be specified as DUMMY, NULLFILE, or an empty file. When both FPXIN DD and DASYSIN DD are specified, DASYSIN DD is ignored.

For details of the control statements, see [“Utility control statements for OPC” on page 696](#).

SYSIN DD

Defines the input control data set that contains the IMS IFP utility control statements. A minimum of four statements is required, as shown below:

```
TYPE TOOL
AREA areaname
EXIT ANALYZE
GO
```

where *areaname* is the name of an area to be processed. Only one area per job step can be specified.

For additional information about these IMS IFP utility control statements, see *IMS Database Utilities* for the version of IMS that you are using.

DASYSIN DD

Defines the control statement file that contains the data collection options, image copy specifications and reporting options. This data set can reside on a direct-access device or be routed through the input stream.

For details of the control statements, see [“Utility control statements for OPC” on page 696](#).

FPXMFIL DD

Defines the default message output. The output can go to a data set or to a printer, routed through the output stream. If the DD statement is specified explicitly or implicitly by the MFILE keyword of the ANALYZE control statement, and if the DD statement for the specified DD name is not coded in the JCL stream, the DD is allocated dynamically to the system output stream with SYSOUT=*. If the DD name is overridden by the MFILE keyword with a different DD name, the DD statement for the specified DD name must be coded in the JCL stream.

FPXRPTS DD

Defines the default report output. The output can go to a data set or to a printer, routed through the output stream. If the DD statement is specified explicitly or implicitly by the RFILE keyword of the ANALYZE control statement and if the DD statement for the specified DD name is not coded in the JCL stream, the DD is allocated dynamically to the system output stream with SYSOUT=*. If the DD statement is overridden by the RFILE keyword with a different DD name, and the DD statement for the specified DD name is not coded in the JCL stream, the DD is allocated dynamically to the system output stream with SYSOUT=*.

FPXSPRT DD

Defines the report output data set when SENSOR=YES is specified. This output data set contains the Sensor Data Statistics report that can be used to verify which sensor data is stored in the Sensor Data repository of IMS Tools KB. The following DCB attributes of the file are set by FPO:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set. You can specify NULLFILE or DUMMY for this data set.

The FPXSPRT data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//FPXSPRT DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPO dynamically allocates the data set by using SYSOUT=*.

HFPFPRINT DD

Defines the message output data set when FULLSTEP=YES is specified. This output data set contains the HFP messages, which are the processing messages issued by FPA. The following DCB attributes of the file are set by FPA:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set. Do not specify DUMMY for this data set.

The HFPPRINT data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPPRINT DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*.

reportdd DD

Defines the output for the following reports:

- DEDB Area Analysis reports
- UOW report
- UOW Exception report
- Largest Database Records report
- Consolidated Largest Database Records report
- Segment Length Distribution report
- DB Record Length Distribution report
- RBA of Used IOVF CI report

The output can go to a data set or to a printer, routed through the output stream. FPXRPTS is the default DD name, however, the DD name can be overridden by the REPORTDD keyword specification of the REPORT control statement.

This statement is used only when you specify the TYPRUN=RPT and the FULLSTEP=YES keywords, or the TYPRUN=FS and the FULLSTEP=YES keywords.

reportdd DD is similar to the *reportdd* DD of the Analyze function of FPA.

SYSPRINT DD

Defines the system message output. For OPC with SYSIN DD, SYSPRINT is used by OPC as the default message data set until it is (optionally) overridden by the MFILE= keyword of the ANALYZE control statement.

Warning: You cannot allocate the OPC SYSPRINT DD output stream to a data set because SYSPRINT DD is also used by IMS. If you specify SYSPRINT DD as a data set, some output might become overridden or unpredictable open or close errors might occur. Therefore, you must specify SYSPRINT DD as SYSOUT.

DFSUDUMP DD

Defines the default output image copy data set. DFSUDUMP is used by OPC as the default primary image copy data set until it is (optionally) overridden by the IC1OUTDD keyword of the ANALYZE control statement. The output can reside on a tape or on a direct-access device. If the DD statement is overridden by the IC1OUTDD keyword with a different DD name, the DD statement for the specified DD name must be coded in the JCL stream.

Do not specify DISP=MOD for this DD statement.

The secondary image copy data set is (optionally) specified by the IC2OUTDD keyword of the ANALYZE control statement, the specified DD name must be coded in the JCL stream.

SNAPPIT DD

Defines the output for the CI SNAP dump produced by OPC.

The CI SNAP dump option is ignored if the DD statement is not provided.

The output can go to a data set or to a printer, routed through the output stream.

FPXAUOWC DD

Each record in this data set specifies a range of UOW numbers of an area, which defines a UOW group of the area. The information about the UOW group is recorded in the FPXAHST2 data set or in a sensor data record set in the order that is specified in the FPXAUOWC data set, even if the UOW ranges overlap or are in reverse order.

The data set can reside in the JCL input stream. The data set can also be defined as a sequential data set or as a member of a partitioned data set. If a data set is specified, the data set must have the following attributes:

- DSORG=PS
- RECFM=FB
- LRECL=80
- BLKSIZE=*a multiple of 80*

If the data set is empty, all UOWs of the area are regarded as the target UOW range.

When you specify the FPXAHST2 DD statement, you must also specify the FPXAUOWC DD statement. If an FPXAUOWC DD statement is not specified, the FPXAHST2 DD statement is ignored.

When SENSOR_UOW=YES is specified, the data elements that will be stored differ by the FPXAUOWC DD specification as follows:

- If you specify the FPXAUOWC DD statement, data elements are collected at the UOW group level.
- If you do not specify the FPXAUOWC DD statement, data elements are collected at the UOW level.

The FPXAUOWC DD is similar to the HFPAUOWC DD of the Analyze function of FPA.

For information about defining a UOW group, see [“UOW group definition record layout” on page 1191](#).

FPXAHST DD

Defines an output data set that contains historical records. The analysis report generation process adds one record for each DEDB area that it processes. The record contains the key space utilization and performance information for that area. This statement is used only when you specify the TYPRUN=RPT keyword and the FULLSTEP=YES keyword.

If this feature is used, the FPXAHST data set must be preallocated with the following attributes:

- DSORG=PS
- RECFM=FB
- LRECL=100
- BLKSIZE=*user-specified*

The DD statement in the JOB stream is specified as:

```
//FPXAHST DD DSN=HPFP.HISTORY,DISP=(MOD,KEEP,KEEP)
```

The DD name can be (optionally) overridden by the HISTORYDD keyword with a different DD name of the ANALYZE control statement.

The FPXAHST DD is similar to the HFPAHST DD of the Analyze function of FPA.

FPXAHST2 DD

Defines an output data set that contains historical records. The analysis report generation process adds one record for each UOW group that is specified by FPXAUOWC DD. The record contains the key space utilization and performance information for the area, which is based on the information obtained from the specified UOW range. This statement is used only when you specify the FPXAUOWC DD statement, the TYPRUN=RPT keyword, and the FULLSTEP=YES keyword. If FPXAUOWC DD statement is not specified, this statement is ignored.

If this feature is used, the FPXAHST2 data set must be preallocated with the following attributes:

- DSORG=PS
- RECFM=FB
- LRECL=200
- BLKSIZE=user-specified

The DD statement in the JOB stream is specified as:

```
//FPXAHST2 DD DSN=HPFP.HISTORY2,DISP=(MOD,KEEP,KEEP)
```

The FPXAHST2 DD is similar to the HFPAHST2 DD of the Analyze function of FPA.

FPXARERG DD

Defines an output data set that contains the reorganization control statements for the High-Speed DEDB Direct Reorganization utility (DBFUHDRO).

This statement is used only when you specify the TYPRUN=RPT and the FULLSTEP=YES keywords, or the TYPRUN=FS and the FULLSTEP=YES keywords.

The following DCB attributes of the file are set by the Analyze process:

- RECFM=PS
- LRECL=80
- BLKSIZE=800

Do not code the DCB information in your JCL statement. Do not specify DISP=MOD for this DD statement.

The FPXARERG DD is similar to the HFPARERG DD of the Analyze function of FPA.

TARnnnnn DD

Defines the tuning aid record data set. A TARnnnnn DD statement specifies the tuning aid record data set for one area.

When TARECGEN=YES is specified, TARnnnnn DD statements define the output data sets for generating tuning aid records. If this statement is not specified for an area, the tuning aid record for that area is not generated.

A TARnnnnn DD statement is for areas in the range of 1 - 2048. The value of nnnnn must be right-aligned, with leading zeros if necessary. When you code multiple TARnnnnn DD statements, do not specify the same data set more than once; each TARnnnnn DD statement must point to a different data set.

This data set can be allocated dynamically by specifying the TARDSNMASK keyword of the ANALYZE statement.

For a description and space requirement of the tuning aid records, see [“Tuning aid records” on page 231](#).

DAR120 DD

Defines the output data set containing the free space and area information records. The data set can reside on either a direct-access device or a tape. When FULLSTEP=YES is specified, this DD statement is not used.

Space requirements vary depending on the size of the areas and the number of free space elements. (Approximation: *number of CIs in area* x 1.5 x 20 bytes.) Do not specify DISP=MOD for this DD statement.

When DAR120 is so big that the allocated space is not adequate, a B37 abend occurs.

DAR13SO DD

Defines the output data set containing the segment information records for all segments that reside in the root addressable part of the database. The data set can reside on either a direct-access device or a tape. When FULLSTEP=YES is specified, this DD statement is not used.

Space requirements vary depending on the size of the areas and the number of segments in the areas. (Approximation: *number of segments* x 40 bytes.) Do not specify DISP=MOD for this DD statement.

When DAR13SO is so big that the allocated space is not adequate, a B37 abend occurs.

SORT12 DD

Defines the output data set for the SORT control statements for the SORT12 step. The data set must reside on a direct-access device. When FULLSTEP=YES is specified, this DD statement is not used.

Space requirements are very small (1 track suffices). Do not specify DISP=MOD for this DD statement.

DCB information is hard-coded in OPC. Do *not* code DCB information in your JCL.

SORT3CS DD

Defines the output data set for the SORT control statements for the sort before pointer resolution. The data set must reside on a direct-access device. When FULLSTEP=YES is specified, this DD statement is not used.

Space requirements are very small (1 track suffices). Do not specify DISP=MOD for this DD statement.

DCB information is hard-coded in OPC. Do *not* code DCB information in your JCL.

DAR3CSO DD

Defines the output data set containing all pointer information records and the segment information records for all segments that reside in IOVF and SDEP. The data set can reside on either a direct-access device or a tape. When FULLSTEP=YES is specified, this DD statement is not used.

Space requirements vary depending on the size of the areas and the number of segments in the areas. (Approximation: *number of segments* x 1.5 x 26 bytes.) Do not specify DISP=MOD for this DD statement.

When DAR3CSO is so big that the allocated space is not adequate, a B37 abend occurs.

IMSDALIB DD

An optional DD statement that defines the library that contains DFSMDA dynamic allocation members for the RECON data sets.

RECON1/RECON2/RECON3 DD

Defines the DBRC RECON data sets. You must specify these DD statements when you specify both IC=YES and ICNOTIFY=YES keywords, or both ITKBSRVR=*servername* and DBRC=YES keywords. RECON*n* data sets must be the same data sets that are used in the IMS control region.

When you specify IC=YES and ICNOTIFY=YES, OPC updates RECON*n* data sets because OPC registers the image copy data set to DBRC. When you specify ITKBSRVR=*servername* and DBRC=YES, OPC retrieves the RECON ID from RECON data sets.

If these statements are omitted when the data sets are necessary and if DFSMDA members are available, the utility uses the DFSMDA members and dynamically allocates the data sets.

TOOLSNP1

Defines a snap dump output data set used for diagnostic purposes. All you need to provide is the DSN and space allocation. OPC assigns the correct DCB parameters internally. Usually, //TOOLSNP1 DD SYSOUT=* is sufficient.

TOOLSNP2

Defines a secondary diagnostic output file subject to the same restrictions as TOOLSNP1.

HFPABEND DD

Defines a snap dump output data set used for diagnostic purposes. This data set might be used when FULLSTEP=YES is specified and the FPO job ends abnormally. Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPO dynamically allocates the data set by using SYSOUT=*

SYSOUT DD

Defines the output data set that is used by both OPC and DFSORT for message functions. OPC writes images of its control statements that are specified in DASYSIN DD. When the FULLSTEP=YES keyword is specified, this output data set also contains the message that is produced by DFSORT.

Note on dynamic allocation:

When DFSORT is needed and this statement is not specified in the JCL statement, OPC dynamically allocates the data set by using SYSOUT=*.

i001WKnn DD

An optional DD statement. These are intermediate storage data sets that are used by DFSORT. For more information about coding the SORTWKnn DD statements, see the *DFSORT Application Programming Guide*. The adequate size of storage data sets is 1.5 to 2.5 times the input file size. This statement is used only when you specify the FULLSTEP=YES keyword.

Allocating twice the space used by the SORTIN data set is usually adequate for each work data set.

Do not specify DISP=MOD for these DD statements.

In the analyze step, the size of each input file is as follows:

i:

A: Always used

30 x the number of pointers in the area

D: If SDEP=YES is specified

30 x the number of SDEP segments in the area

G: If SNAPPIT DD is specified

112 x the number of pointers in which pointer errors are detected

J: If TYPRUN=RPT is specified

X x (the number of non-SDEP segments in the areas + the number of free space elements in the areas + the number of references to other CIs)

Where X is:

- When LARGEREC_REPORT=NO is specified, X is calculated as follows:

34 + (the maximum segment level x 4)

However, if X is smaller than 48, 48 is used for X.

- When LARGEREC_REPORT=YES is specified, X is calculated as follows:

34 + (the maximum segment level x 4) + root key length

However, if X is smaller than 48, 48 is used for X.

J: If TYPRUN=FS is specified

23 x (the number of free space elements in the area + the number of references to other CIs)

K: If TYPRUN=RPT is specified

X x the number of segments in the area

Where X is:

- When TYPRUN=MODEL, LARGEREC=Y, or TARECGEN=Y is specified, X is calculated as follows:

34 + (the maximum segment level x 4) + root key length

- In other cases, X is calculated as follows:

34 + (the maximum segment level x 4)

M: If TYPRUN=RPT or TYPRUN=PTRALL is specified

X x the number of IOVF segments in the area

Where X is:

- When TYPRUN=MODEL, LARGEREC=Y, or TARECGEN=Y is specified, X is calculated as follows:

34 + root key length

- In other cases, X is 34.

Q: Always used

34 x the number of segments in the area

Note on dynamic allocation:

If this statement is not specified in the JCL statement when these data sets are necessary, FPO allocates them dynamically.

!001SMSG DD

This DD statement is an optional DD statement. However, this DD statement is required when DFSORT is used. This statement defines an output data set for storing DFSORT messages. For the condition for running DFSORT and the naming rules for the DD names, see the description of !001WKnn DD.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, OPC dynamically allocates the data set by using SYSOUT=*

ICEIN DD

This DD statement is an optional DD statement. This statement specifies the input data set that contains the control statements for IMS HP Image Copy. The data set can reside on a tape, a direct-access volume, or it can be routed through the input stream (DD * or DD DATA). This data set is used only when you specify IC=(YES,ICEIN). If the DD statement is not coded and if IC=(YES,ICEIN) is specified, the default control statements are used. For details about the ICEIN control statements, see the *IMS High Performance Image Copy User's Guide*.

ICEPRINT DD

This DD statement is an optional DD statement. This statement specifies the output data set that contains the reports that are produced by IMS HP Image Copy. The data set can reside on a tape, a direct-access volume, or a printer, or it can be routed through the output stream (SYSOUT). This data set is used only when you specify IC=(YES,ICEIN). If the DD statement is not coded and IC=(YES,ICEIN) is specified, a SYSOUT stream is dynamically allocated for the DD.

ICERPRT DD

This DD statement is an optional DD statement. This statement specifies the output data set that contains the Image Copy Dump Process report, which is produced by IMS HP Image Copy. It can reside on a tape, a direct-access volume, or a printer, or it can be routed through the output stream (SYSOUT). This data set is used only when you specify IC=(YES,ICEIN).

dataout1 DD

This DD statement is an optional DD statement. This statement defines the primary image copy data set. One DD statement is required for each data set that is to be dumped. The DD name can be any 1- to 8-character string. The data set specified by the DD statement must be on either a direct-access device or a tape device. This data set is used only when you specify IC=(YES,ICEIN).

For details of the format and the characteristics of the image copy data set, see the *IMS High Performance Image Copy User's Guide*. You can request IMS HP Image Copy to allocate the image copy data set dynamically by omitting this DD statement and specifying the allocation parameters that are associated with the data set in the ICEIN data set.

dataout2 DD

This DD statement is an optional DD statement. This statement defines the secondary image copy data set. This statement is required only if the associated ICEIN control statement requests the secondary image copy. The data set specified by the DD statement must be on either a direct-access device or a tape device. This data set is used only when you specify IC=(YES,ICEIN).

For details of the format and the characteristics of the image copy data set, see the *IMS High Performance Image Copy User's Guide*.

D2Xnnnn DD
D3Cnnnn DD
D3SRnnnn DD
D3SInnnn DD
D4Pnnnn DD
D4Snnnn DD
WPRNnnn DD

Define the intermediate work data sets. FPO uses up to three sets of intermediate work data sets, one set for processing one area. For each set, the same *nnnn* value is used. *nnnn* must be right-aligned with leading zeros. These statements are used only when you specify the FULLSTEP=YES keyword.

- D2Xnnnn data sets contain the free space and area information records.
- D3Cnnnn data sets contain all non-SDEP pointer information records.
- D3SRnnnn data sets contain the segment information records for all segments that reside in the RAP part of the database.
- D3SInnnn data sets contain the segment information records for all segments that reside in the IOVF part of the database.
- D4Pnnnn data sets contain all SDEP pointer information records.
- D4Snnnn data sets contain the segment information records for all segments that reside in the SDEP part of the database.
- WPRNnnn data sets contain analysis reports.

Do not specify DISP=MOD for these DD statements. DCB information is hard-coded in OPC. Do not code DCB information in your JCL.

Note on dynamic allocation:

If these statements are not specified in the JCL statements when these data sets are necessary, FPO allocates them dynamically.

WSNAP DD

This DD statement is an optional DD statement. It defines the intermediate work data set for snap dumps. This statement is used only when you specify the FULLSTEP=YES keyword.

Do not specify DISP=MOD for this DD statement. DCB information is hard-coded in OPC. Do not code DCB information in your JCL.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when this data set is necessary, FPO allocates it dynamically.

Option control statement for OPC

The option control statement defines the user description of the OVERRIDE control statement, which specifies information for setting temporary override values of product message defaults.

The option control statements must start with a line that contains the OPTIONS command. The control statement that can be specified is:

OVERRIDE control statement

The OVERRIDE control statement specifies information for setting temporary override values of product message defaults. This control statement is optional. The OVERRIDE control statement can be specified in FPXOPTS DD.

OVERRIDE statement keywords

The keywords associated with the OVERRIDE statement are not available for site default options. For the format of the OVERRIDE statement, see [“OVERRIDE control statement”](#) on page 850.

For the OVERRIDE statement, you can specify the keywords that are listed in the following table.

Table 80. *OVERRIDE statement keywords*

Keyword	Required or optional	Description	Topic
MSG_LIM	Optional	Specifies the maximum number of repetitive messages that are to be issued before suppression.	“MSG_LIM keyword” on page 897
MSG_NUM	Required	Specifies the message number that is to be suppressed.	“MSG_NUM keyword” on page 898

Related reference

[Keyword reference for FPO](#)

These topics explain the control statements and keywords of FPO and their syntax rules.

[Syntax rules for FPXOPTS](#)

FPXOPTS control statements and their associated keywords can be coded in free format in columns 1 through 72, subject to certain syntax rules.

Utility control statements for OPC

The utility control statements for OPC define the user description of the GLOBAL control statement, the ANALYZE control statement and two optional control statements, the REPORT control statement and the THRESHOLD control statement, all of which control the OPC process.

GLOBAL control statement

The GLOBAL control statement defines the job step keyword values. If specified, GLOBAL must be the first command in the input stream. The GLOBAL control statement can be specified only in FPXIN DD.

ANALYZE control statement

The ANALYZE control statement defines information that control how OPC runs. The ANALYZE control statement can be specified in FPXIN DD or in DASYSIN DD. In FPXIN DD, the ANALYZE control statement must start with the ANALYZE command, whereas the ANALYZE control statement must not start with the ANALYZE command when used in DASYSIN DD.

REPORT control statement

The REPORT control statement defines the parameter value that is related to the analysis reports. The REPORT control statement can be specified only in FPXIN DD.

THRESHOLD control statement

The THRESHOLD control statement defines the threshold parameter values, which cause warning messages to be generated when the actual values do not meet the specified criteria. The THRESHOLD control statement can be specified only in FPXIN DD.

OPC allows you to specify site default options.

Subsections:

- [“GLOBAL statement keywords” on page 696](#)
- [“ANALYZE statement keywords” on page 697](#)
- [“REPORT statement keywords” on page 699](#)
- [“THRESHOLD statement keywords” on page 700](#)

GLOBAL statement keywords

The GLOBAL control statement specifies the job step keyword values. This control statement is optional.

The keywords associated with the GLOBAL statement are available for site default options.

For the GLOBAL statement, you can specify the keywords that are listed in the following table. For the format of the GLOBAL statement, see [“GLOBAL control statement” on page 847](#).

Table 81. GLOBAL statement keywords

Keyword	Used in FPXIN DD	Used in DASYSIN DD	Description	Topic
ADXCFCGRP	Yes (Optional)	No	Specifies the name of the Autonomics Director XCF group.	“ADXCFCGRP keyword” on page 861
DBRC	Yes (Optional)	No	Specifies whether to store reports that are associated with a RECON to the IMS Tools KB repository.	“DBRC keyword (for GLOBAL)” on page 868
ITKBLOAD	Yes (Optional)	No	Specifies the name of the load module library for IMS Tools KB.	“ITKBLOAD keyword” on page 890
ITKBSRVR	Yes (Optional)	No	Specifies the name of the IMS Tools KB server XCF group.	“ITKBSRVR keyword” on page 891
KEYWORD_ DUPCHK	Yes (Optional)	No	Specifies to end the job with an error message when a duplicate keyword is found.	KEYWORD_DUPCHK keyword

ANALYZE statement keywords

The ANALYZE control statement specifies information that control how OPC runs. This control statement is mandatory.

The keywords associated with the ANALYZE statement are available for site default options except for the AREA keyword.

For the ANALYZE statement, you can specify the keywords that are listed in the following table. For the format of the ANALYZE statement, see [“ANALYZE control statement” on page 837](#).

Table 82. ANALYZE statement keywords

Keyword	Used in FPXIN DD	Used in DASYSIN DD	Description	Topic
ARDIGIT	Yes (Optional)	No	Specifies the digit of the area number field for the UOW group definition records that are defined in the FPXAUOWC file.	“ARDIGIT keyword” on page 863
AREA	Yes (Required)	No	Specifies the name of the area to be processed.	“AREA keyword” on page 864
DBRC	Yes (Optional)	Yes (Optional)	Specifies that image copies be registered to DBRC. DBRC is alias keyword for ICNOTIFY.	“DBRC keyword (for GLOBAL)” on page 868
DOVFNULPTR _MSGLVL	Yes (Optional)	No	Specifies that the utility issues a warning message if the first DOVF CI in any UOW contains a null pointer for the current overflow CI.	DOVFNULPTR _MSGLVL keyword
ERRORACT	Yes (Optional)	Yes (Optional)	Specifies the action for OPC to take when MAXERROR is reached.	“ERRORACT keyword” on page 870
EXC_RBASEFS	Yes (Optional)	No	Specifies a criterion. If a RAP CI contains free space that is greater than the percentage specified, and the RAP CI also uses overflow CIs, a reorganization control statement is generated for the UOW, and the UOW is printed in the UOW Exception report.	“EXC_RBASEFS keyword” on page 872
EXC_RDOVFFS	Yes (Optional)	No	Specifies a criterion. If the DOVF part of a UOW contains free space that is greater than the percentage specified, and the UOW also uses IOVF CIs, a reorganization control statement is generated for the UOW and the UOW is printed in the UOW Exception report.	“EXC_RDOVFFS keyword” on page 873
EXC_UDOVFFS	Yes (Optional)	No	Specifies a criterion. If the DOVF part of a UOW contains free space that is less than the percentage specified, the UOW is printed in the UOW Exception report.	“EXC_UDOVFFS keyword” on page 873
FULLSTEP	Yes (Optional)	No	Specifies whether OPC continues to process the pointer verification data collected from an online DEDB area.	“FULLSTEP keyword” on page 882

Table 82. ANALYZE statement keywords (continued)

Keyword	Used in FPXIN DD	Used in DASYSIN DD	Description	Topic
HISTORYDD	Yes (Optional)	No	Specifies the ddname of the DD statement for the output History file data sets that are specified in the JCL.	“HISTORYDD keyword” on page 882
IC	Yes (Optional)	Yes (Optional)	Specifies that an IMS standard image copy be taken while scanning the database area.	“IC keyword” on page 883
ICNOTIFY	Yes (Optional)	Yes (Optional)	Specifies that image copies be registered to DBRC.	“ICNOTIFY keyword” on page 885
IC1OUTDD	Yes (Optional)	Yes (Optional)	Specifies the DD name of the output file to which the primary image copy data set is to be written.	“IC1OUTDD keyword” on page 886
IC2OUTDD	Yes (Optional)	Yes (Optional)	Specifies the DD name of the output file to which the secondary image copy data set is to be written.	“IC2OUTDD keyword” on page 886
IOVFPOOL	Yes (Optional)	Yes (Optional)	Specifies the size of the IOVF buffer pool.	“IOVFPOOL keyword” on page 888
MAXERROR	Yes (Optional)	Yes (Optional)	Specifies the maximum number of pointer errors that might be encountered before processing ends abnormally.	“MAXERROR keyword” on page 894
MAXTWINCTR	Yes (Optional)	Yes (Optional)	Specifies the threshold value for the number of twin segments.	“MAXTWINCTR keyword” on page 895
MFILE	Yes (Optional)	Yes (Optional)	Specifies the DD name of the output file to which OPC messages are to be written.	“MFILE keyword” on page 896
PCHKONLY	Yes (Optional)	Yes (Optional)	Specifies whether to evaluate only the control statement keywords or to also check pointer validation of an online DEDB area.	“PCHKONLY keyword” on page 901
PROC_AFT_ERROR	Yes (Optional)	No	Specifies whether to end the job step when an area-specific error is encountered or to continue processing subsequent areas.	PROC_AFT_ERROR keyword
RCEXLOAD	Yes (Optional)	No	Specifies whether to call the return code exit routine FFXRCEX0 from the STEPLIB or JOBLIB data set concatenation.	“RCEXLOAD keyword” on page 904
RFILE	Yes (Optional)	Yes (Optional)	Specifies the DD name of the output file to which OPC reports are to be written.	“RFILE keyword” on page 908
RGROUP	Yes (Optional)	No	Specifies to fill the value in the Repository Group field in the FPXAHST2 file or in a sensor data record set.	“RGROUP keyword” on page 910
SDEP	Yes (Optional)	Yes (Optional)	Specifies whether or not run totals and SDEP pointer integrity is to be verified and any errors reported.	“SDEP keyword” on page 912
SDEP_ORPHAN_MSGLVL	Yes (Optional)	No	Specifies the error message level for an orphan SDEP segment.	SDEP_ORPHAN_MSGLVL keyword
SDEP_PTRIOVF_MSGLVL	Yes (Optional)	Yes (Optional)	Specifies the severity level of the message that is issued when detecting an SDEP pointer that points to the IOVF portion.	SDEP_PTRIOVF_MSGLVL keyword
SENSOR	Yes (Optional)	No	Specifies whether to store the sensor data in the Sensor Data repository of IMS Tools KB.	“SENSOR keyword” on page 922
SENSOR_DBREC	Yes (Optional)	No	Specifies whether to store the data elements that are related to database records and root segment distribution to the Sensor Data repository of IMS Tools KB.	“SENSOR_DBREC keyword” on page 923
SENSOR_PTRERR	Yes (Optional)	No	Specifies to store sensor data in the Sensor Data repository of IMS Tools KB even when pointer errors are detected in the area.	“SENSOR_PTRERR keyword” on page 924
SENSOR_UOW	Yes (Optional)	No	Specifies whether to store the data elements that are related to UOW to the Sensor Data repository of IMS Tools KB.	“SENSOR_UOW keyword” on page 925

Table 82. ANALYZE statement keywords (continued)

Keyword	Used in FPXIN DD	Used in DASYSIN DD	Description	Topic
SENSOR_UOW_ REPORT	Yes (Optional)	No	Specifies whether to print the data elements that are related to UOW in the Sensor Data Statistics report.	SENSOR_UOW_ REPORT keyword
STAT	Yes (Optional)	Yes (Optional)	Specifies whether or not run totals and statistics are to be generated.	“STAT keyword” on page 928
SYNTBL	Yes (Optional)	No	Specifies the number of reporting entries in the Synonym Chain Analysis table.	“SYNTBL keyword” on page 932
TARDSNMASK	Yes (Optional)	No	Specifies the data set name or the mask to use when dynamically allocating cataloged pre-existing tuning aid record data sets.	“TARDSNMASK keyword” on page 932
TARECGEN	Yes (Optional)	No	Specifies whether to generate the tuning aid records.	“TARECGEN keyword” on page 933
TYPRUN	Yes (Optional)	Yes (Optional)	Specifies the operating mode.	“TYPRUN keyword” on page 935

REPORT statement keywords

The REPORT control statement specifies the parameter values related to the analysis reports. This control statement is effective only when the FULLSTEP=YES keyword is specified with the TYPRUN=RPT or the TYPRUN=FS keyword.

The keywords associated with the REPORT statement are available for site default options.

The REPORT control statement is similar to that of the Analyze function of FPA.

For the REPORT statement, you can specify the keywords that are shown in the following table. For the format of the REPORT statement, see [“REPORT control statement” on page 856](#).

Table 83. REPORT statement keywords

Keyword	Used in FPXIN DD	Used in DASYSIN DD	Description	Topic
DBRLDIST	Yes (Optional)	No	Specifies whether to generate the DB Record Length Distribution report.	“DBRLDIST keyword” on page 868
IOVFCI_RBA_ REPORT	Yes (Optional)	No	Specifies whether to generate the RBA of Used IOVF CI report.	IOVFCI_RBA_ REPORT keyword
LARGEREC_ COUNT	Yes (Optional)	No	Specifies the number of reporting entries in the Largest Database Records report and the Consolidated Largest Database Records report.	LARGEREC_ COUNT keyword
LARGEREC_ REPORT	Yes (Optional)	No	Specifies whether to generate the Largest Database Records report.	LARGEREC_ REPORT keyword
REPORTDD	Yes (Optional)	No	Specifies the DD name of the output data set or output stream, to which OPC writes the following reports: <ul style="list-style-type: none"> • DEDB Area Analysis reports • Largest Database Records report • UOW report • UOW Exception report 	“REPORTDD keyword” on page 907
SEGLDIST	Yes (Optional)	No	Specifies whether to generate the Segment Length Distribution report.	“SEGLDIST keyword” on page 918
SEGLSTAT	Yes (Optional)	No	Specifies the format of the DB Record Profile Analysis report, which is one of the DEDB Area Analysis reports.	“SEGLSTAT keyword” on page 918

Table 83. REPORT statement keywords (continued)

Keyword	Used in FPXIN DD	Used in DASYSIN DD	Description	Topic
STARTUOW	Yes (Optional)	No	Specifies the number that defines the first UOW to be reported in the UOW report and the UOW Exception report.	“STARTUOW keyword” on page 928
STOPUOW	Yes (Optional)	No	Specifies the number that defines the last UOW to be reported in the UOW report and the UOW Exception report.	“STOPUOW keyword” on page 930
UOW_REPORT	Yes (Optional)	No	Specifies whether to generate the UOW report.	“UOW_REPORT keyword” on page 939

THRESHOLD statement keywords

The THRESHOLD control statement specifies the threshold parameter values, which cause warning messages to be generated when the actual values do not meet the specified criteria. If you do not specify the TYPRUN=RPT keyword and the FULLSTEP=YES keyword, this control statement is ignored.

The keywords associated with the THRESHOLD statement are not available for site default options.

The THRESHOLD control statement is similar to that of the Analyze function of FPA.

For the THRESHOLD statement, you can specify the keywords that are listed in the following table. For the format of the THRESHOLD statement, see [“THRESHOLD control statement” on page 859](#).

Table 84. THRESHOLD statement keywords

Keyword	Used in FPXIN DD	Used in DASYSIN DD	Description	Topic
DOVFFS_PCT	Yes (Optional)	No	Specifies a threshold setting for the percentage of free space in DOVF. If the percentage of free space in DOVF is less than the value that you specified, a warning message is issued.	“DOVFFS_PCT keyword” on page 869
FS_DOVFIOVF	Yes (Optional)	No	Specifies a threshold setting for the free space in DOVF and in IOVF. If the free space in DOVF is greater than the value that you specified, and if the free space in IOVF is less than the value that you specified, a warning message is issued.	“FS_DOVFIOVF keyword” on page 880
FS_RAADOVF	Yes (Optional)	No	Specifies a threshold setting for the free space in RAA BASE and in DOVF. If the free space in RAA BASE is greater than the value that you specified, and if the free space in DOVF is less than the value that you specified, a warning message is issued.	“FS_RAADOVF keyword” on page 880
FS_RAADOVFIOVF	Yes (Optional)	No	Specifies a threshold setting for the free space in RAA BASE, DOVF, and IOVF. If the free space in RAA BASE is greater than the value that you specified, and if the free space in either DOVF or IOVF is less than the value that you specified, a warning message is issued.	“FS_RAADOVFIOVF keyword” on page 881
FS_RAATIOVF	Yes (Optional)	No	Specifies a threshold setting for the free space in RAA BASE and in IOVF. If the free space in RAA BASE is greater than the value that you specified, and if the free space in IOVF is less than the value that you specified, a warning message is issued.	“FS_RAATIOVF keyword” on page 881
IOVFFS_PCT	Yes (Optional)	No	Specifies a threshold setting for the percentage of free space in IOVF. If the percentage is less than the user-specified value, a warning message is issued.	“IOVFFS_PCT keyword” on page 887
IOVFUSED_PCT	Yes (Optional)	No	Specifies a threshold setting for the percentage of IOVF CIs that are used. If the percentage is greater than the value that you specified, a warning message is issued.	“IOVFUSED_PCT keyword” on page 888

Table 84. THRESHOLD statement keywords (continued)

Keyword	Used in FPXIN DD	Used in DASYSIN DD	Description	Topic
RAAFS_PCT	Yes (Optional)	No	Specifies a threshold setting for the percentage of free space in RAA BASE. If the percentage is less than the value that you specified, a warning message is issued.	“RAAFS_PCT keyword” on page 903
RAPOVFL_PCT	Yes (Optional)	No	Specifies a threshold setting for the percentage of RAA BASE CIs that use either DOVF or IOVF. If the percentage is greater than the value that you specified, a warning message is issued.	“RAPOVFL_PCT keyword” on page 903
RECIO_AVG	Yes (Optional)	No	Specifies a threshold setting for the average number of I/Os that can be done to access a database record. If the number is greater than the value that you specified, a warning message is issued.	“RECIO_AVG keyword” on page 905
RECIO_MAX	Yes (Optional)	No	Specifies a threshold setting for the maximum number of I/Os that can be done to access a database record. If the number is greater than the value that you specified, a warning message is issued.	“RECIO_MAX keyword” on page 906
RECIOVF_PCT	Yes (Optional)	No	Specifies a threshold setting for the percentage of records that use IOVF CIs. If the percentage is greater than the value that you specified, a warning message is issued.	“RECIOVF_PCT keyword” on page 906
ROOTIO_AVG	Yes (Optional)	No	Specifies a threshold setting for the average number of I/Os to get access to a root segment. If the number is greater than the value that you specified, a warning message is issued.	“ROOTIO_AVG keyword” on page 910
ROOTIO_MAX	Yes (Optional)	No	Specifies a threshold setting for the maximum number of I/Os that can be done to get access to a root segment. If the number is greater than the value that you specified, a warning message is issued.	“ROOTIO_MAX keyword” on page 911
SYN_LEN_AVG	Yes (Optional)	No	Specifies a threshold setting for the average synonym chain length. If the length is greater than the value that you specified, a warning message is issued.	“SYN_LEN_AVG keyword” on page 930
SYN_MAX	Yes (Optional)	No	Specifies a threshold setting for the maximum synonym chain length. If the length is greater than the value that you specified, a warning message is issued.	“SYN_MAX keyword” on page 931
SYN_PCT_LEN	Yes (Optional)	No	Specifies a threshold setting for the percentage of synonym chains and the synonym chain length. If the percentage of the exceeding synonym chain length that you specified is greater than the value that you specified, a warning message is issued.	“SYN_PCT_LEN keyword” on page 931
UOWDOVF_PCT	Yes (Optional)	No	Specifies a threshold setting for the percentage of UOWs that use DOVF. If the percentage is greater than the value that you specified, a warning message is issued.	“UOWDOVF_PCT keyword” on page 937
UOWIOVF_AVG	Yes (Optional)	No	Specifies a threshold setting for the average number of IOVF CIs that are used by a UOW. If the number is greater than the value that you specified, a warning message is issued.	“UOWIOVF_AVG keyword” on page 937
UOWIOVF_MAX	Yes (Optional)	No	Specifies a threshold setting for the maximum number of IOVF CIs that are used by a UOW. If the number is greater than the value that you specified, a warning message is issued.	“UOWIOVF_MAX keyword” on page 938
UOWIOVF_PCT	Yes (Optional)	No	Specifies a threshold setting for the percentage of UOWs that use IOVF. If the percentage is greater than the value that you specified, a warning message is issued.	“UOWIOVF_PCT keyword” on page 938

Related reference

[Keyword reference for FPO](#)

These topics explain the control statements and keywords of FPO and their syntax rules.

Syntax rules for FPXIN

FPXIN control statements and their associated keywords can be coded in free format in columns 1 through 72, subject to certain syntax rules.

Syntax rules for DASYSIN

Keywords and the associated values can be coded in free format (columns 1 - 72), subject to certain syntax rules.

Related information

Site default options support for FPO

The site default options are control statements that are specified in a member of a partitioned data set and that are used to override the system default option values, which can be changed without assembling and link-editing.

Output for OPC

These topics describe the output (that is, reports) that are generated by OPC, their formats and data elements, and how to get and interpret them.

FPXRPTS DD data set

The FPXRPTS DD data set contains the reports that are generated by OPC. The reports that are generated in this data set differ by the input DD statement that you use.

Subsections:

- [“FPXRPTS DD data set when FPXIN DD is used” on page 702](#)
- [“FPXRPTS DD data set when DASYSIN DD is used” on page 710](#)

FPXRPTS DD data set when FPXIN DD is used

FPXRPTS is the default value for both the RFILE keyword of the ANALYZE control statement and the REPORTDD keyword of the REPORT control statement.

The Online Pointer Checker Statistics report, the Online Pointer Scan report, and the Process Summary report are produced in the data set or in the output stream depending on the RFILE keyword of the ANALYZE control statement.

When either TYPRUN=RPT and FULLSTEP=YES, or TYPRUN=PTRALL and FULLSTEP=YES are specified, the Resolve Pointers report is produced in the data set or in the output stream depending on the RFILE keyword of the ANALYZE control statement.

When the TYPRUN=FS keyword and the FULLSTEP=YES keyword are specified, the following reports are produced in the data set or in the output stream, depending on the REPORTDD keyword of the REPORT control statement:

- Processing Summary of Analysis report
- DEDB Area Analysis report:
 - Freespace Analysis report
 - Summary section
 - Overview and RAA BASE section
 - DOVF and IOVF sections
- UOW report (without DB record information)
- UOW Exception report (without DB record information)
- RBA of Used IOVF CI report

When the TYPRUN=RPT keyword and the FULLSTEP=YES keyword are specified, the following reports are produced in the data set or in the output stream, depending on the REPORTDD keyword of the REPORT control statement:

- Processing Summary of Analysis report
- DEDB Area Analysis reports:
 - Freespace Analysis report
 - Summary section
 - Overview and RAA BASE section
 - DOVF and IOVF sections
 - DB Record Profile Analysis report
 - DB Record Placement Analysis report
 - Segment Placement Analysis report
 - Segment I/O Analysis report
 - Synonym Chain Analysis report
- Largest Database Records report
- Consolidated Largest Database Records report
- DB Record Length Distribution report
- Segment Length Distribution report
- UOW report
- UOW Exception report
- RBA of Used IOVF CI report

When IC=(YES,ICEIN) is specified, the Image Copy NOTIFY report is produced in the data set or in the output stream depending on the RFILE keyword of the ANALYZE control statement. If IMS HP Image Copy processing fails or when the specified area is not registered to DBRC, the Image Copy NOTIFY report is not produced.

The following sections explain each report in detail:

- [“Online Pointer Checker Statistics” on page 703](#)
- [“Online Pointer Scan report” on page 704](#)
- [“Resolve Pointers report” on page 706](#)
- [“Processing Summary of Analysis report” on page 706](#)
- [“DEDB Area Analysis reports” on page 707](#)
- [“Largest Database Records report” on page 707](#)
- [“Consolidated Largest Database Records report” on page 707](#)
- [“DB Record Length Distribution report” on page 707](#)
- [“Segment Length Distribution report” on page 707](#)
- [“UOW report” on page 708](#)
- [“UOW Exception report” on page 708](#)
- [“RBA of Used IOVF CI report” on page 708](#)
- [“Process Summary report” on page 708](#)
- [“Image Copy NOTIFY report” on page 709](#)

Online Pointer Checker Statistics

The Online Pointer Checker Statistics report shows the information of the area definition, and the result of pointer checking.

The following figure shows an example of the report.

```

IMS HPFP UTILITIES - OPC                "ONLINE POINTER CHECKER STATISTICS"                PAGE: 1
5698-FPP V2R1                          2019-11-22 12:36:41

Area Information
Database name..... DEDBJN23
Area name..... DB23AR1
Randomizer name..... RMODTS2
Area ID..... 1
UOW parameter of DBD..... (5,2)
ROOT parameter of DBD..... (95,30)
CI size..... 1024
Number of UOWs in Root Addressable Part..... 65
Number of RAPs in a UOW..... 3
Number of CIs in DOVF of a UOW..... 2
Number of CIs in IOVF..... 150
Number of IOVF Space Map CIs..... 2
Number of Free CIs in IOVF..... 148
RAA RBA Range..... x'00000800' to x'00051C00'
IOVF RBA Range..... x'00051C00' to x'00077400'
REORG UOW Range..... x'00077400' to x'00078800'
SDEP Range..... x'00078800' to x'000F7800'
Global DMAC XVAL..... Cycle Count x'00000001' RBA x'00078800'
Global DMAC NXTS..... Cycle Count x'00000001' RBA x'0007EC00'
Local DMAC NXTS..... Cycle Count x'00000001' RBA x'0007EC00'
SDEP Logical Begin TS..... x'D194CC50C010E341'
End of Area RBA..... x'000F7800'
Segments Defined..... 6
Hierarchical Levels..... 3
Maximum IOA Length..... 900
Image Copy..... 0

```

Figure 289. Online Pointer Checker Statistics report (Part 1 of 2)

```

IMS HPFP UTILITIES - OPC                "ONLINE POINTER CHECKER STATISTICS"                PAGE: 2
5698-FPP V2R1                          2019-11-22 12:36:41

Area level statistics:
Segment Segment Segment Segments
Code Name Hierarchy Read
 1 ROOTSEG1 1 168
 2 SDEPSEG1 2 40 (See "SDEP segment details")
 3 DD1 2 280
 4 DD11 3 261
 5 DD12 3 0
 6 DD2 2 60
-----
809

SDEP segment details (For Segments Read)
Valid segment : 19
Logically deleted segment : 0
In-doubt segment : 0
Aborted segment : 0
SDEP control segment : 21
-----
40

IMS HPFP UTILITIES - OPC                "ONLINE POINTER CHECKER STATISTICS"                PAGE: 3
5698-FPP V2R1                          2019-11-22 12:36:41

PTRSCAN detected errors: 0

Type 10 records written: 1
Type 11 records written: 2
Type 2F records written: 422
Type 2R records written: 201
Type 2T records written: 1
Type 3C0 records written: 120
Type 3C1 records written: 360
Type 3C2 records written: 289
Type 3C3 records written: 241
Type 3C4 records written: 26
Type 3I records written: 0
Type 3R records written: 0
Type 3S records written: 769
Type 4P records written: 19
Type 4S records written: 19

```

Figure 290. Online Pointer Checker Statistics report (Part 2 of 2)

Online Pointer Scan report

The Online Pointer Scan report shows information about pointer errors.

The following figure shows an example of the report.

nn

The segment code.

segname

The segment name.

rba

The relative byte address.

Root key value: *root_key_value*

The value of the root key (in hexadecimal).

Source SC *nn* Name *segmname* RBA *x'rba'*

Information about the source segment.

nn

The segment code.

segname

The segment name.

rba

The relative byte address.

Space map element: *content*

The content of the IOVF space map element in which the problem was found (in hexadecimal).

Space map element offset: *offset*

The offset to the IOVF space map element in which the problem was found (in hexadecimal).

SSP number of the incorrect pointer: *ssp_number*

The SSP number of the subset pointer that has the incorrect pointer.

Target SC *nn* Name *segmname* RBA *x'rba'*

Target SC *nn* Name *segmname* RBA *x'rba'* <- Invalid pointer value.

Information about the target segment.

nn

The segment code.

segname

The segment name.

rba

The relative byte address.

The remainder of the CI is skipped

The Online Pointer Checker skipped checking subsequent pointers in the CI because of the pointer error.

To look up the meaning of each Online Pointer Scan report message, see [“Online Pointer Scan report messages”](#) on page 1425.

Resolve Pointers report

This report contains the processing information of pointer validation.

If the Analyze function of FPA, which is called by OPC internally when FULLSTEP=YES is specified, detects any pointer errors, the error messages are issued in this report.

For DDNAME and DSNAME, N/A is shown because Online Pointer Checker is run for an online DEDB area.

Processing Summary of Analysis report

This report contains the processing information of the analysis report.

The report is produced by the Analyze function of FPA, which is called by OPC internally when TYPRUN=RPT and FULLSTEP=YES are specified, or when TYPRUN=FS and FULLSTEP=YES are specified.

DEDB Area Analysis reports

The DEDB Area Analysis reports include the following reports:

- Freespace Analysis report
- DB Record Profile Analysis report
- DB Record Placement Analysis report
- Segment Placement Analysis report
- Segment I/O Analysis report
- Synonym Chain Analysis report

Note: The header of each report is "DEDB Area Analysis Reports."

The DEDB Area Analysis reports are produced by the Analyze function of FPA, which is called by OPC internally when TYPRUN=RPT and FULLSTEP=YES are specified. When TYPRUN=FS and FULLSTEP=YES are specified, only the Freespace Analysis report is produced.

Largest Database Records report

The optional report for largest database records analysis provides a description for each of the large database records, starting with the largest database record.

The Largest Database Records report is produced by the Analyze function of FPA, which is called by OPC internally when TYPRUN=RPT, FULLSTEP=YES, and LARGEREC_REPORT=YES are specified for the REPORT control statement.

Consolidated Largest Database Records report

You can use this optional report to analyze the largest database records. The report provides information about each of the large database records, starting with the largest database record.

The difference between this report and the Largest Database Records report is that the Largest Database Records report provides information about the large database records for each area whereas the Consolidated Largest Database Records report consolidates information about the large database records found in multiple areas into a single report.

The Consolidated Largest Database Records report is produced by the Analyze function of FPA, which is internally called by OPC, when the following keywords are specified:

- TYPRUN=RPT.
- FULLSTEP=YES.
- LARGEREC_REPORT=YES.
- The second parameter of LARGEREC_COUNT keyword, which specifies the number of reporting entries in the Consolidated Largest Database Record report, specifies a value that is equal to or greater than 1.

DB Record Length Distribution report

This report contains database record information.

This report is produced by the Analyze function of FPA, which is called by OPC internally when the following keywords are specified:

- TYPRUN=RPT
- FULLSTEP=YES
- DBRLDIST=YES

Segment Length Distribution report

This report contains segment length information.

This report is produced by the Analyze function of FPA, which is called by OPC internally when the following keywords are specified:

- TYPRUN=RPT
- FULLSTEP=YES
- SEGLDIST=YES

UOW report

The optional report for UOW analysis provides detailed information about each UOW.

The UOW report is produced by the Analyze function of FPA, which is called by OPC internally when TYPRUN=RPT and FULLSTEP=YES are specified, or when TYPRUN=FS and FULLSTEP=YES are specified.

UOW Exception report

Entries in this report are generated when one of the following user-specified reporting parameters is exceeded:

- The parameter exceeded *nn*% free space in a RAP CI, and overflow CIs are used. The threshold value is defined by the EXC_RBASEFS keyword on the ANALYZE statement (OPC) or the PTRCHK statement (OER).
- The parameter exceeded *nn*% free space in DOVF, and the UOW uses IOVF CIs. The threshold value is defined by the EXC_RDOVFS keyword on the ANALYZE statement (OPC) or the PTRCHK statement (OER).
- The parameter fell below *nn*% free space in the DOVF part of a UOW. The threshold value is defined by the EXC_UDOVFFS keyword on the ANALYZE statement (OPC) or the PTRCHK statement (OER).

The UOW exception report is produced by the Analyze function of FPA, which is called by OPC internally when TYPRUN=RPT and FULLSTEP=YES are specified, or when TYPRUN=FS and FULLSTEP=YES are specified.

RBA of Used IOVF CI report

This optional report for IOVF analysis provides a list of RBAs of used IOVF CIs for each UOW.

The RBA of Used IOVF CI report is produced by the Analyze function of FPA, which is called by OPC internally when TYPRUN=RPT or TYPRUN=FS, FULLSTEP=YES are specified. To request this report, IOVFCI_RBA_REPORT=YES must also be specified for the REPORT control statement.

Process Summary report

The Process Summary report summarizes the result of processing for each area.

The following figure shows an example of the report.

IMS HPFP UTILITIES - OPC		"Process Summary Report"				PAGE: 1
5698-FPP V2R1						2019-11-22 22:09:58
DBDNAME	AREANAME	AREA NO	START DATE/TIME	END DATE/TIME	STATUS (If not blank, see messages in the message data sets.)	
DEDBJN23	DB23AR1	1	2019-11-22 22:10:00	2019-11-22 22:10:12		
DEDBJN23	DB23AR2	2	2019-11-22 22:11:02	2019-11-22 22:11:18	Processing ended with warnings.	
DEDBJN23	DB23AR3	3	2019-11-22 22:12:17	2019-11-22 22:12:18		
DEDBJN23	DB23AR4	4	2019-11-22 22:12:19	2019-11-22 22:12:20		

Figure 292. Process Summary report

DBDNAME

Shows the name of the database.

AREANAME

Shows the name of the area. The area names are displayed in processed order.

AREA NO

Shows the area number.

START DATE/TIME

Shows the time stamp when the area was connected.

END DATE/TIME

Shows the time stamp when the area was disconnected.

STATUS

Shows the processing result for each area.

If OPC ended successfully without errors, this column is blank. Otherwise, one of the following messages is displayed in this column:

Processing ended with warnings.

This message indicates that the processing ended with return code of 4. No pointer errors were detected.

Processing ended with errors.

This message indicates that the processing ended with return code of 8.

Pointer errors were detected.

This message indicates that the processing ended with return code of 4. One or more pointer errors were detected.

The following areas not processed because PROC_AFT_ERROR=STOP.

This message indicates that the processing ended with errors in the previous area. This area and the subsequent areas are not processed because PROC_AFT_ERROR=STOP is specified. The status for the subsequent areas is blank.

Processing did not complete.

This message indicates that the processing for the area did not complete because the processing ended abnormally, or an error, which prevented from processing the subsequent areas, was encountered. The status for the subsequent areas is blank.

Processing was skipped.

This message indicates that the processing for this area was skipped because the area was stopped. This message is displayed only when AREA=ALL is specified.

For details about these messages, see the messages produced in the FPXMFIL data set or in the SYSPRINT data set.

Image Copy NOTIFY report

The Image Copy NOTIFY report contains the NOTIFY.IC control card.

This report is generated when OPC or OER is run with the following specifications:

- OPC with IC=(YES,ICEIN)
- OER running with OPC with PTRCHK=YES and IC=(YES,ICEIN)

The following figure shows an example of the report.

```

IMS HFPF UTILITIES - OPC                                "Image Copy NOTIFY Report"                                PAGE: 1
5698-FPP V2R1                                          2019-11-22 14:38:05

PROCESSING INFORMATION
- NOTIFY.IC command was issued.
- Database Recovery Utility return code: 00
NOTIFY.IC DBD(DEBJN23) AREA(DB23AR1 ) USID(0000000000) -
RUNTIME('2019.326 14:38:09.4 +09:00') -
STOPTIME('2019.326 14:38:09.7 +09:00') -
CIC -
ICDSN(HFPF.DB23AR1.IC) -
FILESEQ(0001) UNIT(3390) -
VOLLIST(AAAAAA) -
RECDCT(0000000130)

```

Figure 293. Image Copy NOTIFY report (OPC)

FPXRPTS DD data set when DASYSIN DD is used

The Online Pointer Scan report is generated in this data set.

Related reference

Resolve Pointers report

The Resolve Pointers report contains the processing information of pointer validation.

Process Summary of Analysis report

The Process Summary of Analysis report contains the processing information of the analysis report.

DEDB Area Analysis reports

The DEDB Area Analysis reports include the comprehensive analysis information of an area.

Largest Database Records report

The optional report for the largest database records analysis provides a description for each of the large database records, starting with the largest database record.

Consolidated Largest Database Records report

You can use this optional report to analyze the largest database records. The report provides information about each of the large database records, starting with the largest database record.

UOW report

The optional report for UOW analysis provides detailed information about each UOW.

UOW Exception report

The UOW Exception report contains the analysis information of the UOW that satisfies the criteria specified by the HFPAPARM keywords.

FPXSPRT DD data set

The FPXSPRT DD data set contains the Sensor Data Statistics report that is generated by OPC when the sensor data is stored in the Sensor Data repository of IMS Tools KB.

This topic describes the FPXSPRT DD data set, which is generated by OPC only when FPXIN DD is used. For OPC with DASYSIN DD, no FPXSPRT DD data set is generated.

Sensor Data Statistics report

The Sensor Data Statistics report contains the names and the values of the data elements that are stored in the Sensor Data repository of IMS Tools KB.

For a list of the data elements that are collected by IMS HP Fast Path Utilities, see the topics about data elements in the *Tools Base Policy Services User's Guide*.

The following figures show an example of the Sensor Data Statistics report. The data elements that are underlined are printed only when SENSOR_DBREC=YES is specified.

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1

=====
AREA DEFINITION
=====

Data Element Name	Value
DB_AREAEDEF_CISIZE	4,096
DB_AREAEDEF_UOW1	20
DB_AREAEDEF_UOW2	10
DB_AREAEDEF_ROOT1	1,500
DB_AREAEDEF_ROOT2	1,000
DB_AREAEDEF_NUM_SDEP_CIS	10,000

=====
FREE SPACE PERCENTAGE STATISTICS
=====

Data Element Name	Value
DB_PCT_BYTES_FS_RAA	50%
DB_PCT_BYTES_FS_DOVF	50%
DB_PCT_BYTES_FS_IOVF	50%
DB_PCT_BYTES_FS_SDEP	52%

=====
OVERFLOW USAGE STATISTICS
=====

Data Element Name	Value
DB_PCT_NUM_UOW_USE_DOVF	50%
DB_AVG_NUM_DOVFCI_BY_UOW	678.99
DB_MAX_NUM_DOVFCI_BY_UOW	9,999
DB_PCT_NUM_UOW_USE_IOVF	50%
DB_NUM_UOW_USE_IOVF	9,999
DB_AVG_NUM_IOVFCI_BY_UOW	678.99
DB_MAX_NUM_IOVFCI_BY_UOW	9,999
DB_MIN_NUM_IOVFCI_BY_UOW	9,999
DB_PCT_NUM_IOVFCI_USED	50%
DB_PCT_NUM_RAPCI_OVFL	50%

=====
SEGMENT OCCURRENCE STATISTICS
=====

Data Element Name	Value
DB_NUM_SEG	12,345,678

Figure 294. Sensor Data Statistics report (Part 1 of 2)

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1

```

=====
DATABASE RECORD STATISTICS
=====
Data Element Name          Value
-----
DB_NUM_ROOT                111,111
DB_AVG_DBREC_LENGTH        111.00
DB_MAX_DBREC_LENGTH        111,111
DB_MIN_DBREC_LENGTH        111,111
DB_PCT_NUM_DBREC_IOVF     50%
=====

RANDOMIZING STATISTICS
=====
Data Element Name          Value
-----
DB_AVG_LEN_SYNONYM_CHAIN   1.11
DB_MAX_LEN_SYNONYM_CHAIN   99,999
=====

PHYSICAL I/O STATISTICS
=====
Data Element Name          Value
-----
DB_ESTIMATED_DBREC_IO      1.25
DB_ESTIMATED_ROOT_IO      1.25
DB_AVG_DBREC_IO            678.99
DB_MAX_DBREC_IO            99,999
DB_AVG_ROOT_IO             678.99
DB_MAX_ROOT_IO             99,999
=====

UOW STATISTICS INFORMATION
=====
Data Element Name          Value
-----
DB_FLAG_UOW_DATA           N
DB_FLAG_UOW_GROUP_DATA    N
DB_THRESHOLD_RBASDFS      30%
DB_THRESHOLD_RDVFFS       50%
DB_NUM_UOW_RFS_COND       50
DB_PCT_NUM_UOW_RFS_COND   10%
=====

```

Figure 295. Sensor Data Statistics report (Part 2 of 2)

The following figures show another example of the Sensor Data Statistics report. In this report, logical pages 3 and 4 contain data element values for the first UOW in the area that is specified by the IAREA keyword. This report is printed if both of the following conditions are met:

- SENSOR_UOW=YES is specified
- FPXAUOWC DD statement is not specified

In actual reports, data element values for other UOWs are printed in the subsequent pages.

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1

=====

AREA DEFINITION

=====

Data Element Name	Value
DB_AREAEDEF_CISIZE	4,096
DB_AREAEDEF_UOW1	20
DB_AREAEDEF_UOW2	10
DB_AREAEDEF_ROOT1	1,500
DB_AREAEDEF_ROOT2	1,000
DB_AREAEDEF_NUM_SDEP_CIS	10,000

=====

FREE SPACE PERCENTAGE STATISTICS

=====

Data Element Name	Value
DB_PCT_BYTES_FS_RAA	50%
DB_PCT_BYTES_FS_DOVF	50%
DB_PCT_BYTES_FS_IOVF	50%
DB_PCT_BYTES_FS_SDEP	52%

=====

OVERFLOW USAGE STATISTICS

=====

Data Element Name	Value
DB_PCT_NUM_UOW_USE_DOVF	50%
DB_AVG_NUM_DOVFCI_BY_UOW	678.99
DB_MAX_NUM_DOVFCI_BY_UOW	9,999
DB_PCT_NUM_UOW_USE_IOVF	50%
DB_NUM_UOW_USE_IOVF	9,999
DB_AVG_NUM_IOVFCI_BY_UOW	678.99
DB_MAX_NUM_IOVFCI_BY_UOW	9,999
DB_MIN_NUM_IOVFCI_BY_UOW	9,999
DB_PCT_NUM_IOVFCI_USED	50%
DB_PCT_NUM_RAPCI_OVFL	50%

=====

SEGMENT OCCURRENCE STATISTICS

=====

Data Element Name	Value
DB_NUM_SEG	12,345,678

Figure 296. Sensor Data Statistics report when SENSOR_UOW=YES (without FPXAUOWC DD) (Part 1 of 3)

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1

=====

DATABASE RECORD STATISTICS

=====

Data Element Name	Value
DB_NUM_ROOT	111,111
DB_AVG_DBREC_LENGTH	111.00
DB_MAX_DBREC_LENGTH	111,111
DB_MIN_DBREC_LENGTH	111,111
DB_PCT_NUM_DBREC_IOVF	50%

=====

RANDOMIZING STATISTICS

=====

Data Element Name	Value
DB_AVG_LEN_SYNONYM_CHAIN	1.11
DB_MAX_LEN_SYNONYM_CHAIN	99,999

=====

PHYSICAL I/O STATISTICS

=====

Data Element Name	Value
DB_ESTIMATED_DBREC_IO	1.25
DB_ESTIMATED_ROOT_IO	1.25
DB_AVG_DBREC_IO	678.99
DB_MAX_DBREC_IO	99,999
DB_AVG_ROOT_IO	678.99
DB_MAX_ROOT_IO	99,999

=====

UOW STATISTICS INFORMATION

=====

Data Element Name	Value
DB_FLAG_UOW_DATA	Y
DB_FLAG_UOW_GROUP_DATA	N
DB_THRESHOLD_RBASEFS	30%
DB_THRESHOLD_RDOVFFS	50%
DB_NUM_UOW_RFS_COND	50
DB_PCT_NUM_UOW_RFS_COND	10%

=====

REPOSITORY GROUP INFORMATION

=====

Data Element Name	Value
DB_SENSOR_DATA_GROUP_ID	IMS1

Figure 297. Sensor Data Statistics report when SENSOR_UOW=YES (without FPXAUOWC DD) (Part 2 of 3)

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1, UOW NUMBER: 0

=====

FREE SPACE PERCENTAGE STATISTICS

=====

Data Element Name	Value
DBU_PCT_BYTES_FS_RAA	80%
DBU_PCT_BYTES_FS_DOVF	80%
DBU_PCT_BYTES_FS_IOVF	95%
DBU_PCT_USABLE_RAAFS	80%
DBU_PCT_USABLE_DOVFFS	80%
DBU_PCT_USABLE_IOVFFS	95%
DBU_PCT_RAP_ROOTSZFS	0%
DBU_MAX_PCT_BYTES_RAPFS	0%
DBU_FLAG_UOW_USING_OVFL	Y
DBU_FLAG_UOW_USING_IOVF	Y

=====

OVERFLOW USAGE STATISTICS

=====

Data Element Name	Value
DBU_NUM_DOVFCI_BY_UOW	10
DBU_NUM_IOVFCI_BY_UOW	3
DBU_PCT_NUM_RAPCI_OVFL	100%

=====

DATABASE RECORD STATISTICS

=====

Data Element Name	Value
DBU_NUM_ROOT	120
DBU_AVG_DBREC_LENGTH	106.00
DBU_MAX_DBREC_LENGTH	582
DBU_MIN_DBREC_LENGTH	102
DBU_PCT_NUM_DBREC_IOVF	18%

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1, UOW NUMBER: 0

=====

RANDOMIZING STATISTICS

=====

Data Element Name	Value
DBU_AVG_LEN_SYN_CHAIND	120.00
DBU_MAX_LEN_SYN_CHAIN	120

=====

PHYSICAL I/O STATISTICS

=====

Data Element Name	Value
DBU_AVG_DBREC_IO	7.06
DBU_MAX_DBREC_IO	13
DBU_AVG_ROOT_IO	7.06
DBU_MAX_ROOT_IO	13

Figure 298. Sensor Data Statistics report when SENSOR_UOW=YES (without FPXAUOWC DD) (Part 3 of 3)

The following figures show another example of the Sensor Data Statistics report. In this report, logical pages 3 and 4 contain data element values for a UOW group that is specified by the HFPXAUOWC DD statement. This report is printed if both of the following conditions are met:

- SENSOR_UOW=YES is specified
- FPXAUOWC DD statement is specified

In actual reports, if multiple UOW groups are specified, data element values for other UOW groups are printed in the subsequent pages.

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1

```
=====
AREA DEFINITION
=====
```

Data Element Name	Value
DB_AREAEDEF_CISIZE	4,096
DB_AREAEDEF_UOW1	20
DB_AREAEDEF_UOW2	10
DB_AREAEDEF_ROOT1	1,500
DB_AREAEDEF_ROOT2	1,000
DB_AREAEDEF_NUM_SDEP_CIS	10,000

```
=====
FREE SPACE PERCENTAGE STATISTICS
=====
```

Data Element Name	Value
DB_PCT_BYTES_FS_RAA	50%
DB_PCT_BYTES_FS_DOVF	50%
DB_PCT_BYTES_FS_IOVF	50%
DB_PCT_BYTES_FS_SDEP	52%

```
=====
OVERFLOW USAGE STATISTICS
=====
```

Data Element Name	Value
DB_PCT_NUM_UOW_USE_DOVF	50%
DB_AVG_NUM_DOVFCI_BY_UOW	678.99
DB_MAX_NUM_DOVFCI_BY_UOW	9,999
DB_PCT_NUM_UOW_USE_IOVF	50%
DB_NUM_UOW_USE_IOVF	9,999
DB_AVG_NUM_IOVFCI_BY_UOW	678.99
DB_MAX_NUM_IOVFCI_BY_UOW	9,999
DB_MIN_NUM_IOVFCI_BY_UOW	9,999
DB_PCT_NUM_IOVFCI_USED	50%
DB_PCT_NUM_RAPCI_OVFL	50%

```
=====
SEGMENT OCCURRENCE STATISTICS
=====
```

Data Element Name	Value
DB_NUM_SEG	12,345,678

Figure 299. Sensor Data Statistics report when SENSOR_UOW=YES (with FPXAUOWC DD) (Part 1 of 3)

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1

=====

DATABASE RECORD STATISTICS

=====

Data Element Name	Value
DB_NUM_ROOT	111,111
DB_AVG_DBREC_LENGTH	111.00
DB_MAX_DBREC_LENGTH	111,111
DB_MIN_DBREC_LENGTH	111,111
DB_PCT_NUM_DBREC_IOVF	50%

=====

RANDOMIZING STATISTICS

=====

Data Element Name	Value
DB_AVG_LEN_SYNONYM_CHAIN	1.11
DB_MAX_LEN_SYNONYM_CHAIN	99,999

=====

PHYSICAL I/O STATISTICS

=====

Data Element Name	Value
DB_ESTIMATED_DBREC_IO	1.25
DB_ESTIMATED_ROOT_IO	1.25
DB_AVG_DBREC_IO	678.99
DB_MAX_DBREC_IO	99,999
DB_AVG_ROOT_IO	678.99
DB_MAX_ROOT_IO	99,999

=====

UOW STATISTICS INFORMATION

=====

Data Element Name	Value
DB_FLAG_UOW_DATA	N
DB_FLAG_UOW_GROUP_DATA	Y
DB_NUM_UOW_GROUPS	2
DB_THRESHOLD_RBASEFS	30%
DB_THRESHOLD_RDOVFFS	50%
DB_NUM_UOW_RFS_COND	50
DB_PCT_NUM_UOW_RFS_COND	10%

=====

REPOSITORY GROUP INFORMATION

=====

Data Element Name	Value
DB_SENSOR_DATA_GROUP_ID	IMS1

Figure 300. Sensor Data Statistics report when SENSOR_UOW=YES (with FPXAUOWC DD) (Part 2 of 3)

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1, UOW GROUP (STARTUOW: 10001, STOPUOW: 20000)

=====

FREE SPACE PERCENTAGE STATISTICS

=====

Data Element Name	Value
DEBUG_PCT_BYTES_FS_RAA	80%
DEBUG_PCT_BYTES_FS_DOVF	80%
DEBUG_PCT_BYTES_FS_IOVF	95%
DEBUG_PCT_USABLE_RAAFS	80%
DEBUG_PCT_USABLE_DOVFFS	80%
DEBUG_PCT_USABLE_IOVFFS	95%
DEBUG_PCT_RAP_ROOTSZFS	0%

=====

OVERFLOW USAGE STATISTICS

=====

Data Element Name	Value
DEBUG_PCT_NUM_UOW_DOVF	80%
DEBUG_AVG_NUM_DOVFCCI	5
DEBUG_MAX_NUM_DOVFCCI	10
DEBUG_PCT_NUM_UOW_IOVF	80%
DEBUG_AVG_NUM_IOVFCCI	10
DEBUG_MAX_NUM_IOVFCCI	50
DEBUG_PCT_NUM_RAPCI_OVFL	100%

=====

DATABASE RECORD USAGE STATISTICS

=====

Data Element Name	Value
DEBUG_NUM_ROOT	120
DEBUG_AVG_DBREC_LENGTH	106.00
DEBUG_MAX_DBREC_LENGTH	582
DEBUG_MIN_DBREC_LENGTH	102
DEBUG_PCT_NUM_DBREC_IOVF	18%

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1, UOW GROUP (STARTUOW: 10001, STOPUOW: 20000)

=====

RANDOMIZING STATISTICS

=====

Data Element Name	Value
DEBUG_AVG_LEN_SYN_CHAIN	120.00
DEBUG_MAX_LEN_SYN_CHAIN	120

=====

PHYSICAL I/O STATISTICS

=====

Data Element Name	Value
DEBUG_AVG_DBREC_IO	7.06
DEBUG_MAX_DBREC_IO	13
DEBUG_AVG_ROOT_IO	7.06
DEBUG_MAX_ROOT_IO	13

Figure 301. Sensor Data Statistics report when SENSOR_UOW=YES (with FPXAUOWC DD) (Part 3 of 3)

SYSPRINT/FPXMFIL DD data set

The IMS system messages and control statement analysis messages are produced in the system output stream that is specified on the SYSPRINT DD statement.

OPC messages (except control statement analysis messages) can be routed to another output data set or to the system output stream by coding the MFILE parameter of the ANALYZE control statement. If the MFILE parameter is not specified, they are produced in the default message data set FPXMFIL for OPC with FPXIN DD, or in SYSPRINT for OPC with DASYSIN DD.

SNAPPIT DD data set

The SNAPPIT data set contains the CI SNAP dump produced by the OPC utility.

When FULLSTEP=NO and SNAPPIT DD are specified, and if any pointer errors are detected in the area, the CI Map/CI Dump report is produced (by OPC) in the SNAPPIT data set.

When the TYPRUN=RPT keyword, FULLSTEP=YES keyword, and SNAPPIT DD are specified, and if any pointer errors are detected in the area, the following reports are produced in the SNAPPIT data set:

- Pointer Chain Reconstruction report (produced by the Analyze function of FPA)
- CI Map/CI Dump report (produced by OPC and the Analyze function of FPA)

Subsections:

- [“Pointer Chain Reconstruction report” on page 719](#)
- [“CI MAP/CI Dump report” on page 719](#)

Pointer Chain Reconstruction report

This report shows the following information:

- A list of all pointers to user-specified targets that were detected during the analysis
- CI dump number of the CI dump that contains the user-specified target

This report is produced by the Analyze function of FPA, which is called by OPC internally when FULLSTEP=YES and SNAPPIT DD is specified.

CI MAP/CI Dump report

The CI Map/CI Dump report is used to analyze DEDB area CIs to determine the best way to repair them. The report provides the following information:

- CI dump that is produced during online pointer check
- CI map and CI dump that is produced during in-depth pointer check of the Analyze function of FPA when FULLSTEP=YES is specified:
 - CI map, which is a list of CI headers, all the segments, free space elements, scrap bytes, unknown area, and CI trailer that the Analyze process found in the CI
 - CI dump that is logically formatted based on the CI map

The following figure shows an example of a CI Dump report that is produced during online pointer check.

```

IMS HFPF UTILITIES - OPC                                " CI dump report"                                PAGE: 1
5698-FPP V2R1                                          2019-11-22 12:36:41

SNAP OF ERROR CI AT RBA X'00008400'
00000000 00E51005 C4C2E3F1 40404040 02000000 00010000 800C0062 20220000 02002DE2 |.V...DBT1 .....S|
00000020 F1F2F2F0 F0F0F0F0 F2C1F0F2 4060C9D5 C9E3C9C1 D340D3D6 C1C46040 E2C5D84B |122000002A02 -INITIAL LOAD- SEQ.|
00000040 40C4C5D7 4B40E2C3 C1D5C6C9 C5D3C440 40404040 40404040 40404040 40404040 |DEP. SCANFIELD .....^WanC y.....|
00000060 40404040 40404040 40404040 40404040 B0E68195 C36AA804 02000000 00010000 |d.....S122000002A03 -INITIA|
00000080 840C0062 20220000 02002DE2 F1F2F2F0 F0F0F0F0 F2C1F0F3 4060C9D5 C9E3C9C1 |L LOAD- SEQ. DEP. SCANFIELD ^Wan|
000000A0 D340D3D6 C1C46040 E2C5D84B 40C4C5D7 4B40E2C3 C1D5C6C9 C5D3C440 40404040 |C y.....|
000000C0 40404040 40404040 40404040 40404040 40404040 40404040 B0E68195 |.....|
000000E0 C36AA804 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
00000100 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
00000120 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
00000140 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
00000160 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
00000180 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
000001A0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
000001C0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
000001E0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
00000200 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
00000220 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
00000240 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
00000260 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
00000280 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
000002A0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
000002C0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
000002E0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
00000300 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
00000320 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
00000340 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
00000360 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
00000380 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
000003A0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
000003C0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....|
000003E0 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 |.....d...9.9...|

```

Figure 302. CI Dump report produced during online pointer check

The following figure shows an example of a CI Map/CI Dump report that is produced during in-depth pointer check of the Analyze function of FPA.

IMS HPFP UTILITIES - FPA ANALYZE										"CI map / CI dump report"										PAGE: 1	
5698-FPP V2R1																				2019-11-22 23:02:23	
DBNAME: DEDBJN22		AREA#: 00001		AREANAME: DB22AR0		DSNAME: HPFP01.HFPREG.IMS81TMP.MDB22AR0.ADS1		DUMP NO. 0001													
CI-NO:	STORAGE	TYPE(HEX)	RBA	ADDRESS	FLAGS	SC	COUNTERS	LENGTH	ROOT#	TYPE(CHAR)											
BACB000	C8C5C1C4C5D94040		00001800	0BAAE0C8	000000	00	0000000000000000	0008	0000	(HEADER)											
BACB02C	D9D6D6E3E2C5C7F1		00001808	0BAAE0D0	000000	01	0000000000000000	01A6	0000	(ROOTSEG1)											
BACB058	C6E2C54040404040		000019AE	0BAAE276	000000	00	0000000000000000	0052	0000	(FSE)											
BACB084	C4C4F24040404040		00001A00	0BAAE2C8	000000	04	0000000000000000	003D	0000	(DD2)											
BACB0B0	C4C4F14040404040		00001A3D	0BAAE305	000000	03	0000000000000000	003D	0000	(DD1)											
BACB0DC	C4C4F34040404040		00001A7A	0BAAE342	000000	05	0000000000000000	003D	0000	(DD3)											
BACB108	C4C4F34040404040		00001AB7	0BAAE37F	000000	05	0000000000000000	003D	0000	(DD3)											
BACB134	C4C4F14040404040		00001AF4	0BAAE3BC	000000	03	0000000000000000	003D	0000	(DD1)											
BACB160	C4C4F24040404040		00001B31	0BAAE3F9	000000	04	0000000000000000	003D	0000	(DD2)											
BACB18C	C4C4F34040404040		00001B6E	0BAAE436	000000	05	0000000000000000	003D	0000	(DD3)											
BACB1B8	C4C4F24040404040		00001BAB	0BAAE473	000000	04	0000000000000000	003D	0000	(DD2)											
BACB1E4	C6E2C54040404040		00001BE8	0BAAE4B0	000000	00	0000000000000000	000B	0000	(FSE)											
BACB210	E3D9C1C9D3C5D940		00001BF3	0BAAE4BB	000000	00	0000000000000000	000D	0000	(TRAILER)											

IMS HPFP UTILITIES - FPA ANALYZE										"CI map / CI dump report"										PAGE: 2	
5698-FPP V2R1																				2019-11-22 23:02:23	
DBNAME: DEDBJN22		AREA#: 00001		AREANAME: DB22AR0		DSNAME: HPFP01.HFPREG.IMS81TMP.MDB22AR0.ADS1		DUMP NO. 0001													
CI-NO:	RBA	OFFSET																			
1800	0000	03E80202	00006000																		
1808	0008	01FF0000	00000000	00000000	00000000	5F980000	5F5B0000	60080000	00000000	*.Y....-	*										
0028	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000168	*.....?..?\$.-.....*	*										
0048	10220000	041D09F1	F2F2F0F0	F0F0F0F0	F4C44060	C9D5C9E3	60D3D6C1	C440E4D6	*.....R1220000004D -INIT-LOAD UO*	*											
0068	E64840F0	40C14BD7	4B40F040	F140C4C5	D74BD6C6	D3E64040	40404040	40404040	*W. 0 A.P. 0 1 DEP.OFLW	*											
0088	40404040	40404040	40404040	40404040	40404040	40404040	40404040	40404040	*	*											
LINES	000018A8-00001988	SAME AS ABOVE																			
01A8	40404040	4040																			
19AE	01AE	80000052	00000000	00000000	00000000	00000000	00000000	00000000	00000000	*.....*	*										
01CE	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	*.....*	*										
01EE	00000000	00000000	00000000	00000000	00000000	0000															
1A00	0200	04000000	5E2A0037	32220000	02006DC4	F2F2F2F0	F0F0F0F0	F2C1F0F6	4060C9D5	*.....;.....D222000002A06 -IN*	*										
0220	C9E3C9C1	D340D3D6	C1C46040	C4C4F240	40404040	40404040	40404040	40404040	40	*ITIAL LOAD- DD2	*										
1A3D	023D	03000000	5E670037	31220000	02006DC4	F1F2F2F0	F0F0F0F0	F2C1F0F6	4060C9D5	*.....;.....D122000002A06 -IN*	*										
025D	C9E3C9C1	D340D3D6	C1C46040	C4C4F140	40404040	40404040	40404040	40404040	40	*ITIAL LOAD- DD1	*										
1A7A	027A	05000000	00000037	33220000	02006DC4	F3F2F2F0	F0F0F0F0	F2C1F0F6	4060C9D5	*.....;.....D322000002A06 -IN*	*										
029A	C9E3C9C1	D340D3D6	C1C46040	C4C4F340	40404040	40404040	40404040	40404040	40	*ITIAL LOAD- DD3	*										
1AB7	02B7	05000000	1A7A0037	33220000	02004DC4	F3F2F2F0	F0F0F0F0	F2C1F0F4	4060C9D5	*.....;.....(D322000002A04 -IN*	*										
02D7	C9E3C9C1	D340D3D6	C1C46040	C4C4F340	40404040	40404040	40404040	40404040	40	*ITIAL LOAD- DD3	*										
1AF4	02F4	03000000	1A3D0037	31220000	02004DC4	F1F2F2F0	F0F0F0F0	F2C1F0F4	4060C9D5	*.....;.....(D122000002A04 -IN*	*										
0314	C9E3C9C1	D340D3D6	C1C46040	C4C4F140	40404040	40404040	40404040	40404040	40	*ITIAL LOAD- DD1	*										
1B31	0331	04000000	1A000037	32220000	02004DC4	F2F2F2F0	F0F0F0F0	F2C1F0F4	4060C9D5	*.....;.....(D222000002A04 -IN*	*										
0351	C9E3C9C1	D340D3D6	C1C46040	C4C4F240	40404040	40404040	40404040	40404040	40	*ITIAL LOAD- DD2	*										
1B6E	036E	05000000	1AB70037	33220000	02002DC4	F3F2F2F0	F0F0F0F0	F2C1F0F2	4060C9D5	*.....;.....D322000002A02 -IN*	*										
038E	C9E3C9C1	D340D3D6	C1C46040	C4C4F340	40404040	40404040	40404040	40404040	40	*ITIAL LOAD- DD3	*										

Figure 303. CI Map/CI Dump report produced during in-depth pointer check of the Analyze function of FPA

Related reference

Pointer Chain Reconstruction report

The Pointer Chain Reconstruction report contains the pointer information and the CI dump number.

CI Map/CI Dump report

The CI Map/CI Dump report is used to analyze DEDB area CIs to determine the best way to repair them.

Examples for OPC

There are many ways to run the OPC utility. The examples provided in the following topics show some of the typical ways to run the OPC utility.

You can find additional JCL examples that help you to monitor and tune DEDB areas in “Online Pointer Checker JCL examples for monitoring and tuning DEDBs” on page 1157.

Example 1: Running fast scan of database integrity

The following figure shows example JCL for running fast scan of database integrity in an online area.

No analysis reports are generated for this example.

```

//ANALYZE EXEC FPUTIL, DBD=DEDBJN23, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//        DD DISP=SHR, DSN=IMSVS.SDFSRESL
//FPXIN   DD *
ANALYZE  AREA=DB23AR0
          TYPRUN=PTRSCAN
/*

```

Figure 304. Running fast scan of database integrity

The SYSPRINT and SYSUDUMP DD statements are coded in the FPUTIL cataloged procedure. The FPXRPTS DD is dynamically allocated to SYSOUT=*.

This example JCL stream runs similarly to the following JCL stream, which is for OPC with SYSIN and DASYSIN DD.

```

//ANALYZE EXEC FPUTIL, DBD=DEDBJN23, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//        DD DISP=SHR, DSN=IMSVS.SDFSRESL
//SYSIN   DD *
TYPE TOOL
EXIT ANALYZE
AREA DB23AR0
GO
/*
//DASYSIN DD *
TYPRUN=PTRSCAN
/*
//SYSOUT  DD SYSOUT=*

```

Figure 305. Running fast scan of database integrity with SYSIN and DASYSIN DD

Example 2: Running fast scan of database integrity and collecting pointer verification data

The following figure shows example JCL for running fast scan of database integrity and collecting pointer verification data in an online area.

No analysis reports are generated for this example. The pointer verification record data is generated for further analysis.

```

//ANALYZE EXEC FPUTIL, DBD=DEDBJN23, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//        DD DISP=SHR, DSN=IMSVS.SDFSRESL
//FPXIN   DD *
ANALYZE  AREA=DB23AR0
          TYPRUN=PTRALL
          SDEP=YES
/*
//DAR13S0 DD DISP=(NEW, CATLG), DSN=HPFP.TSSDA.DA13S, UNIT=SYSDA,
//        SPACE=(CYL, (50, 5), RLSE)
//DAR3CS0 DD DISP=(NEW, CATLG), DSN=HPFP.TSSDA.DA3CS, UNIT=SYSDA,
//        SPACE=(CYL, (50, 5), RLSE)
//SORT3CS DD DISP=(NEW, PASS), DSN=&SORT3CS, UNIT=SYSDA,
//        SPACE=(TRK, (1, 1), RLSE)

```

Figure 306. Running fast scan of database integrity and collecting pointer verification data

The SYSPRINT and SYSUDUMP DD statements are coded in the FPUTIL cataloged procedure. The FPXRPTS DD is dynamically allocated to SYSOUT=*.

This example JCL stream runs similarly to the following JCL stream, which is for OPC with SYSIN and DASYSIN DD.

```

//ANALYZE EXEC FPUTIL, DBD=DEDBJN23, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//        DD DISP=SHR, DSN=IMSVS.SDFSRESL
//SYSIN   DD *
TYPE TOOL
EXIT ANALYZE
AREA DB23AR0
GO
/*
//DASYSIN DD *
TYPRUN=PTRALL, SDEP=YES
/*
//DAR13SO DD DISP=(NEW, CATLG), DSN=HPFP.TSSDA.DA13S, UNIT=SYSDA,
//        SPACE=(CYL, (50, 5), RLSE)
//DAR3CSO DD DISP=(NEW, CATLG), DSN=HPFP.TSSDA.DA3CS, UNIT=SYSDA,
//        SPACE=(CYL, (50, 5), RLSE)
//SORT3CS DD DISP=(NEW, PASS), DSN=&SORT3CS, UNIT=SYSDA,
//        SPACE=(TRK, (1, 1), RLSE)
//SYSOUT  DD SYSOUT=*

```

Figure 307. Running fast scan of database integrity and collecting pointer verification data with SYSIN and DASYSIN DD

Example 3: Running cross-reference validation of pointers

The following figure shows example JCL for running cross-reference validation of pointers.

No pointer verification record data is generated.

```

//ANALYZE EXEC FPUTIL, DBD=DEDBJN23, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//        DD DISP=SHR, DSN=IMSVS.SDFSRESL
//FPXIN   DD *
ANALYZE AREA=DB23AR0
        FULLSTEP=YES
        TYPRUN=PTRALL
        SDEP=YES
/*
//FPXRPTS DD SYSOUT=*
//SYSOUT  DD SYSOUT=*

```

Figure 308. Running cross-reference validation of pointers

The SYSPRINT and SYSUDUMP DD statements are coded in the FPUTIL cataloged procedure.

Example 4: Running cross-reference validation of pointers and generating analysis reports

The following figure shows example JCL for running cross-reference validation of pointers in an online area, and generating analysis reports.

No pointer verification record data is generated.

```

//ANALYZE EXEC FPUTIL, DBD=DEDBJN23, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//        DD DISP=SHR, DSN=IMSVS.SDFSRESL
//FPXIN   DD *
ANALYZE AREA=DB23AR0
        FULLSTEP=YES
        TYPRUN=RPT
        SDEP=YES
REPORT  REPORTDD=FPXRPTS
THRESHOLD DOVFFS_PCT=25
/*
//FPXRPTS DD SYSOUT=*
//SYSOUT  DD SYSOUT=*

```

Figure 309. Running cross-reference validation of pointers and generating analysis reports

The SYSPRINT and SYSUDUMP DD statements are coded in the FPUTIL cataloged procedure.

Example 5: Collecting segment information records for database tuning

The following figure shows example JCL for collecting segment information records from an online area. The collected records can be used to tune the database by using DEDB Tuning Aid.

Tip: If you want to tune the database by using the FPA Tune function instead of DEDB Tuning Aid, see “Example 5: Simulating changes in area definitions of online areas” on page 257.

```
//*****  
//** OPC TYPRUN=MODEL **  
//*****  
//ANALYZE EXEC FPUTIL,DBD=DEDBJN23,IMSID=IMS1  
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0  
// DD DISP=SHR,DSN=IMSVS.SDFSRESL  
//FPXRPTS DD SYSOUT=A  
//FPXMFIL DD SYSOUT=A  
//SYSPRINT DD SYSOUT=A  
//FPXIN DD *  
ANALYZE AREA=DB23AR1  
FULLSTEP=NO  
TYPRUN=MODEL  
SDEP=YES  
REPORT REPORTDD=FPXRPTS  
  
/*  
//DAR120 DD DISP=(NEW,CATLG),DSN=HPFP.TSSDA.DA12,  
// UNIT=SYSDA,SPACE=(CYL,(1,1),RLSE)  
//DAR13SO DD DISP=(NEW,CATLG),DSN=HPFP.TSSDA.DA13S,  
// UNIT=SYSDA,SPACE=(CYL,(1,1),RLSE)  
//DAR3CSO DD DISP=(NEW,CATLG),DSN=HPFP.TSSDA.DA3CS,  
// UNIT=SYSDA,SPACE=(CYL,(1,1),RLSE)  
//SORT12 DD DISP=(NEW,CATLG),DSN=HPFP.TSSDA.SORT12,  
// UNIT=SYSDA,SPACE=(CYL,(1,1),RLSE)  
//SORT3CS DD DISP=(NEW,CATLG),DSN=HPFP.TSSDA.SORT3CS,  
// UNIT=SYSDA,SPACE=(CYL,(1,1),RLSE)  
//*  
//*****  
//** SORT DA3CS RECORDS **  
//*****  
//SORT3CS EXEC PGM=SORT,REGION=2048K,PARM='CORE=MAX'  
//SYSOUT DD SYSOUT=A  
//SYSIN DD DISP=SHR,DSN=HPFP.TSSDA.SORT3CS  
//SORTIN DD DISP=(OLD,DELETE,KEEP),DSN=HPFP.TSSDA.DA3CS  
//SORTOUT DD DISP=(NEW,CATLG,DELETE),DSN=HPFP.TSSDA.SDA3CS,  
// UNIT=SYSDA,SPACE=(CYL,(1,1),RLSE)  
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(2,2))  
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(2,2))  
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(2,2))  
//*  
//*****  
//** RESOLVE 3,C AND 3,S RECORDS **  
//*****  
//FABADA3 EXEC PGM=FABADA3,REGION=768K  
//SYSPRINT DD SYSOUT=A  
//DAR13SI DD DISP=SHR,DSN=HPFP.TSSDA.DA13S  
//DAR3CSI DD DISP=SHR,DSN=HPFP.TSSDA.SDA3CS  
//DAR4ERRS DD DISP=(NEW,CATLG),DSN=HPFP.TSSDA.DA4ERRS,  
// UNIT=SYSDA,SPACE=(CYL,(2,2),RLSE)  
//DAR130 DD DISP=(NEW,CATLG),DSN=HPFP.TSSDA.RDA13,  
// UNIT=SYSDA,SPACE=(CYL,(1,1),RLSE)  
//SORT4CTL DD DISP=(NEW,PASS),DSN=&SORT4,  
// UNIT=SYSDA,SPACE=(TRK,(2,1),RLSE)  
//SORTR13 DD DISP=(NEW,PASS),DSN=&SORTR13,  
// UNIT=SYSDA,SPACE=(TRK,(2,1),RLSE)  
//*
```

Figure 310. Collecting segment information records for database tuning with DEDB Tuning Aid (Part 1 of 2)

```

//*****
//** SORT DA4ERRS RECORDS          **
//*****
//SORT4 EXEC PGM=SORT,REGION=2048K,PARM='CORE=MAX',
// COND=(8,GE,FABADA3)
//SYSOUT DD SYSOUT=A
//SYSIN DD DISP=(OLD,DELETE),DSN=&SORT4
//SORTIN DD DISP=(OLD,DELETE,KEEP),DSN=HPFP.TSSDA.DA4ERRS
//SORTOUT DD DISP=(NEW,CATLG,DELETE),DSN=HPFP.TSSDA.SDA4ERRS,
// UNIT=SYSDA,SPACE=(TRK,(2,1),RLSE)
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//*
//*****
//** GENERATE CI MAP/DUMP & REPORT **
//** FOR IMAGE COPY INPUT USE: //DFSUDUMP DD DSN= **
//** FOR VSAM INPUT USE: //DARVSAM DD DSN= **
//** ** **
//** ERROR DATA **
//** FOR SYSTEM INPUT USE: //DAR4ERRS DD DSN= **
//** PARM='DAR4ERRS' **
//** OR PARM=' ' **
//** FOR USER INPUT USE: //DAR4USER DD DSN= **
//** PARM='DAR4USER' **
//** FOR BOTH INPUT USE: //DAR4ERRS DD DSN= **
//** //DAR4USER DD DSN= **
//** PARM='BOTH' **
//*****
//FABADA4 EXEC PGM=FABADA4,REGION=1200K,COND=(8,GE,FABADA3)
//ACBLIB DD DISP=SHR,DSN=IMSVS.ACBLIB
//DARVSAM DD DISP=SHR,DSN=HPFP.ONLINE.TSSAR01
//SYSOUT DD SYSOUT=A
//SYSPRINT DD SYSOUT=A
//REPORTS DD SYSOUT=A
//SNAPPIT DD SYSOUT=A
//DAR4ERRS DD DISP=(OLD,DELETE,KEEP),DSN=HPFP.TSSDA.SDA4ERRS
//DAR13SI DD DISP=(OLD,DELETE,KEEP),DSN=HPFP.TSSDA.DA13S
//SORTIN DD DISP=(OLD,DELETE,KEEP),DSN=HPFP.TSSDA.SDA3CS
//SORT3CS DD DISP=SHR,DSN=HPFP.TSSDA.SORT3CS
//SORTOUT DD DISP=(NEW,DELETE),DSN=&SDA3CI,
// UNIT=SYSDA,SPACE=(TRK,(2,1),RLSE)
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(2,2))
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(2,2))
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(2,2))
//*
//*****
//** SORT DA13 RECORDS          **
//*****
//SORTR13 EXEC PGM=SORT,REGION=2048K,PARM='CORE=MAX'
//SYSOUT DD SYSOUT=A
//SYSIN DD DISP=(OLD,DELETE),DSN=&SORTR13
//SORTIN DD DISP=(OLD,DELETE,KEEP),DSN=HPFP.TSSDA.RDA13
//SORTOUT DD DISP=(NEW,CATLG,DELETE),DSN=HPFP.TSSDA.SRDA13,
// UNIT=SYSDA,SPACE=(CYL,(1,1),RLSE)
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(2,2))
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(2,2))
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(2,2))
//*
//

```

Figure 311. Collecting segment information records for database tuning with DEDB Tuning Aid (Part 2 of 2)

The following example JCL is for tuning a database by using DEDB Tuning Aid.

```

//*****
//** EXTRACT/FORMAT SEGMENT DATA **
//*****
//FABBME1 EXEC PGM=FABBME1,REGION=768K
//SYSPRINT DD SYSOUT=A
//ACBLIB DD DISP=SHR,DSN=IMSVS.TEST.ACBLIB
//RMODLIB DD DISP=SHR,DSN=IMSVS.PGMLIB
//MEDARI DD DISP=OLD,DSN=HPFP.TSSDA.SRDA13
//MESORTCD DD DISP=(NEW,PASS),DSN=&&SORT13,
// UNIT=SYSDA,SPACE=(TRK,(1,1))
//MEDARO DD DISP=(NEW,CATLG),DSN=HPFP.TSSME.ME13,
// UNIT=SYSDA,SPACE=(CYL,(2,1),RLSE)
//SYSIN DD *,DCB=BLKSIZE=80
DBDNAME=DEDBJN23,AREA=1,RPT=YES
/*
//*****
//** SORT ME13 RECORDS **
//*****
//SORT13 EXEC PGM=SORT,PARM='CORE=MAX',COND=(4,LT),
// REGION=2048K
//SYSOUT DD SYSOUT=A
//SYSIN DD DISP=(OLD,DELETE),DSN=&&SORT13
//SORTIN DD DISP=(OLD,DELETE,KEEP),DSN=HPFP.TSSME.ME13
//SORTOUT DD DISP=(NEW,CATLG,DELETE),DSN=HPFP.TSSME.SME13,
// UNIT=SYSDA,SPACE=(CYL,(2,1),RLSE)
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(2,3))
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(2,3))
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(2,3))
/*
//*****
//** BUILD AREA MODEL **
//*****
//FABBME3 EXEC PGM=FABBME3,PARM='L',REGION=768K,COND=(4,LT)
//SYSPRINT DD SYSOUT=A
//MEDARI DD DISP=(OLD,DELETE,KEEP),DSN=HPFP.TSSME.SME13
//MEDAR120 DD DISP=(NEW,CATLG,DELETE),DSN=HPFP.TSSME.DA12,
// UNIT=SYSDA,SPACE=(CYL,(2,1),RLSE),
// DCB=BLKSIZE=23000
//MEDAR30 DD DISP=(NEW,CATLG,DELETE),DSN=HPFP.TSSME.DA3,
// UNIT=SYSDA,SPACE=(CYL,(1,1),RLSE),
// DCB=BLKSIZE=23000
//SORT12 DD DISP=(NEW,PASS),DSN=&SORT12,
// UNIT=SYSDA,SPACE=(TRK,(1,1))
//SORT3 DD DISP=(NEW,PASS),DSN=&SORT3,
// UNIT=SYSDA,SPACE=(TRK,(1,1))
/*

```

Figure 312. Running DEDB Tuning Aid with segment information records collected by OPC (Part 1 of 2)

```

//*****
//** SORT MODELLER DA12 RECORDS **
//*****
//SORT12 EXEC PGM=SORT,PARM='CORE=MAX',COND=(4,LT),
//          REGION=2048K
//SYSOUT DD SYSOUT=A
//SYSIN DD DISP=(OLD,DELETE),DSN=&SORT12
//SORTIN DD DISP=(OLD,DELETE,KEEP),DSN=HPFP.TSSME.DA12
//SORTOUT DD DISP=(NEW,CATLG,DELETE),DSN=HPFP.TSSME.SDA12,
//          UNIT=SYSDA,SPACE=(CYL,(2,1),RLSE)
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(2,3))
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(2,3))
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(2,3))
//*
//*****
//** SORT DA3 FILE **
//** **
//** NOTE: STEP CAN BE OMITTED IF **
//** MODELLER OUTPUT (STEP FABME1) **
//** IS ONLY FOR ONE AREA **
//*****
//SORT3 EXEC PGM=SORT,PARM='CORE=MAX',COND=(4,LT),
//          REGION=2048K
//SYSOUT DD SYSOUT=A
//SYSIN DD DISP=(OLD,DELETE),DSN=&SORT3
//SORTIN DD DISP=(OLD,DELETE,KEEP),DSN=HPFP.TSSME.DA3
//SORTOUT DD DISP=(NEW,CATLG,DELETE),DSN=HPFP.TSSME.SDA3,
//          UNIT=SYSDA,SPACE=(CYL,(1,1),RLSE)
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(2,3))
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(2,3))
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(2,3))
//*
//*****
//** GENERATE REPORTS **
//** **
//** NOTE: ALTER DD STMT DAR13I AS REQUIRED **
//** IF STEP SORT3 WAS OMITTED **
//*****
//FABADA5 EXEC PGM=FABADA5,REGION=512K,COND=(4,LT)
//SYSPRINT DD SYSOUT=A
//REPORTS DD SYSOUT=A
//UOWRPT DD SYSOUT=A
//UOWEXCP DD SYSOUT=A
//DAR12I DD DISP=(OLD,DELETE,KEEP),DSN=HPFP.TSSME.SDA12
//DAR13I DD DISP=(OLD,DELETE,KEEP),DSN=HPFP.TSSME.SDA3
//*
//

```

Figure 313. Running DEDB Tuning Aid with segment information records collected by OPC (Part 2 of 2)

Example 6: Processing multiple areas and taking image copies

The following figure shows example JCL for running OPC for multiple areas and generating an image copy for each area by using IMS HP Image Copy.

```
//ANALYZE EXEC FPUTIL,DBD=DEDBJN23,IMSID=IMS1
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=HPS.SHPSLMD0
//        DD DISP=SHR,DSN=IMS.SDFSRESL
//FPXIN   DD *
ANALYZE  AREA=(DB23AR1,DB23AR2),
          FULLSTEP=YES,
          TYPRUN=RPT,
          IC=(YES,ICEIN),
          ICNOTIFY=YES
REPORT   REPORTDD=FPXRPTS
/*
//ICEIN   DD *
GLOBAL   COMP=Y,COMPRTN=FABJCOMP4
CIC      DBD=DEDBJN23,
          AREA=DB23AR1,
          ICOUT=ICOUT001
CIC      DBD=DEDBJN23,
          AREA=DB23AR2,
          ICOUT=ICOUT002
/*
//FPXRPTS DD SYSOUT=*
//SYSOUT  DD SYSOUT=*
//ICERPRT DD SYSOUT=*
```

Figure 314. Processing multiple areas and taking image copies

Example 7: Running cross-reference validation of pointers, generating analysis reports, and storing sensor data in the IMS Tools KB repository

The following figure shows example JCL for running cross-reference validation of pointers in an online area, generating the analysis reports, and storing the sensor data in the IMS Tools KB repository.

No pointer verification record data set is generated.

In addition to the default data elements that are collected by the SENSOR keyword, the data elements that are related to database records and root segment distribution are stored (SENSOR_DBREC=YES specification).

```
//ANALYZE EXEC FPUTIL,DBD=DEDBJN23,IMSID=IMS1
/*
//FPU.STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=ITB.SHKTL0AD
//FPXMFIL   DD SYSOUT=*
//FPXRPTS   DD SYSOUT=*
//FPXSPRT   DD SYSOUT=*
//RECON1    DD DISP=SHR,DSN=IMSVS.RECON1
//RECON2    DD DISP=SHR,DSN=IMSVS.RECON2
//RECON3    DD DISP=SHR,DSN=IMSVS.RECON3
//FPXIN     DD *
GLOBAL
  DBRC=YES,
  ITKBSRVR=FPQSVR01,
  ADXCFGRP=ADSVR01
ANALYZE
  AREA=ALL,
  TYPRUN=RPT,
  FULLSTEP=YES,
  SENSOR=YES,
  SENSOR_DBREC=YES
/*
```

Figure 315. Running cross-reference validation of pointers, generating analysis reports, and storing sensor data in the IMS Tools KB repository

Example 8: Generating reports that contain SDEP record information, prefix length, and data flags

The following figure shows example JCL for generating the reports that contain SDEP record information, prefix length, and data flags to facilitate diagnosis of the database.

In this example:

- SDEP=YES for the ANALYZE command specifies that a Segment Placement Analysis report that includes SDEP information is generated.
- SEGLSTAT=DETAILED keyword specifies that a DB Record Profile Analysis report that includes prefix length and data flags is generated.

```
//ANALYZE      EXEC FPUTIL,DBD=DEDBJN23,IMSID=IMS1
//*
//FPU.STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//            DD DISP=SHR,DSN=IMSVS.SDFSRESL
//FPXMFILE DD SYSOUT=*
//FPXRPTS DD SYSOUT=*
//FPXSPRT DD SYSOUT=*
//FPXIN DD *
GLOBAL
ITKBSRVR=FPQSVR01
ANALYZE
    AREA=ALL,
    TYPRUN=RPT,
    SDEP=YES,
    FULLSTEP=YES
REPORT
    SEGLSTAT=DETAILED
/*
```

Figure 316. Generating reports that contain SDEP record information, prefix length, and data flags

Chapter 21. Extracting a DEDB online with FPO

You can extract segment data from an online DEDB by using the Online Data Extract utility (also referred to as ODE) of FPO.

Topics:

- [“Functions of ODE” on page 729](#)
- [“Restrictions of ODE” on page 731](#)
- [“Input and output for ODE” on page 731](#)
- [“Considerations for running ODE” on page 732](#)
- [“Running ODE” on page 732](#)
- [“Region size requirements for ODE” on page 733](#)
- [“DD statements for ODE” on page 733](#)
- [“Option control statement for ODE” on page 736](#)
- [“Utility control statements for ODE” on page 737](#)
- [“Output for ODE” on page 741](#)
- [“Examples for ODE” on page 744](#)

Functions of ODE

ODE extracts segment data from an online DEDB area and writes the output to a sequential file.

The sequential file can be in one of the following formats:

- Standard format
- IMS DBT Unload/Reload format
- TFMT Unload/Reload format
- TRIMAR Unload/Reload format
- User format

ODE can be used as an alternative to writing your own application to extract DEDB data. Other potential uses of ODE include:

- Record reporting
- Extraction of records to populate a test database

Each invocation of ODE specifies the area name as a parameter, and ODE will access only this area. However, multiple areas can be processed sequentially in a single run, similar to the DBFUMSCO and DBFUMDLO utilities.

ODE selects data for specified segments only. The resulting extracted data does not include parents or children of the specified segment, unless they are also specified.

Segment selection

Segment selection is done using a SELECT control statement.

Selection criteria can be based on:

- Key value
- Segment data content (based on a positional offset, operator, and data type)

If the selection criteria specifies the root segment, and the specification is an equal compare on the entire key field, ODE invokes the randomizer and determines which UOWs must be read to satisfy the selection criteria. For other selections, ODE scans UOWs sequentially to search for the valid segments.

You can specify up to eight comparisons per segment SELECT statement. If multiple comparisons are specified, they operate as Boolean AND operations. Each comparison supports:

- EQ, NE, GT, GE, LT, and LE operations
- Hexadecimal, character, and packed decimal data types

The same segment name can appear in more than one segment selection specification. Each segment specification is evaluated independently for a match, which allows for Boolean OR logic between SELECT statements.

SDEP segments

SDEP segment types are supported. The SDEP timestamp, added automatically by IMS at the end of every SDEP segment, is a valid segment selection argument.

SDEPs can be processed physically or hierarchically. If SDEPs are processed physically, SDEPs can be retrieved even if the parent root is deleted. The utility might return logically deleted SDEP segments when an SDEP timestamp is specified, depending on the time and comparison you select. If SDEPs are processed hierarchically, SDEPs whose parent root is deleted and SDEPs that are logically deleted cannot be retrieved.

If an SDEP timestamp is not specified, any SDEP segments with a timestamp prior to the DMAC Logical Begin Timestamp are not returned. This facility can prove useful to you when recovering from SDEP processing errors by retrieving logically deleted SDEP segments.

Note: IMS 5.1 SDEP data is not supported and will be ignored if found in the SDEP portion of an area. Warning message FPX0006W is produced in this case.

Segment output

An OUTPUT control statement controls what data from a segment match is written to the output file.

OUTPUT specifies segment name and selection criteria. Up to eight groups of fields per output segment can be specified. Unless SELECT SEG=ALL or SEG=* is specified, an OUTPUT statement is required for each selected segment.

If the option to write output in IMS DBT unload/reload format is selected, the entire segment is written to the output file. If this option is not selected, the data is written in a file format called standard or STD.

You can take one of the following actions:

- Skip a number of segment occurrences
- Stop selection after a number of segment occurrences
- Select every Nth occurrence of a segment

Sort program

You can have ODE write the output file in a format suitable for offline sorting.

Related reference

[Extracted records layout](#)

Extracted records are created by the FPA Extract function and the FPO ODE utility.

Run-level versus area-level control

ODE can be run using area- or run-level control.

With run-level control, there is one set of selection criteria for *all* areas (the FPXIN DD statement and the FPXCTL DD statement) and one output for all areas.

With area-level control, you specify selection criteria for *each* area. There is no FPXIN DD statement and FPXCTL DD statement. Instead, each area has its own input DD name, which is the same as the name of the area.

Note: If both FPXCTL and *areaname* DD statements are coded in the JCL, ODE will not attempt to process the *areaname* DD statements. Instead, ODE will execute using run-level control. This contrasts with the output file side of processing, where the presence of both FPXOFILE DD statements and OFILE keywords can be used to direct (and stack) output on an area-by-area basis. If FPXIN DD is specified as an input, ODE always runs by using the run-level control. Both FPXCTL and *areaname* DD are ignored.

In the case of run-level control, all areas can be written to the same output file, and in the case of area-level control, each area can be written to an area-specific output file.

Restrictions of ODE

Certain restrictions apply to using ODE.

- The IMS IFP utility must execute ODE in a separate job step. The IMS IFP utility cannot switch from one utility or one database to another within the same job step.
- Only one utility at a time can be active against an area.
- If you issue /STA AREA for a shared VSO area defined with the PRELOAD attribute, you can receive IMS message DFS2651A although no utility appears to be running. This is because ODE cannot run concurrently with the preload process.
- The segment edit/compression routine interface environment that is created by ODE conforms to the published interface as defined in *IMS Exit Routines* with one exception. When the segment edit/compression routine is invoked by IMS, register 1 contains the address of EPST. However, when ODE calls a segment edit/compression routine by specifying the EXPAND keyword in the FPXCTL statement, register 1 is set to 0, which indicates that it was not called by IMS.
- If the selection criteria specify the root segment, and the specification is an equal compare on the entire key field, ODE invokes the randomizer and determines which UOWs must be read to satisfy the selection criteria. When a randomizer is invoked by IMS, register 10 contains the address of the EPST, and register 11 contains the address of the ESCD. Some user-written or user-customized randomizers are designed to use these addresses to gain access to IMS control blocks other than those passed as part of the published interface. When ODE issues a call to a randomizing module, register 10 is set to -1 and register 11 is set to 0.

Input and output for ODE

ODE extracts segment data from online DEDB areas.

The following figure shows input and output for ODE.

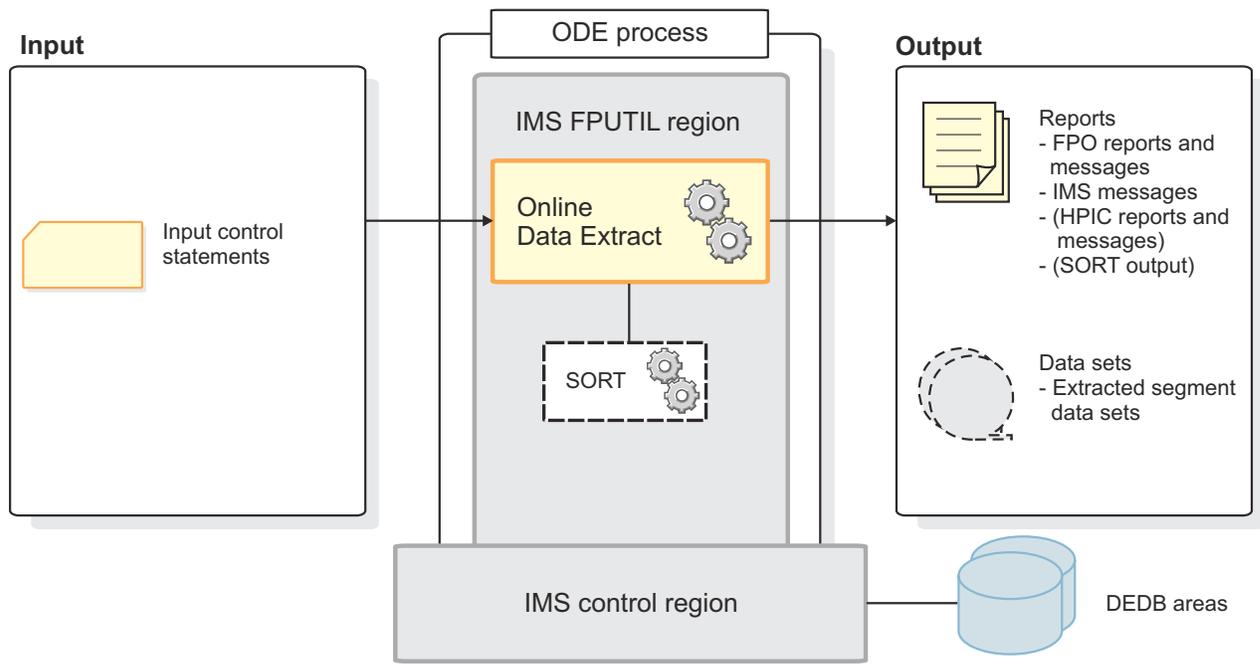


Figure 317. Input and output for ODE

Considerations for running ODE

This topic describes what you need to consider before running ODE.

Performance considerations

ODE is intended to favor online system performance as much as possible, while retaining a consistent view of IMS data.

UOW locking is used.

ODE scans an entire area. To the extent possible, selection criteria are applied within a UOW to limit pointer chasing and minimize IOVF CI access.

SDEP CIs are not read unless the segment selection criteria specify an SDEP segment or unless SEG=ALL or SEG=* is specified. In this case, the entire SDEP portion of the area is read, without locking. If SDEPs are processed physically, before reading the first SDEP CI, a request to write out the data in buffer to DASD (harden) up to the high water mark CI is made; this causes all systems to write out their current SDEP buffers. SDEPs inserted after this point will not be returned in this run. However, if SDEPs are processed hierarchically, SDEPs that are inserted while ODE is running will be returned and more I/O overhead will be expected than SDEP physical process.

If the selection criteria specify the root segment, and the specification is an equal compare on the entire key field, ODE invokes the randomizer and determines which UOWs must be read to satisfy the selection criteria. In this case, only the necessary UOWs are read. All selection criteria for the run must involve a full root key compare for this option to take effect. Detection is automatic, and there are no control statements required, nor is it necessary to include the randomizer load module in STEPLIB of the run.

Running ODE

ODE runs as an IMS IFP utility, using standard JCL and control statements. You need to specify an EXEC statement and DD statements that define the input and output data sets in your JCL.

Procedure

To run ODE, complete the following steps.

Note: If you want to change the return codes of ODE, you need to create the return code user exit routines.

1. Specify the EXEC statement in one of the following formats:

```
//name EXEC procname
```

where *procname* is the IMS-supplied IFP procedure FPUTIL or any other IFP utility supporting procedure.

```
//name EXEC PGM=DFSRR00,PARM='IFP,dbdname,DBF#FPU0'
```

where *dbdname* is the DBDNAME associated with the DEDB.

Related reading:

- For information about the FPUTIL utility, see *IMS System Definition* for the version of IMS that you are using.
 - For information about the DEDB online utility commands, see *IMS Database Utilities* for the version of IMS that you are using.
2. Specify the DD statements to define input data sets and output data sets.
 3. Run the JCL.

Recovery and restart: You cannot restart ODE. ODE can be stopped using the /STOP REGION command. STOP takes effect on the next I/O call.

Related reference

[Return code user exit routine \(FPXRCEX0\)](#)

IMS Fast Path Online Tools supports a return code user exit routine that can be used to change the return codes of OPC, ODE, or OER.

[Examples for ODE](#)

There are many ways to run the ODE utility. The examples provided in the following topics show some of the typical ways to run the ODE utility.

[Tips and best practices for the FPO Online Data Extract utility](#)

The FPO Online Data Extract utility (ODE) supports keywords that further simplify management of DEDBs.

Region size requirements for ODE

The region size for ODE depends on the size of a UOW and the keyword specifications.

Because ODE operates in a virtual storage region as an IMS IFP utility, the region size required to run ODE is larger than the following values:

- For the buffer sets portion for UOW processing, ODE requires the storage calculated by the following formula:

```
Buffer sets:          UOW1 * CISZ * 5
```

- For the IOVFPPOOL-dependent portion, ODE requires the storage calculated by the following formulas:

```
Work buffer:          (UOW1 + IOVFPPOOL) * CISZ * 10
IOVF private buffer:  IOVFPPOOL * CISZ * 5
```

DD statements for ODE

DD statements for ODE determine the input and output data sets and specify how to run ODE.

The following table summarizes the DD statements for ODE.

Table 85. DD statements for ODE

DDNAME	Use	Format	Required or optional
JOBLIB or STEPLIB	Input	PDS	Required
FPXOPTS	Input	LRECL=80	Optional
FPXIN, SYSIN, FPXCTL, or <i>areaname</i>	Input	LRECL=80	Required
FPXMFIL	Output	LRECL=133	Optional
FPXOFIL	Output	PS	Required
FPXRPTS	Output	LRECL=133	Optional
SYSPRINT	Output	LRECL=121	Required
ODESORT	Output	LRECL=80	Optional
TOOLSNP1	Output	LRECL=125	Optional
TOOLSNP2	Output	LRECL=125	Optional
SYSOUT	Output		Optional

STEPLIB DD

Defines the library that contains the ODE utility. The load library that contains the segment edit/compression routine must be concatenated when the EXPAND keyword is specified in the FPXCTL statement.

Considerations for calling a randomizer through batch interface:

If the database uses a randomizer that uses a table or another load module, you must specify the load module library that contains the table or the load module. Such randomizers include randomizers that were generated by IBM IMS Sequential Randomizer Generator for z/OS as multiple modules (SRTYPE=MULTI). For more information about the batch interface for the randomizer, see [“Restrictions of ODE” on page 731](#).

FPXOPTS DD

Defines the input data set that contains the option control statements for the process. FPXOPTS usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte fixed-length records. Block size, if coded, must be a multiple of 80. It can be specified as DUMMY, NULLFILE, or an empty file.

For details of the option control statements, see [“Option control statement for ODE” on page 736](#).

FPXIN DD

Defines the input data set that contains the ODE utility control statements for the run-level control. FPXIN DD and SYSIN DD are mutually exclusive. FPXIN DD specifies control statements that are needed for the run process. It cannot be specified as DUMMY, NULLFILE, or an empty file. When both FPXIN DD and FPXCTL DD are specified, FPXCTL DD is ignored.

For details of the control statements, see [“Utility control statements for ODE” on page 737](#).

SYSIN DD

Defines the input control data set that contains the IMS IFP utility control statements. SYSIN DD and FPXIN DD are mutually exclusive. A minimum of four statements is required, as shown in the following example:

```
TYPE TOOL
AREA areaname
EXIT EXTRACT
GO
```

where *areaname* is the name of an area to be processed. Multiple areas can be processed during execution of a single step.

For additional information about these IMS IFP utility control statements, see *IMS Database Utilities* for the version of IMS that you are using.

FPXCTL DD

Defines the input data set that contains the ODE utility control statements for the run-level control. If FPXIN DD is specified, FPXCTL DD is ignored. If the FPXCTL DD and FPXIN DD are not included, the result is area-level control.

For the details of these utility control statements, see [“Utility control statements for ODE” on page 737.](#)

areaname DD

Defines the input data set that contains the ODE utility control statements for the area-level control. The type of control is determined by the *areaname* DD statement included in the JCL. If the FPXCTL DD or FPXIN DD is included, the result is run-level control. If the FPXCTL DD or FPXIN DD is *not* included, the result is area-level control, in which case the DD names match the area names specified on the AREA SYSIN control statements.

For the details of these utility control statements, see [“Utility control statements for ODE” on page 737.](#)

FPXMFIL DD

Defines the default message output for ODE with FPXIN DD. The output can go to a data set or to a printer, routed through the output stream. If the DD statement is specified explicitly or implicitly by the MFILE keyword of the FPXCTL control statement, and the DD statement for the specified DD name is not coded in the JCL stream, the DD is allocated dynamically to the system output stream with SYSOUT=*. If the DD name is overridden by the MFILE keyword with a different DD name, the DD statement for the specified DD name must be coded in the JCL stream.

FPXOFIL DD

Defines the default output data set for the extracted data. ODE checks for the existence of either FPXOFIL DD or ddname DD specified by the OFIL keyword of the FPXCTL control statement. If neither is found, an error message is issued and processing ends. If both are present, the OFIL keyword overrides the FPXOFIL DD specification for that area (area-level control) or for that run (run-level control). The output file is opened when it is first referenced and closed at end-of-job; this is so output from multiple areas can be concatenated into a single output file.

The data set for FPXOFIL DD resides on either a direct-access device or a tape. Standard labels must be used for this data set. The DCB attributes are calculated by FPXOXIT0. RECFM is VB.

- For 3380, the BLKSIZE is 23476.
- For 3390, the BLKSIZE is 27998.
- For 9345, the BLKSIZE is 22928.
- For Tape, the BLKSIZE is 32760.

When BatchPipe is applied to the FPXOFIL file, ODE does nothing for BLKSIZE and LRECL.

Do not specify DISP=MOD for this DD statement.

FPXRPTS DD

Defines the default report output. The output can go to a data set or to a printer, routed through the output stream. If the DD statement is specified explicitly or implicitly by the RFILE keyword of the FPXCTL control statement and if the DD statement for the specified DD name is not coded in the JCL stream, the DD is allocated dynamically to the system output stream with SYSOUT=*. If the DD statement is overridden by the RFILE keyword with a different DD name, and the DD statement for the specified DD name is not coded in the JCL stream, the DD is allocated dynamically to the system output stream with SYSOUT=*.

SYSPRINT DD

Defines the system message output. For ODE with SYSIN DD, SYSPRINT is used by ODE as the default message data set until it is (optionally) overridden by the MFILE= keyword of the FPXCTL control statement.

Warning: You cannot allocate the ODE SYSPRINT DD output stream to a data set because SYSPRINT DD is also used by IMS. If you specify SYSPRINT DD as a data set, some output might be overridden or unpredictable open/close errors might occur. You must specify SYSPRINT DD as SYSOUT.

ODESORT DD

Defines the output data set that contains the sort control statement when SORT=YES. DCB parameter is not needed. ODE internally sets necessary DCB attributes. Do not specify DISP=MOD for this DD statement.

If SORT=YES is specified and ODESORT DD is not specified in JCL, ODE internally sorts extracted data.

TOOLSNP1

Defines a snap dump output data set used for diagnostic purposes. All you need to provide is the DSN and space allocation. ODE assigns the correct DCB parameters internally. Usually, //TOOLSNP1 DD SYSOUT=* is sufficient.

TOOLSNP2

Defines a secondary diagnostic output file subject to the same restrictions as TOOLSNP1.

SYSOUT DD

Defines the output and is used by both ODE and DFSORT for message functions. ODE writes images of its control statements that are specified in FPXCTL DD or *areaname* DD to SYSOUT. DFSORT writes its messages to SYSOUT.

Option control statement for ODE

The option control statement defines the user description of the OVERRIDE control statement, which specifies information for setting temporary override values of product message defaults.

The option control statements must start with a line that contains the OPTIONS command. The control statement that can be specified is:

OVERRIDE control statement

The OVERRIDE control statement specifies information for setting temporary override values of product message defaults. This control statement is optional. The OVERRIDE control statement can be specified in FPXOPTS DD.

OVERRIDE statement keywords

The keywords associated with the OVERRIDE statement are not available for site default options. For the format of the OVERRIDE statement, see [“OVERRIDE control statement” on page 850](#).

For the OVERRIDE statement, you can specify the keywords that are listed in the following table.

Table 86. *OVERRIDE* statement keywords

Keyword	Required or optional	Description	Topic
MSG_LIM	Optional	Specifies the maximum number of repetitive messages that are to be issued before suppression.	“MSG_LIM keyword” on page 897
MSG_NUM	Required	Specifies the message number that is to be suppressed.	“MSG_NUM keyword” on page 898

Related reference

[Keyword reference for FPO](#)

These topics explain the control statements and keywords of FPO and their syntax rules.

[Syntax rules for FPXOPTS](#)

FPXOPTS control statements and their associated keywords can be coded in free format in columns 1 through 72, subject to certain syntax rules.

Utility control statements for ODE

The utility control statements for ODE define the user description of the EXTRACT control statement, FPXCTL control statement, SELECT control statement, OUTPUT control statement, and USER_RECORD control statement, all of which control the ODE process.

All control statements that can be specified to run an ODE job are explained in this topic. The control statements that can be specified are:

GLOBAL control statement

The GLOBAL control statement defines the job step keyword values. If specified, GLOBAL must be the first command in the input stream. The GLOBAL control statement can be specified only in FPXIN DD.

EXTRACT control statement

The EXTRACT control statement defines the areas to be extracted. The EXTRACT control statement can be specified only in FPXIN DD.

FPXCTL control statement

The FPXCTL control statement defines control information that influences how a run is processed.

SELECT control statement

The SELECT control statement defines segment selection criteria.

OUTPUT control statement

The OUTPUT control statement defines the format of the data that is to be extracted.

USER_RECORD control statement

The USER_RECORD control statement defines the user-defined header and the trailer records that are to be written to an extract file. The USER_RECORD control statement can be specified only in FPXIN DD.

ODE allows you to specify site default options.

Subsections:

- [“GLOBAL statement keywords” on page 737](#)
- [“EXTRACT statement keywords” on page 738](#)
- [“FPXCTL statement keywords” on page 738](#)
- [“SELECT statement keywords” on page 739](#)
- [“OUTPUT statement keywords” on page 739](#)
- [“USER_RECORD statement keywords” on page 740](#)

GLOBAL statement keywords

The GLOBAL control statement specifies the job step keyword values. This control statement is optional. The keyword associated with the GLOBAL statement are available for site default options.

For the GLOBAL statement, you can specify the keyword that is listed in the following table. For the format of the GLOBAL statement, see [“GLOBAL control statement” on page 847](#).

Table 87. GLOBAL statement keywords

Keyword	Used in FPXIN DD	Used in FPXCTL DD or areaname DD	Description	Topic
KEYWORD_DUPCHK	Yes (Optional)	No	Specifies to end the job with an error message when a duplicate keyword is found.	KEYWORD_DUPCHK keyword

EXTRACT statement keywords

The EXTRACT statement specifies the areas to be extracted.

For the EXTRACT statement, you can specify the keywords that are listed in the following table. For the format of the EXTRACT statement, see [“EXTRACT control statement” on page 843](#).

Table 88. EXTRACT statement keywords

Keyword	Used in FPXIN DD	Used in FPXCTL DD or areaname DD	Description	Topic
AREA	Yes (Required)	No	Specifies one or more input areas for ODE process.	“AREA keyword” on page 864
PROC_AFT_ERROR	Yes (Optional)	No	Specifies whether to end the job step when an area-specific error is encountered or to continue processing subsequent areas.	“PROC_AFT_ERROR keyword” on page 902
RCEXLOAD	Yes (Optional)	No	Specifies whether to call the return code exit routine FPXRCEX0 from the STEPLIB or JOBLIB data set concatenation.	“RCEXLOAD keyword” on page 904

FPXCTL statement keywords

The FPXCTL control statement specifies control information that influences how the run is to be processed. This control statement is optional. If it is omitted, the indicated defaults are taken for each of the keywords.

The keywords associated with the FPXCTL statement are available for site default options except for the SEGREC_PREFIX and the SEGREC_SUFFIX keywords.

For the FPXCTL statement, you can specify the keywords that are listed in the following table. For the format of the FPXCTL statement, see [“FPXCTL control statement” on page 844](#).

Table 89. FPXCTL statement keywords

Keyword	Used in FPXIN DD	Used in FPXCTL DD or areaname DD	Description	Topic
CASE	Yes (Optional)	Yes (Optional)	Allows case sensitivity with character specification in the FIELD= and ANDFIELD= keywords.	“CASE keyword” on page 866
EXPAND	Yes (Optional)	Yes (Optional)	Specifies whether compressed data is to be expanded during extraction.	“EXPAND keyword” on page 874
INDOUBT	Yes (Optional)	Yes (Optional)	Specifies whether in-doubt SDEP segments are to be selected.	“INDOUBT keyword” on page 887
IOVFPOOL	Yes (Optional)	Yes (Optional)	Specifies the number of IOVF buffers per buffer set.	“IOVFPOOL keyword” on page 888
MAXTWINCTR	Yes (Optional)	Yes (Optional)	Specifies the threshold value for the number of twin segments.	“MAXTWINCTR keyword” on page 895
MFILE	Yes (Optional)	Yes (Optional)	Specifies the DD name of the output file to which utility messages are to be written.	“MFILE keyword” on page 896
OFIL	Yes (Optional)	Yes (Optional)	Specifies the DD name of the output file to which the extracted data is to be written.	“OFIL keyword” on page 898
OLRECL	Yes (Optional)	Yes (Optional)	Specifies the LRECL of FPXOFIL.	“OLRECL keyword” on page 899
OUTPUT	Yes (Optional)	Yes (Optional)	Specifies the format of the output file.	“OUTPUT keyword” on page 899

Table 89. FPXCTL statement keywords (continued)

Keyword	Used in FPXIN DD	Used in FPXCTL DD or areaname DD	Description	Topic
PCHKONLY	Yes (Optional)	Yes (Optional)	Specifies whether to evaluate only the control statement keywords or to also extract segment data from online DEDB area.	“PCHKONLY keyword” on page 901
RFILE	Yes (Optional)	No	Specifies the DD name of the output file to which ODE reports are to be written.	“RFILE keyword” on page 908
SDEP_BYPASS _MSGLVL	Yes (Optional)	Yes (Optional)	Specifies the severity level of the message that the utility issues when the SDEP is defined but SDEP segment data is not extracted.	SDEP_BYPASS _MSGLVL keyword
SDEPHIER	Yes (Optional)	Yes (Optional)	Specifies whether SDEP segment records should be extracted hierarchically.	“SDEPHIER keyword” on page 913
SEGREC_PREFIX	Yes (Optional)	No	Specifies the contents of the fields within the prefix portion of the segment records, which are to be written to an extract file.	“SEGREC_PREFIX keyword” on page 919
SEGREC_SUFFIX	Yes (Optional)	No	Specifies the contents of the fields within the suffix portion of the segment records, which are to be written to an extract file.	“SEGREC_SUFFIX keyword” on page 920
SORT	Yes (Optional)	Yes (Optional)	Specifies how extracted information is to be sorted for output.	“SORT keyword” on page 927
STAT	Yes (Optional)	Yes (Optional)	Specifies whether or not extraction statistics are to be generated.	“STAT keyword” on page 928

SELECT statement keywords

The SELECT control statement specifies the segment name and the selection criteria for the segment.

SELECT is optional; if it is omitted, the indicated defaults are taken for each of the keywords if the corresponding OUTPUT statement is coded for the segment. If both SELECT and OUTPUT are omitted for a specific segment, then no selection or extraction criteria is generated for it.

The keywords associated with the SELECT statement are not available for site default options.

For the SELECT statement, you can specify the keywords that are listed in the following table. For the format of the SELECT statement, see [“SELECT control statement” on page 857](#).

Table 90. SELECT statement keywords

Keyword	Used in FPXIN DD	Used in FPXCTL DD or areaname DD	Description	Topic
SEG	Yes (Required)	Yes (Required)	Specifies what segments are to be selected.	“SEG keyword” on page 916
FIELD	Yes (Optional)	Yes (Optional)	Specifies the criteria used to select segments for extraction.	“FIELD keyword” on page 875
ANDFIELD	Yes (Optional)	Yes (Optional)	Specifies additional selection criteria (used in conjunction with the FIELD keyword criteria) for selecting segments for extraction.	“ANDFIELD keyword” on page 862

OUTPUT statement keywords

The OUTPUT control statement defines what segments are to be included in the output file (OFILE).

OUTPUT is optional. If specific keywords are omitted, the indicated defaults are taken for each of the keywords.

The keywords associated with the OUTPUT statement are not available for site default options.

For the OUTPUT statement, you can specify the keywords that are listed in the following table. For the format of the OUTPUT statement, see [“OUTPUT control statement”](#) on page 848.

Table 91. OUTPUT statement keywords

Keyword	Used in FPXIN DD	Used in FPXCTL DD or areaname DD	Description	Topic
SEG	Yes (Required)	Yes (Required)	Specifies which segments are to be included in the output.	“SEG keyword” on page 916
FIELDS	Yes (Optional)	Yes (Optional)	Specifies what data from a selected segment is to be written to the output file.	“FIELDS keyword (for OUTPUT)” on page 876
STOPAFT	Yes (Optional)	Yes (Optional)	Specifies the number of segments to stop the selection of segments. For DDEP segments, after a specific number of segment occurrences under a parent segment, for root segments, after a specific number of root segments in a synonym chain under a RAP, and for SDEP segments, after a specific number of segments starting from the first CI of SDEP part.	“STOPAFT keyword” on page 929
SKIP	Yes (Optional)	Yes (Optional)	Specifies the number of segment occurrences to skip. For DDEP segments, a specific number of segment occurrences under a parent, for root segments, a specific number of root segment occurrences in a synonym chain under a RAP, and for SDEP segments, a specific number of segments starting from the first CI of the SDEP part.	“SKIP keyword” on page 926
EVERY	Yes (Optional)	Yes (Optional)	Selects every Nth occurrence of a segment starting from the first segment. For DDEP segments, every Nth occurrence of a segment under a parent, for root segments, every Nth occurrence of a root segment in a synonym chain under a RAP, and for SDEP segments, every Nth segment starting from the first CI of the SDEP part.	“EVERY keyword” on page 871
AFTERTIME	Yes (Optional)	Yes (Optional)	Specifies the starting time stamp value for SDEP selections.	“AFTERTIME keyword” on page 862

USER_RECORD statement keywords

The USER_RECORD control statement specifies the user-defined header and the trailer records that are to be written to an extract file.

USER_RECORD is optional; if it is omitted, no user-defined header and trailer records are written.

The keywords associated with the USER_RECORD statement are not available for site default options.

For the USER_RECORD statement, you can specify the keywords that are listed in the following table. For the format of the USER_RECORD statement, see [“USER_RECORD control statement”](#) on page 861.

Table 92. USER_RECORD statement keywords

Keyword	Used in FPXIN DD	Used in FPXCTL DD or areaname DD	Description	Topic
BREAK	Yes (Optional)	No	Specifies the level and the timing for the trigger event for user-defined header and trailer of an extract file.	“BREAK keyword” on page 865
FIELDS	Yes (Required)	No	Specifies the user-defined header records, or the user-defined trailer records that are written to the output file.	“FIELDS keyword (for USER_RECORD)” on page 877

Related reference

[Keyword reference for FPO](#)

These topics explain the control statements and keywords of FPO and their syntax rules.

Syntax rules for FPXIN

FPXIN control statements and their associated keywords can be coded in free format in columns 1 through 72, subject to certain syntax rules.

Syntax rules for FPXCTL and areaname

Control statements and their associated keywords are free form and are coded in columns 1 through 80, subject to certain syntax rules.

Related information

Site default options support for FPO

The site default options are control statements that are specified in a member of a partitioned data set and that are used to override the system default option values, which can be changed without assembling and link-editing.

Output for ODE

The following topics describe the output generated by ODE, its format and data elements, and how to interpret them.

FPXRPTS DD data set

The FPXRPTS DD data set contains the reports that are generated by ODE.

This topic describes the FPXRPTS DD data set, which is generated by ODE only when FPXIN DD is used. For ODE with FPXCTL DD or *areaname* DD, no FPXRPTS DD data set is generated.

The Process Summary report is produced in the data set or written to the output stream that is determined by the RFILE keyword of the FPXCTL control statement (FPXRPTS is the default value for the RFILE keyword.)

Process Summary report

The Process Summary report summarizes the result of processing for each area.

The following figure shows an example of the report.

IMS HPFP UTILITIES - ODE		"Process Summary Report"				PAGE: 1
5698-FPP V2R1						2019-11-22 22:09:58
DBDNAME	AREANAME	AREA NO	START DATE/TIME	END DATE/TIME	STATUS (If not blank, see messages in the message data sets.)	
DEDBJN23	DB23AR1	1	2019-11-22 22:10:00	2019-11-22 22:10:12		
DEDBJN23	DB23AR2	2	2019-11-22 22:11:02	2019-11-22 22:11:18	Processing ended with warnings.	
DEDBJN23	DB23AR3	3	2019-11-22 22:12:17	2019-11-22 22:12:18		
DEDBJN23	DB23AR4	4	2019-11-22 22:12:19	2019-11-22 22:12:20		

Figure 318. Process Summary report (ODE)

DBDNAME

Shows the name of the database.

AREANAME

Shows the name of the area. The area names are displayed in processed order.

AREA NO

Shows the area number.

START DATE/TIME

Shows the time stamp when the area was connected.

END DATE/TIME

Shows the time stamp when the area was disconnected.

STATUS

Shows the processing result for each area.

If ODE ended successfully without errors, this column is blank. Otherwise, one of the following messages is displayed in this column:

Processing ended with warnings.

This message indicates that the processing ended with return code of 4. No pointer errors were detected.

Processing ended with errors.

This message indicates that the processing ended with return code of 8.

Pointer errors were detected.

This message indicates that the processing ended with return code of 4. One or more pointer errors were detected.

The following areas not processed because PROC_AFT_ERROR=STOP.

This message indicates that the processing ended with errors in the previous area. This area and the subsequent areas are not processed because PROC_AFT_ERROR=STOP is specified. The status for the subsequent areas is blank.

Processing did not complete.

This message indicates that the processing for the area did not complete because the processing ended abnormally, or an error, which prevented from processing the subsequent areas, was encountered. The status for the subsequent areas is blank.

Processing was skipped.

This message indicates that the processing for this area was skipped because the area was stopped. This message is displayed only when AREA=ALL is specified.

For details about these messages, see the messages produced in the FPXMFIL data set or in the SYSPRINT data set.

FPXOFIL DD data set

The FPXOFIL DD data set contains the data of segments that are extracted by ODE.

FPXOFIL is the default value for the OFIL keyword of the FPXCTL control statement.

The sequential file can be in one of the following formats:

- Standard format
- IMS DBT Unload/Reload format
- TFMT Unload/Reload format
- TRIMAR Unload/Reload format

Related reference

[Extracted records layout](#)

Extracted records are created by the FPA Extract function and the FPO ODE utility.

SYSPRINT/FPXMFIL DD data set

The IMS system messages and control statement analysis messages are produced in the system output stream that is specified on the SYSPRINT DD statement.

The following ODE messages can be routed to another output data set or to the system output stream by coding the MFILE parameter of the FPXCTL control statement. If the MFILE parameter is not specified, they are produced in the default message data set FPXMFIL for ODE with FPXIN DD, or in SYSPRINT for ODE with FPXCTL DD or *areaname* DD.

- ODE messages except control statement analysis messages
- ODE Area Level Statistics
- ODE Database Level Statistics

The following figure shows an example of the ODE Area Level Statistics and Database level Statistics.

5698-FPP

DATE: 11/22/2019 12:30:05

ODE - V2R1

FPX0039I Starting Online Fast Path utility processing:

(cont) AREA name = DB22AR0 AREA number = *****1

FPX0054I Area level statistics:

Segment Code	Segment Name	Segment Hierarchy	Segments Read	Segments Selected	Segments Written
1	ROOTSEG1	1	2	2	2
2	SDSEG1	2	6	3	3 (See "SDEP segment details")
3	DD1	2	6	6	6
4	DD2	2	6	6	6
5	DD3	2	7	7	7
6	DD4	2	0	0	0
7	DD43	3	0	0	0
8	DD44	4	0	0	0
9	DD45	5	0	0	0
10	DD46	6	0	0	0
11	DD47	7	0	0	0
12	DD48	8	0	0	0
13	DD49	9	0	0	0
14	DD410	10	0	0	0
15	DD411	11	0	0	0
16	DD412	12	0	0	0
17	DD413	13	0	0	0
18	DD414	14	0	0	0
19	DD415	15	0	0	0
20	DD5	2	0	0	0
21	DD53	3	0	0	0
22	DD54	4	0	0	0
23	DD55	5	0	0	0
24	DD56	6	0	0	0
25	DD57	7	0	0	0
26	DD58	8	0	0	0
27	DD59	9	0	0	0
28	DD510	10	0	0	0
29	DD511	11	0	0	0
30	DD512	12	0	0	0
31	DD513	13	0	0	0
32	DD514	14	0	0	0
33	DD515	15	0	0	0
34	DD6	2	0	0	0
35	DD63	3	0	0	0
35	DD63	3	0	0	0
36	DD64	4	0	0	0
			-----	-----	-----
			27	24	24

SDEP segment details (For Segments Read)

Valid segment	:	2
Logically deleted segment	:	1
In-doubt segment	:	0
Aborted segment	:	0
SDEP control segment	:	3

		6

Figure 319. Online Data Extract Statistics report (Part 1 of 2)

```

FPX0055I Extract process completed successfully.
IMS HPFP UTILITIES "ONLINE DATA EXTRACT STATISTICS"
5698-FPP DATE: 11/22/2019 12:30:05 PAGE: 2
FPX0056I Database level statistics: ODE - V2R1
(cont) DEDB name = DEDBJN22
Segment Segment Segment Segments Segments Segments
Code Name Hierarchy Read Selected Written
1 ROOTSEG1 1 2 2 2
2 SDESEGMM1 2 6 3
3 DD1 2 6 6
4 DD2 2 6 6
5 DD3 2 7 7
6 DD4 2 0 0
7 DD43 3 0 0
8 DD44 4 0 0
9 DD45 5 0 0
10 DD46 6 0 0
11 DD47 7 0 0
12 DD48 8 0 0
13 DD49 9 0 0
14 DD410 10 0 0
15 DD411 11 0 0
16 DD412 12 0 0
17 DD413 13 0 0
18 DD414 14 0 0
19 DD415 15 0 0
20 DD5 2 0 0
21 DD53 3 0 0
22 DD54 4 0 0
23 DD55 5 0 0
24 DD56 6 0 0
25 DD57 7 0 0
26 DD58 8 0 0
27 DD59 9 0 0
28 DD510 10 0 0
29 DD511 11 0 0
30 DD512 12 0 0
31 DD513 13 0 0
32 DD514 14 0 0
33 DD515 15 0 0
34 DD6 2 0 0
35 DD63 3 0 0
36 DD64 4 0 0
-----
27 24 24
*****

```

Figure 320. Online Data Extract Statistics report (Part 2 of 2)

The section named 'SDEP segment details (For Segments Read)' indicates the number of the following SDEP segments that are read between LB and LE.

Valid segment

Shows the number of valid SDEP segments that are read.

Logically deleted segment

Shows the number of logically deleted SDEP segments that are read.

In-doubt segment

Shows the number of in-doubt SDEP segments that are read.

Aborted segment

Shows the number of aborted SDEP segments that are read.

SDEP control segment

Shows the number of control statements for SDEP segments that are read. The SDEP control segments are internal segments that are used by IMS for control.

Examples for ODE

There are many ways to run the ODE utility. The examples provided in the following topics show some of the typical ways to run the ODE utility.

Example 1: Running ODE with run-level control

The following figure shows example JCL for extracting data from multiple online areas. With run-level control, there is one set of selection criteria for all areas (the FPXIN DD statement or FPXCTL DD statement) and one output for all areas.

The format of the output file is standard format, and every segment defined in the DBD is extracted.

```

//EXTRACT EXEC FPUTIL,REGION=1024K,DBD=dbdname,IMSID=IMS1
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
//FPXOFILE DD SYSOUT=*
//FPXIN DD *
EXTRACT AREA=(AREA1,AREA2,AREA3)
FPXCTL OUTPUT=STD
OUTPUT SEG=ALL
/*

```

Figure 321. Run-level control using FPXIN DD

The SYSPRINT and SYSUDUMP DD statements are coded in the FPUTIL cataloged procedure.

This example JCL stream runs similarly to the following JCL stream, which is for ODE with SYSIN and FPXCTL DD.

```

//EXTRACT EXEC FPUTIL,REGION=1024K,DBD=dbdname,IMSID=IMS1
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
//FPXOFILE DD DISP=(NEW,CATLG),DSN=HPFP.FPXOFILE,UNIT=SYSDA,
// SPACE=(CYL,(5,5),RLSE)
//SYSOUT DD SYSOUT=*
//FPXCTL DD *
FPXCTL OUTPUT=STD
OUTPUT SEG=ALL
/*
//SYSIN DD *
TYPE TOOL
EXIT EXTRACT
AREA AREA1
GO
EXIT EXTRACT
AREA AREA2
GO
EXIT EXTRACT
AREA AREA3
GO
/*

```

Figure 322. Run-level control using SYSIN and FPXCTL DD

Example 2: Running ODE with area-level control

The following figure shows example JCL for extracting data from multiple online areas. With area-level control, you specify selection criteria for each area. There is no FPXIN DD statement and FPXCTL DD statement. Instead, each area has its own input DD name, which is the same as the name of the area.

The format of the output file is standard format, and every segment defined in the DBD is extracted. Each area has its own output file.

```

//EXTRACT EXEC FPUTIL,REGION=1024K,DBD=dbname,IMSID=IMS1
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
//A DD DISP=(NEW,CATLG),DSN=HPFP.TSSDA.OFILEA,UNIT=SYSDA,
// SPACE=(CYL,(5,5),RLSE)
//B DD DISP=(NEW,CATLG),DSN=HPFP.TSSDA.OFILEB,UNIT=SYSDA,
// SPACE=(CYL,(5,5),RLSE)
//C DD DISP=(NEW,CATLG),DSN=HPFP.TSSDA.OFILEC,UNIT=SYSDA,
// SPACE=(CYL,(5,5),RLSE)
//SYSOUT DD SYSOUT=*
//AREA1 DD *
FPXCTL OUTPUT=STD,OFILE=A
OUTPUT SEG=ALL
//AREA2 DD *
FPXCTL OUTPUT=STD,OFILE=B
OUTPUT SEG=ALL
//AREA3 DD *
FPXCTL OUTPUT=STD,OFILE=C
OUTPUT SEG=ALL
/*
//SYSIN DD *
TYPE TOOL *
EXIT EXTRACT
AREA AREA1
GO
EXIT EXTRACT
AREA AREA2
GO
EXIT EXTRACT
AREA AREA3
GO
/*

```

Figure 323. Area-level control

The SYSPRINT and SYSUDUMP DD statements are coded in the FPUTIL cataloged procedure.

Chapter 22. Extending a DEDB online with FPO

You can extend the size of an online DEDB area by using the Online Area Extender utility (also referred to as OAE) of FPO.

Topics:

- [“Functions of OAE” on page 747](#)
- [“Restrictions of OAE” on page 748](#)
- [“Input and output for OAE” on page 748](#)
- [“Considerations for running OAE” on page 749](#)
- [“Running OAE” on page 754](#)
- [“Region size requirements for OAE” on page 760](#)
- [“DD statements for OAE” on page 761](#)
- [“Option control statement for OAE” on page 763](#)
- [“Utility control statements for OAE” on page 764](#)
- [“Output for OAE” on page 767](#)
- [“OAE Rename program” on page 771](#)
- [“Examples for OAE” on page 772](#)

Functions of OAE

Occasionally, it becomes necessary to increase the size of a DEDB area. In the past, the DEDB area had to be taken offline in order to change the DBD/ACB and to resize the area. This was potentially disruptive in environments where database availability was crucial.

OAE lets you extend the size of the DEDB area dynamically by enlarging the sequential dependent (SDEP) portion, the independent overflow (IOVF) portion of the DEDB area, or both. Since this is accomplished without taking the DEDB area offline, the process is not disruptive to applications accessing the area.

OAE extends the size of the DEDB area in two ways:

- Shadow expansion enables you to specify the target area data sets that you want to extend. Original area data set is made UNAVAIL status with DBRC at the end of OAE.
- Original expansion extends the original area data set.
 - With `EXTEND_TYPE=COPY`, OAE copies the data to the temporary ADS, then copies the data back from the temporary ADS to the extended original area data set.
 - With `EXTEND_TYPE=RENAME`, OAE copies the data to the temporary ADS, deletes the original ADS, then renames the temporary ADS to the original one by using the OAE Rename program. The area will be temporarily stopped in all sharing IMS systems during the rename process.

When you extend only the SDEP portion of the DEDB area, you can optionally extend the image copy data sets in the same job step. You can also extend the image copy data sets in the same job step when you extend the IOVF portion if you are on IMS 14 with APAR PI43928 or IMS 15 or later. This option is called the *image copy extension option*. To use this option, the image copy data set must be created before the OAE job and it must be registered in the RECON data sets. By having OAE extend the image copy data sets, you can use the extended image copy data sets to recover the area that is extended by OAE. Shadow expansion and original expansion both support this option. This option can be requested by the `ICEXTEND` keyword in `SYSIN DD` for shadow expansion, and `ICEXTEND=YES` in `FPXIN DD` for original expansion.

In a data-sharing IMS environment, as part of updating the IMS control blocks (DMAC) with the new AREA structure information, all IMS systems are notified by OAE of the AREA structure change.

Restrictions of OAE

Certain restrictions apply to using OAE.

- The IMS IFP utility must execute OAE in a separate job step. The IMS IFP utility cannot switch from one utility or one database to another within the same job step.
- Only one utility at a time can be active against an area.
- The utility cannot extend RSR-recovered areas.
- Areas that are defined as VSO areas by the Virtual Storage Option must be removed from the virtual storage before area extension.
- The original expansion with EXTEND_TYPE=RENAME needs an additional process step, the OAE Rename program. During the rename process, the area is temporarily stopped in all sharing IMS systems.

Input and output for OAE

OAE has two types, the original expansion and the shadow expansion.

The following figure shows input and output for the OAE original expansion. OAE accepts FPXIN DD as the input and extends the original data set.

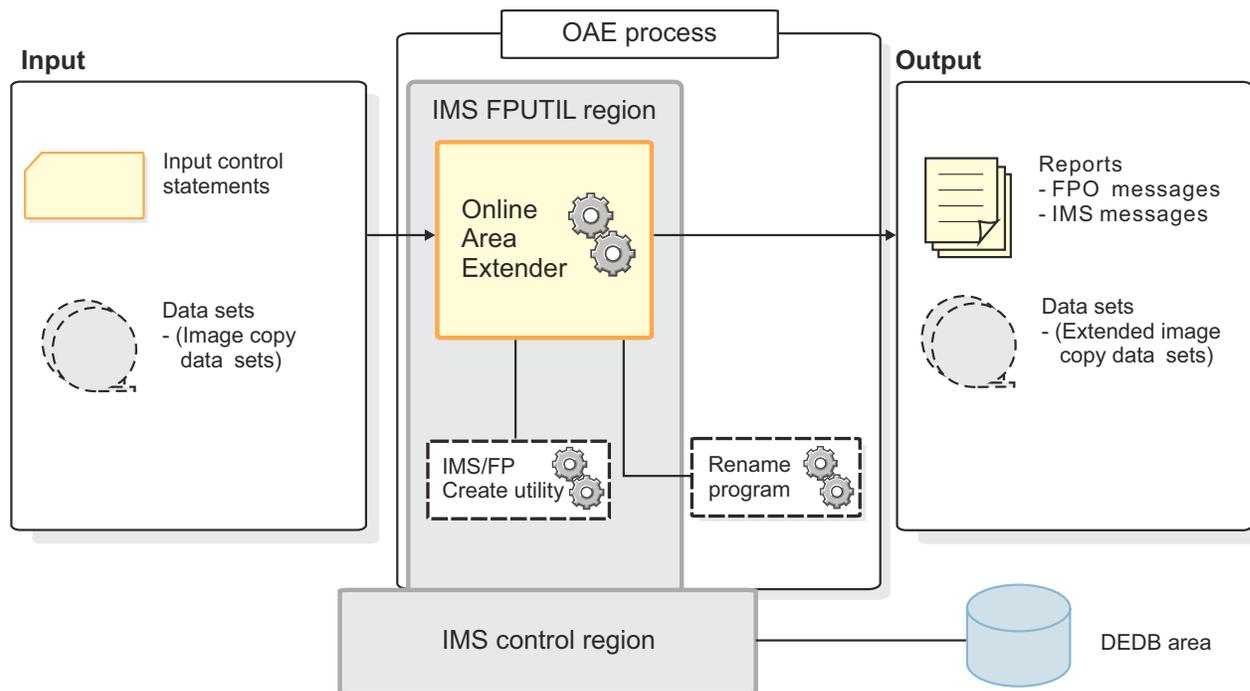


Figure 324. Input and output for OAE (Original expansion)

The following figure shows input and output for the OAE shadow expansion. OAE accepts SYSIN DD as the input, and the target data set is different from the original data set.

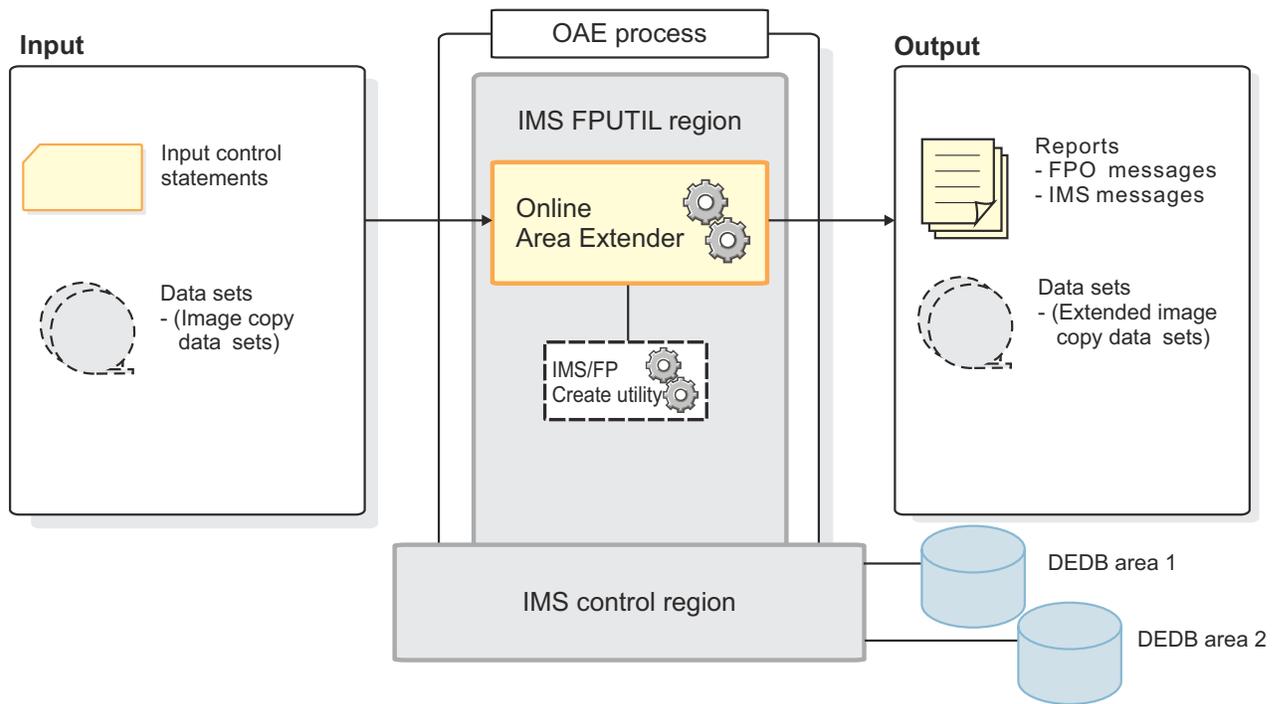


Figure 325. Input and output for OAE (Shadow expansion)

During execution, the data from the current ADS is copied to the new ADS for shadow expansion, to the temporary ADS for original expansion. All additional control intervals are formatted as SDEP control intervals and are made available to online processes.

If IOVF extension has been requested, the appropriate control intervals will be subsequently formatted as IOVF control intervals, and are made available to online processes.

Considerations for running OAE

The following topics describe what you need to consider before running OAE.

Considerations on area extension

When you invoke OAE with shadow expansion, you need to pass it the DDname associated with a new, larger, preallocated area data set (ADS) that has been registered with DBRC.

The size of the new input ADSs can be determined as follows:

- For SDEP extension, calculate the size of the current RAP/IOVF/RUOW portion of the area, then add the size of the new SDEP portion.
- For IOVF extension, calculate the size of the new RAP/IOVF/RUOW portion of the area, then add the size of the current SDEP portion.
- For concurrent extension of IOVF and SDEP, calculate the size of the new RAP/IOVF/RUOW portion of the area, then add the size of the new SDEP portion.

When you use the original expansion, you do not need to allocate a larger data set or register it as a new ADS with DBRC processes.

Before proceeding, the topics of expanding the SDEP area, the IOVF area, and both these portions of the area, are explored more fully.

Subsections:

- [“SDEP extension” on page 750](#)
- [“IOVF extension” on page 750](#)

- [“Concurrent extension of IOVF and SDEP” on page 751](#)
- [“Considerations for further extension of an area” on page 752](#)

SDEP extension

If you are expanding the SDEP portion of the area, the ACBLIB DD statement, if specified, is ignored. No changes are allowed to any area definitions including the ROOT and UOW parameters used in DBDGEN.

Shadow expansion

Allocate a target ADS. The size of the target ADS is the size of the current ADS plus the size of the new SDEP portion.

Original expansion

Specify the number of CIs for SDEP extension, by using the SDEP_CI keyword.

Notes:

- The SDEP portion of the area might also change as a result of IOVF extension. The new size of the SDEP portion of the ADS is the difference between the new RAP/IOVF/RUOW portion of the area and the size of the ADS.
- If the SDEP portion is in a wraparound state, the new SDEP portion cannot be used after OAE expands the area. To resolve the wraparound state, use the SDEP SCAN/DELETE utility. Optionally, if you can take the database offline, use the FPA Unload function with the RELOCATE_SDEP=YES option to unload the area and then use the FPA Reload function to reload the area.

For how to process SDEP segments, see [Chapter 48, “Determining how to process SDEP segments,” on page 1205](#).

IOVF extension

Before talking about the IOVF extension, it should be noted that the SDEP portion might be changed as well, either intentionally or unintentionally. For this reason, be sure to read the SDEP discussion and the discussion of concurrent extension of IOVF and SDEP, as well as this IOVF discussion.

The size of the IOVF extension is related to the difference between parameters associated with the ROOT= parameter in the original DBD definition of an area and the corresponding parameters in the expanded DBD definition.

During the IOVF extension, the IOVF portion of the area and the Reorg UOW (although the RUOW is no longer used, it is retained for compatibility purposes) are extended INTO the first part of the SDEP portion of the area.

This results in the start of the SDEP portion of the area being pushed further out into the area to make room for the IOVF extension. The CIs within the SDEP portion, which eventually will become IOVF CIs, are referred to as *Cannibalized SDEP CIs* (CSC) for the purpose of this discussion. The CSC for an area are those SDEP CIs which lie between the current first SDEP RBA and the new first SDEP RBA created as a result of IOVF extension.

IOVF extension into the CSC portion is not allowed by the utility if either of the following situations is true at execution time:

- The CSC portion of the area contains SDEP segments that are currently active.
- There is still the potential to add new active SDEP statements to the CSC portion because one or the following SDEP pointers lie within the CSC portion:
 - The Oldest Active SDEP RBA
 - The Next Available SDEP RBA

All active SDEP segments must be deleted from the CSC portion of the area, and the potential to add new SDEP segments to the CSC portion of the area must be eliminated PRIOR to running OAE. The potential to add such SDEP segments can be eliminated by taking the following actions:

- Inserting a sufficient number of additional SDEP segments such that the Next Available SDEP RBA is moved below the CSC portion.
- Using the SDEP Delete Utility to delete all of the SDEP segments with RBAs within the CSC portion.

You do not need to perform these actions if all of the following conditions are met:

- APAR PI73397 is applied to IMS 14.
- APAR PI82362 is applied to IMS Fast Path Solution Pack.
- You used the SDEP Delete utility with the QUITCI parameter immediately before running OAE and deleted all active segments.
- No active SDEP segments exist in the CSC portion.

To determine the extents of SDEP segments that must be deleted, use the number of CIs by which the IOVF portion of the area is to be expanded. Delete all SDEP segments currently residing in this SDEP portion of the area, from the beginning of the SDEPs by the number of IOVF CIs by which the area will be expanded + 1. This will allow the IOVF and REORG UOW to expand into the current SDEP portion of the area.

If SDEP segments reside in this portion of the area, the utility will end abnormally.

Shadow expansion

Specify a new IOVF size by the ROOT= parameter or use a new ACBLIB as input to the OAE utility. Because the size of the RAP area cannot be changed, increase the values of both the number2 and the overflow2 parameters on the ROOT= parameter from the DBD definition by *equal amounts* such that the difference between the two does not change from the original allocation.

Important: Do not change the UOW= parameter.

Original expansion

For IOVF extension, use the IOVF_ROOT keyword to specify the new ROOT parameter in the DBD definition, or use the IOVF_UOW keyword to specify the number of IOVF UOWs to extend. When you specify the IOVF_ROOT keyword, increase the values of both the *number2* and the *overflow2* parameters from the ROOT= parameter in the DBD definition by *equal amounts* such that the difference between the two does not change from the original allocation.

Warning: The pointers from Root segments to dependent SDEP segments are not corrected when the SDEP portion of the area is converted to IOVF CIs as a result of IOVF extension. Consequently, these pointers will be invalid, and processes which make use of this dependence will not work.

The invalidation of these pointers will be transparent to most users, who do not make use of them, but process SDEP segments independently of their actual Root parentage. However, you do need to consider the implications of these pointers being invalid should your processes require them.

Concurrent extension of IOVF and SDEP

OAE can also extend both the SDEP portion and the IOVF portion concurrently.

Shadow expansion

A concurrent extension of both the IOVF and SDEP portions of the area can be accomplished by taking both of the following actions:

- Code the ROOT= parameters in the DBD to allow for the desired IOVF extension.
- Code the VSAM allocation to increase the ADS by an amount larger than would be required to accommodate the expanded IOVF. The additional amount is used to expand the SDEP portion.

It should be noted that shadow expansion does allow you to perform an IOVF extension without increasing the ADS VSAM allocation at all, or without increasing it enough to completely accommodate the expanded IOVF. Either of these situations would result in a net decrease of the SDEP area. If OAE determines that the SDEP portion will be decreased, one of the following results:

- If the SDEPWARN control statement has been included as a safeguard against a decreasing SDEP, an error message is issued, and processing ends with return code 8 prior to actually performing the extension.
- If the SDEPWARN control statement has not been included, an informational message indicating the decreasing SDEP is issued, but processing continues.

Original expansion

A concurrent extension of both the IOVF and SDEP portions of the area can be accomplished by taking both of the following actions:

- For IOVF extension, use the IOVF_ROOT keyword to specify the new ROOT parameter in the DBD definition, or use the IOVF_UOW keyword to specify the number of IOVF UOWs to extend.
- For SDEP extension, code the SDEP_CI parameter to specify the number of SDEP CIs to extend.

Original expansion increases the ADS VSAM allocation with the added size, therefore a net decrease of SDEP portion never occurs.

Considerations for further extension of an area

An expanded area can be extended at some future point.

- A DEDB member for which an SDEP extension has been performed does not need to be activated before the next extension.
- For the DEDB member in ACBLIB for which an IOVF extension has been performed, consider bringing online the DEDB member in the ACBLIB before extending the area a second time.

A new ACBLIB is required when bringing the DEDB member online. Therefore, before bringing the DEDB member online, run the DBDGEN utility to obtain an updated DBD, and then run the ACBGEN utility to generate a new ACBLIB. Offline tools, such as FPA, also require a new ACBLIB as input.

If you do not bring the new DEDB member online, the following message, which notifies that discrepancies are found during open processing, is issued by IMS when the area is accessed for the first time. However, open processing continues because the discrepancies indicate to IMS that you used an accepted procedure to increase the size of the IOVF.

```
DFS3703I SIZE OF INDEPENDENT OVERFLOW HAS BEEN INCREASED FOR AREA
```

Considerations on running OAE

When you submit your JCL and control statements, a word of caution is in order. Remember that once the utility is successfully run, the DEDB area is extended, and it cannot be undone. In other words, once the area is extended, you cannot go back to a smaller area.

Warning: In view of the irrevocability of the utility, proceed with caution.

Consider first running with the TEST keyword, which verifies the correctness of your parameters and shows you the structure of the new area—without actually performing the extension.

You might also choose to specify the SDEPWARN keyword to run OAE with the shadow expansion. Using this keyword, if you inadvertently make the SDEP area smaller than it was, OAE issues a warning message and ends abnormally. If the SDEP area remains the same or is increased, no warning message is issued and processing continues.

Considerations on image copy extension

OAE can extend both the DEDB area and the image copies of the area in a single job step. OAE extends the size of the image copy data sets that are created before the area extension by inserting dummy SDEP CIs in the extended portion. By extending the image copies, you can later use the extended image copies to recover the DEDB area.

To extend an image copy data set, the following conditions must be met:

- The image copy must be registered in the RECON data sets.
- The DD statements that specify the target extended image copy data sets must be coded in the JCL stream.
- Only the SDEP portion of the area is extended.

OAE can extend the primary image copy, the secondary image copy, or both.

Review the following considerations before you extend image copies with OAE:

- OAE extends the latest image copy data set that is registered in the RECON data sets. If the image copy data set is written on a tape volume, you must mount the tape volume.
- OAE can extend the image copy that is compressed with compression routine FABJCOMPx, which is provided by IMS HP Image Copy.
- OAE cannot extend IC2 image copies that are created by the IMS Database Image Copy 2 Utility (DFSUDMT0). Also, OAE cannot extend image copies that are created by the flash copy process.
- For shadow expansion, if the size of the target area data set is the same as the size of the original area data set, image copy data sets of the area can be used for recovery without having them extended. Therefore, when the size is the same, OAE does not extend the image copy data set.
- If both the primary image copy and the secondary image copy are registered in the RECON data sets and if OAE extends only one of them due to an error, OAE invalidates the image copy that is not extended in the RECON data sets to maintain consistency.

Considerations on recovery and restart

This topic describes considerations for the restart and the recovery methods of the tool.

Recovery

As part of updating the IMS control blocks (DMAC) with the new AREA structure information, all data-sharing IMS systems are notified by OAE of the AREA structure change. If this notification process fails, the AREA is marked recovery needed, and all ADS(s) are stopped. The AREA can then be recovered, but only into a data set at least as large as the input ADS. After recovery, the AREA will be partially extended, and the utility must be restarted to complete the process from the point of failure.

If OAE extended the image copy that is taken before the OAE run along with the DEDB area, you can use that extended image copy to recover the extended DEDB area.

One method for recovery is to recover the DEDB area by using the extended image copy data set with the point-in-time recovery (PITR) of IMS Database Recovery Facility. When you do so, be careful about the time that you specify for recovery. If you specify a recovery time that is after the OAE run, the extended DEDB area recovers completely. However, if you specify a recovery time that is earlier than the OAE run, the recovered DEDB area will have the size of the extended DEDB area (because the extended image copy is used for recovery), but the DMAC information will be that of before the OAE run because the log updates during the OAE job are not applied. In this case, if you want to resize the recovered DEDB area to the size before the OAE run, use the DEDB Area Data Set Create utility (DBFUMR10) to copy the recovered DEDB area data set to a smaller data set.

Restart

If the utility ends prematurely, restart the utility by using the same input and the following parameters:

- For shadow expansion, specify the RESTART parameter on the EXIT statement of SYSIN DD.
- For original expansion, specify the RUNTYPE=RESTART keyword on the EXTEND statement of FPXIN DD.

The area is accessible to the online environment until OAE is restarted.

See [“Recovering and restarting OAE” on page 758](#) when restart fails.

Important: OAE cannot extend image copies in the restart process. When restart processing ends, create an image copy of the extended area as soon as possible.

APF authorization

When you run OAE for the original expansion with EXTEND_TYPE=RENAME, load libraries for FPO and the IMS Tools Online System Interface must be APF-authorized.

DBRC SCI registration exit routine

If you are using the RECON data set, in which an IMSplex name and a DBRC group are set, you must use the DBRC SCI registration exit routine (DSPSCIX0) to pass the IMSplex name and the DBRC group to DBRC. You must specify DSPSCIX0 to use the parallel RECON access.

The SCI registration exit routine is called by DBRC before registering with the SCI. The exit routine supplies the IMSplex name and the DBRC group needed for SCI registration. For information about the DBRC SCI registration exit routine, see *IMS Exit Routines* for the version of IMS that you are using.

Performance considerations

The create step of OAE execution impacts on system performance because the IMS/FP Create utility, DBFUMRIO, is run. For shadow expansions, you can improve the performance by pre-formatting the target area data sets so that the format phase of the DBFUMRIO utility is skipped.

Once the data is copied to the extended ADS(s), there is negligible impact upon online performance during the remainder of the utility run.

Original expansion with EXTEND_TYPE=COPY has more impact on system performance because the IMS/FP Create utility, DBFUMRIO, is run twice.

Running OAE

OAE operates on an area level. Each invocation of OAE specifies the area name as a parameter, and OAE will access only this area. However, multiple areas can be processed sequentially in a single run, similar to the DBFUMRIO utility.

Step 1: Preparatory steps for OAE

Before running OAE, you must perform the preparatory steps.

About this task

The steps you take depend on whether you want to extend the IOVF or the SDEP portion of the area.

For details about extending the IOVF and the SDEP portions of the area, see [“Considerations on area extension”](#) on page 749.

Important: Step [“1”](#) on page 754 through Step [“5”](#) on page 755 are for allocating a larger data set, which is required only for the shadow expansion. If you specify the ROOT parameter in the SYSIN DD, skip Step [“3”](#) on page 755 through Step [“5”](#) on page 755. If you have decided to use the original expansion, start from Step [“6”](#) on page 755.

Procedure

1. Define a new data set.

The data set is defined by the VSAM definition. The size of the CI must be identical to that defined for the other data sets of the same area. If the allocated space for the data set is not equal to or greater than that defined for the other data sets of the same area, the OAE utility will determine this and will end abnormally.

2. The new ADS must be registered with DBRC.

Use the DBRC INIT.ADS command with the UNAVAIL parameter to register the new ADS with DBRC. This command can be issued by one of the following ways:

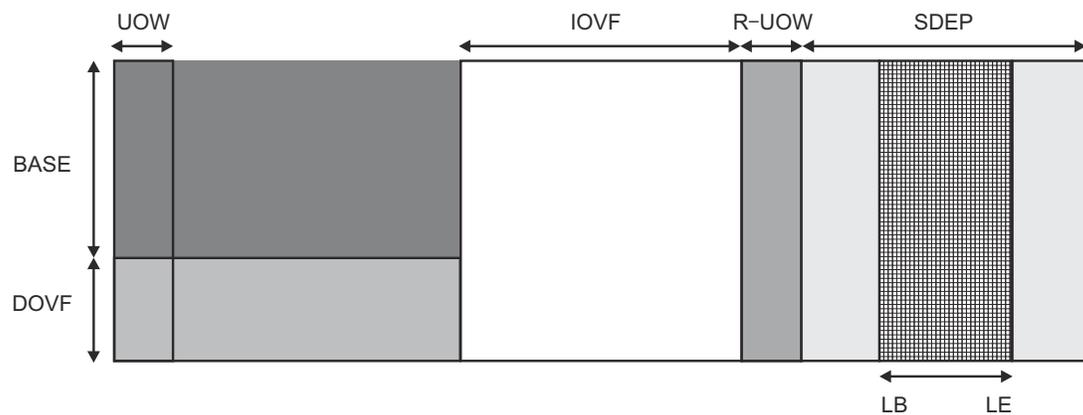
- Use of the Database Recovery Control Utility (DSPURX00) in batch
- Use of the /RMINIT IMS command

Related reading: For information about the INIT.ADS command, the DSPURX00 utility, and the /RMINIT command, see *IMS Commands* for the version of IMS that you are using.

3. (For IOVF extension only) Change the ROOT parameter in the database descriptors (DBDs) in the DBDLIB for the area(s) to be extended.
4. (For IOVF extension only) Run DBDGEN.
5. (For IOVF extension only) Run ACBGEN.
6. If necessary, run the SDEP SCAN/DLET utilities.

The SDEP SCAN/DLET utilities are used to delete all of the SDEP segments with RBAs within the Cannibalized SDEP CIs (CSC) portion. The CSC for an area are those SDEP CIs which lie between the current first SDEP RBA and the new first SDEP RBA created as a result of IOVF extension, as shown in the following figure.

Before expansion



After expansion

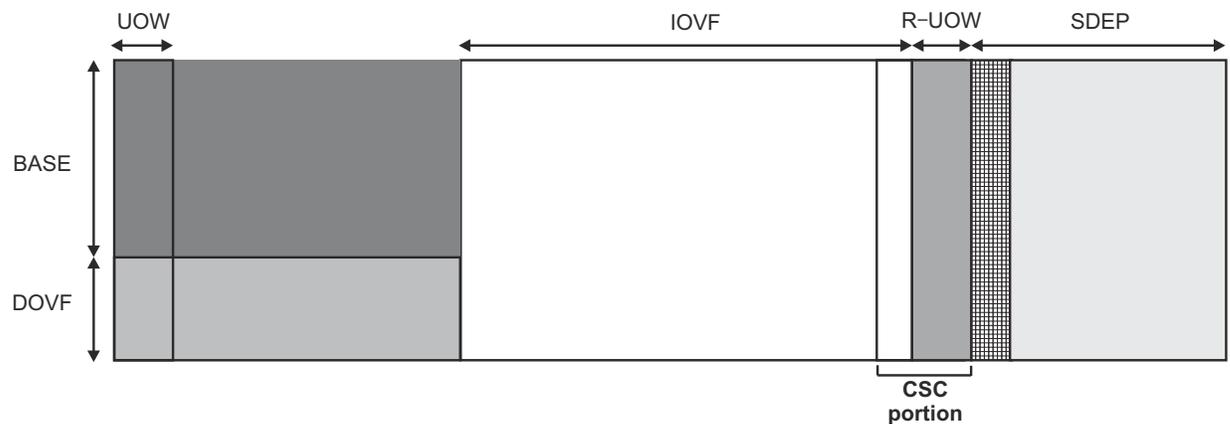


Figure 326. CSC portion that is created as a result of IOVF extension

If SDEP segments exist in the CSC portion, you must run the SDEP SCAN/DLET utilities before extending the area with OAE.

Note: You do not need to run the SDEP SCAN/DLET utilities when you want to extend the IOVF portion of the area that has no SDEP segments in the CSC portion or has no SDEP definition in DBD, or when you want to extend only the SDEP portion.

To check if SDEP segments with RBAs exist within the CSC portion, follow these steps:

a) Run OPC or ODM to obtain the current SDEP LB and LE values.

An SDEP LB value that is larger than the SDEP LE value indicates that SDEP segments with RBAs exist within the CSC portion. If the SDEP LB value is smaller than the SDEP LE value, continue with the next step.

b) Calculate the first SDEP RBA (that is, the first SDEP RBA value after the extension) from the UOW and the ROOT parameters.

c) A calculated first SDEP RBA value that is larger than the current SDEP LB value indicates that SDEP segments with RBAs exist within the CSC portion. Otherwise, SDEP segments with RBAs do not exist in the CSC portion.

Related reading: For a JCL example to extend IOVF portion of an area that has SDEP segments, see [“Example 6: Extending the IOVF portion of the area that has SDEP segments” on page 775.](#)

7. Issue the /VUNLOAD IMS command to remove the area from VSO (If applicable).

Note that the current area must be removed from Virtual Storage Option (VSO) before running OAE.

Related reading: For information about the /VUNLOAD command, see *IMS Commands* for the version of IMS that you are using.

Step 2: Coding and running JCL

The OAE program runs as an IFP utility application, using standard JCL and control statements. You need to specify an EXEC statement and DD statements that define the input and output data sets in your JCL.

About this task

You must execute the OAE utility in a separate job step. The program cannot switch from one database to another or from one area to another within the same job step.

Procedure

1. Specify the EXEC statement in one of the following formats:

```
//name EXEC procname
```

where *procname* is the IMS-supplied IFP procedure FPUTIL or any other IFP utility supporting procedure.

```
//name EXEC PGM=DFSRR00,PARM='IFP,dbdname,DBF#FPU0'
```

where *dbdname* is the DBDNAME associated with the DEDB.

Related reading:

- For information about the FPUTIL utility, see *IMS System Definition* for the version of IMS that you are using.
- For information about the DEDB online utility commands, see *IMS Database Utilities* for the version of IMS that you are using.

2. Use the REGION= parameter on either the JOB or the EXEC statement to specify the region size.

Region size varies with:

- UOW size (CI number x CI size)
- CI size
- (CI number x CI size) for the rest of the buffer after IOVF in case of SDEP

If there is no SDEP portion in the DBD source definition of an area, OAE acquires (GETMAIN) buffers after IOVF, up to the high allocated RBA. Specify the region size big enough to contain the buffer acquired.

The required region size increases when you extend image copy data sets along with the DEDB area:

- For shadow expansion, an additional 11 MB is required
 - For original expansion, an additional 7 MB is required
3. Specify the DD statements to define input data sets and output data sets.
For shadow expansion, specify the control statements for SYSIN DD. For original expansion, specify the control statements for FPXIN DD.

Shadow expansion

The JCL for shadow expansion is very similar to that of the ADS Create utility, DBFUMRIO. You can choose to use your Create utility JCL as a model for your OAE JCL, noting that three items have been added:

- //ACBLIB
- TYPE TOOL
- EXIT EXTEND

Original expansion

The original expansion process defines and allocates a larger data set, and registers it temporarily as a new ADS with DBRC processes. It requires a new //FPXIN DD statement instead of //SYSIN DD. You need to add only the //FPXIN DD statement to the utility JCL. //SYSIN DD should not be specified.

Note: The original expansion with the EXTEND_TYPE=RENAME option needs an FPXBINTP DD statement. The subsequent OAE Rename program step will rename the temporary ADS to the original ADS.

4. Run the JCL.

Related reference

Examples for OAE

There are many ways to run the OAE utility. The examples provided in the following topics show some of the typical ways to run the OAE utility.

Tips and best practices for the FPO Online Area Extender utility

The FPO Online Area Extender utility (OAE) supports several keywords that improve the performance of OAE jobs and that further simplify management of DEDBs.

Step 3: Post execution steps for OAE

Once you have successfully run OAE and the IOVF portion is extended, bring the new DEDB member in ACBLIB online. IMS will function normally without an ACBLIB switch, but the DBD/ACB IMS is executing with will not match the structure of the current DEDB area.

Procedure

The steps for activating the new DEDB member depend on where the DEDB member is at the end of OAE processing.

- If your new DEDB member is in the *staging ACBLIB* for IMS, do the following to activate the ACBLIB:
 - a) Run the Online Change utility (OLCUTL) to copy the staging ACBLIB to the inactive ACBLIB.
 - b) Issue the IMS /MODIFY command to activate the inactive ACBLIB.
- If your new DEDB member is in the *inactive ACBLIB* for IMS, to the inactive ACBLIB, perform IMS Online Change to activate the inactive ACBLIB.
- If your new DEDB member is in your *own ACBLIB*, do as follows:
 - a) Perform an ACBGEN of the new DBD into the staging ACB library.
 - b) Run the Online Change utility (OLCUTL) to copy the staging ACBLIB to the inactive ACBLIB.
 - c) Issue the IMS /MODIFY command to activate the inactive ACBLIB.

What to do next

Finally, if you had removed the area from VSO before running, be sure to reload it into VSO at this point.

Related reading:

- For more information about the Online Change utility, see *IMS System Utilities* for the version of IMS that you are using.
- For more information about the /MODIFY command, see *IMS Commands* for the version of IMS that you are using.

Recovering and restarting OAE

These topics describe restart procedures for Online Area Extender.

When the shadow or the original expansion fails

This topic describes the steps you should take when the shadow or the original expansion has failed.

About this task

If IMS, OS, or system fails, then restart the IMS online system first. For shadow expansion, rerun OAE by adding the RESTART option, and for original expansion, change the RUNTYPE=RESTART parameter.

Procedure

If one of the following messages is issued in the OAE restart process, follow the instructions:

FPX0091E OAE restart not processed. Follow the RESTART procedure in the OAE User's Guide.

- a. Check the status of the original ADS in DBRC.
 - If the original ADS is in AVAIL status in DBRC:
 - i) If the temporary ADS is registered with DBRC, check its status in DBRC:
 - If the temporary ADS is in AVAIL status, separate the temporary ADS from the IMS online system by using the /STOP ADS command, and then unregister the temporary ADS with DBRC.
 - If the temporary ADS is in UNAVAIL status in DBRC, unregister the temporary ADS with DBRC.
 - ii) If DELETE is not specified on the DSNAME keyword, delete the temporary ADS cluster if exists.
 - If the original ADS is in UNAVAIL status in DBRC, complete the following steps:
 - i) Run the DEDB Area Data Set Create utility (DBFUMRIO) by specifying the DD name of the original ADS.
 - ii) Separate the temporary ADS from the IMS online system by using the /STOP ADS command.
 - iii) Unregister the temporary ADS with DBRC.
 - iv) If DELETE is not specified on the DSNAME keyword, delete the temporary ADS cluster if exists.
- b. If ICEXTEND=YES is specified, delete and redefine the target extended image copy data sets that are specified as DISP=NEW on FPXBIC01 DD, FPXBIC02 DD, or both.
- c. Rerun OAE without the RUNTYPE=RESTART option.

FPX0292E OAE restart is not required. Remove the RESTART parameter from the EXTEND statement.

- a. If ICEXTEND is specified, delete and redefine the target extended image copy data sets that are specified as DISP=NEW on FPXBIC01 DD, FPXBIC02 DD, or both.
- b. Rerun OAE without the RESTART option.

FPX0136W The OAE RESTART process extended the temporary ADS. It did not copy the temporary ADS back to the original ADS.

- a. If the extended temporary ADS is used instead of the original ADS, do nothing.
- b. Identify the original ADSs from the subtexts of message FPX0136W.
- c. Redefine the original ADS clusters if they do not exist.
- d. Run the DEDB Area Data Set Create utility (DBFUMRIO) by specifying the DD names of the original ADSs.
- e. Separate the temporary ADS from the IMS online system by using the /STOP ADS command.
- f. Unregister the temporary ADS with DBRC.
- g. Delete the temporary ADS cluster.

DFS3736A VALIDATION CALL FAILED FOR AREA=areaname DD=ddname CODE=xx

- a. Check the status of the original ADS in DBRC.
 - If the original ADS is in AVAIL status in DBRC, separate the target ADS from the IMS online system by using the /STOP ADS command.
 - If the original ADS is in UNAVAIL status in DBRC, run the DEDB Area Data Set Create utility (DBFUMRIO) by specifying the DD name of the original ADS, and then separate the target ADS from the IMS online system by using the /STOP ADS command.
- b. If ICEXTEND=YES is specified, delete and redefine the target extended image copy data sets that are specified as DISP=NEW on FPXBIC01 DD, FPXBIC02 DD, or both.
- c. Rerun OAE without the RESTART option.

When the OAE Rename program fails after the original expansion with EXTEND_TYPE=RENAME

This topic describes the steps to take when the OAE Rename program fails after the original expansion with EXTEND_TYPE=RENAME.

About this task

Refer to both the original ADS name and temporary ADS name in the FPXBINTP data set printed in SYSPRINT DD. Check and find one of following conditions in sequence, and follow the description:

FPX0151E Incorrect FPXBINTP file was specified.

Exceeded the FPXBINTP creation time limit, which is three minutes

If this error message is issued due to exceeding the FPXBINTP creation time limit, which is 3 minutes, and when the temporary ADS name can be renamed after a lapse of 3 minutes, add the TIMECHECK=NO statement in the OAE Rename program and rerun the JCL. Or go through the recovery procedure from Step “1” on page 760.

FPX0188E Load module library is not APF-authorized.

Check the setting of IMS Tools Online System Interface. Correct the setting of IMS Tools Online System Interface and rerun the JCL or go through the recovery procedure from Step “2” on page 760. For information about IMS Tools Online System Interface, see the *Tools Base IMS Tools Common Services User's Guide*.

FPX0191E TOITIME time over. REQUEST command=xxxxxxx

Go through the recovery procedure from Step “2” on page 760.

When both the original ADS and temporary ADS exist:

1. Check the AREA status under all IMS systems that use the area.
2. If all areas are not stopped under all IMS systems, then go through the recovery procedure from Step “2” on page 760.

When the original ADS does not exist and the temporary ADS exists:

Go through the recovery procedure from Step “5” on page 760.

When the original ADS exists and the temporary ADS does not exist:

1. Check the status of the original and temporary ADSs in DBRC.
2. If the temporary ADS records exist in DBRC, go through the recovery procedure from Step [“6.a” on page 760](#).
3. If the status of the original ADS is "UNAVAIL" in DBRC, go through the recovery procedure from Step [“6.b” on page 760](#).
4. If the AREA status under all IMS systems that use the area are still stopped, go through the recovery procedure from Step [“7” on page 760](#).
5. If the AREA status under all IMS systems that use the area are started, there is nothing to do.

Procedure

The following steps describe the manual recovery procedure.

1. Check IMS Tools Online System Interface environment.
2. Stop the area in all IMS systems.
Issue `"/STOP areaname GLOBAL"`.
3. Check the area status in all IMS systems.
Issue `"/DISPLAY areaname"`.
4. Delete the original ADS.
5. Rename the temporary ADS to the original ADS.
6. Update DBRC.
 - a) Delete the temporary ADS record.
 - b) Change the original ADS status to "AVAILABLE".
7. Start the area in all IMS systems.
Issue `"/START areaname"`.

Region size requirements for OAE

The region size for OAE depends on the size of a UOW.

Because OAE operates in a virtual storage region as an IMS IFP utility, the region size required to run OAE is larger than the following values:

- OAE requires the buffer set and IOVF work buffer calculated by the following formulas:

```
Buffer set:      (UOW1 + 5) * CISZ
IOVF work buffer: CISZ * 125
```

For the original expansion, an additional work buffer is required for the DEDB Area Data Set Create utility's use:

```
Work buffer: (UOW1 + 7) * CISZ
```

- The region size required by OAE also depends on whether the image copy extension option is used (ICEXTEND keyword). The required region size increases when you extend image copy data sets along with the DEDB area:
 - For shadow expansion, an additional 11 MB is required
 - For original expansion, an additional 7 MB is required

DD statements for OAE

DD statements for OAE determine the input and output data sets and specify how to run OAE.

The following table summarizes the DD statements.

Table 93. OAE DD statements

DDNAME	Use	Format	Required or optional
JOBLIB or STEPLIB	Input	PDS	Required
ACBLIB	Input	PDS and member	Required if either of the following conditions is met: <ul style="list-style-type: none"> Shadow expansion: Extend the IOVF portion without the ROOT=(aaaaa,bbbb) keyword Original expansion: IOVF_ROOT=ACBLIB is specified Requirement: ACBLIB DD must contain the DBD member name.
RECONn	Input/ Output		Required when both of the following conditions are met: <ul style="list-style-type: none"> DFSMDA members are not used Original expansion or image copy extension is requested
SYSPRINT	Output	LRECL=121	Required
SYSOUT	Output		Optional
FPXOPTS	Input	LRECL=80	Optional
SYSIN or FPXIN	Input	LRECL=80	SYSIN required for shadow expansion, FPXIN required for original expansion
FPXBINTP	Output	PS RECFM=VB	Required for original expansion with EXTEND_TYPE=RENAME
FPXBIC01	Output	PS	Required when extending primary image copy data set
FPXBIC02	Output	PS	Required when extending secondary image copy data set
FPXBXRTP	Output	PS	Optional

STEPLIB DD

Defines the library that contains the OAE utility. The data set that contains the DBRC SCI registration exit routine (DSPSCIX0) must be concatenated when using the RECON data set in which an IMSplex name is set.

If the image copy extension option is specified and the image copy data set to extend is compressed with the compression routine FABJCMpx (provided by IMS HP Image Copy), the load library of IMS HP Image Copy must be concatenated.

The IMS Tools Base library (SHKTLOAD) is required if you specify the ITKBSRVR keyword.

ACBLIB DD

Identifies the ACBLIB and DEDB member to be used. This DD must also contain the name of the DBD member.

This DD statement is required for extending the IOVF portion when either of the following conditions is met:

- Shadow expansion: Extend the IOVF portion without the `ROOT=(aaaaa,bbbb)` keyword
- Original expansion: `IOVF_ROOT=ACBLIB` is specified

This DD statement is ignored when either of the following conditions is met:

- SDEP expansion
- IOVF expansion with one of the following keywords specified:
 - Shadow expansion: `ROOT=(aaaaa,bbbb)` keyword
 - Original expansion: `IOVF_ROOT=(number2,overflow2)` keyword or `IOVF_UOW` keyword

This DD statement must specify the ACBLIB library that contains the member of the present DEDB.

RECON n DD

Defines the DBRC RECON data sets. These DD statements must be specified when DFSMDA members are not used and original expansion or image copy extension is requested. The RECON n data sets must be the same data sets as those used by the IMS control region because OAE refers to and updates the records in the RECON data sets.

If these statements are not specified in the JCL statement when the data sets are necessary, the data sets are allocated dynamically by using DFSMDA members.

SYSPRINT DD

Defines the data set that contains output messages.



Attention: You cannot allocate the OAE SYSPRINT DD output stream to a data set because SYSPRINT DD is also used by IMS. If you specify SYSPRINT DD as a data set, some output might be overridden or unpredictable open/close errors might occur. You must specify SYSPRINT DD as SYSOUT.

SYSOUT DD

Defines an output data set; OAE writes images of its control statements to SYSOUT.

SYSIN DD

Required DD statement for the shadow expansion process. SYSIN DD and FPXIN DD are mutually exclusive. SYSIN DD describes the input control data set that contains the IMS IFP utility control statements. A minimum of five statements is required, as shown in the following example:

```
TYPE TOOL
AREA areaname
DDNAME ddname
.
.
EXIT EXTEND ...
GO
```

FPXOPTS DD

Defines the input data set that contains the option control statements for the process. FPXOPTS usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte fixed-length records. Block size, if coded, must be a multiple of 80. It can be specified as DUMMY, NULLFILE, or an empty file.

For details of the option control statements, see [“Option control statement for OAE”](#) on page 763.

FPXIN DD

Required DD statement for the original expansion process. FPXIN DD and SYSIN DD are mutually exclusive. FPXIN DD specifies control statements for the run process. It is not allowed to be specified as DUMMY, NULLFILE, or empty file.

FPXBINTP DD

Required DD statement for the original expansion process with the EXTEND_TYPE=RENAME option. It contains the interface information for the subsequent step, which will be used for the rename process. Do not specify DCB parameters except RECFM=VB. Do not specify DISP=MOD for this DD statement.

FPXBIC01 DD and FPXBIC02 DD

Defines the target extended image copy data sets when the image copy extension option is specified. The extended image copy data sets can be used to recover the extended DEDB area.

FPXBIC01 DD is used to extend the latest primary image copy data set that is registered in the RECON data sets. FPXBIC02 DD is used to extend the latest secondary image copy data set that is registered in the RECON data sets. After extending the image copies, OAE modifies the last image record in the RECON data sets to replace the information of the original image copy with the information of the extended image copy.

If you want OAE to perform both area extension and image copy extension, you must specify FPXBIC01 DD, FPXBIC02 DD, or both DD statements in the JCL stream.

Do not code the DCB information in the JCL statement. You must not specify DISP=MOD for these DD statements.

Considerations:

- If both FPXBIC01 DD and FPXBIC02 DD are not specified in the JCL stream, OAE does not extend the image copies, even if the primary image copy and the secondary image copy are registered in the RECON data sets. When only one of the target extended image copy data sets is specified correctly, OAE extends the DEDB area, extends the image copy that is specified by the correct FPXBIC0x DD, and invalidates the other image copy (that is, either the primary image copy or the secondary image copy) in the RECON data sets.
- If both FPXBIC01 DD and FPXBIC02 DD are specified as DUMMY or NULLFILE, OAE does not extend the image copies. If only one of the FPXBIC0x DDs is specified correctly and the other specifies DUMMY or NULLFILE, OAE extends the DEDB area, extends the image copy that is specified by the valid FPXBIC0x DD, and invalidates the other image copy in the RECON data sets.
- The target extended image copy data sets must not be stacked data sets.
- FPXBIC01 DD and FPXBIC02 DD must specify different data sets.
- The data set names of the original image copy data and the target extended image copy must be different.
- The data set names can contain a reference to a generation data group (GDG), with or without a relative generation number such as "(+1)".

FPXBX RTP DD

Defines the SYSPRINT data set for the DSPURXRT module.

If this DD statement is not specified, the OAE utility allocates this DD data set dynamically.

When the OAE utility extends image copies, this data set must be large enough to contain the result of the LIST.HISTORY command of DBRC.

Do not specify DUMMY or NULLFILE for this DD statement. Do not code the DCB information in the JCL.

Option control statement for OAE

The option control statement defines the user description of the OVERRIDE control statement, which specifies information for setting temporary override values of product message defaults.

The option control statements must start with a line that contains the OPTIONS command. The control statement that can be specified is:

OVERRIDE control statement

The OVERRIDE control statement specifies information for setting temporary override values of product message defaults. This control statement is optional. The OVERRIDE control statement can be specified in FPXOPTS DD.

OVERRIDE statement keywords

The keywords associated with the OVERRIDE statement are not available for site default options. For the format of the OVERRIDE statement, see [“OVERRIDE control statement”](#) on page 850.

For the OVERRIDE statement, you can specify the keywords that are listed in the following table.

Table 94. OVERRIDE statement keywords

Keyword	Required or optional	Description	Topic
MSG_LIM	Optional	Specifies the maximum number of repetitive messages that are to be issued before suppression.	“MSG_LIM keyword” on page 897
MSG_NUM	Required	Specifies the message number that is to be suppressed.	“MSG_NUM keyword” on page 898

Related reference

[Keyword reference for FPO](#)

These topics explain the control statements and keywords of FPO and their syntax rules.

[Syntax rules for FPXOPTS](#)

FPXOPTS control statements and their associated keywords can be coded in free format in columns 1 through 72, subject to certain syntax rules.

Utility control statements for OAE

The utility control statements for OAE define the user description of the control statement specified in SYSIN DD and the EXTEND control statement specified in FPXIN DD, both of which control the OAE process.

SYSIN DD data set (Shadow expansion)

The SYSIN DD data set contains the control commands of OAE for the shadow expansion.

For shadow expansion, OAE runs as an IFP utility, by using standard JCL and control statements, plus one or more additional files unique to OAE.

All the following control commands of the SYSIN data set are required for the shadow expansion:

Table 95. Control commands for the SYSIN data set

Command name	Description
TYPE TOOL	This indicates that a DBTOOLS utility is to be run.
AREA	Identifies the area to be processed. Using standard IFP utility support, you can specify more than one AREA per run. Each sequence of AREA and GO statements results in a new invocation of the utility.
DDNAME	Specifies the DDNAME(s) of the (new) area data set(s) where the area will reside after the extension. The area data set(s) must be registered in DBRC, with a maximum of 6 DDNAMEs specified. The total number of preexisting and new ADS(s) cannot exceed 7, since during utility execution, both the preexisting and the new ADS(s) can be open concurrently.

Table 95. Control commands for the SYSIN data set (continued)

Command name	Description
EXIT	<p>This input control statement contains important keywords.</p> <p>EXTEND Specifies that the OAE utility is to be invoked to perform the extension.</p> <p>ICEXTEND Specifies whether to extend the latest image copy data set that is registered in the RECON data sets, along with the DEDB area.</p> <p>Restriction: IOVF and ICEXTEND are mutually exclusive in IMS 14. This restriction can be removed by applying APAR PI43928 to IMS 14.</p> <p>ICERR_CONT Specifies whether to continue extending the DEDB area when image copy extension fails.</p> <p>This keyword is valid only when the ICEXTEND keyword is specified.</p> <p>IOVF or SDEP Specifies which portion of the data area is to be extended. Remember, if you specify IOVF here, you must include an ACBLIB DD statement in your JCL.</p> <p>RESTART Specifies that this run of the utility is a restart. See “Restart” on page 753 for a discussion of when you would want to specify the RESTART keyword.</p> <p>ROOT=(aaaaa,bbbb) Specifies the value to change the ROOT parameter in the database descriptors (DBDs) in the ACBLIB for the area(s) that is to be extended. This keyword syntax is the same as the ROOT= parameter of IMS DBD definition. This keyword must be accompanied with the IOVF keyword. If you are extending only the SDEP portion, this keyword, even if it is specified, is ignored.</p> <p>SDEPWARN Specifies that a warning is to be issued and processing is to end abnormally if the SDEP portion of the area is decreased rather than increased. This is a safety feature.</p> <p>If it is your intent to decrease the SDEP portion of the area, remove the SDEPWARN keyword from your control statements.</p> <p>TEST Specifies that this is just a trial run. All the input parameters are only verified for correctness. Output statistics are generated showing the structure of the new area without actually performing the extension.</p>
GO	<p>This control command is used to signal the end of the control commands and indicates that the extension should be performed.</p>

FPXIN DD data set (Original expansion)

The FPXIN DD data set contains the control statements of OAE for the original expansion.

The control statements and their parameters that can be specified to run an OAE job for the original expansion are explained in this topic. The control statements that can be specified are:

GLOBAL control statement

The GLOBAL control statement defines the job step keyword values. If specified, GLOBAL must be the first command in the input stream. The GLOBAL control statement can be specified only in FPXIN DD.

EXTEND control statement

The EXTEND control statement specifies information that controls how the OAE run is to be processed. This control statement is mandatory for the original expansion.

Subsections:

- [“GLOBAL statement keywords” on page 766](#)
- [“EXTEND statement keywords” on page 766](#)

GLOBAL statement keywords

The GLOBAL control statement specifies the job step keyword values. This control statement is optional.

The keywords associated with the GLOBAL statement are available for site default options.

For the GLOBAL statement, you can specify the keywords that are listed in the following table. For the format of the GLOBAL statement, see [“GLOBAL control statement” on page 847](#).

Table 96. GLOBAL statement keywords

Keyword	Required or optional	Description	Topic
DBRC	Optional	Specifies whether to store reports that are associated with a RECON to the IMS Tools KB repository.	“DBRC keyword (for GLOBAL)” on page 868
ITKBLOAD	Optional	Specifies the name of the load module library for IMS Tools KB.	“ITKBLOAD keyword” on page 890
ITKBSRVR	Optional	Specifies the name of the IMS Tools KB server XCF group.	“ITKBSRVR keyword” on page 891

EXTEND statement keywords

The EXTEND control statement specifies information that controls how the OAE run is to be processed.

The keywords associated with the EXTEND statement are not available for site default options.

For the EXTEND statement, you can specify the keywords that are listed in the following table. For the format of the EXTEND statement, see [“EXTEND control statement” on page 842](#).

Table 97. EXTEND statement keywords

Keyword	Required or optional	Description	Topic
AREA	Required	Specifies the area to process.	“AREA keyword” on page 864
DATACLAS	Optional	Specifies the name of the SMS data class for the temporary area data set.	“DATACLAS keyword” on page 867
DDNAME	Required	Specifies the DDNAME of the temporary area data set where the area will be copied.	“DDNAME keyword” on page 869
DSNAME	Required	Specifies the data set name of the temporary area data set where the area will be copied.	“DSNAME keyword” on page 870
EXTEND_TYPE	Optional	Specifies the extend type.	“EXTEND_TYPE keyword” on page 875
ICEXTEND	Optional	Specifies whether to extend the latest image copy data set that is registered in the RECON data sets, along with the DEDB area.	“ICEXTEND keyword” on page 884
ICERR_CONT	Optional	Specifies whether to continue extending the DEDB area when image copy extension fails.	“ICERR_CONT keyword” on page 884

Table 97. EXTEND statement keywords (continued)

Keyword	Required or optional	Description	Topic
IOVF_ROOT	Optional	Specifies which extension information is to be used when the IOVF portion is extended. Either the IOVF_ROOT, IOVF_UOW, or the SDEP_CI keyword must be specified.	“IOVF_ROOT keyword” on page 889
IOVF_UOW	Optional	Specifies the number of IOVF UOWs to extend when extending the IOVF portion. Either the IOVF_UOW, IOVF_ROOT, or SDEP_CI keyword must be specified.	“IOVF_UOW keyword” on page 889
MAXVOLS	Optional	Specifies the maximum number of volumes for the extended original area data set. This keyword is effective only in SMS environments.	“MAXVOLS keyword” on page 896
MGMTCLAS	Optional	Specifies the name of the SMS management class for the temporary area data set.	“MGMTCLAS keyword” on page 897
PCHKONLY	Optional	Specifies whether to evaluate only the control statement keywords or to also extend the online DEDB area.	“PCHKONLY keyword” on page 901
RUNTYPE	Optional	Specifies the run mode type.	“RUNTYPE keyword” on page 911
SDEP_CI	Optional	Specifies the number of extended SDEP CIs when the SDEP portion is extended. Either the SDEP_CI, IOVF_UOW, or the IOVF_ROOT keyword must be specified.	“SDEP_CI keyword” on page 913
STORCLAS	Optional	Specifies the name of the SMS storage class for the temporary area data set.	“STORCLAS keyword” on page 930
VOLUME	Required	Specifies, for non-SMS environments, one or more volume-serial identifiers for the temporary area data set.	“VOLUME keyword” on page 939
VOLUME_EXT	Optional	Specifies one or more volume serial identifiers for the extended original area data set.	“VOLUME_EXT keyword” on page 939

Related reference

[Keyword reference for FPO](#)

These topics explain the control statements and keywords of FPO and their syntax rules.

[Syntax rules for FPIXIN](#)

FPIXIN control statements and their associated keywords can be coded in free format in columns 1 through 72, subject to certain syntax rules.

Output for OAE

The following topics describe the outputs that are generated by OAE.

Shadow expansion

After the OAE utility is run, you will have a new ADS or MADS for the area, and all preexisting ADS(s) are closed and removed from DBRC.

Original expansion

The current ADSs are once closed and deleted. Then they are redefined by extending their space, and then copied from the temporary ADS by calling the Create Utility internally. Thus the output is the same ADSs but those that are extended. The temporary ADS is deleted at the end of OAE.

SYSPRINT DD data set

The SYSPRINT DD data set contains the statistics report that is generated by the OAE run.

Statistics report

Output statistics are generated showing the structure of the new area. A sample of the output statistics is shown in the following figure; it corresponds to the sample job in [Figure 335 on page 776](#), and assumes that the DBD sources are as follows:

Original AREA statement from DBD source:

```
AREA DD1=DB21AR1,SIZE=(512),UOW=(2,1),ROOT=(2,1)
```

New AREA statement from DBD source:

```
AREA DD1=DB21AR1,SIZE=(512),UOW=(2,1),ROOT=(4,3)
```

FPX0099I Online Area Extender statistics:

```

Original Area Datasets
-----
DDNames          Dataset Names
DB21AR11         DB21AR11
DB21AR12         DB21AR12
Extended Area Datasets
-----
DDNames          Dataset Names
DB21AR13         DB21AR13
DB21AR14         DB21AR14
                  Original Area
                  Datasets
                  -----
                  Extended Area
                  Datasets
                  -----

New RAP and IOVF Size          5,120
Min. allocated Area Size      75,264

First SDEP RBA (HEX)          00001800
Oldest SDEP RBA (HEX)        00007200
Next Avail. SDEP RBA (HEX)   00009200

Area Dataset Size             50,176
CI Size                        512
CIs / UOW                       2
DOVF CIs / UOW                 1

RAP Size (BLKS)                 2
RAP Size (Bytes)                1,024
First IOVF CI #                  4
# of Space Map CIs              1
IOVF Usable Size (BLKS)         1
IOVF Usable Size (BYTES)       512
IOVF Usable Size Change (BLKS) 4

First Reorg CI #                6

First SDEP CI #                 8
SDEP Size (BLKS)                90
SDEP Size (Bytes)              46,080
SDEP Size Change (BLKS)        INCREASE

```

Figure 327. OAE sample output statistics

This discussion focuses on the statistics report, breaking the report down into its component parts.

- The following identifies the DD names and data set names associated with the active ADS(S) for the area. These ADS(s) will become inactive after a successful OAE run.

```

Original Area Datasets
-----
DDNames          Dataset Names

```

DB21AR11	DB21AR11
DB21AR12	DB21AR12

- The following identifies the DD names and data set names associated with the extended ADS(S) for the area. These ADS(s) will become active after a successful OAE run.

```

Extended Area Datasets
-----
DDNames          Dataset Names
-----
DB21AR13         DB21AR13
DB21AR14         DB21AR14
  
```

- The following shows the size of the RAP/IOVF/RUOW portion of the area as a result of the new values coded for ROOT=(number2,overflow2) for the extended ADS.

	Original Area Datasets -----	Extended Area Datasets -----
New RAP and IOVF Size		5,120

- The following shows the VSAM allocation for the smallest of the Extended ADS data sets indicated in the preceding figure.

Min. allocated Area Size	75,264
--------------------------	--------

- The following shows the First SDEP RBA of the extended ADS, and the Oldest SDEP RBA and the Next Available SDEP RBA of the original ADS. These three values can be used to determine whether an IOVF extension will overlap with the CSC portion of the area.

First SDEP RBA (HEX)		00001800
Oldest SDEP RBA (HEX)	00007200	
Next Avail. SDEP RBA (HEX)	00009200	

- This shows the CI and UOW-related statistics for both the original and extended ADS.

Area Dataset Size	50,176	75,264
CI Size	512	512
CIs / UOW	2	2
DOVF CIs / UOW	1	1

- The following shows the RAP-related statistics for both the original and extended ADS.

RAP Size (BLKS)	2	2
RAP Size (Bytes)	1,024	1,024

- The following are IOVF-related statistics for both the original and extended ADS.

First IOVF CI #	4	4
# of Space Map CIs	1	1
IOVF Usable Size (BLKS)	1	5
IOVF Usable Size (BYTES)	512	2,560
IOVF Usable Size Change (BLKS)		4

- The following shows RUOW-related statistics for both the original and extended ADS.

First Reorg CI #	6	10
------------------	---	----

- The following shows SDEP-related statistics for both the original and extended ADS. The final line shows whether there was a net increase or decrease in the SDEP area.

For the statistics which indicate a CI number (First IOVF CI#, First REORG #, First SDEP CI #), the number indicates any offset from the number 1. For example, a value of 6 for the First IOVF CI # indicates that the CI is the seventh (7th).

First SDEP CI #	8	12
SDEP Size (BLKS)	90	135

SDEP Size (Bytes)	46,080		69,120
SDEP Size Change (BLKS)		INCREASE	45

If image copies are extended along with DEDB area extension, the following additional information is printed in the SYSPRINT data set.

- The following figure shows the original last image record, which is the last image record in the RECON data sets before OAE extends the latest image copy data set.

```
Original last image record
-----
IMAGE
2019.319 14:53:58.092851 +09:00 LISTING OF RECON
  RUN = 2019.319 14:53:28.165510 +09:00* RECORD COUNT = 12120
  STOP = 0000.000 00:00:00.000000 +00:00 BATCH USID=0000000001

IC1
DSN=HPFP.DB21AR1.IC1.G0001V00 FILE SEQ=0001
UNIT=3400 VOLS DEF=0001 VOLS USED=0001
VOLSER=HFPVOL

IC2
DSN=HPFP.DB21AR1.IC2.G0001V00 FILE SEQ=0001
UNIT=3400 VOLS DEF=0001 VOLS USED=0001
VOLSER=HFPVOL
```

- The following figure shows the extended last image record, which is the last image record in the RECON data sets after OAE extended the latest image copy data set.

```
Extended last image record
-----
IMAGE
2019.319 14:54:03.163345 +09:00 LISTING OF RECON
  RUN = 2019.319 14:53:28.165510 +09:00* RECORD COUNT =12720
  STOP = 0000.000 00:00:00.000000 +00:00 BATCH USID=0000000001

IC1
DSN=HPFP.DB21AR1.IC1.G0002V00 FILE SEQ=0001
UNIT=3390 VOLS DEF=0001 VOLS USED=0001
VOLSER=VOLIC1

IC2
DSN=HPFP.DB21AR1.IC2.G0002V00 FILE SEQ=0001
UNIT=3390 VOLS DEF=0001 VOLS USED=0001
VOLSER=VOLIC2
```

- The following figure shows the CHANGE.IC command list. These commands were issued to update the last image record in the RECON data sets after OAE extended the latest image copy data set.

```
CHANGE.IC command list
-----
CHANGE.IC DBD(DEDBJN21) AREA(DB21AR1 )
RECTIME('19319145328165510+0900')
FILESEQ(0001) RECDCT(0000012720) UNIT(3390 )
ICDSN(HPFP.DB21AR1.IC1.G0002V00 )
VOLLIST(VOLIC1)

CHANGE.IC DBD(DEDBJN21) AREA(DB21AR1 )
RECTIME('19319145328165510+0900')
FILESEQ2(0001) RECDCT(0000012720) UNIT2(3390 )
ICDSN(HPFP.DB21AR1.IC2.G0002V00 )
VOLLIST2(VOLIC2)
```

FPXBIC01 and FPXBIC02 DD data sets

FPXBIC01 and FPXBIC02 DD data sets contain the image copy data that is extended by OAE. This data can be used to recover the DEDB area extended by OAE.

These data sets are produced only when the image copy extension option is specified.

OAE Rename program

To complete the OAE process for the original expansion with EXTEND_TYPE=RENAME, rename will be done by the OAE Rename program, which should be the step that immediately succeeds the OAE program.

This program gets the interface information that was generated by the OAE original expansion process with the EXTEND_TYPE=RENAME option, deletes the original ADS, then renames the temporary ADS to the original one.

Running the OAE Rename program

The OAE Rename program is run as a standard z/OS batch job, and it should be the step that immediately succeeds the OAE original expansion program run with the EXTEND_TYPE=RENAME option.

Procedure

1. Specify the EXEC statement in the following format:

```
// EXEC PGM=FPXBTOA0,REGION=rrrrM
```

where *rrrr* is the size of the region.

2. Specify the DD statements to define input data sets and output data sets.
3. Run the JCL.

DD statements for the OAE Rename program

The DD statements for the OAE Rename program define how the program is processed.

The following table summarizes the DD statements.

Table 98. DD statements for the OAE Rename program

DDNAME	Use	Format	Required or optional
STEPLIB	Input	PDS	Required
FPXBINTP	Input/ Output	PS RECFM=VB	Required
SYSPRINT	Output	LRECL=133	Required
FPXIN	Input	LRECL=80	Required

STEPLIB DD

Defines the library that contains the OAE Rename program. The library must be APF-authorized. The IMS Tools Base library (SFOILOAD, SGLXLOAD) must be concatenated.

FPXBINTP DD

Defines the input data set that contains the interface information that is produced by the preceding step of the OAE process. The data set will be updated by the program to set an internal indicator.

SYSPRINT DD

Defines the output. If this DD is not specified, SYSPRINT DD will be allocated dynamically with the SYSOUT=* attribute.

FPXIN DD

Defines the input control data set that contains the OAE Rename program control statements.

Utility control statements for the OAE Rename program

This topic describes the utility control statements of the OAE Rename program.

The control statement of the OAE Rename program is required.

The keywords that can be used for the OAE Rename program must be specified in the FPXIN data set.

Keywords for the OAE Rename program

For the OAE Rename program keywords, you can specify the keywords that are listed in the following table. For the details of each keyword, see the related topics.

Table 99. OAE Rename program keywords

Keyword	Required or optional	Description
TOIXCFGR	Optional	Identifies the XCF group that the IMS Tools Online System Interface uses to interface with OAE.
TOITIME	Optional	Specifies the execution time value that is used for /DIS AREA requests.
TIMECHECK	Optional	Specifies whether to verify the time stamp of the interface information in the FPXBINTP DD file.

Related reference

[Keyword reference for FPO](#)

These topics explain the control statements and keywords of FPO and their syntax rules.

[Syntax rules for the OAE Rename program](#)

The keywords and their associated parameter values for the OAE Rename program can be coded in a free format (columns 1-71), subject to certain syntax rules.

Examples for OAE

There are many ways to run the OAE utility. The examples provided in the following topics show some of the typical ways to run the OAE utility.

Examples: Original expansion

These topics provide examples for original expansion.

Example 1: Extending IOVF portion in SMS environment

The following figure shows example JCL to extend the IOVF portion in SMS environment.

```
//EXTEND EXEC FPUTIL,
// DBD=DEDBJN21,           <--- IDENTIFY DATABASE
// IMSID=IMS1
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//SYSOUT DD SYSOUT=*,FREE=END   <--- MANDATORY
//SYSPRINT DD SYSOUT=*,FREE=END <--- MANDATORY
//ACBLIB DD DSN=IMSVS.ACBLIBB(DEDBJN21),DISP=SHR
//FPXIN DD *
*
* EXTEND parameter
*
EXTEND
AREA=DB21AR1
IOVF_ROOT=(10,5)
DDNAME=DBOAETMP
DSNAME=IMSVS.USER.DB21AR1
/*
//RECON1 DD DISP=SHR,DSN=IMSVS.RECON1
//RECON2 DD DISP=SHR,DSN=IMSVS.RECON2
//RECON3 DD DISP=SHR,DSN=IMSVS.RECON3
```

Figure 328. Extending IOVF portion in an SMS environment

Example 2: Extending both IOVF and SDEP portions in SMS environment

The following figure shows example JCL to extend both IOVF and SDEP portions in SMS environment.

```
//EXTEND EXEC FPUTIL,
// DBD=DEDBJN21,           <--- IDENTIFY DATABASE
// IMSID=IMS1
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//SYSOUT DD SYSOUT=*,FREE=END      <--- MANDATORY
//SYSPRINT DD SYSOUT=*,FREE=END   <--- MANDATORY
//ACBLIB DD DSN=IMSESA.ACBLIBB(DEDBJN21),DISP=SHR
//FPXIN DD *
*
* EXTEND parameter
*
EXTEND
AREA=DB21AR1
IOVF_ROOT=(10,5)
SDEP_CI=200
DDNAME=DBOAETMP
DSNAME=IMSVS.USER.DB21AR1
DATACLAS=SMSTMP
/*
//RECON1 DD DISP=SHR,DSN=IMSVS.RECON1
//RECON2 DD DISP=SHR,DSN=IMSVS.RECON2
//RECON3 DD DISP=SHR,DSN=IMSVS.RECON3
```

Figure 329. Extending both IOVF and SDEP portions in an SMS environment

Example 3: Extending the portions with ACBLIB under non-SMS environment

The following figure shows example JCL to extend the portions with ACBLIB under non-SMS environment.

The temporary data set, IMSVS.USER.DB21AR1 is reused.

```
//EXTEND EXEC FPUTIL,
// DBD=DEDBJN21,           <--- IDENTIFY DATABASE
// IMSID=IMS1
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//SYSOUT DD SYSOUT=*,FREE=END      <--- MANDATORY
//SYSPRINT DD SYSOUT=*,FREE=END   <--- MANDATORY
//ACBLIB DD DSN=IMSESA.ACBLIBB(DEDBJN21),DISP=SHR
//FPXIN DD *
*
* EXTEND parameter
*
EXTEND
AREA=DB21AR1
IOVF_ROOT=ACBLIB
DDNAME=DBOAETMP
DSNAME=(IMSVS.USER.DB21AR1,DELETE)
VOLUME=VOLTMP
/*
//RECON1 DD DISP=SHR,DSN=IMSVS.RECON1
//RECON2 DD DISP=SHR,DSN=IMSVS.RECON2
//RECON3 DD DISP=SHR,DSN=IMSVS.RECON3
```

Figure 330. Extending the portions with ACBLIB under non-SMS environment

Example 4: Verifying the parameter to extend the SDEP portion in SMS environment

The following figure shows example JCL to verify the parameter to extend the SDEP portion in an SMS environment.

```
//EXTEND EXEC FPUTIL ,
// DBD=DEDBJN21,                <--- IDENTIFY DATABASE
// IMSID=IMS1
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//SYSOUT DD SYSOUT=*,FREE=END    <--- MANDATORY
//SYSPRINT DD SYSOUT=*,FREE=END  <--- MANDATORY
//ACBLIB DD DSN=IMSESA.ACBLIBB(DEDBJN21),DISP=SHR
//FPXIN DD *
*
* EXTEND parameter
*
EXTEND
  AREA=DB21AR1
  SDEP_CI=2000
  DDNAME=DBOAEtmp
  DSNNAME=(IMSVS.USER.DB21AR1,DELETE)
  RUNTYPE=TEST
/*
```

Figure 331. Verifying the parameter to extend the SDEP portion in an SMS environment

Example 5: Extending the IOVF portion with EXTEND_TYPE=RENAME

The following figure shows example JCL to extend the IOVF portions with EXTEND_TYPE=RENAME.

After OAE, the subsequent rename program must be run.

```
//*=====*
/* OAE EXTEND_TYPE=RENAME *
//*=====*
//EXTEND EXEC FPUTIL, DBD=DEDBJN21, IMSID=imsid
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//          DD DISP=SHR, DSN=IMSVS.SDFSRESL
//          DD DISP=SHR, DSN=IMSVS.PGMLIB
//RECON1 DD DISP=SHR, DSN=IMSVS.RECON1
//RECON2 DD DISP=SHR, DSN=IMSVS.RECON2
//FPXBINTP DD DSN=&&FPXBINTP, DISP=(NEW,PASS),
//          UNIT=SYSDA, SPACE=(TRK,(1,1))
//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//ACBLIB DD DISP=SHR, DSN=IMSVS.ACBLIB(DEDBJN21)
//FPXIN DD *
EXTEND
  AREA=DB21AR1
  IOVF_ROOT=(10,5)
  DDNAME=DBOAEtmp
  DSNNAME=(IMSVS.USER.DB21AR1,DELETE)
  EXTEND_TYPE=RENAME
/*
//*=====*
/* RENAME PGM *
//*=====*
//RENAME EXEC PGM=FPXBTOA0
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//          DD DISP=SHR, DSN=TOSI.SFOILOAD
//          DD DISP=SHR, DSN=IMSVS.SDFSRESL
//SYSPRINT DD SYSOUT=*
//FPXBINTP DD DISP=(OLD,DELETE), DSN=&&FPXBINTP
//FPXIN DD *
TOIXCFGR=xxxxx
/*
```

Figure 332. Extending the IOVF portion with EXTEND_TYPE=RENAME

Example 6: Extending the IOVF portion of the area that has SDEP segments

The following figure shows example JCL to extend the IOVF portion of the area that has SDEP segments.

Before running OAE, run OPC to obtain the SDEP LB and LE values.

Assume that the following RBA values were obtained from OPC run:

- SDEP LB value: X'00078800'
- SDEP LE value: X'0007EC00'

The first SDEP RBA after the extension can be computed from the UOW and the ROOT parameters. Assume that the computed value is X'0007D800'.

From the SDEP LB and LE values (before the extension) and the first SDEP RBA (after the extension), you can identify whether SDEP segments with RBAs exist within the CSC portion. In this sample scenario, SDEP segments with RBAs exist within the CSC portion, so these SDEP segments must be retrieved and deleted by using the SDEP SCAN/DLET utilities or other similar utilities before running OAE.

Related reading: For information about the CSC portion, see [“Step 1: Preparatory steps for OAE” on page 754](#).

The following JCL example runs SDEP SCAN/DLET utilities before extending the area with OAE.

```
//*-----*
//* Scan SDEP
//*-----*
//SDEPSCA EXEC FPUTIL,DBD=DEDBJN23,IMSID=IMSID
//STEPLIB DD DISP=SHR,DSN=IMSVS.SDFSRESL
//SCANCOPY DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//SYSIN DD *
ERRORACTION SCANRUN
TYPE SCAN
AREA DB23AR1
STOPRBA X'000000010007D800'
GO
/*
//*-----*
//* Delete SDEP
//*-----*
//SDEPDEL EXEC FPUTIL,DBD=DEDBJN23,IMSID=IMSID
//STEPLIB DD DISP=SHR,DSN=IMSVS.SDFSRESL
//SYSIN DD *
ERRORACTION SCANRUN
TYPE DELETE
AREA DB23AR1
STOPRBA X'000000010007D800'
GO
/*
/*=====*
/* OAE Original expansion (IOVF/SDEP expansion) *
/*=====*
//OAE EXEC FPUTIL,DBD=DEDBJN23,IMSID=IMSID
//FPU.STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//RECON1 DD DSN=IMSVS.RECON1,DISP=SHR
//RECON2 DD DSN=IMSVS.RECON2,DISP=SHR
//RECON3 DD DSN=IMSVS.RECON3,DISP=SHR
//ACBLIB DD DISP=SHR,
// DSN=IMSESA.ACBLIB(DEDBJN23)
//FPXIN DD *
EXTEND
AREA=DB23AR1
IOVF_ROOT=(99,34)
SDEP_CI=10
VOLUME=VOLTMP
DSNAME=(IMSVS.USER.DB21AR1,DELETE)
DDNAME=DB23AR1T
/*
```

Figure 333. Extending the IOVF portion of the area that has SDEP segments

Example 7: Extending the SDEP portion of the area and the latest image copy data sets

The following figure shows example JCL to extend the SDEP portion of the area. At the same time, OAE extends the latest image copy data sets that are registered in the RECON data sets.

```
//EXTEND EXEC FPUTIL, DBD=DEDBJN21, IMSID=imsid
//*
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//          DD DISP=SHR, DSN=IMSVS.SDFSRESL
//          DD DISP=SHR, DSN=IMSVS.PGMLIB
//          DD DISP=SHR, DSN=HPS.SHPSLMD0 <--- Edit/comp rtn
//SYSOUT DD SYSOUT=*, FREE=END
//SYSPRINT DD SYSOUT=*, FREE=END
//ACBLIB DD DISP=SHR, DSN=IMSVS.ACBLIB(DEDBJN21)
//FPXBIC01 DD DISP=SHR, DSN=HPFP.DB21AR1.IC1(+1)
//FPXBIC02 DD DISP=SHR, DSN=HPFP.DB21AR1.IC2(+1)
//FPXIN DD *
*
*   EXTEND parameter
*
EXTEND
AREA=DB21AR1
SDEP_CI=600
ICEXTEND=YES
DDNAME=DBOAETMP
DSNAME=IMSVS.USER.DB21AR1
DATACLAS=SMSTMP
/*
//RECON1 DD DISP=SHR, DSN=IMSVS.RECON1
//RECON2 DD DISP=SHR, DSN=IMSVS.RECON2
//RECON3 DD DISP=SHR, DSN=IMSVS.RECON3
```

Figure 334. Extending the SDEP portion of the area and the latest image copy data sets (original expansion)

Examples: Shadow expansion

The following topics provide examples for shadow expansion.

Example 1: Extending IOVF portion

The following figure shows example JCL to extend the IOVF portion.

```
//*****
//*-----
//* OAE STEP
//*-----
//*****
//EXTEND EXEC FPUTIL,
// DBD=DEDBJN21, <--- IDENTIFY DATABASE
// IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//          DD DISP=SHR, DSN=IMSVS.SDFSRESL
//          DD DISP=SHR, DSN=IMSVS.PGMLIB
//SYSOUT DD SYSOUT=*, FREE=END <--- MANDATORY
//SYSPRINT DD SYSOUT=*, FREE=END <--- MANDATORY
//ACBLIB DD DSN=IMSVS.ACBLIBB(DEDBJN21), DISP=SHR
//* <-- START SYSIN STREAM
//* (OAE UTILITY CONTROL STATEMENTS)
//SYSIN DD *
* ONLINE TOOL
TYPE TOOL
* TARGET DEDB AREA
AREA DB21AR1
* TARGET AREA DATA SETS (ADS)
DDNAME DB21AR13
DDNAME DB21AR14
* EXIT PROGRAM NAME: EXTEND, CONTROL KEYWORDS
EXIT EXTEND IOVF
* INITIATE OAE PROCESSING
GO
/* <-- END SYSIN STREAM
```

Figure 335. Extending IOVF portion

Example 2: Extending the SDEP portion of the area and the latest image copy data sets

The following figure shows example JCL to extend the SDEP portion of the area. At the same time, OAE extends the latest image copy data sets that are registered in the RECON data sets.

```
//EXTEND EXEC FPUTIL,DBD=DEDBJN21,IMSID=imsid
//*
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=IMSVS.PGMLIB
// DD DISP=SHR,DSN=HPS.SHPSLMD0 <--- Edit/comp rtn
//SYSOUT DD SYSOUT=*,FREE=END
//SYSPRINT DD SYSOUT=*,FREE=END
//ACBLIB DD DISP=SHR,DSN=IMSVS.ACBLIB(DEDBJN21)
//FPXBIC01 DD DISP=SHR,DSN=HPFP.DB21AR1.IC1(+1)
//FPXBIC02 DD DISP=SHR,DSN=HPFP.DB21AR1.IC2(+1)
//* <-- START SYSIN STREAM
//* (OAE UTILITY CONTROL STATEMENTS)
//SYSIN DD *
*
TYPE TOOL
*
AREA DB21AR1
*
DDNAME DB21AR13
DDNAME DB21AR14
*
EXIT EXTEND SDEP ICEXTEND
*
GO
/*
//RECON1 DD DISP=SHR,DSN=IMSVS.RECON1
//RECON2 DD DISP=SHR,DSN=IMSVS.RECON2
//RECON3 DD DISP=SHR,DSN=IMSVS.RECON3
```

Figure 336. Extending the SDEP portion of the area and the latest image copy data sets (shadow expansion)

Chapter 23. Reorganizing a DEDB online with FPO

You can reorganize an online DEDB by using the Online Expert Reorganization utility (also referred to as OER) of FPO.

Topics:

- [“Functions of OER” on page 779](#)
- [“Input and output for OER” on page 782](#)
- [“Running OER” on page 784](#)
- [“Region size requirements for OER” on page 785](#)
- [“DD statements for OER” on page 785](#)
- [“Option control statement for OER” on page 795](#)
- [“Utility control statements for OER” on page 796](#)
- [“Output for OER” on page 803](#)
- [“Examples for OER” on page 817](#)

Functions of OER

OER reorganizes a set of UOWs in a DEDB area on the basis of a ranked set of UOWs.

OER includes the UOW Evaluation phase where the reorganization free space score (RFS score) for a set of UOWs of the DEDB area is determined. In the UOW Ranking phase that follows the UOW Evaluation phase, OER ranks the set of UOWs according to the RFS score. Finally, the highly ranked UOWs are reorganized up to the user-specified number of UOWs.

OER reduces the possibility of having too many or too few UOWs reorganized by ranking the set of UOWs in the DEDB area, and allows the user to select how the subset of the ranked set of UOWs is chosen. Consequently, the user can limit the size of the subset of UOWs that will be reorganized before the reorganization is done.

OER can run OPC after the subset of UOWs have been reorganized and can produce reports that contain the space statistics of each UOW that has been reorganized.

Reorganization of a ranked set of UOWs

OER goes through the phases to reorganize a ranked set of UOWs. These phases include the statistics gathering phase, the UOW Evaluation phase, the UOW Ranking phases, and the UOW Reorganization phase.

Statistics gathering phase

In the statistics gathering phase, OER runs Area Sensor to gather statistics for a set of UOWs in the DEDB area. The statistics gathered is used in the subsequent UOW Evaluation phase.

UOW Evaluation phase

In the UOW Evaluation phase, each UOW is evaluated by the RBASEFS and RDOVFFS criteria. These criteria measure the free space percentage in the CIs where the overflow data might be relocated by reorganization. These criteria are specified by the RBASEFS and RDOVFFS parameters respectively:

- The RBASEFS parameter specifies a percentage of free space in a RAP CI. If the UOW contains at least one RAP CI that contains free space that is greater than the percentage that is specified, and the RAP CI also uses a DOVF CI or an IOVF CI, the UOW can be selected as a candidate for the reorganization (See

Figure 337 on page 780). RBASEFS=100 has a special meaning; if RBASEFS=100, this parameter is not used when selecting the candidate UOWs.

- The RDOVFFS parameter specifies a percentage of free space in the DOVF section in a UOW. If the free space percentage in the DOVF section in a UOW is greater than the percentage that is specified, and the UOW also uses IOVF CIs, the UOW can be selected as a candidate for the reorganization (See Figure 338 on page 780). RDOVFFS=100 has a special meaning; if RDOVFFS=100, this parameter is not used when selecting the candidate UOWs.

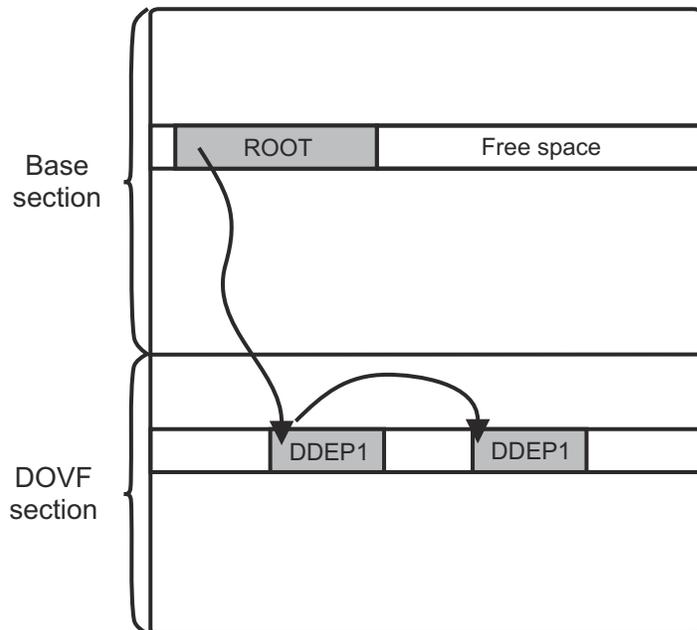


Figure 337. A UOW that satisfies the RBASEFS condition

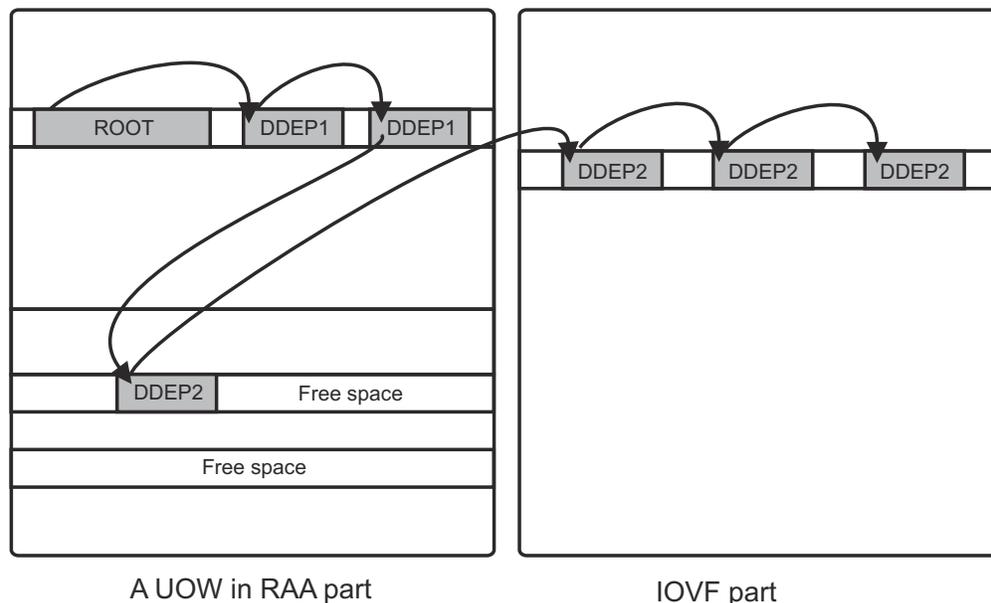


Figure 338. A UOW that satisfies the RDOVFFS condition

In the UOW Evaluation phase, an integer called Reorganization Free Space Score (RFS Score) is calculated for each of the candidate UOWs that are selected by the RBASEFS and RDOVFFS criteria. The RFS score is a combined score that is based on the RBASEFS and RDOVFFS criteria.

UOW Ranking phase

In the UOW Ranking phase, OER ranks, according to the RFS score, the set of candidate UOWs that is selected by the RBASEFS and RDOVFFS criteria.

UOW Reorganization phase

In the UOW Reorganization phase, OER reorganizes the ranked set of UOWs.

- To specify the maximum number of UOWs to reorganize, use the REORGLIM keyword. This number is referred to as *reorganization limit*. If you specify the reorganization limit, highly ranked UOWs are reorganized up to the reorganization limit. The reorganization limit can be supplied as a number (quantity) of UOWs or as a percentage of the UOWs defined for the DEDB area.
- To enable segment shunting, use the INDD DD statement. Segment shunting is the capability to reorganize specified segments directly into dependent overflow (DOVF) or independent overflow (IOVF), bypassing copying these segments into the root anchor point control intervals (RAP CI) even though space might currently exist in the RAP CI. Segment shunting allows you to potentially reorganize the UOWs and retain space in the RAP CI for new inserts.

OER internally calls the IMS DEDB High-Speed Direct Reorganization utility (also referred to as HSR) to reorganize the selected subset of UOWs and to enable segment shunting.

Related reference

RBASEFS keyword

The RBASEFS keyword specifies a criterion to select UOWs to reorganize.

RDOVFFS keyword

The RDOVFFS keyword specifies a criterion to select UOWs to reorganize.

REORGLIM keyword

The REORGLIM keyword specifies the reorganization limit for the amount of UOWs that can be reorganized in the job.

The number of UOWs to be reorganized

The number of UOWs to be reorganized is determined by the combination of the REORG control statement parameters.

In this topic, it is assumed that YES is specified in the first parameter of the RFSSCAN keyword of the REORG control statement. This enables the UOW Evaluation and UOW Ranking functions of OER.

The number of UOWs to be reorganized is determined by the combination of the following parameters of the REORG control statement:

- RBASEFS
- RDOVFFS
- REORGLIM

Both the RBASEFS and the RDOVFFS parameters can be used to evaluate each UOW in the area. If at least one of RBASEFS and RDOVFFS is less than 100, the set of UOWs are classified into two subsets:

- The group of UOWs that satisfy the RBASEFS condition or the RDOVFFS condition (SUBSET-1)
- The group of UOWs that satisfy neither the RBASEFS condition nor the RDOVFFS condition

UOWs in SUBSET-1 are selected as initial candidates for the reorganization. If value 100 is specified for both the RBASEFS and the RDOVFFS parameters, all UOWs in the area are classified as SUBSET-1, that is, all UOWs are selected as candidates for the reorganization.

The REORGLIM parameter is used to limit the number of UOWs that are actually reorganized in the OER job. The parameter works on the ranked UOWs in SUBSET-1.

If the maximum number of UOWs determined by the REORGLIM parameter is larger than the number of UOWs in SUBSET-1, all UOWs in SUBSET-1 are reorganized, but no other UOWs are reorganized.

Reports for reorganization result

When a reorganization job is completed, a report that contains information about the job is produced.

This report contains the following information:

- A brief summary of the definition of DEDB area processed
- The OER processing options used
- The result of UOW Evaluation
- The result of the reorganization
- The statistics of space usage for each UOW that was selected for the reorganization

Related reference

[FPXRPTS DD data set](#)

The Reorganization Processing Summary report, the UOW Space Statistics report, and the Process Summary report are produced in the data set or written to the output stream that is determined by the RFILE keyword of the REORG control statement.

Pointer check during online reorganization

OER can optionally invoke OPC after the selected UOWs have been reorganized.

OPC is invoked when PTRCHK=YES is specified on the REORG control statement.

Almost all options for the stand-alone OPC can be specified.

Related reference

[PTRCHK control statement](#)

The PTRCHK statement specifies the options for OPC that runs after all selected UOWs have been reorganized.

[REORG control statement](#)

The REORG statement specifies information that controls how the OER run is to be processed.

Input and output for OER

OER includes the Area Sensor component, the UOW Evaluation component, the UOW Ranking component, and the UOW Reorganization component.

OER can optionally run OPC if the selected UOWs have been reorganized.

The following figure shows input and output for OER and its general data flow.

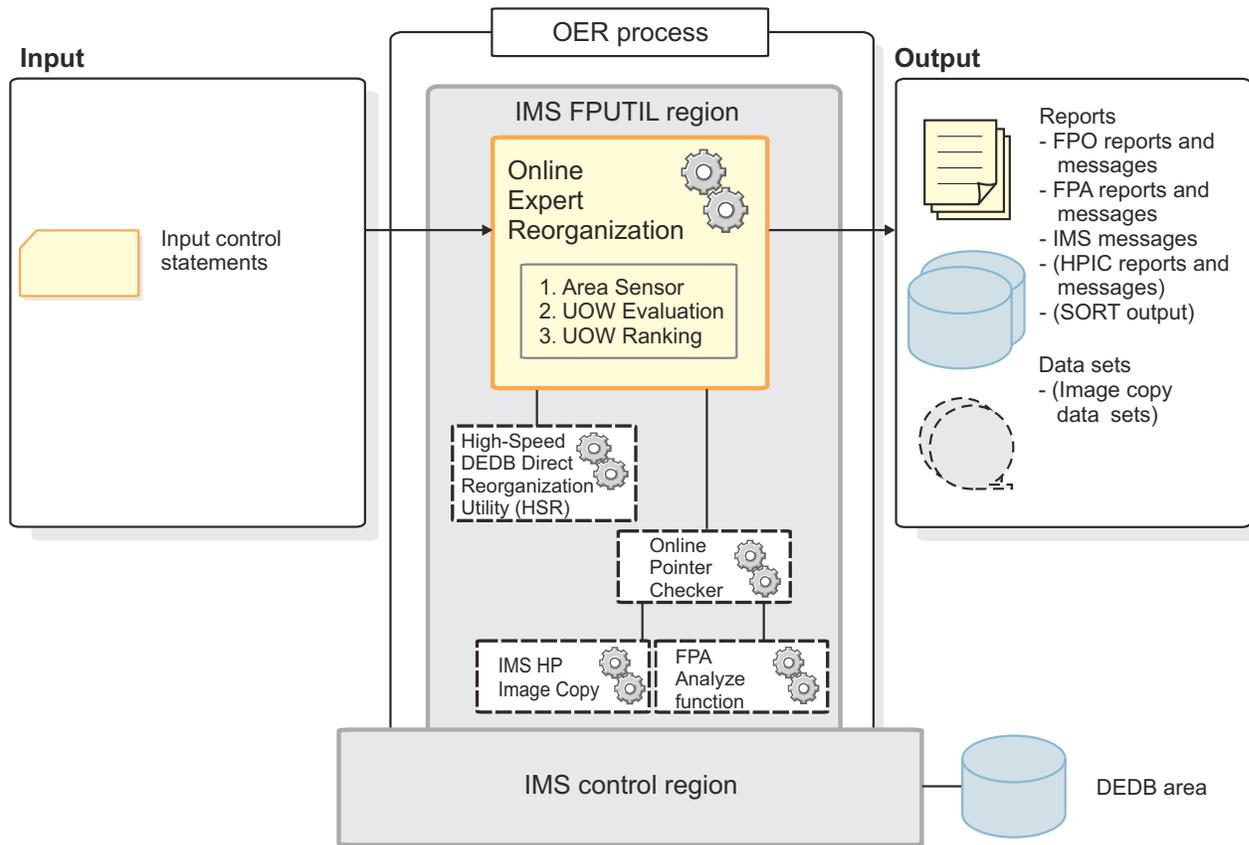


Figure 339. Input and output for OER

The UOW Evaluation component of OER selects a set of UOWs within a DEDB area on the basis of the statistics that is gathered by Area Sensor and on the RBASEFS and RDOVFFS criteria. Then, the RFS score is calculated for each UOW in the set of UOWs.

When the set of UOWs is evaluated, the UOW Ranking component ranks the set of UOWs according to the RFS score.

Then, the UOW Reorganization component selects a subset of the ranked set of UOWs on the basis of the rank order for the ranked set of UOWs. The subset is selected so that higher ranked UOWs are included up to the user-specified number. Finally, the UOW Reorganization component reorganizes the subset of UOWs.

If a pointer check is requested, OER can then invoke OPC to do the pointer checking or to produce statistics report of the UOWs in the area. OPC can generate records that will be used by the batch DEDB Pointer Checker to produce detailed area analysis reports. OPC can also take a concurrent image copy of the area. Instead of running OPC, the Area Sensor can be run to gather statistics of UOWs that were reorganized. The statistics can also be gathered as part of the OPC process.

The statistics gathered by the Area Sensor or OPC can be presented as a report if it is requested. The statistics are reported only for those UOWs that have been reorganized.

Running OER

OER is run by using standard JCL and control statements. You need to specify an EXEC statement and DD statements that define the input and output data sets in your JCL.

About this task

Cataloged procedures FPXAPC1, FPXAPC2, and FPXAPC3 are provided for OPC and OER. These procedures are extensions of the IMS supplied cataloged procedure FPUTIL. You can use these procedures to run the batch DEDB Pointer Checker after you run OPC or OER.

The following cataloged procedures are stored in the SHFPSAMP sample library, and can be used as follows:

- FPXAPC1 to run OPC or OER with TYPRUN=RPT
- FPXAPC2 to run OPC or OER with TYPRUN=FS
- FPXAPC3 to run OPC or OER with TYPRUN=PTRALL

Tip: If you want to change the return codes of OER, you need to create the return code user exit routines.

Procedure

To run OER, complete the following steps:

1. Specify the EXEC statement in one of the following formats:

```
//name EXEC procname
```

where *procname* is the IMS-supplied IFP procedure FPUTIL or any other IFP utility supporting procedure.

```
//name EXEC PGM=DFSRRRC00,PARM='IFP,dbdname,DBF#FPU0'
```

where *dbdname* is the DBDNAME associated with the DEDB.

Related reading:

- For information about the FPUTIL utility, see *IMS System Definition* for the version of IMS that you are using.
- For information about the DEDB online utility commands, see *IMS Database Utilities* for the version of IMS that you are using.

2. Specify the DD statements to define input data sets and output data sets.
3. Run the JCL.

Related reference

[Return code user exit routine \(FPXRCEX0\)](#)

IMS Fast Path Online Tools supports a return code user exit routine that can be used to change the return codes of OPC, ODE, or OER.

Examples for OER

There are many ways to run the OER utility. The examples provided in the following topics show some of the typical ways to run the OER utility.

[Tips and best practices for the FPO Online Expert Reorganization utility](#)

The FPO Online Expert Reorganization utility (OER) supports several keywords that improve the performance of OER jobs and that further simplify management of DEDBs.

Region size requirements for OER

The region size for OER depends on the size of a UOW and the keyword specifications.

Because OER operates in a virtual storage region as an IMS IFP utility, the region size required to run OER is larger than the following values:

- If the PTRCHK=YES keyword is specified, OER requires the storage calculated by the following formulas:

```
Buffer sets:          UOW1 * CISZ * 5
Work buffer:         (UOW1 + IOVFP00L) * CISZ * 10
IOVF private buffer: IOVFP00L * CISZ * 5
```

In calculating the buffer sets, if the value of UOW1 is smaller than 120, 120 is used instead of UOW1. For the work buffer, the UOW1 value is used even if the value is smaller than 120.

- If the RFSSCAN=(YES,*) keyword or the RFSSCAN=(*,YES) keyword is specified, OER requires the storage calculated by the following formula:

```
Buffer set:  UOW1 * CISZ
```

If the value of UOW1 is smaller than 120, 120 is used instead of UOW1.

- If the return code user exit routine is specified, the required region size of OER is also affected by the MAXERROR keyword as follows:

```
Message and return code table: MAXERROR * 8
```

- If the PTRCHK=YES and FULLSTEP=YES keywords are specified, OER internally calls the FPA Analyze function to perform a FULL mode pointer checking and generate complete analysis reports. For the additional region size required by the FPA Analyze function, see [“Region size requirements for the Analyze function”](#) on page 200.
- To generate the RBA of Used IOVF CI report (generated when PTRCHK=YES, FULLSTEP=YES, TYPRUN=RPT or FS, and IOVFCI_RBA_REPORT=YES are specified), OER requires the storage calculated by the following formulas:

```
IOVF space map buffer: 4 * 119 * the number of space map CIs
Internal work buffer:  8 * maximum UOW + (8+64) * maximum number of IOVF CI
```

The MAXBUFFS keyword specifies the number of buffer sets that are used in the reorganization. The meaning of MAXBUFFS is the same as that of the BUFNO command for the High-Speed DEDB Direct Reorganization utility (DBFUHDR0).

DD statements for OER

DD statements for OER determine the input and output data sets and specify how to run OER.

Optional output data sets used by OPC are determined by the TYPRUN= parameter of the PTRCHK control statement. For details, see [“Data sets required for each operating mode”](#) on page 680.

The following table summarizes the DD statements for OER.

DDNAME	Use	Format	Required or optional
JOBLIB or STEPLIB	Input	PDS	Required
FPXOPTS	Input	LRECL=80	Optional
FPXIN	Input	LRECL=80	Required

Table 100. DD statements for OER (continued)

DDNAME	Use	Format	Required or optional
FPXMFILE	Output	LRECL=133	Optional
FPXRPTS	Output	LRECL=133	Optional
FPXSPRT	Output	LRECL=133	Optional
HFPPRINT	Output	LRECL=133	Optional
<i>reportdd</i>	Output	LRECL=133	Optional
SYSPRINT	Output	LRECL=121	Required
DAR12O	Output		Required when PTRCHK=YES and FULLSTEP=NO
DAR13SO	Output		Required when PTRCHK=YES and FULLSTEP=NO
DFSUDUMP (<i>ic1outdd</i> or <i>ic2outdd</i>)	Output	IC1 or IC2	Optional
SORT12	Output	LRECL=80	Required when PTRCHK=YES and FULLSTEP=NO
SORT3CS	Output	LRECL=80	Required when PTRCHK=YES and FULLSTEP=NO
SNAPPIT	Output	LRECL=133	Optional
DAR3CSO	Output		Required when PTRCHK=YES and FULLSTEP=NO
FPXAUOWC	Input	LRECL=80	Required when FPXAHST2 DD is specified
FPXAHST	Output	DISP=MOD	Optional
FPXAHST2	Output	LRECL=133	Required when FPXAUOWC DD is specified
FPXARERG	Output	LRECL=80	Optional
TAR <i>nnnn</i>	Output	PS	Required when both of the following conditions are met: <ul style="list-style-type: none"> • TARECGEN=YES or TARECGEN=ONLY • TARSDNMASK keyword is not specified
HFPABEND	Output	LRECL=125	Optional
IMSDALIB	Input	PDS	Optional
RECON <i>n</i>	Input/ Output		Required when all of the following conditions are met: <ul style="list-style-type: none"> • IC=YES and ICNOTIFY=YES are specified, or ITKBSRVR=<i>servername</i> and DBRC=YES are specified. • DFSMDA member for RECON<i>n</i> data sets are not used.
SYSOUT	Output	LRECL=121	Optional
<i>i001WKnn</i>	Input/ Output		Optional
<i>i001SMG</i>	Output	LRECL=133	Optional
ICEIN	Input	LRECL=80	Optional
ICEPRINT	Output	LRECL=133	Optional

Table 100. DD statements for OER (continued)

DDNAME	Use	Format	Required or optional
ICERPRT	Output	LRECL=133	Optional
<i>dataout1</i>	Output	PS	Optional
<i>dataout2</i>	Output	PS	Optional
INDD	Input	LRECL=80	Optional
D2Xnnnn	Input/ Output		Optional
D3Cnnnn	Input/ Output		Optional
D3SRnnnn	Input/ Output		Optional
D3SInnnn	Input/ Output		Optional
D4Pnnnn	Input/ Output		Optional
D4Snnnn	Input/ Output		Optional
WPRNnnn	Input/ Output		Optional
WSNAP	Input/ Output	LRECL=133	Optional

STEPLIB DD

Defines the library that contains the OER utility.

- The load library of IMS HP Image Copy must be concatenated if you specify the PTRCHK=YES keyword with either of the following specifications:
 - IC=(YES,*comprtn*) option
 - IC=(YES,ICEIN) option
- The IMS Tools Base library (SHKTLOAD) is required if you specify the ADXCFGRP keyword, the ITKBSRVR keyword, or the SENSOR=YES option.
- The library that contains DFSMDA dynamic allocation members for the RECON data sets.

The data set that contains the DBRC SCI registration exit routine (DSPSCIX0) must be concatenated when using the RECON data set in which an IMSplex name is set.

FPXOPTS DD

Defines the input data set that contains the option control statements for the process. FPXOPTS usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte fixed-length records. Block size, if coded, must be a multiple of 80. It can be specified as DUMMY, NULLFILE, or an empty file.

For details of the option control statements, see [“Option control statement for OER”](#) on page 795.

FPXIN DD

Defines control statements for OER. FPXIN describes the area to be reorganized, and it can contain optional user requests that define the operating mode. FPXIN usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte fixed-length records. Block size, if coded, must be a multiple of 80.

For details of control statements, see [“Utility control statements for OER”](#) on page 796.

FPXMFIL DD

Defines the default message output. The output can go to a data set or to a printer, routed through the output stream. If the DD statement is specified explicitly or implicitly by the MFILE keyword of the REORG control statement, and the DD statement for the specified DD name is not coded in the JCL stream, the DD is allocated dynamically to the system output stream with SYSOUT=*. If the DD name is overridden by the MFILE keyword with a different DD name, the DD statement for the specified DD name must be coded in the JCL stream.

FPXRPTS DD

Defines the default report output. The output can go to a data set or to a printer, routed through the output stream. If the DD statement is specified explicitly or implicitly by the RFILE keyword of the REORG control statement and the DD statement for the specified DD name is not coded in the JCL stream, the DD is allocated dynamically to the system output stream with SYSOUT=*.

FPXSPRT DD

Defines the report output data set when SENSOR=YES is specified. This output data set contains the Sensor Data Statistics report that can be used to verify which sensor data is stored in the Sensor Data repository of IMS Tools KB. The following DCB attributes of the file are set by FPO:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set. You can specify NULLFILE or DUMMY for this data set.

The FPXSPRT data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//FPXSPRT DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPO dynamically allocates the data set by using SYSOUT=*.

HFPPRINT DD

Defines the message output data set when FULLSTEP=YES is specified. This output data set contains the HFP messages, which are the processing messages issued by FPA. The following DCB attributes of the file are set by FPA:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set. Do not specify DUMMY for this data set.

The HFPPRINT data set contains 133-byte, fixed-length records. It is recommended that you code your DD statement as follows:

```
//HFPPRINT DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPA dynamically allocates the data set by using SYSOUT=*.

reportdd DD

Defines the output for the following reports:

- DEDB Area Analysis reports
- UOW report
- UOW Exception report

- Largest Database Records report
- Consolidated Largest Database Records report
- Segment Length Distribution report
- DB Record Length Distribution report
- RBA of Used IOVF CI report

The output can go to a data set or to a printer, routed through the output stream. FPXRPTS is the default DD name, however, the DD name can be overridden by the REPORTDD keyword specification of the REPORT control statement.

This statement is used only when you specify the TYPRUN=RPT and the FULLSTEP=YES keywords, or the TYPRUN=FS and the FULLSTEP=YES keywords.

reportdd DD is similar to the *reportdd* DD of the Analyze function of FPA.

SYSPRINT DD

Defines the system message output.

Warning: You cannot allocate the OER SYSPRINT DD output stream to a data set because SYSPRINT DD is also used by the IMS system. If you specify SYSPRINT DD as a data set, some output might be overridden or unpredictable open or close errors might occur. Therefore, you must specify SYSPRINT DD as SYSOUT.

DAR120 DD

Defines the output data set containing the free space and area information records. The data set can reside on either a direct-access device or a tape. When FULLSTEP=YES is specified, this DD statement is not used.

Space requirements vary depending on the size of the areas and the number of free space elements. (Approximation: *number of CIs in area* x 1.5 x 20 bytes.) Do not specify DISP=MOD for this DD statement.

When DAR120 is so big that the allocated space is not adequate, a B37 abend occurs.

DAR13SO DD

Defines the output data set containing the segment information records for all segments that reside in the root addressable part of the database. The data set can reside on either a direct-access device or a tape. When FULLSTEP=YES is specified, this DD statement is not used.

Space requirements vary depending on the size of the areas and the number of segments in the areas. (Approximation: *number of segments* x 40 bytes.) Do not specify DISP=MOD for this DD statement.

When DAR13SO is so big that the allocated space is not adequate, a B37 abend occurs.

DFSUDUMP DD

Defines the default output image copy data set. DFSUDUMP is used by OPC as the default primary image copy data set until it is (optionally) overridden by the IC1OUTDD keyword of the ANALYZE control statement. The output can reside on a tape or on a direct-access device. If the DD statement is overridden by the IC1OUTDD keyword with a different DD name, the DD statement for the specified DD name must be coded in the JCL stream.

Do not specify DISP=MOD for this DD statement.

The secondary image copy data set is (optionally) specified by the IC2OUTDD keyword of the ANALYZE control statement, the specified DD name must be coded in the JCL stream.

SORT12 DD

Defines the output data set for the SORT control statements for the SORT12 step. The data set must reside on a direct-access device. When FULLSTEP=YES is specified, this DD statement is not used.

Space requirements are very small (1 track suffices). Do not specify DISP=MOD for this DD statement.

DCB information is hard-coded in OPC. Do *not* code DCB information in your JCL.

SORT3CS DD

Defines the output data set for the SORT control statements for the sort before pointer resolution. The data set must reside on a direct-access device. When FULLSTEP=YES is specified, this DD statement is not used.

Space requirements are very small (1 track suffices). Do not specify DISP=MOD for this DD statement.

DCB information is hard-coded in OPC. Do *not* code DCB information in your JCL.

SNAPPIT DD

Defines the output for the CI SNAP dump produced by OPC.

The CI SNAP dump option is ignored if the DD statement is not provided.

The output can go to a data set or to a printer, routed through the output stream.

DAR3CSO DD

Defines the output data set containing all pointer information records and the segment information records for all segments that reside in IOVF and SDEP. The data set can reside on either a direct-access device or a tape. When FULLSTEP=YES is specified, this DD statement is not used.

Space requirements vary depending on the size of the areas and the number of segments in the areas. (Approximation: *number of segments* x 1.5 x 26 bytes.) Do not specify DISP=MOD for this DD statement.

When DAR3CSO is so big that the allocated space is not adequate, a B37 abend occurs.

FPXAUOWC DD

Each record in this data set specifies a range of UOW numbers of an area, which defines a UOW group of the area. The information about the UOW group is recorded in the FPXAHST2 data set or in a sensor data record set in the order that is specified in the FPXAUOWC data set, even if the UOW ranges overlap or are in reverse order.

The data set can reside in the JCL input stream. The data set can also be defined as a sequential data set or as a member of a partitioned data set. If a data set is specified, the data set must have the following attributes:

- DSORG=PS
- RECFM=FB
- LRECL=80
- BLKSIZE=*a multiple of 80*

If the data set is empty, all UOWs of the area are regarded as the target UOW range.

When you specify the FPXAHST2 DD statement, you must also specify the FPXAUOWC DD statement. If an FPXAUOWC DD statement is not specified, the FPXAHST2 DD statement is ignored.

When SENSOR_UOW=YES is specified, the data elements that will be stored differ by the FPXAUOWC DD specification as follows:

- If you specify the FPXAUOWC DD statement, data elements are collected at the UOW group level.
- If you do not specify the FPXAUOWC DD statement, data elements are collected at the UOW level.

The FPXAUOWC DD is similar to the HFPXAUOWC DD of the Analyze function of FPA.

For information about defining a UOW group, see [“UOW group definition record layout” on page 1191](#).

FPXAHST DD

Defines an output data set that contains historical records. The analysis report generation process adds one record for each DEDB area that it processes. The record contains the key space utilization and performance information for that area. This statement is used only when you specify the TYPRUN=RPT keyword and the FULLSTEP=YES keyword.

If this feature is used, the FPXAHST data set must be preallocated with the following attributes:

- DSORG=PS
- RECFM=FB
- LRECL=100

- `BLKSIZE=user-specified`

The DD statement in the JOB stream is specified as:

```
//FPXAHST DD DSN=HPFP.HISTORY,DISP=(MOD,KEEP,KEEP)
```

The DD name can be (optionally) overridden by the HISTORYDD keyword with a different DD name of the ANALYZE control statement.

The FPXAHST DD is similar to the HFPAHST DD of the Analyze function of FPA.

FPXAHST2 DD

Defines an output data set that contains historical records. The analysis report generation process adds one record for each UOW group that is specified by FPXAUOWC DD. The record contains the key space utilization and performance information for the area, which is based on the information obtained from the specified UOW range. This statement is used only when you specify the FPXAUOWC DD statement, the TYPRUN=RPT keyword, and the FULLSTEP=YES keyword. If FPXAUOWC DD statement is not specified, this statement is ignored.

If this feature is used, the FPXAHST2 data set must be preallocated with the following attributes:

- `DSORG=PS`
- `RECFM=FB`
- `LRECL=200`
- `BLKSIZE=user-specified`

The DD statement in the JOB stream is specified as:

```
//FPXAHST2 DD DSN=HPFP.HISTORY2,DISP=(MOD,KEEP,KEEP)
```

The FPXAHST2 DD is similar to the HFPAHST2 DD of the Analyze function of FPA.

FPXARERG DD

Defines an output data set that contains the reorganization control statements for the High-Speed DEDB Direct Reorganization utility (DBFUHDRO).

This statement is used only when you specify the TYPRUN=RPT and the FULLSTEP=YES keywords, or the TYPRUN=FS and the FULLSTEP=YES keywords.

The following DCB attributes of the file are set by the Analyze process:

- `RECFM=PS`
- `LRECL=80`
- `BLKSIZE=800`

Do not code the DCB information in your JCL statement. Do not specify `DISP=MOD` for this DD statement.

The FPXARERG DD is similar to the HFPARERG DD of the Analyze function of FPA.

TARnnnnn DD

Defines the tuning aid record data set. A `TARnnnnn` DD statement specifies the tuning aid record data set for one area.

When `TARECGEN=YES` is specified, `TARnnnnn` DD statements define the output data sets for generating tuning aid records. If this statement is not specified for an area, the tuning aid record for that area is not generated.

A `TARnnnnn` DD statement is for areas in the range of 1 - 2048. The value of `nnnnn` must be right-aligned, with leading zeros if necessary. When you code multiple `TARnnnnn` DD statements, do not specify the same data set more than once; each `TARnnnnn` DD statement must point to a different data set.

This data set can be allocated dynamically by specifying the `TARDSNMASK` keyword of the ANALYZE statement.

For a description and space requirement of the tuning aid records, see [“Tuning aid records”](#) on page 231.

HFPABEND DD

Defines a snap dump output data set used for diagnostic purposes. This data set might be used when FULLSTEP=YES is specified and the FPO job ends abnormally. Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, FPO dynamically allocates the data set by using SYSOUT=*

IMSDALIB DD

An optional DD statement that defines the library that contains DFSMDA dynamic allocation members for the RECON data sets.

RECON1/RECON2/RECON3 DD

Defines the DBRC RECON data sets. You must specify these DD statements when you specify both IC=YES and ICNOTIFY=YES keywords, or both ITKBSRVR=*servername* and DBRC=YES keywords. RECON*n* data sets must be the same data sets that are used in the IMS control region.

When you specify IC=YES and ICNOTIFY=YES, OER updates RECON*n* data sets because OER registers the image copy data set to DBRC. When you specify ITKBSRVR=*servername* and DBRC=YES, OER retrieves the RECON ID from RECON data sets.

If these statements are omitted when the data sets are necessary and if DFSMDA members are available, the utility uses the DFSMDA members and dynamically allocates the data sets.

SYSOUT DD

Defines the output data set that is used by DFSORT for message functions when the FULLSTEP=YES keyword is specified.

Note on dynamic allocation:

When DFSORT is needed and this statement is not specified in the JCL statement, OER dynamically allocates the data set by using SYSOUT=*

i001WK*nn* DD

An optional DD statement. These are intermediate storage data sets that are used by DFSORT. For more information about coding the SORTWK*nn* DD statements, see the *DFSORT Application Programming Guide*. The adequate size of storage data sets is 1.5 to 2.5 times the input file size. This statement is used only when you specify the FULLSTEP=YES keyword.

Allocating twice the space used by the SORTIN data set is usually adequate for each work data set.

Do not specify DISP=MOD for these DD statements.

In the analyze step, the size of each input file is as follows:

i:

A: Always used

30 x the number of pointers in the area

D: If SDEP=YES is specified

30 x the number of SDEP segments in the area

G: If SNAPPIT DD is specified

112 x the number of pointers in which pointer errors are detected

J: If TYPRUN=RPT is specified

X x (the number of non-SDEP segments in the areas + the number of free space elements in the areas + the number of references to other CIs)

Where X is:

- When LARGEREC_REPORT=NO is specified, X is calculated as follows:

34 + (the maximum segment level x 4)

However, if X is smaller than 48, 48 is used for X .

- When LARGEREC_REPORT=YES is specified, X is calculated as follows:

$$34 + (\text{the maximum segment level} \times 4) + \text{root key length}$$

However, if X is smaller than 48, 48 is used for X .

J: If TYPRUN=FS is specified

$23 \times (\text{the number of free space elements in the area} + \text{the number of references to other CIs})$

K: If TYPRUN=RPT is specified

$X \times \text{the number of segments in the area}$

Where X is:

- When TYPRUN=MODEL, LARGEREC=Y, or TARECGEN=Y is specified, X is calculated as follows:

$$34 + (\text{the maximum segment level} \times 4) + \text{root key length}$$

- In other cases, X is calculated as follows:

$$34 + (\text{the maximum segment level} \times 4)$$

M: If TYPRUN=RPT or TYPRUN=PTRALL is specified

$X \times \text{the number of IOVF segments in the area}$

Where X is:

- When TYPRUN=MODEL, LARGEREC=Y, or TARECGEN=Y is specified, X is calculated as follows:

$$34 + \text{root key length}$$

- In other cases, X is 34.

Q: Always used

$34 \times \text{the number of segments in the area}$

Note on dynamic allocation:

If this statement is not specified in the JCL statement when these data sets are necessary, FPO allocates them dynamically.

i001SMMSG DD

This DD statement is an optional DD statement. However, this DD statement is required when DFSORT is used. This statement defines an output data set for storing DFSORT messages. For the condition for running DFSORT and the naming rules for the DD names, see the description of i001WKnn DD.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, OPC dynamically allocates the data set by using SYSOUT=*

ICEIN DD

This DD statement is an optional DD statement. This statement specifies the input data set that contains the control statements for IMS HP Image Copy. The data set can reside on a tape, a direct-access volume, or it can be routed through the input stream (DD * or DD DATA). This data set is used only when you specify PTRCHK=YES and IC=(YES,ICEIN). If the DD statement is not coded and if PTRCHK=YES and IC=(YES,ICEIN) are specified, the default control statements are used. For details about the ICEIN control statements, see the *IMS High Performance Image Copy User's Guide*.

ICEPRINT DD

This DD statement is an optional DD statement. This statement specifies the output data set that contains the reports that are produced by IMS HP Image Copy. The data set can reside on a tape, a direct-access volume, or a printer, or it can be routed through the output stream (SYSOUT). This data set is used only when you specify PTRCHK=YES and IC=(YES,ICEIN). If the DD statement is not coded and PTRCHK=YES and IC=(YES,ICEIN) are specified, a SYSOUT stream is dynamically allocated for the DD.

ICERPRT DD

This DD statement is an optional DD statement. This statement specifies the output data set that contains the Image Copy Dump Process report, which is produced by IMS HP Image Copy. It can reside on a tape, a direct-access volume, or a printer, or it can be routed through the output stream (SYSOUT). This data set is used only when you specify PTRCHK=YES and IC=(YES,ICEIN).

dataout1 DD

This DD statement is an optional DD statement. This statement defines the primary image copy data set. One DD statement is required for each data set that is to be dumped. The DD name can be any 1- to 8-character string. The data set specified by the DD statement must be on either a direct-access device or a tape device. This data set is used only when you specify PTRCHK=YES and IC=(YES,ICEIN).

For details of the format and the characteristics of the image copy data set, see the *IMS High Performance Image Copy User's Guide*. You can request IMS HP Image Copy to allocate the image copy data set dynamically by omitting this DD statement and specifying the allocation parameters that are associated with the data set in the ICEIN data set.

dataout2 DD

This DD statement is an optional DD statement. This statement defines the secondary image copy data set. This statement is required only if the associated ICEIN control statement requests the secondary image copy. The data set specified by the DD statement must be on either a direct-access device or a tape device. This data set is used only when you specify PTRCHK=YES and IC=(YES,ICEIN).

For details of the format and the characteristics of the image copy data set, see the *IMS High Performance Image Copy User's Guide*.

INDD DD

Defines the segment shunting input. You can specify which segments to shunt using this data set. This data set is similar to the INDD data set for the IMS DEDB High-Speed Direct Reorganization utility (HSR).

This data set must be a fixed block with a logical record length (LRECL) of 80, a block length of 80, and one input card per record. You can also specify NULLFILE or DUMMY, or an empty data set. The input card must be left-aligned or the card is ignored. This data set is required only for those interested in the segment shunting function.

The only segments that can be shunted are direct dependent segments. Root segments and SDEPs cannot be shunted.

The INDD data set consists of three input card types:

ERRACTN=TERM

An optional input card type. The card specifies that if any cards in the INDD data set contain invalid data, the job ends with return code 8 and reorganization will not occur. If you include this card, it must be the first card in the INDD data set.

TERM is the default value for this card. Even if ERRACTN=TERM is omitted or a value that is supported by HSR is coded (ERRACTN=CONT or ERRACTN=EXIT), the OER utility always applies ERRACTN=TERM.

AREA=areaname

Starts the input stream for a single area. The input stream for this area terminates at the next AREA card, or at the end of the file. All of the following cards in the input stream are considered segments to shunt for the associated AREA card.

You can specify multiple AREA cards. The areas you specify with INDD AREA cards must also be specified with the AREA keyword of FPXIN DD.

segmentname

Contains the segment names associated with the prior AREA card.

D2Xnnnn DD
D3Cnnnn DD
D3SRnnnn DD
D3SInnnn DD
D4Pnnnn DD
D4Snnnn DD
WPRNnnn DD

Define the intermediate work data sets. FPO uses up to three sets of intermediate work data sets, one set for processing one area. For each set, the same *nnnn* value is used. *nnnn* must be right-aligned with leading zeros. These statements are used only when you specify the FULLSTEP=YES keyword.

- D2Xnnnn data sets contain the free space and area information records.
- D3Cnnnn data sets contain all non-SDEP pointer information records.
- D3SRnnnn data sets contain the segment information records for all segments that reside in the RAP part of the database.
- D3SInnnn data sets contain the segment information records for all segments that reside in the IOVF part of the database.
- D4Pnnnn data sets contain all SDEP pointer information records.
- D4Snnnn data sets contain the segment information records for all segments that reside in the SDEP part of the database.
- WPRNnnn data sets contain analysis reports.

Do not specify DISP=MOD for these DD statements. DCB information is hard-coded in OPC. Do not code DCB information in your JCL.

Note on dynamic allocation:

If these statements are not specified in the JCL statements when these data sets are necessary, FPO allocates them dynamically.

WSNAP DD

This DD statement is an optional DD statement. It defines the intermediate work data set for snap dumps. This statement is used only when you specify the FULLSTEP=YES keyword.

Do not specify DISP=MOD for this DD statement. DCB information is hard-coded in OPC. Do not code DCB information in your JCL.

Note on dynamic allocation:

If this statement is not specified in the JCL statements when this data set is necessary, FPO allocates it dynamically.

Option control statement for OER

The option control statement defines the user description of the OVERRIDE control statement, which specifies information for setting temporary override values of product message defaults.

The option control statements must start with a line that contains the OPTIONS command. The control statement that can be specified is:

OVERRIDE control statement

The OVERRIDE control statement specifies information for setting temporary override values of product message defaults. This control statement is optional. The OVERRIDE control statement can be specified in FPXOPTS DD.

OVERRIDE statement keywords

The keywords associated with the OVERRIDE statement are not available for site default options. For the format of the OVERRIDE statement, see [“OVERRIDE control statement”](#) on page 850.

For the OVERRIDE statement, you can specify the keywords that are listed in the following table.

Table 101. *OVERRIDE* statement keywords

Keyword	Required or optional	Description	Topic
MSG_LIM	Optional	Specifies the maximum number of repetitive messages that are to be issued before suppression.	“MSG_LIM keyword” on page 897
MSG_NUM	Required	Specifies the message number that is to be suppressed.	“MSG_NUM keyword” on page 898

Related reference

[Keyword reference for FPO](#)

These topics explain the control statements and keywords of FPO and their syntax rules.

[Syntax rules for FPXOPTS](#)

FPXOPTS control statements and their associated keywords can be coded in free format in columns 1 through 72, subject to certain syntax rules.

Utility control statements for OER

The utility control statements for OER define the user description of the GLOBAL control statement, the REORG control statement, the PTRCHK control statement, the REPORT control statement, and the THRESHOLD control statement, all of which control the OER process.

All control statements that can be specified to run an OER job are explained in this topic. The control statements that can be specified are:

GLOBAL control statement

The GLOBAL control statement defines the job step keyword values. If specified, GLOBAL must be the first command in the input stream. The GLOBAL control statement can be specified only in FPXIN DD.

REORG control statement

The REORG control statement specifies information that control how the OER run is to be processed.

PTRCHK control statement

The PTRCHK control statement specifies the options for OPC that runs after all selected UOWs have been reorganized.

REPORT control statement

The REPORT control statement defines the parameter value that is related to the analysis reports.

THRESHOLD control statement

The THRESHOLD control statement defines the threshold parameter values, which cause warning messages to be generated when the actual values do not meet the specified criteria.

OER allows you to specify site default options.

Subsections:

- [“GLOBAL statement keywords” on page 796](#)
- [“REORG statement keywords” on page 797](#)
- [“PTRCHK statement keywords” on page 798](#)
- [“REPORT statement keywords” on page 800](#)
- [“THRESHOLD statement keyword” on page 801](#)

GLOBAL statement keywords

The GLOBAL control statement specifies the job step keyword values. This control statement is optional.

The keywords associated with the GLOBAL statement are available for site default options.

For the GLOBAL statement, you can specify the keywords that are listed in the following table. For the format of the GLOBAL statement, see [“GLOBAL control statement” on page 847](#).

Table 102. GLOBAL statement keywords

Keyword	Required or optional	Description	Topic
ADXCFCGRP	Optional	Specifies the name of the Autonomics Director XCF group.	“ADXCFCGRP keyword” on page 861
DBRC	Optional	Specifies whether to store reports that are associated with a RECON in the IMS Tools KB repository.	“DBRC keyword (for GLOBAL)” on page 868
ITKBDATA	Optional	Specifies the type of additional data to store in the IMS Tools Base IMS Tools Knowledge Base repository.	“ITKBDATA keyword” on page 890
ITKBLOAD	Optional	Specifies the name of the load module library for IMS Tools KB.	“ITKBLOAD keyword” on page 890
ITKBSRVR	Optional	Specifies the name of the IMS Tools KB server XCF group.	“ITKBSRVR keyword” on page 891
KEYWORD_DUPCHK	Optional	Specifies to end the job with an error message when a duplicate keyword is found.	KEYWORD_DUPCHK keyword

REORG statement keywords

The REORG control statement specifies information that controls how the OER run is to be processed. This control statement is mandatory.

The keywords associated with the REORG statement are available for site default options except for the AREA keyword.

For the REORG statement, you can specify the keyword that is listed in the following table. For the format of the REORG statement, see [“REORG control statement” on page 854](#).

Table 103. REORG statement keywords

Keyword	Required or optional	Description	Topic
AREA	Required	Specifies the name of the area to be reorganized.	“AREA keyword” on page 864
ERRORACT	Optional	Specifies the action that OER takes when one of the following errors occurs: <ul style="list-style-type: none"> An error that occurs while the Area Sensor is running The number of errors that occur during the UOW reorganization exceeds the value that is specified on the MAXFAILS keyword of the REORG control statement The number of errors that occurred in OPC reaches the value that is specified on the MAXERROR keyword of the PTRCHK control statement 	“ERRORACT keyword” on page 870
FPBP64U	Optional	Specifies whether to enable the Fast Path 64-bit buffer manager for the IMS High-Speed DEDB Direct Reorganization utility (DBFUHDRO).	“FPBP64U keyword” on page 879
MAXBUFFS	Optional	Specifies the number of buffer sets that are used in the reorganization.	“MAXBUFFS keyword” on page 894
MAXFAILS	Optional	Specifies the maximum allowable number of UOWs that fail to be reorganized, before the utility is ends abnormally.	“MAXFAILS keyword” on page 895
MFILE	Optional	Specifies the DD name of the output data set or output stream to which OER messages and OPC messages are to be written.	“MFILE keyword” on page 896
PCHKONLY	Optional	Specifies whether to evaluate only the control statement keywords or also to reorganize a set of UOWs in an online DEDB area.	“PCHKONLY keyword” on page 901
PROC_AFT_ERROR	Optional	Specifies whether to end the job step when an area-specific error is encountered or to continue processing subsequent areas.	PROC_AFT_ERROR keyword
PTRCHK	Optional	Specifies whether to run OPC after the area is reorganized.	“PTRCHK keyword” on page 902

Table 103. REORG statement keywords (continued)

Keyword	Required or optional	Description	Topic
RBASEFS	Optional	Specifies a criterion to select UOWs to reorganize.	“RBASEFS keyword” on page 904
RCEXLOAD	Optional	Specifies whether to call the return code exit routine FPXRCEX0 from the STEPLIB or JOBLIB data set concatenation.	“RCEXLOAD keyword” on page 904
RDOVFFS	Optional	Specifies a criterion to select UOWs to reorganize.	“RDOVFFS keyword” on page 905
REORGLIM	Optional	Specifies the reorganization limit for the amount of UOWs that can be reorganized in the job.	“REORGLIM keyword” on page 906
REORGMSG	Optional	Specifies whether to generate detailed reorganization messages.	“REORGMSG keyword” on page 907
RFILE	Optional	Specifies the DD name of the output data set or output stream to which OER reports are to be written.	“RFILE keyword” on page 908
RFSSCAN	Optional	Specifies whether to gather the statistics before and after reorganization by using Area Sensor.	“RFSSCAN keyword” on page 909
RFSSTAT	Optional	Specifies whether to generate the UOW Space Statistics report before and after reorganization.	“RFSSTAT keyword” on page 909

PTRCHK statement keywords

The PTRCHK control statement specifies the options for OPC that runs after all selected UOWs have been reorganized. The acceptable values and their meanings for the keyword parameter are the same as those for the stand-alone OPC except for the TYPRUN and CHKUOW keywords.

The PTRCHK control statement can be specified in FPXIN even if PTRCHK=YES is not specified in the REORG control statement. But in that case, PTRCHK control statement is ignored.

The keywords associated with the PTRCHK statement are available for site default options.

For the PTRCHK statement, you can specify the keyword that is listed in the following table. For the format of the PTRCHK statement, see [“PTRCHK control statement” on page 850](#).

Table 104. PTRCHK statement keywords

Keyword	Required or optional	Description	Topic
ARDIGIT	Optional	Specifies the digit of the area number field for the UOW group definition records that are defined in the FPXAUOWC file.	“ARDIGIT keyword” on page 863
CHKUOW	Optional	Specifies whether to do the pointer checking for all UOWs in the area or for the UOWs that have been reorganized.	“CHKUOW keyword” on page 866
DBRC	Optional	Specifies that image copies be registered to DBRC. DBRC registration of the area is required in order to DBRC register an image copy. DBRC is alias keyword for ICNOTIFY.	“DBRC keyword (for GLOBAL)” on page 868
DOVFNULPTR_MSGLVL	Optional	Specifies that the utility issues a warning message if the first DOVF CI in any UOW contains a null pointer for the current overflow CI.	DOVFNULPTR_MSGLVL keyword
EXC_RBASEFS	Optional	Specifies a criterion. If a RAP CI contains free space that is greater than the percentage specified, and the RAP CI also uses overflow CIs, a reorganization control statement is generated for the UOW, and the UOW is printed in the UOW Exception report.	“EXC_RBASEFS keyword” on page 872
EXC_RDOVFFS	Optional	Specifies a criterion. If the DOVF part of a UOW contains free space that is greater than the percentage specified, and the UOW also uses IOVF CIs, a reorganization control statement is generated for the UOW and the UOW is printed in the UOW Exception report.	“EXC_RDOVFFS keyword” on page 873

Table 104. PTRCHK statement keywords (continued)

Keyword	Required or optional	Description	Topic
EXC_UDOVFFS	Optional	Specifies a criterion. If the DOVF part of a UOW contains free space that is less than the percentage specified, the UOW is printed in the UOW Exception report.	“EXC_UDOVFFS keyword” on page 873
FULLSTEP	Optional	Specifies whether OPC continues to process the pointer verification data collected from an online DEDB area.	“FULLSTEP keyword” on page 882
HISTORYDD	Optional	Specifies the ddname of the DD statement for the output History file data sets that are specified in the JCL.	“HISTORYDD keyword” on page 882
IC	Optional	Specifies that an IMS standard image copy be taken while scanning the database area.	“IC keyword” on page 883
ICNOTIFY	Optional	Specifies that image copies be registered to DBRC.	“ICNOTIFY keyword” on page 885
IC1OUTDD	Optional	Specifies the DD name of the output file to which the primary image copy data set is to be written.	“IC1OUTDD keyword” on page 886
IC2OUTDD	Optional	Specifies the DD name of the output file to which the secondary image copy data set is to be written.	“IC2OUTDD keyword” on page 886
IOVFPOOL	Optional	Specifies the size of the IOVF buffer pool. To allow for overlapped read/extract I/O, twice this number of buffers are actually obtained. The keyword is optional and should be required only for severely disorganized areas.	“IOVFPOOL keyword” on page 888
MAXERROR	Optional	Specifies the maximum number of pointer errors that might be encountered before processing ends abnormally.	“MAXERROR keyword” on page 894
MAXTWINCTR	Optional	Specifies the threshold value for the number of twin segments.	“MAXTWINCTR keyword” on page 895
RGROUP	Optional	Specifies to fill the value in the Repository Group field in the FPXAHST2 file or in a sensor data record set.	“RGROUP keyword” on page 910
SDEP	Optional	Specifies whether or not run totals and SDEP pointer integrity is to be verified and any errors reported.	“SDEP keyword” on page 912
SDEP_ORPHAN_MSGLVL	Optional	Specifies the error message level for an orphan SDEP segment.	SDEP_ORPHAN_MSGLVL keyword
SDEP_PTRIOVF_MSGLVL	Optional	Specifies the severity level of the message that is issued when detecting an SDEP pointer that points to the IOVF portion.	SDEP_PTRIOVF_MSGLVL keyword
SENSOR	Optional	Specifies whether to store the sensor data in the Sensor Data repository of IMS Tools KB.	“SENSOR keyword” on page 922
SENSOR_DBREC	Optional	Specifies whether to store the data elements that are related to database records and root segment distribution to the Sensor Data repository of IMS Tools KB.	“SENSOR_DBREC keyword” on page 923
SENSOR_PTRERR	Optional	Specifies to store the sensor data in the Sensor Data repository of IMS Tools KB even when pointer errors are detected in the area.	“SENSOR_PTRERR keyword” on page 924
SENSOR_UOW	Optional	Specifies whether to store the data elements that are related to UOW to the Sensor Data repository of IMS Tools KB.	“SENSOR_UOW keyword” on page 925
SENSOR_UOW_REPORT	Optional	Specifies whether to print the data elements that are related to UOW in the Sensor Data Statistics report.	SENSOR_UOW_REPORT keyword
STAT	Optional	Specifies whether or not run totals and statistics are to be generated.	“STAT keyword” on page 928
SYNTBL	Optional	Specifies the number of reporting entries in the Synonym Chain Analysis table.	“SYNTBL keyword” on page 932

Table 104. PTRCHK statement keywords (continued)

Keyword	Required or optional	Description	Topic
TARDSNMASK	Optional	Specifies the data set name or the mask to use when dynamically allocating cataloged pre-existing tuning aid record data sets.	“TARDSNMASK keyword” on page 932
TARECGEN	Optional	Specifies whether to generate the tuning aid records.	“TARECGEN keyword” on page 933
TYPRUN	Optional	Specifies the operating mode.	“TYPRUN keyword” on page 935

REPORT statement keywords

The REPORT control statement specifies the parameter values related to the analysis reports. This control statement is effective only when the PTRCHK=YES keyword is specified on the REORG control statement and either of the following keyword combinations exists on the PTRCHK control statement:

- FULLSTEP=YES and TYPRUN=RPT
- FULLSTEP=YES and TYPRUN=FS

The keywords associated with the REPORT statement are available for site default options.

The REPORT control statement is similar to that of the Analyze function of FPA.

For the REPORT statement, you can specify the keywords that are listed in the following table. For the format of the REPORT statement, see [“REPORT control statement” on page 856](#).

Table 105. REPORT statement keywords

Keyword	Required or optional	Description	Topic
DBRLDIST	Optional	Specifies whether to generate the DB Record Length Distribution report.	“DBRLDIST keyword” on page 868
IOVFCI_RBA_REPORT	Optional	Specifies whether to generate the RBA of Used IOVF CI report.	IOVFCI_RBA_REPORT keyword
LARGEREC_COUNT	Optional	Specifies the number of reporting entries in the Largest Database Records report and the Consolidated Largest Database Records report.	LARGEREC_COUNT keyword
LARGEREC_REPORT	Optional	Specifies whether to generate the Largest Database Records report.	LARGEREC_REPORT keyword
REPORTDD	Optional	Specifies the DD name of the output data set or output stream, to which OPC writes the following reports: <ul style="list-style-type: none"> • DEDB Area Analysis reports • Largest Database Records report • UOW report • UOW Exception report 	“REPORTDD keyword” on page 907
SEGLDIST	Optional	Specifies whether to generate the Segment Length Distribution report.	“SEGLDIST keyword” on page 918
SEGLSTAT	Optional	Specifies the format of the DB Record Profile Analysis report, which is one of the DEDB Area Analysis reports.	“SEGLSTAT keyword” on page 918
STARTUOW	Optional	Specifies the number that defines the first UOW to be reported in the UOW report and the UOW Exception report.	“STARTUOW keyword” on page 928
STOPUOW	Optional	Specifies the number that defines the last UOW to be reported in the UOW report and the UOW Exception report.	“STOPUOW keyword” on page 930

Table 105. REPORT statement keywords (continued)

Keyword	Required or optional	Description	Topic
UOW_REPORT	Optional	Specifies whether to generate the UOW report.	"UOW_REPORT keyword" on page 939

THRESHOLD statement keyword

The THRESHOLD control statement specifies the threshold parameter values, which cause warning messages to be generated when the actual values do not meet the specified criteria. If you do not specify the PTRCHK=YES keyword in the REORG control statement, and the TYPRUN=RPT keyword and the FULLSTEP=YES keyword in the PTRCHK control statement, this control statement is ignored.

The keywords associated with the THRESHOLD statement are not available for site default options.

The THRESHOLD control statement is similar to that of the Analyze function of FPA.

For the THRESHOLD statement, you can specify the keywords that are listed in the following table. For the format of the THRESHOLD statement, see ["THRESHOLD control statement" on page 859](#).

Table 106. THRESHOLD statement keywords

Keyword	Required or optional	Description	Topic
DOVFFS_PCT	Optional	Specifies a threshold setting for the percentage of free space in DOVF. If the percentage of free space in DOVF is less than the value that you specified, a warning message is issued.	"DOVFFS_PCT keyword" on page 869
FS_DOVFIOVF	Optional	Specifies a threshold setting for the free space in DOVF and in IOVF. If the free space in DOVF is greater than the value that you specified, and if the free space in IOVF is less than the value that you specified, a warning message is issued.	"FS_DOVFIOVF keyword" on page 880
FS_RAADOVF	Optional	Specifies a threshold setting for the free space in RAA BASE and in DOVF. If the free space in RAA BASE is greater than the value that you specified, and if the free space in DOVF is less than the value that you specified, a warning message is issued.	"FS_RAADOVF keyword" on page 880
FS_RAADOVFIOVF	Optional	Specifies a threshold setting for the free space in RAA BASE, DOVF, and IOVF. If the free space in RAA BASE is greater than the value that you specified, and if the free space in either DOVF or IOVF is less than the value that you specified, a warning message is issued.	"FS_RAADOVFIOVF keyword" on page 881
FS_RAATIOVF	Optional	Specifies a threshold setting for the free space in RAA BASE and in IOVF. If the free space in RAA BASE is greater than the value that you specified, and if the free space in IOVF is less than the value that you specified, a warning message is issued.	"FS_RAATIOVF keyword" on page 881
IOVFFS_PCT	Optional	Specifies a threshold setting for the percentage of free space in IOVF. If the percentage is less than the user-specified value, a warning message is issued.	"IOVFFS_PCT keyword" on page 887
IOVFUSED_PCT	Optional	Specifies a threshold setting for the percentage of IOVF CIs that are used. If the percentage is greater than the value that you specified, a warning message is issued.	"IOVFUSED_PCT keyword" on page 888
RAAFS_PCT	Optional	Specifies a threshold setting for the percentage of free space in RAA BASE. If the percentage is less than the value that you specified, a warning message is issued.	"RAAFS_PCT keyword" on page 903
RAPOVFL_PCT	Optional	Specifies a threshold setting for the percentage of RAA BASE CIs that use either DOVF or IOVF. If the percentage is greater than the value that you specified, a warning message is issued.	"RAPOVFL_PCT keyword" on page 903
RECIO_AVG	Optional	Specifies a threshold setting for the average number of I/Os that can be done to access a database record. If the number is greater than the value that you specified, a warning message is issued.	"RECIO_AVG keyword" on page 905

Table 106. THRESHOLD statement keywords (continued)

Keyword	Required or optional	Description	Topic
RECIO_MAX	Optional	Specifies a threshold setting for the maximum number of I/Os that can be done to access a database record. If the number is greater than the value that you specified, a warning message is issued.	“RECIO_MAX keyword” on page 906
RECIOVF_PCT	Optional	Specifies a threshold setting for the percentage of records that use IOVF CIs. If the percentage is greater than the value that you specified, a warning message is issued.	“RECIOVF_PCT keyword” on page 906
ROOTIO_AVG	Optional	Specifies a threshold setting for the average number of I/Os to get access to a root segment. If the number is greater than the value that you specified, a warning message is issued.	“ROOTIO_AVG keyword” on page 910
ROOTIO_MAX	Optional	Specifies a threshold setting for the maximum number of I/Os that can be done to get access to a root segment. If the number is greater than the value that you specified, a warning message is issued.	“ROOTIO_MAX keyword” on page 911
SYN_LEN_AVG	Optional	Specifies a threshold setting for the average synonym chain length. If the length is greater than the value that you specified, a warning message is issued.	“SYN_LEN_AVG keyword” on page 930
SYN_MAX	Optional	Specifies a threshold setting for the maximum synonym chain length. If the length is greater than the value that you specified, a warning message is issued.	“SYN_MAX keyword” on page 931
SYN_PCT_LEN	Optional	Specifies a threshold setting for the percentage of synonym chains and the synonym chain length. If the percentage of the exceeding synonym chain length that you specified is greater than the value that you specified, a warning message is issued.	“SYN_PCT_LEN keyword” on page 931
UOWDOVF_PCT	Optional	Specifies a threshold setting for the percentage of UOWs that use DOVF. If the percentage is greater than the value that you specified, a warning message is issued.	“UOWDOVF_PCT keyword” on page 937
UOWIOVF_AVG	Optional	Specifies a threshold setting for the average number of IOVF CIs that are used by a UOW. If the number is greater than the value that you specified, a warning message is issued.	“UOWIOVF_AVG keyword” on page 937
UOWIOVF_MAX	Optional	Specifies a threshold setting for the maximum number of IOVF CIs that are used by a UOW. If the number is greater than the value that you specified, a warning message is issued.	“UOWIOVF_MAX keyword” on page 938
UOWIOVF_PCT	Optional	Specifies a threshold setting for the percentage of UOWs that use IOVF. If the percentage is greater than the value that you specified, a warning message is issued.	“UOWIOVF_PCT keyword” on page 938

Related reference

[Keyword reference for FPO](#)

These topics explain the control statements and keywords of FPO and their syntax rules.

[Syntax rules for FPXIN](#)

FPXIN control statements and their associated keywords can be coded in free format in columns 1 through 72, subject to certain syntax rules.

Related information

[Site default options support for FPO](#)

The site default options are control statements that are specified in a member of a partitioned data set and that are used to override the system default option values, which can be changed without assembling and link-editing.

Output for OER

These topics describe the outputs that are generated by OER, their formats, and data elements.

FPXRPTS DD data set

The Reorganization Processing Summary report, the UOW Space Statistics report, and the Process Summary report are produced in the data set or written to the output stream that is determined by the RFILE keyword of the REORG control statement.

The Online Pointer Checker Statistics report and the Online Pointer Scan report are produced by OPC when PTRCHK=YES is specified.

Detailed area statistics reports and the Image Copy NOTIFY report can be produced by OPC that runs under OER with certain specifications.

Subsections:

- [“Reorganization Processing Summary report” on page 803](#)
- [“UOW Space Statistics report” on page 804](#)
- [“Online Pointer Checker Statistics report” on page 805](#)
- [“Online Pointer Scan report” on page 806](#)
- [“Detailed area statistics reports” on page 806](#)
- [“Process Summary report” on page 807](#)
- [“Image Copy NOTIFY report” on page 808](#)

Reorganization Processing Summary report

The Reorganization Processing Summary report shows the information of the area definition, the reorganization processing options, the result of UOW evaluation, and the result of reorganization. This report is printed regardless of what is specified for the RFSSTAT parameter of the REORG control statement.

The following figure shows an example of the report.

Area Information	
Database name.....	DEDBJN23
Area name.....	DB23AR0
UOW parameter of DBD.....	(32,16)
ROOT parameter of DBD.....	(20000,9975)
CI size.....	4096
Number of UOWs in Root Addressable Part.....	10025
Number of RAPs in a UOW.....	16
Number of CIs in DOVF of a UOW.....	16
Number of CIs in IOVF.....	319200
Number of IOVF Space Map CIs.....	2660
Processing Options	
RBASEFS.....	15%
RDOVFFS.....	15%
Free Space Scan (RFSSCAN)	
before reorganization.....	YES
after reorganization.....	YES
Reorganization Limit (REORGLIM).....	20
Number of failures allowed (MAXFAILS).....	5
Maximum number of buffer sets (MAXBUFFS).....	3
Result of UOW Evaluation	
Number of UOWs that match RBASEFS.....	32
Number of UOWs that match RDOVFFS.....	25
Number of UOWs that match both RBASEFS and RDOVFFS....	25
Number of UOWs that use at least one IOVF CI.....	20
Number of IOVF CIs used before Reorganization	91255 (28% of total)
Result of Reorganization	
Number of UOWs successfully reorganized.....	20
Number of UOWs skipped.....	0
Number of UOWs failed in reorganization.....	0
Number of IOVF CIs used after Reorganization	48500 (15% of total)
Number of IOVF CIs reclaimed	42755 (13% of total)

Figure 340. Reorganization Processing Summary report

UOW Space Statistics report

The UOW Space Statistics report is produced when RFSSTAT=YES is effective. The statistics are reported only for the UOWs that are selected for reorganization unless REORGLIM=0 or REORGLIM=0% is specified. If REORGLIM=0 or REORGLIM=0% is specified, the statistics for all UOWs are reported.

Figure 341 on page 805 shows the UOW Space Statistics report when RFSSCAN=(YES,YES) is specified.

The left-most column with the label 'UOW No.' shows the UOW number. The first row of the entry for each UOW shows the statistics before the reorganization; the second row shows the statistics after the reorganization.

The column with the label 'Number of CIs' is a collection of sub-columns, each of which show a number of CIs that satisfy a certain condition. The left three sub-columns show the number of CIs used in the RAA Base section of the UOW, in the DOVF section of the UOW, and in the IOVF part. The next sub-column with label 'OVERFLW' shows the number of RAP CIs that uses at least one overflow CI in the DOVF section or in the IOVF part. The sub-column with the label of 'RBASEFS' shows the number of RAP CIs that satisfy the condition specified by the RBASEFS parameter.

The next column with label 'PCT CI Free Space' shows the free space percentages of RAP CIs and the DOVF section. The sub-column named 'RAP CIs' consists of sub-columns 'OVERFLW' and 'All CIs'. The sub-column named 'OVERFLW' shows the percentage of the average, maximum, and minimum free space of RAP CIs that overflowed into DOVF or IOVF. The sub-column named 'All CIs' shows the percentage of the average, maximum, and minimum free space of all RAP CIs. Every RAP CI is included in the calculation of these values whether it is overflowed into DOVF or IOVF. The sub-column named 'DOVF CIs' shows the free space percentage, which is the average value of all DOVF CIs.

The next column 'Reorg Result Flags' shows the result of reorganization of each of the selected UOWs. If the flag character 'F' is printed, it means that the reorganization for the UOW failed.

UOW Space Statistics before and after Reorganization:
Upper row for each UOW shows the statistics before reorganization
Lower row for each UOW shows the statistics after reorganization

UOW No.	Number of CIs						PCT CI			Free Space			DOVF CIs Avg	Reorg Result Flags
	RAP	Used CIs DOVF	IOVF	OVERFLW	RAP CIs	RBASEFS	Avg	Max	Min	Avg	Max	Min		
23	24	17	63	24	24	66	77	48	66	77	48	38		
	24	20	32	21	21	13	27	1	14	32	1	14		
21	22	18	49	17	17	68	77	25	68	77	25	30		
	22	20	27	16	16	15	26	2	29	75	2	14		
27	28	16	66	22	22	64	77	47	63	77	47	40		
	28	20	36	22	22	9	25	3	21	70	3	18		
25	26	15	61	24	24	66	74	50	67	74	50	45		
	26	20	33	20	20	12	23	1	18	73	1	16		
19	20	16	39	18	18	60	77	34	61	77	34	41		
	20	20	16	15	15	10	27	1	17	70	1	17		
15	16	16	37	14	14	62	73	35	62	73	35	43		
	16	20	19	14	14	14	35	0	20	66	0	16		
28	30	19	60	28	28	38	75	22	39	76	22	37		
	30	20	38	27	27	13	30	0	16	76	0	17		
13	14	16	21	13	13	67	75	57	67	75	57	45		
	14	20	4	12	12	13	27	0	18	63	0	14		
24	26	15	53	22	22	41	75	25	43	75	25	49		
	26	20	28	20	20	9	22	0	16	74	0	14		
26	28	12	64	25	25	39	76	23	40	76	23	58		
	28	20	38	25	25	15	38	1	19	61	1	16		
20	22	16	45	20	20	40	62	28	41	67	28	41		
	22	20	25	15	15	11	21	1	15	67	1	16		
22	24	16	44	21	21	40	76	23	43	76	23	47		
	24	20	19	20	20	11	27	0	17	70	0	16		
18	20	16	37	19	19	41	67	23	42	67	23	41		
	20	20	17	15	15	14	23	5	15	57	2	15		
.	
.	
.	

Figure 341. UOW Space Statistics report when RFSSCAN=(YES,YES)

The following figure shows the UOW Space Statistics report when RFSSCAN=(YES,NO) is specified.

UOW No.	Number of CIs						PCT CI			Free Space			DOVF CIs Avg	Reorg Result Flags
	RAP	Used CIs DOVF	IOVF	OVERFLW	RAP CIs	RBASEFS	Avg	Max	Min	Avg	Max	Min		
23	24	17	63	24	24	66	77	48	66	77	48	38		
21	22	18	49	17	17	68	77	25	68	77	25	30		
27	28	16	66	22	22	64	77	47	63	77	47	40		
25	26	15	61	24	24	66	74	50	67	74	50	45		
19	20	16	39	18	18	60	77	34	61	77	34	41		
15	16	16	37	14	14	62	73	35	62	73	35	43		
28	30	19	60	28	28	38	75	22	39	76	22	37		
13	14	16	21	13	13	67	75	57	67	75	57	45		
24	26	15	53	22	22	41	75	25	43	75	25	49		
26	28	12	64	25	25	39	76	23	40	76	23	58		
20	22	16	45	20	20	40	62	28	41	67	28	41		
22	24	16	44	21	21	40	76	23	43	76	23	47		
18	20	16	37	19	19	41	67	23	42	67	23	41		
.	
.	
.	

Figure 342. UOW Space Statistics report when RFSSCAN=(YES,NO)

Online Pointer Checker Statistics report

The Online Pointer Checker Statistics report contains information about the area definitions and results from checking pointers.

The Online Pointer Checker Statistics report is produced by OPC, which is internally called by OER when PTRCHK=YES is specified.

Online Pointer Scan report

The Online Pointer Scan report contains information about pointer errors.

The Online Pointer Scan report is produced by OPC, which is internally called by OER when PTRCHK=YES is specified.

Detailed area statistics reports

If PTRCHK=YES keyword is specified for the REORG control statement, and the TYPRUN=FS keyword and the FULLSTEP=YES keyword are specified for the PTRCHK control statement, the following reports are produced in the data set or in the output stream, depending on the REPORTDD keyword of the REPORT control statement:

- Processing Summary of Analysis report
- DEDB Area Analysis report:
 - Freespace Analysis report
 - Summary section
 - Overview and RAA BASE section
 - DOVF and IOVF sections
- UOW report (without DB record information)
- UOW Exception report (without DB record information)
- RBA of Used IOVF CI report

If PTRCHK=YES keyword is specified for the REORG control statement, and the TYPRUN=RPT keyword and the FULLSTEP=YES keyword are specified for the PTRCHK control statement, the following reports are produced in the data set or in the output stream, depending on the REPORTDD keyword of the REPORT control statement:

- Processing Summary of Analysis report
- DEDB Area Analysis reports:
 - Freespace Analysis report
 - Summary section
 - Overview and RAA BASE section
 - DOVF and IOVF sections
 - DB Record Profile Analysis report
 - DB Record Placement Analysis report
 - Segment Placement Analysis report
 - Segment I/O Analysis report
 - Synonym Chain Analysis report
- Largest Database Records report (when LARGEREC_REPORT=YES)
- Consolidated Largest Database Records report (when LARGEREC_REPORT=YES and LARGEREC_COUNT=(value1,value2))
- DB Record Length Distribution report (when DBRLDIST=YES)
- Segment Length Distribution report (when SEGLDIST=YES)
- UOW report
- UOW Exception report
- RBA of Used IOVF CI report

These reports provide the database administrator with the information of the overall health of the DEDB area that has been just reorganized.

Process Summary report

The Process Summary report summarizes the result of processing for each area.

The following figure shows an example of the report.

IMS HPFP UTILITIES - OER 5698-FPP V2R1		"Process Summary Report"				PAGE: 1 2019-11-22 19:06:24
DBDNAME	AREANAME	AREA NO	START DATE/TIME	END DATE/TIME	STATUS (If not blank, see messages in the message data sets.)	
DEDBJN23	DB23AR2	2	2019-11-22 19:06:24	2019-11-22 19:06:25	Processing ended with warnings.	
DEDBJN23	DB23AR1	1	2019-11-22 19:06:29	2019-11-22 19:06:30	Processing ended with errors.	
DEDBJN23	DB23AR4	4	2019-11-22 19:06:33	2019-11-22 19:06:33		
			2019-11-22 19:06:36	2019-11-22 19:06:40		
DEDBJN23	DB23AR3	3	2019-11-22 19:06:40	2019-11-22 19:06:41		
			2019-11-22 19:06:41	2019-11-22 19:06:44		

Figure 343. Process Summary report (OER)

DBDNAME

Shows the name of the database.

AREANAME

Shows the name of the area. The area names are displayed in processed order.

AREA NO

Shows the area number.

START DATE/TIME

Shows the time stamp when the area was connected.

END DATE/TIME

Shows the time stamp when the area was disconnected.

STATUS

Shows the processing result for each area.

If OER ended successfully without errors, this column is blank. Otherwise, one of the following messages is displayed in this column:

Processing ended with warnings.

This message indicates that the processing ended with return code of 4. No pointer errors were detected during OPC processing that ran under OER.

Processing ended with errors.

This message indicates that the processing ended with return code of 8.

Pointer errors were detected.

This message indicates that the processing ended with return code of 4. One or more pointer errors were detected during OPC processing that ran under OER.

The following areas not processed because PROC_AFT_ERROR=STOP.

This message indicates that the processing ended with errors in the previous area. This area and the subsequent areas are not processed because PROC_AFT_ERROR=STOP is specified. The status for the subsequent areas is blank.

Processing did not complete.

This message indicates that the processing for the area did not complete because the processing ended abnormally, or an error, which prevented from processing the subsequent areas, was encountered. The status for the subsequent areas is blank.

Processing was skipped.

This message indicates that the processing for this area was skipped because the area was stopped. This message is displayed only when AREA=ALL is specified.

For details about these messages, see the messages produced in the FPXMFIL data set or in the SYSPRINT data set.

Because OER connects to and disconnects from an area more than once, the time stamp values for START DATE/TIME and END DATE/TIME for an area might be shown in multiple lines.

Image Copy NOTIFY report

The Image Copy NOTIFY report contains the NOTIFY.IC control card.

The Image Copy NOTIFY report is produced by OPC, which is internally called by OER when PTRCHK=YES is specified.

If PTRCHK=YES keyword is specified for the REORG control statement, and the IC=(YES,ICEIN) keyword is specified for the PTRCHK control statement, Image Copy NOTIFY report is produced in the data set or in the output stream depending on the RFILE keyword of the REORG control statement. If IMS HP Image Copy processing fails or when the specified area is not registered to DBRC, the Image Copy NOTIFY report is not produced.

FPXSPRT DD data set

The FPXSPRT DD data set contains the Sensor Data Statistics report that is generated by OPC when the sensor data is stored in the Sensor Data repository of IMS Tools KB.

This topic describes the FPXSPRT DD data set, which is generated by OPC only when FPXIN DD is used. For OPC with DASYSIN DD, no FPXSPRT DD data set is generated.

Sensor Data Statistics report

The Sensor Data Statistics report contains the names and the values of the data elements that are stored in the Sensor Data repository of IMS Tools KB.

For a list of the data elements that are collected by IMS HP Fast Path Utilities, see the topics about data elements in the *Tools Base Policy Services User's Guide*.

The following figure shows an example of the Sensor Data Statistics report. The data elements that are underlined are printed only when SENSOR_DBREC=YES is specified.

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1

=====

AREA DEFINITION

=====

Data Element Name	Value
DB_AREAEDEF_CISIZE	4,096
DB_AREAEDEF_UOW1	20
DB_AREAEDEF_UOW2	10
DB_AREAEDEF_ROOT1	1,500
DB_AREAEDEF_ROOT2	1,000
DB_AREAEDEF_NUM_SDEP_CIS	10,000

=====

FREE SPACE PERCENTAGE STATISTICS

=====

Data Element Name	Value
DB_PCT_BYTES_FS_RAA	50%
DB_PCT_BYTES_FS_DOVF	50%
DB_PCT_BYTES_FS_IOVF	50%
DB_PCT_BYTES_FS_SDEP	52%

=====

OVERFLOW USAGE STATISTICS

=====

Data Element Name	Value
DB_PCT_NUM_UOW_USE_DOVF	50%
DB_AVG_NUM_DOVFCI_BY_UOW	678.99
DB_MAX_NUM_DOVFCI_BY_UOW	9,999
DB_PCT_NUM_UOW_USE_IOVF	50%
DB_NUM_UOW_USE_IOVF	9,999
DB_AVG_NUM_IOVFCI_BY_UOW	678.99
DB_MAX_NUM_IOVFCI_BY_UOW	9,999
DB_MIN_NUM_IOVFCI_BY_UOW	9,999
DB_PCT_NUM_IOVFCI_USED	50%
DB_PCT_NUM_RAPCI_OVFL	50%

=====

SEGMENT OCCURRENCE STATISTICS

=====

Data Element Name	Value
DB_NUM_SEG	12,345,678

Figure 344. Sensor Data Statistics report (Part 1 of 2)

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1

```

=====
DATABASE RECORD STATISTICS
=====
Data Element Name          Value
-----
DB_NUM_ROOT                111,111
DB_AVG_DBREC_LENGTH       111.00
DB_MAX_DBREC_LENGTH       111,111
DB_MIN_DBREC_LENGTH       111,111
DB_PCT_NUM_DBREC_IOVF     50%
=====

RANDOMIZING STATISTICS
=====
Data Element Name          Value
-----
DB_AVG_LEN_SYNONYM_CHAIN   1.11
DB_MAX_LEN_SYNONYM_CHAIN   99,999
=====

PHYSICAL I/O STATISTICS
=====
Data Element Name          Value
-----
DB_ESTIMATED_DBREC_IO      1.25
DB_ESTIMATED_ROOT_IO      1.25
DB_AVG_DBREC_IO            678.99
DB_MAX_DBREC_IO            99,999
DB_AVG_ROOT_IO             678.99
DB_MAX_ROOT_IO             99,999
=====

UOW STATISTICS INFORMATION
=====
Data Element Name          Value
-----
DB_FLAG_UOW_DATA           N
DB_FLAG_UOW_GROUP_DATA     N
DB_THRESHOLD_RBSEFS        30%
DB_THRESHOLD_RDVFFS        50%
DB_NUM_UOW_RFS_COND        50
DB_PCT_NUM_UOW_RFS_COND    10%
=====

```

Figure 345. Sensor Data Statistics report (Part 2 of 2)

The following figures show another example of the Sensor Data Statistics report. In this report, logical pages 3 and 4 contain data element values for the first UOW in the area that is specified by the IAREA keyword. This report is printed if both of the following conditions are met:

- SENSOR_UOW=YES is specified
- FPXAUOWC DD statement is not specified

In actual reports, data element values for other UOWs are printed in the subsequent pages.

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1

```
=====
AREA DEFINITION
=====
```

Data Element Name	Value
DB_AREAEDEF_CISIZE	4,096
DB_AREAEDEF_UOW1	20
DB_AREAEDEF_UOW2	10
DB_AREAEDEF_ROOT1	1,500
DB_AREAEDEF_ROOT2	1,000
DB_AREAEDEF_NUM_SDEP_CIS	10,000

```
=====
FREE SPACE PERCENTAGE STATISTICS
=====
```

Data Element Name	Value
DB_PCT_BYTES_FS_RAA	50%
DB_PCT_BYTES_FS_DOVF	50%
DB_PCT_BYTES_FS_IOVF	50%
DB_PCT_BYTES_FS_SDEP	52%

```
=====
OVERFLOW USAGE STATISTICS
=====
```

Data Element Name	Value
DB_PCT_NUM_UOW_USE_DOVF	50%
DB_AVG_NUM_DOVFCI_BY_UOW	678.99
DB_MAX_NUM_DOVFCI_BY_UOW	9,999
DB_PCT_NUM_UOW_USE_IOVF	50%
DB_NUM_UOW_USE_IOVF	9,999
DB_AVG_NUM_IOVFCI_BY_UOW	678.99
DB_MAX_NUM_IOVFCI_BY_UOW	9,999
DB_MIN_NUM_IOVFCI_BY_UOW	9,999
DB_PCT_NUM_IOVFCI_USED	50%
DB_PCT_NUM_RAPCI_OVFL	50%

```
=====
SEGMENT OCCURRENCE STATISTICS
=====
```

Data Element Name	Value
DB_NUM_SEG	12,345,678

Figure 346. Sensor Data Statistics report when SENSOR_UOW=YES (without FPXAUOWC DD) (Part 1 of 3)

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1

=====

DATABASE RECORD STATISTICS

=====

Data Element Name	Value
DB_NUM_ROOT	111,111
DB_AVG_DBREC_LENGTH	111.00
DB_MAX_DBREC_LENGTH	111,111
DB_MIN_DBREC_LENGTH	111,111
DB_PCT_NUM_DBREC_IOVF	50%

=====

RANDOMIZING STATISTICS

=====

Data Element Name	Value
DB_AVG_LEN_SYNONYM_CHAIN	1.11
DB_MAX_LEN_SYNONYM_CHAIN	99,999

=====

PHYSICAL I/O STATISTICS

=====

Data Element Name	Value
DB_ESTIMATED_DBREC_IO	1.25
DB_ESTIMATED_ROOT_IO	1.25
DB_AVG_DBREC_IO	678.99
DB_MAX_DBREC_IO	99,999
DB_AVG_ROOT_IO	678.99
DB_MAX_ROOT_IO	99,999

=====

UOW STATISTICS INFORMATION

=====

Data Element Name	Value
DB_FLAG_UOW_DATA	Y
DB_FLAG_UOW_GROUP_DATA	N
DB_THRESHOLD_RBSEFS	30%
DB_THRESHOLD_RDOVFFS	50%
DB_NUM_UOW_RFS_COND	50
DB_PCT_NUM_UOW_RFS_COND	10%

=====

REPOSITORY GROUP INFORMATION

=====

Data Element Name	Value
DB_SENSOR_DATA_GROUP_ID	IMS1

Figure 347. Sensor Data Statistics report when SENSOR_UOW=YES (without FPXAUOWC DD) (Part 2 of 3)

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1, UOW NUMBER: 0

=====

FREE SPACE PERCENTAGE STATISTICS

=====

Data Element Name	Value
DBU_PCT_BYTES_FS_RAA	80%
DBU_PCT_BYTES_FS_DOVF	80%
DBU_PCT_BYTES_FS_IOVF	95%
DBU_PCT_USABLE_RAAFS	80%
DBU_PCT_USABLE_DOVFFS	80%
DBU_PCT_USABLE_IOVFFS	95%
DBU_PCT_RAP_ROOTSZFS	0%
DBU_MAX_PCT_BYTES_RAPFS	0%
DBU_FLAG_UOW_USING_OVFL	Y
DBU_FLAG_UOW_USING_IOVF	Y

=====

OVERFLOW USAGE STATISTICS

=====

Data Element Name	Value
DBU_NUM_DOVFCI_BY_UOW	10
DBU_NUM_IOVFCI_BY_UOW	3
DBU_PCT_NUM_RAPCI_OVFL	100%

=====

DATABASE RECORD STATISTICS

=====

Data Element Name	Value
DBU_NUM_ROOT	120
DBU_AVG_DBREC_LENGTH	106.00
DBU_MAX_DBREC_LENGTH	582
DBU_MIN_DBREC_LENGTH	102
DBU_PCT_NUM_DBREC_IOVF	18%

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1, UOW NUMBER: 0

=====

RANDOMIZING STATISTICS

=====

Data Element Name	Value
DBU_AVG_LEN_SYN_CHAIN	120.00
DBU_MAX_LEN_SYN_CHAIN	120

=====

PHYSICAL I/O STATISTICS

=====

Data Element Name	Value
DBU_AVG_DBREC_IO	7.06
DBU_MAX_DBREC_IO	13
DBU_AVG_ROOT_IO	7.06
DBU_MAX_ROOT_IO	13

Figure 348. Sensor Data Statistics report when SENSOR_UOW=YES (without FPXAUOWC DD) (Part 3 of 3)

The following figures show another example of the Sensor Data Statistics report. In this report, logical pages 3 and 4 contain data element values for a UOW group that is specified by the HFPXAUOWC DD statement. This report is printed if both of the following conditions are met:

- SENSOR_UOW=YES is specified
- FPXAUOWC DD statement is specified

In actual reports, if multiple UOW groups are specified, data element values for other UOW groups are printed in the subsequent pages.

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1

```
=====
AREA DEFINITION
=====
```

Data Element Name	Value
DB_AREAEDEF_CISIZE	4,096
DB_AREAEDEF_UOW1	20
DB_AREAEDEF_UOW2	10
DB_AREAEDEF_ROOT1	1,500
DB_AREAEDEF_ROOT2	1,000
DB_AREAEDEF_NUM_SDEP_CIS	10,000

```
=====
FREE SPACE PERCENTAGE STATISTICS
=====
```

Data Element Name	Value
DB_PCT_BYTES_FS_RAA	50%
DB_PCT_BYTES_FS_DOVF	50%
DB_PCT_BYTES_FS_IOVF	50%
DB_PCT_BYTES_FS_SDEP	52%

```
=====
OVERFLOW USAGE STATISTICS
=====
```

Data Element Name	Value
DB_PCT_NUM_UOW_USE_DOVF	50%
DB_AVG_NUM_DOVFCI_BY_UOW	678.99
DB_MAX_NUM_DOVFCI_BY_UOW	9,999
DB_PCT_NUM_UOW_USE_IOVF	50%
DB_NUM_UOW_USE_IOVF	9,999
DB_AVG_NUM_IOVFCI_BY_UOW	678.99
DB_MAX_NUM_IOVFCI_BY_UOW	9,999
DB_MIN_NUM_IOVFCI_BY_UOW	9,999
DB_PCT_NUM_IOVFCI_USED	50%
DB_PCT_NUM_RAPCI_OVFL	50%

```
=====
SEGMENT OCCURRENCE STATISTICS
=====
```

Data Element Name	Value
DB_NUM_SEG	12,345,678

Figure 349. Sensor Data Statistics report when SENSOR_UOW=YES (with FPXAUOWC DD) (Part 1 of 3)

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1

=====

DATABASE RECORD STATISTICS

=====

Data Element Name	Value
DB_NUM_ROOT	111,111
DB_AVG_DBREC_LENGTH	111.00
DB_MAX_DBREC_LENGTH	111,111
DB_MIN_DBREC_LENGTH	111,111
DB_PCT_NUM_DBREC_IOVF	50%

=====

RANDOMIZING STATISTICS

=====

Data Element Name	Value
DB_AVG_LEN_SYNONYM_CHAIN	1.11
DB_MAX_LEN_SYNONYM_CHAIN	99,999

=====

PHYSICAL I/O STATISTICS

=====

Data Element Name	Value
DB_ESTIMATED_DBREC_IO	1.25
DB_ESTIMATED_ROOT_IO	1.25
DB_AVG_DBREC_IO	678.99
DB_MAX_DBREC_IO	99,999
DB_AVG_ROOT_IO	678.99
DB_MAX_ROOT_IO	99,999

=====

UOW STATISTICS INFORMATION

=====

Data Element Name	Value
DB_FLAG_UOW_DATA	N
DB_FLAG_UOW_GROUP_DATA	Y
DB_NUM_UOW_GROUPS	2
DB_THRESHOLD_RBASEFS	30%
DB_THRESHOLD_RDOVFFS	50%
DB_NUM_UOW_RFS_COND	50
DB_PCT_NUM_UOW_RFS_COND	10%

=====

REPOSITORY GROUP INFORMATION

=====

Data Element Name	Value
DB_SENSOR_DATA_GROUP_ID	IMS1

Figure 350. Sensor Data Statistics report when SENSOR_UOW=YES (with FPXAUOWC DD) (Part 2 of 3)

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1, UOW GROUP (STARTUOW: 10001, STOPUOW: 20000)

=====

FREE SPACE PERCENTAGE STATISTICS

=====

Data Element Name	Value
DEBUG_PCT_BYTES_FS_RAA	80%
DEBUG_PCT_BYTES_FS_DOVF	80%
DEBUG_PCT_BYTES_FS_IOVF	95%
DEBUG_PCT_USABLE_RAASF	80%
DEBUG_PCT_USABLE_DOVFFS	80%
DEBUG_PCT_USABLE_IOVFFS	95%
DEBUG_PCT_RAP_ROOTSZFS	0%

=====

OVERFLOW USAGE STATISTICS

=====

Data Element Name	Value
DEBUG_PCT_NUM_UOW_DOVF	80%
DEBUG_AVG_NUM_DOVFCCI	5
DEBUG_MAX_NUM_DOVFCCI	10
DEBUG_PCT_NUM_UOW_IOVF	80%
DEBUG_AVG_NUM_IOVFCCI	10
DEBUG_MAX_NUM_IOVFCCI	50
DEBUG_PCT_NUM_RAPCI_OVFL	100%

=====

DATABASE RECORD USAGE STATISTICS

=====

Data Element Name	Value
DEBUG_NUM_ROOT	120
DEBUG_AVG_DBREC_LENGTH	106.00
DEBUG_MAX_DBREC_LENGTH	582
DEBUG_MIN_DBREC_LENGTH	102
DEBUG_PCT_NUM_DBREC_IOVF	18%

DBD NAME: DEDBJN23, AREA NAME: DB23AR0 , AREA NUMBER: 1, UOW GROUP (STARTUOW: 10001, STOPUOW: 20000)

=====

RANDOMIZING STATISTICS

=====

Data Element Name	Value
DEBUG_AVG_LEN_SYN_CHAIN	120.00
DEBUG_MAX_LEN_SYN_CHAIN	120

=====

PHYSICAL I/O STATISTICS

=====

Data Element Name	Value
DEBUG_AVG_DBREC_IO	7.06
DEBUG_MAX_DBREC_IO	13
DEBUG_AVG_ROOT_IO	7.06
DEBUG_MAX_ROOT_IO	13

Figure 351. Sensor Data Statistics report when SENSOR_UOW=YES (with FPXAUOWC DD) (Part 3 of 3)

SYSPRINT DD data set

The IMS system messages and control statement analysis messages are produced in the data set or the system output stream that is specified on the SYSPRINT DD statement.

FPXMFIL DD data set

Area Sensor messages, Reorganizer messages, and OPC messages are produced in the FPXMFIL data set.

These messages can be routed to another output data set or to the system output stream by coding the MFILE parameter of the REORG control statement.

Related reference

[REORG control statement](#)

The REORG statement specifies information that controls how the OER run is to be processed.

Examples for OER

There are many ways to run the OER utility. The examples provided in the following topics show some of the typical ways to run the OER utility.

You can find additional JCL examples that help you to monitor and tune DEDB areas in [“Online Expert Reorganization JCL examples for monitoring and tuning DEDBs”](#) on page 1162.

Example 1: Running OER without OPC

The following figure shows example JCL that runs OER without OPC. In this example, the IMS cataloged procedure FPUTIL is used.

The REORG control statement specified in the FPXIN DD statement specifies reorganization options. The name of the area is specified by the AREA keyword. Note that only one DEDB area can be processed in an OER job step.

```
//REORG EXEC FPUTIL,DBD=DEDBJN23,IMSID=IMS1
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
//FPXIN DD *
        REORG AREA=DB23AR0,
              RDOVFFS=20,
              RBASEFS=20,
              REORGLIM=20
/*
```

Figure 352. Example JCL to run OER without OPC

First, the UOWs to be reorganized are limited to the subset of UOWs that satisfy one of the RDOVFFS or RBASEFS condition. Then, UOWs in the subset are ranked on the basis of statistics gathered by the Area Sensor. Second, because REORGLIM=20 is specified, higher ranked 20 UOWs are selected from the ranked subset of UOWs and those selected UOWs are reorganized by this OER job.

The UOW Space Statistics report is produced by default in the FPXRPTS DD. The destination of report output can be changed by coding the RFILE keyword. If RFSSTAT=NO is specified, the UOW Space Statistics report is suppressed. The messages that show the progress of OER processing is produced by default in the data set or the system output stream specified on the SYSPRINT DD. The default can be changed by coding the MFILE keyword.

The following figure shows another example. The reorganization limit (REORGLIM) is specified in percentage and the higher ranked 25% UOWs out of all UOWs in the area are reorganized. The message and report outputs are rerouted by the MFILE and RFILE keywords.

```
//REORG EXEC FPUTIL,DBD=DEDBJN23,IMSID=IMS1
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
//FPXIN DD *
        REORG AREA=DB23AR0,
              RBASEFS=15,
              RDOVFFS=25,
              REORGLIM=25%,
              MFILE=MESSAGES,
              RFILE=REPORTS
/*
//MESSAGES DD SYSOUT=*
//REPORTS DD DSN=HPFPU.OER.REPORTS.DB23AR0,DISP=OLD
```

Figure 353. Example JCL to run OER without OPC

Example 2: Running OER with OPC

The following figure shows example JCL that runs OER with the pointer checking of the reorganized UOWs.

PTRCHK=YES in the REORG control statement specifies that OPC be called after a set of UOWs have been reorganized. The options for OPC, including TYPRUN, which specifies the type of pointer check, are specified in the PTRCHK control statement.

```
//REORG EXEC FPUTIL, DBD=DEDBJN23, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
// DD DISP=SHR, DSN=IMSVS.SDFSRESL
//FPXIN DD *
REORG AREA=DB23AR0,
      RDOVFFS=20,
      RBASEFS=20,
      REORGLIM=20,
      PTRCHK=YES
/*
```

Figure 354. Example JCL to run OER with the pointer checking of the reorganized UOWs

The CHKUOW=REORGED option of the PTRCHK control statement is effective by default, which means that only those UOWs that have been actually reorganized are pointer-checked. In this case, TYPRUN must be PTRSCAN, which is the default.

If every UOW in the area, whether it is reorganized or not, needs to be pointer-checked, the PTRCHK control statement must be coded and the CHKUOW=ALL parameter must be specified. The following figure shows an example JCL to run OER with pointer-checking all UOWs in the area. If a concurrent image copy needs to be taken during the pointer check, CHKUOW=ALL must also be specified.

```
//REORG EXEC FPUTIL, DBD=DEDBJN23, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
// DD DISP=SHR, DSN=IMSVS.SDFSRESL
//FPXIN DD *
REORG AREA=DB23AR0,
      RDOVFFS=20,
      RBASEFS=20,
      REORGLIM=20,
      PTRCHK=YES
PTRCHK CHKUOW=ALL
/*
```

Figure 355. Example JCL to run OER with the pointer checking for all UOWs

OER can also run OPC after a subset of UOWs have been reorganized and can produce reports that contain the space statistics of each UOW in the area. The PTRCHK control statement must be coded, and CHKUOW=ALL, FULLSTEP=YES, and TYPRUN=RPT must be specified.

The following figure shows example JCL for reorganizing the subset of UOWs in an online area, checking pointer validation, and producing complete analysis reports.

```
//REORG EXEC FPUTIL, DBD=DEDBJN23, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
// DD DISP=SHR, DSN=IMSVS.SDFSRESL
//FPXIN DD *
REORG AREA=DB23AR0
PTRCHK YES
PTRCHK TYPRUN=RPT
      FULLSTEP=YES
      SDEP=YES
      CHKUOW=ALL
REPORT REPORTDD=FPXRPTS
THRESHOLD DOVFFS_PCT=25
/*
//FPXRPTS DD SYSOUT=*
//SYSOUT DD SYSOUT=*
```

Figure 356. Running OER with OPC to check pointer validation and generate analysis reports

The SYSPRINT and SYSUDUMP DD statements are coded in the FPUTIL cataloged procedure.

Example 3: Running OER with OPC for multiple areas and taking image copies

The following figure shows example JCL for running OER with OPC for multiple areas and generating an image copy for each area by using IMS HP Image Copy.

```
//REORG EXEC FPUTIL,DBD=DEDBJN23,IMSID=IMS1
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=HPS.SHPSLMD0
// DD DISP=SHR,DSN=IMS.SDFSRESL
//FPXIN DD *
REORG AREA=(DB23AR1,DB23AR2),
PTRCHK=YES
PTRCHK CHKUOW=ALL,
FULLSTEP=YES,
TYPRUN=RPT,
IC=(YES,ICEIN),
ICNOTIFY=YES
REPORT REPORTDD=FPXRPTS
/*
//ICEIN DD *
GLOBAL COMP=Y,COMPRTN=FABJCMP4
CIC DBD=DEDBJN23,
AREA=DB23AR1,
ICOUT=ICOUT001
CIC DBD=DEDBJN23,
AREA=DB23AR2,
ICOUT=ICOUT002
/*
//FPXRPTS DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//ICERPRT DD SYSOUT=*
```

Figure 357. Running OER with OPC for multiple areas and taking image copies

Example 4: Running OER with OPC and storing sensor data in the IMS Tools KB repository

The following figure shows example JCL for running OER with OPC and for storing the sensor data in the IMS Tools KB repository.

In addition to the default data elements that are collected by the SENSOR keyword, the data elements that are related to database records and root segment distribution are stored (SENSOR_DBREC=YES specification).

```

//REORG      EXEC FPUTIL, DBD=DEDBJN23, IMSID=IMS1
//*
//FPU.STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//          DD DISP=SHR, DSN=IMSVS.SDFSRESL
//          DD DISP=SHR, DSN=ITB.SHKTL0AD
//FPXMFILe   DD SYSOUT=*
//FPXRPTS    DD SYSOUT=*
//FPXSPRT    DD SYSOUT=*
//RECON1     DD DISP=SHR, DSN=IMSVS.RECON1
//RECON2     DD DISP=SHR, DSN=IMSVS.RECON2
//RECON3     DD DISP=SHR, DSN=IMSVS.RECON3
//FPXIN      DD *
GLOBAL
  DBRC=YES,
  ITKBSRVR=FPQSVR01,
  ADXCFGRP=ADSVR01
REORG
  AREA=ALL,
  PTRCHK=YES
PTRCHK
  TYPRUN=RPT,
  CHKUOW=ALL,
  FULLSTEP=YES,
  SENSOR=YES,
  SENSOR_DBREC=YES
/*

```

Figure 358. Running OER with OPC and storing sensor data in the IMS Tools KB repository

Example 5: Running OER with the segment shunting function

The following figure shows a JCL example for running OER and enabling the segment shunting function for multiple areas.

In this example:

- The INDD data set specifies to enable segment shunting.
- For area DB23AR1, segments DD1 and DD11 will not be reorganized into the RAP CI, but other segments will be inserted into the RAP CI if space exists.
- For area DB23AR2, segments DD1, DD11, DD12, and DD2 will not be reorganized into the RAP CI, but other segments will be inserted into the RAP CI if space exists.

```

//OER      EXEC FPUTIL, DBD=DEDBJN23, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//          DD DISP=SHR, DSN=IMSVS.SDFSRESL
//          DD DISP=SHR, DSN=IMSVS.PGMLIB
//FPXIN    DD *
REORG     AREA=(DB23AR1, DB23AR2)
PTRCHK    TYPRUN=RPT
          FULLSTEP=YES
          CHKUOW=ALL
/*
//INDD    DD *
AREA=DB23AR1
DD1
DD11
AREA=DB23AR2
DD1
DD11
DD12
DD2
/*

```

Figure 359. Running OER with the segment shunting function

Chapter 24. Printing DMAC blocks in a DEDB online with FPO

You can print DMAC blocks of an online DEDB by using the Online DMAC Print utility (also referred to as ODM) of FPO.

Topics:

- [“Functions of ODM” on page 821](#)
- [“Input and output for ODM” on page 821](#)
- [“Running ODM” on page 822](#)
- [“Region size requirements for ODM” on page 823](#)
- [“DD statements for ODM” on page 823](#)
- [“Option control statement for ODM” on page 824](#)
- [“Utility control statement for ODM” on page 824](#)
- [“Output for ODM” on page 826](#)
- [“Examples for ODM” on page 832](#)

Functions of ODM

The ODM utility reports two kinds of DMAC information; DMAC that is used by the IMS control region in storage and DMAC of available area data sets (physical).

DMAC is an important CI that has the information for ROOT, UOW definition, and SDEP LB/LE values, and so on, for a DEDB area data set. ODM maps Global section and Local section with offset, label, values. It also provides the dump specified by the DMAC length.

When the CIPRT=YES keyword is specified, ODM also provides the second CI information for all available area data sets. The second CI information that provides EQE list, CUSN, and RBA are also reported except for DMAC.

Input and output for ODM

ODM reports the DMAC information.

The following figure shows input and output for ODM.

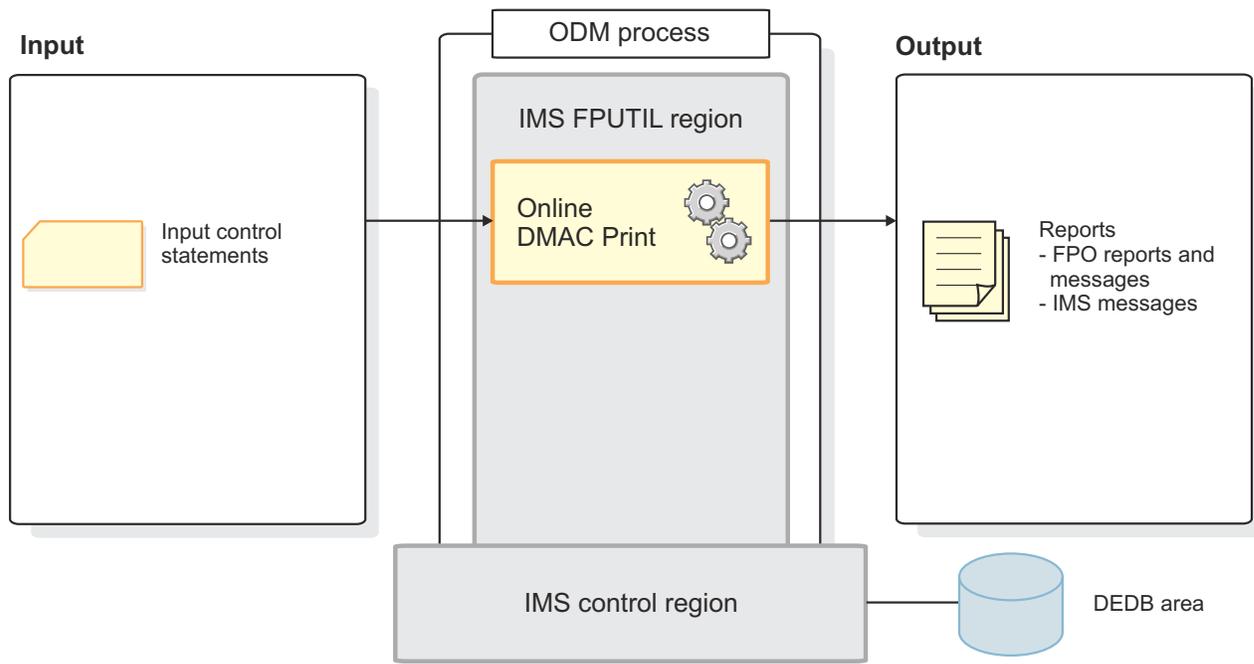


Figure 360. Input and output for ODM

Running ODM

You need to specify an EXEC statement and DD statements that define the input and output data sets in your JCL.

Procedure

1. Specify the EXEC statement in one of the following formats:

```
//name EXEC procname
```

where *procname* is the IMS-supplied IFP procedure FPUTIL or any other IFP utility supporting procedure.

```
//name EXEC PGM=DFSRR00,PARM='IFP,dbdname,DBF#FPU0'
```

where *dbdname* is the DBDNAME associated with the DEDB.

Related reading:

- For information about the FPUTIL utility, see *IMS System Definition* for the version of IMS that you are using.
 - For information about the DEDB online utility commands, see *IMS Database Utilities* for the version of IMS that you are using.
2. Specify the DD statements to define input data sets and output data sets.
 3. Run the JCL.

Related reference

Examples for ODM

There are many ways to run the ODM utility. The examples provided in the following topics show some of the typical ways to run the ODM utility.

[Tips and best practices for the FPO Online DMAC Print utility](#)

The FPO Online DMAC Print utility (ODM) supports a keyword that further simplifies management of DEDBs.

Region size requirements for ODM

The region size for ODM depends on the size of a UOW, and it does not depend on the keyword specifications.

Because ODM operates in a virtual storage region as an IMS IFP utility, the region size required to run ODM is larger than the following value.

ODM requires the work buffer calculated by the following formula:

```
Work buffer: CISZ
```

DD statements for ODM

DD statements for ODM determine the input and output data sets and specify how to run ODM.

The following table summarizes the DD statements for ODM.

Table 107. DD statements for ODM

DDNAME	Use	Format	Required or optional
JOBLIB/STEPLIB	Input		Required
FPXOPTS	Input	LRECL=80	Optional
FPXIN	Input	LRECL=80	Required
FPXMFIL	Output	LRECL=133	Optional
FPXRPTS	Output	LRECL=133	Optional
SYSPRINT	Output	LRECL=121	Required

STEPLIB DD

This statement defines the library that contains the ODM utility.

The IMS Tools Base library (SHKTLOAD) is required if you specify the ITKBSRVR keyword.

FPXOPTS DD

Defines the input data set that contains the option control statements for the process. FPXOPTS usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte fixed-length records. Block size, if coded, must be a multiple of 80. It can be specified as DUMMY, NULLFILE, or an empty file.

For details of the option control statements, see [“Option control statement for ODM”](#) on page 824.

FPXIN DD

Defines control statements containing optional user requests for ODM. FPXIN usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte fixed-length records. Block size, if coded, must be a multiple of 80.

For details of control statements, see [“Utility control statement for ODM”](#) on page 824.

FPXMFIL DD

Defines the default message output. The output can go to a data set or to a printer, routed through the output stream. If the DD statement is specified explicitly or implicitly by the MFILE keyword of the DMACPRT control statement, and the DD statement for the specified DD name is not coded in the JCL stream, the DD is allocated dynamically to the system output stream with SYSOUT=*. If the DD name is overridden by the MFILE keyword with a different DD name, the DD statement for the specified DD name must be coded in the JCL stream.

FPXRPTS DD

Defines the default report output. The output can go to a data set or to a printer, routed through the output stream. If the DD statement is specified explicitly or implicitly by the RFILE keyword of the DMACPRT statement and the DD statement for the specified DD name is not coded in the JCL stream, the DD is allocated dynamically to the system output stream with SYSOUT=*

SYSPRINT DD

Defines the system message output.

Warning: You cannot allocate the ODM SYSPRINT DD output stream to a data set because SYSPRINT DD is also used by the IMS system. If you specify SYSPRINT DD as a data set, some output might be overridden or unpredictable open or close errors might occur. Therefore, you must specify SYSPRINT DD as SYSOUT.

Option control statement for ODM

The option control statement defines the user description of the OVERRIDE control statement, which specifies information for setting temporary override values of product message defaults.

The option control statements must start with a line that contains the OPTIONS command. The control statement that can be specified is:

OVERRIDE control statement

The OVERRIDE control statement specifies information for setting temporary override values of product message defaults. This control statement is optional. The OVERRIDE control statement can be specified in FPXOPTS DD.

OVERRIDE statement keywords

The keywords associated with the OVERRIDE statement are not available for site default options. For the format of the OVERRIDE statement, see [“OVERRIDE control statement” on page 850](#).

For the OVERRIDE statement, you can specify the keywords that are listed in the following table.

Table 108. OVERRIDE statement keywords

Keyword	Required or optional	Description	Topic
MSG_LIM	Optional	Specifies the maximum number of repetitive messages that are to be issued before suppression.	“MSG_LIM keyword” on page 897
MSG_NUM	Required	Specifies the message number that is to be suppressed.	“MSG_NUM keyword” on page 898

Related reference

[Keyword reference for FPO](#)

These topics explain the control statements and keywords of FPO and their syntax rules.

[Syntax rules for FPXOPTS](#)

FPXOPTS control statements and their associated keywords can be coded in free format in columns 1 through 72, subject to certain syntax rules.

Utility control statement for ODM

The utility control statements for ODM define the user description of the GLOBAL control statement and the DMACPRT statement, both of which control the ODM process.

All control statement keywords that can be specified to run an ODM job are explained in this topic.

The control statements that can be specified are:

GLOBAL control statement

The GLOBAL control statement defines the job step keyword values. If specified, GLOBAL must be the first command in the input stream. The GLOBAL control statement can be specified only in FPXIN DD.

DMACPRT control statement

The DMACPRT control statement specifies control information that affects how the ODM run is processed. This control statement is mandatory. The control statements must start with a line that contains the DMACPRT statement.

ODM allows you to specify site default options.

Subsections:

- [“GLOBAL statement keywords” on page 825](#)
- [“DMACPRT statement keywords” on page 825](#)

GLOBAL statement keywords

The GLOBAL control statement specifies the job step keyword values. This control statement is optional.

The keywords associated with the GLOBAL statement are available for site default options.

For the GLOBAL statement, you can specify the keywords that are listed in the following table. For the format of the GLOBAL statement, see [“GLOBAL control statement” on page 847](#).

Table 109. GLOBAL statement keywords

Keyword	Required or optional	Description	Topic
DBRC	Optional	Specifies whether to store reports that are associated with a RECON to the IMS Tools KB repository.	“DBRC keyword (for GLOBAL)” on page 868
ITKBLOAD	Optional	Specifies the name of the load module library for IMS Tools KB.	“ITKBLOAD keyword” on page 890
ITKBSRVR	Optional	Specifies the name of the IMS Tools KB server XCF group.	“ITKBSRVR keyword” on page 891
KEYWORD_DUPCHK	Optional	Specifies to end the job with an error message when a duplicate keyword is found.	KEYWORD_ DUPCHK keyword

DMACPRT statement keywords

The keywords associated with the DMACPRT statement are available for site default options except for the AREA keyword.

For the DMACPRT statement, you can specify the keyword that is listed in the following table. For the format of the DMACPRT statement, see [“DMACPRT control statement” on page 841](#).

Table 110. DMACPRT statement keywords

Keyword	Required or optional	Description	Topic
AREA	Required	Specifies the name of the area where DMAC in storage and the second CI of the active ADSs are printed.	“AREA keyword” on page 864
CIPRT	Optional	Specifies whether the second CI of available ADS(s) is printed or not.	“CIPRT keyword” on page 867
MFILE	Optional	Specifies the DD name of the default report output.	“MFILE keyword” on page 896
PCHKONLY	Optional	Specifies whether to evaluate only the control statement keywords or to run the functions that are designated by the control statements and their parameters.	“PCHKONLY keyword” on page 901
PROC_AFT_ERROR	Optional	Specifies whether to end the job step when an area-specific error is encountered or to continue processing subsequent areas.	PROC_AFT_ERROR keyword
RFILE	Optional	Specifies the default report output.	“RFILE keyword” on page 908

Related reference

[Keyword reference for FPO](#)

These topics explain the control statements and keywords of FPO and their syntax rules.

[Syntax rules for FPXIN](#)

FPXIN control statements and their associated keywords can be coded in free format in columns 1 through 72, subject to certain syntax rules.

Related information

[Site default options support for FPO](#)

The site default options are control statements that are specified in a member of a partitioned data set and that are used to override the system default option values, which can be changed without assembling and link-editing.

Output for ODM

The following topics describe the outputs that are generated by ODM.

FPXRPTS DD data set

The DMACPRT report and the Process Summary report are produced in the data set or written to the output stream that is determined by the RFILE keyword of the DMACPRT control statement.

Subsections:

- [“DMACPRT report” on page 826](#)
- [“Process Summary report” on page 830](#)

DMACPRT report

This report consists of the following sections:

- The following DMACs that are in use by the IMS control region:
 - DMAC mapping
 - DMAC dump
- The second CI of the available area data sets. The following sections are generated by the CIPRT=YES keyword:
 - DMAC mapping
 - DMAC dump

These sections are produced in the data set or written to the output stream that is specified by the RFILE keyword of the DMACPRT control statement. When CIPRT=YES on the DMACPRT control statement is specified, ODM also generates the DMACPRT report for all available area data sets (ADSs).

Area Information section

This section shows the common area information statistics. The area information for 2ND CI (physical) and DMAC (IMS storage) are reported and the differences between them are shown with '*'. The second CI of the first available data set with no EQE is used for 2ND CI (physical). If all available area data sets have EQEs, the second CI of the first available area data set is used for 2ND CI (physical).

The following figure is an example of area information.

DMAC MAPPING FROM DATABASE: DEDBJN23

- AREA NO: 1 AREANAME: DB23AR1 DDNAME: DB23AR11 DSNAME: HPFP.MDB23AR1.ADSFP

- AREA INFORMATION

	2ND CI	DMAC (IMS Storage)	*: DIFFERENCE DETECTED
- IMS VERSION	141	141	
- DBDNAME	DEDBJN23	DEDBJN23	
- AREA NAME	DB23AR1	DB23AR1	
- CI SIZE	1024	1024	
- ROOT	(95,30)	(95,30)	
- UOW	(5,2)	(5,2)	
- BASE			
- NUMBER OF CI'S			
- TOTAL CI'S	195	195	
- DOVF			
- NUMBER OF CI'S			
- TOTAL CI'S	130	130	
- IOVF			
- NUMBER OF CI'S			
- TOTAL CI'S	148	148	
- SDEP			
- NUMBER OF CI'S			
- TOTAL CI'S	508	508	
- START BLOCK#	482	482	
- RANGE			
- FROM	X'00078800'	X'00078800'	
- TO	X'000F7800'	X'000F7800'	
- LOGICAL BEGIN			
- CYCLE COUNT	X'00000001'	X'00000001'	
- RBA	X'00078800'	X'00078800'	
- LOGICAL END			
- CYCLE COUNT	X'00000001'	X'00000001'	
- RBA	X'0007EC00'	X'0007EC00'	
- LOGICAL BEGIN TS	X'D194CC50C010E341'	X'D194CC50C010E341'	

Figure 361. Example of an area information

DMAC Mapping section

This section shows the DMAC information by using the IMS mapping macro. The DMACPRT report shows both the global section and the local section in DMAC. OFFSET shows the offset from the second CI top address. LABEL shows the name of the IMS DBFDMAC macro. VALUE shows the value of the data. COMMENT shows what the data that is quoted from IMS DBFDMAC macro is.

The following figure is an example of the global section of DMAC mapping.

- AREA NO: 1 AREANAME: DB23AR1

- DMAC INFORMATION : DMAC GLOBAL SECTION in IMS storage

OFFSET	LABEL	VALUE	COMMENT
000000	DMACVNO	141	VERSION NUMBER
000003	DMACLKID	8	AREA LOCK ID
			* DMACLKCD C'8' : AREA LOCK ID FOR AREA
000004	DMACDBNM	DEDBJN23	DATA BASE NAME
00000C	DMACNAME	DB23AR1	AREANAME FOR AREA
000014	DMACDBRC	X'03'	DBRC USAGE INDICATOR
			DMACREGR X'80' : AREA REGISTERED WITH DBRC
			DMACWDUI X'40' : DBRC USAGE INDICATOR CHANGED
			DMACSDWRA.. X'08' : ALL LOCAL 2ND TIME.
			(DMACSDWRAPLCL2)
			DMACSDWRA.. X'04' : ALL LOCAL 1ST TIME.
			(DMACSDWRAPLCL1)
			* DMACSDW2 X'02' : COMPAT WRAP 2ND TIME.
			* DMACSDW1 X'01' : COMPAT WRAP 1ST TIME.
000015	DMACDUI	X'11157F0659331F'	DBRC USAGE INDICATOR
00001C	DMACYSI		IMS SUBSYSTEM ID
000024	DMACOCNT	148 X'00000094'	COUNT OF UNUSED INDEPENDENT OVERFLOW CI'S
000028	DMACCUSN	X'0005'	DMAC CUSN
00002A	DMACFLG4	X'88'	FLAG BITS...
			* DMACF4SH X'80' : AREA HAS A SHARABILITY
			DMACF4CC X'40' : DEDB CONVERSION COMPLETED
			DMACF4FC X'20' : FORCE CLOSE DUE TO IRLM FAIL
			DMACF4LR X'10' : LOCK REJECT AT COMMIT-ROLB AND RETRY THE LOCK A
			* DMACF4SD X'08' : AREA HAS SDEP SEGMENT
			DMACF4CS X'04' : AREA HAS A CACHED DEVICE WHICH RECEIVED AN I/O
			DMACCHG X'02' : OLC AREA LEVEL CHANGE
			DMACF4RC X'01' : MAD5 IOT RECOVERY STARTED
00002B	DMACUTIL	X'A0'	UTILITY CONNECT FLAG
			* DMACUTON X'80' : UTILITY ACTIVE ON THIS AREA
			DMACHSSP X'40' : HSSP ACTIVE ON THIS AREA
			* DMACUOWL X'20' : UOW LOCK MODE ON THIS AREA
			DMACPBPU X'10' : PRIVATE BUFFER POOL USER
			DMACRDAH X'08' : HS READ AHEAD
			DMACUTEH X'04' : EXIT EXTEND IN CONTROL
00002C	DMACFBAD	990 X'000003DE'	BLOCK NUMBER OF FIRST BLOCK WHICH WOULD BE BEYOND THE SEQUENTIAL PART.
.....			
.....			
.....			

Figure 362. Example of a DMAC mapping (DMAC global section)

The following figure is an example of the local section of DMAC mapping.

- DMAC INFORMATION : DMAC LOCAL SECTION in IMS storage

OFFSET	LABEL	VALUE	COMMENT
00009C	DMACFLG1	X'01'	FLAG BITS...
			DMACF1AF X'80' : SEQUENTIAL DEP. PART FULL
			DMACF1EP X'40' : I/O ERROR
			DMACF1ST X'20' : AREA STOP REQUEST
			DMACF1RE X'10' : AREA RESTART REQUEST
			DMACF1ER X'08' : ERROR RECOVERY NEEDED.
			DMACF1CP X'04' : WAIT FOR W-M-C
			DMACF1BK X'02' : TEMPORARY BIT FOR BACKOUT.
			* DMACF10P X'01' : AREA IS OPEN
00009D	DMACFLG3	X'00'	FLAG BITS...
			DMACF3SP X'80' : AREA CLOSE IN PROCESS
			DMACNTFY X'40' : NOTIFY HAS BEEN SENT TO DBRC
			DMACF3ER X'20' : DMAC LOG EXIST DURING ERE
			DMACF3RN X'10' : AREA RECOVERY NEEDED REQUEST
			DMACF3CR X'08' : CREATE IS IN PROCESS SDEP
			DMACARCB X'04' : AVA. PHASE REACH BIT
			DMACANYC X'02' : AVA. PHASE NOT YET COMPLETED
			DMACF3AS X'01' : ADS STOP IN PROCESS
00009E	DMACFLG6	X'01'	FLAG BYTE 6
			DMACF6FU X'80' : FORCE DMAC SYNC 2ND CI UPD.
			DMACF6RV X'40' : CLOSE BY IRLM REVERIFICATION
			DMACF60P X'20' : AOCE REOPEN PENDING
			DMACF6WT X'10' : MFLG WAITING FOR WTCT
			DMACF6IP X'08' : DMAC INTERNAL STOP REQUEST
			DMACF6CS X'04' : DMAC CTL BLK SYNC REQUEST
			DMACF6RQ X'02' : LB RECOVERY REQUEST
			* DMACF6UP X'01' : ACCESS INTENT IS UP/EX
00009F	DMACSLV	X'03'	DEDB SHARE LEVEL
			DMACSLV0 X'00' : NO SHARE
			DMACSLV1 X'01' : SHARE AT AREA LEVEL
			DMACSLV2 X'02' : SHARE AT BLOCK LEVEL(LOCAL)
.....			
.....			
.....			

Figure 363. Example of a DMAC mapping (DMAC local section)

DMAC Dump section

This section shows the DMAC dump having the DMACLEN length. The length depends on the version of the IMS system. When the report is for the available area data sets, EQE list, CUSN, and RBA are also reported.

If ODM ended successfully without errors, this column is blank. Otherwise, one of the following messages is displayed in this column:

Processing ended with warnings.

This message indicates that the processing ended with return code of 4. No pointer errors were detected.

Processing ended with errors.

This message indicates that the processing ended with return code of 8.

Pointer errors were detected.

This message indicates that the processing ended with return code of 4. One or more pointer errors were detected.

The following areas not processed because PROC_AFT_ERROR=STOP.

This message indicates that the processing ended with errors in the previous area. This area and the subsequent areas are not processed because PROC_AFT_ERROR=STOP is specified. The status for the subsequent areas is blank.

Processing did not complete.

This message indicates that the processing for the area did not complete because the processing ended abnormally, or an error, which prevented from processing the subsequent areas, was encountered. The status for the subsequent areas is blank.

Processing was skipped.

This message indicates that the processing for this area was skipped because the area was stopped. This message is displayed only when AREA=ALL is specified.

For details about these messages, see the messages produced in the FPXMFIL data set or in the SYSPRINT data set.

SYSPRINT DD data set

IMS system messages and control statement analysis messages are produced in the data set or the system output stream that is specified on the SYSPRINT DD statement.

The following figure shows an example of the control statement analysis messages.

```

FPX0178I Online DMACPR process started.
FPX0234I The following control statements were specified in //FPXIN:
FPX0049I 0.....1.....2.....3.....4.....5.....6.....7.....8
          1234567890123456789012345678901234567890123456789012345678901234567890
          DMACPR
            AREA=(areaname1,areaname2)
            PCHKONLY=NO
            MFILE=FPXMFIL
            RFILE=FPXRFILE
FPX0217I Starting keyword analysis.
FPX0208I DMACPR keywords are specified as follows:
          Keyword          Control Card    DB Level    Global Level    System
          -----          -
          CIPRT
          MFILE             FPXMFIL
          PCHKONLY         NO
          RFILE             FPXRFILE
FPX0218I Completed keyword analysis.
FPX0203I ODM processing options in effect:
          DMACPR OPTIONS
          - CIPRT           : YES
          - MFILE           : FPXMFIL
          - PCHKONLY        : NO
          - RFILE           : FPXRFILE

```

Figure 367. Control statement analysis messages

Related reference

[Utility control statement for ODM](#)

The utility control statements for ODM define the user description of the GLOBAL control statement and the DMACPRT statement, both of which control the ODM process.

FPXMFIL DD data set

ODM messages are produced in the FPXMFIL data set.

These messages can be routed to another output data set or the system output stream by coding the MFILE parameter of the DMACPRT control statement.

The following figure shows an example of the ODM messages.

```
FPX0039I Starting Online Fast Path utility processing:
(cont)   AREA name = areaname1 AREA number = *****1
FPX0181I DMACPRT process ended for AREA name = areaname1.
FPX0039I Starting Online Fast Path utility processing:
(cont)   AREA name = areaname2 AREA number = *****2
FPX0181I DMACPRT process ended for AREA name = areaname2.
FPX0179I Online DMACPRT process ended with RC=00.           (Elapsed Time: 00:00:00.58)
```

Figure 368. ODM messages

Examples for ODM

There are many ways to run the ODM utility. The examples provided in the following topics show some of the typical ways to run the ODM utility.

Example 1: Reporting only DMAC information in IMS storage

The following figure shows example JCL for reporting DMAC information that is used by the IMS control region in storage.

```
//*****
//*      DEDB DMACPRT utility
//*****
//ODM1   EXEC FPUTIL,DBD=dbdname,IMSID=imsid
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//       DD DISP=SHR,DSN=IMSVS.SDFSRESL
//MFILE  DD SYSOUT=*
//RFILE  DD SYSOUT=*
//FPXIN  DD *
*=====
* DMACPRT for two areas
*=====
  DMACPRT
    AREA=(areaname1,areaname2)
    MFILE=MFILE
    RFILE=RFILE
    CIPRT=NO
/*
```

Figure 369. Reporting only DMAC information in IMS storage

Example 2: Reporting two kinds of DMAC information

The following figure shows example JCL for reporting DMAC information that is used by the IMS control region in storage, and DMAC information of available area data sets (physical).

```
//*****  
//*      DEDB DMACPRT utility  
//*****  
//ODM2   EXEC FPUTIL,DBD=dbdname,IMSID=imsid  
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0  
//       DD DISP=SHR,DSN=IMSVS.SDFSRESL  
//FPXIN  DD *  
*=====   
* DMACPRT for all areas defined in DB  
*=====   
  DMACPRT  
    AREA=ALL,  
    CIPRT=YES  
/*
```

Figure 370. Reporting two kinds of DMAC information

Chapter 25. Keyword reference for FPO

These topics explain the control statements and keywords of FPO and their syntax rules.

Topics:

- [“Syntax rules” on page 835](#)
- [“FPO control statements” on page 837](#)
- [“FPO keywords ” on page 861](#)

Syntax rules

These reference topics explain the syntax rules for each statement.

Syntax rules for FPXIN

FPXIN control statements and their associated keywords can be coded in free format in columns 1 through 72, subject to certain syntax rules.

- Control statements must start with a line that contains one of the following commands. This line can be preceded by any number of blank lines or comment lines, which can be specified in any order.

```
GLOBAL
ANALYZE
DMACPRT
EXTEND
EXTRACT
REORG
```

Note: The GLOBAL control statement, if specified, must be the first command in the input stream.

- A keyword that is not followed by an equal sign is regarded as a command name.
- Each keyword must have one or multiple associated values, they are connected by an equal sign. The equal sign must not be preceded or followed by blanks, but the value can be followed by one or more blanks.
- A control statement record can include multiple commands or keywords. For OAE, however, a control statement record can include only one command or one keyword.
- Keywords must be delimited by at least one comma or one blank. The end of a line is treated as a blank. Any number of commas and blanks may be specified between adjacent keywords.
- Keywords and their associated values *cannot* span (or be continued on) multiple control statement records except for those that have multiple associated values.
- If the same keyword is specified for a command more than once, the last one coded is used unless KEYWORD_DUPCHK=YES is specified.
- If multiple values are specified for a keyword, they must be enclosed by parentheses. The left parenthesis must immediately follow the equal sign. Each pair of adjacent values specified within the parentheses is separated by at least one comma or one blank. The values can be specified on multiple lines. If a comma is used to separate two adjacent values, it can be preceded or followed by any number of blanks, but no comment line can be specified within the parentheses. For ODE, only a comma can be used to separate two adjacent values for the SEG keyword, the FIELD keyword, and the ANDFIELD keyword. For OAE, only the values of the VOLUME keyword can be specified on multiple lines.
- A command and the first associated keyword are separated by at least one blank, any number of comment lines, or blank lines.
- A command and a keyword can start in any column.
- A comment must start with * in column 1.

Unless explicitly specified in the description of a particular control statement, none of the control statements or keywords are case-sensitive.

Syntax rules for DASYSIN

Keywords and the associated values can be coded in free format (columns 1 - 72), subject to certain syntax rules.

- The keyword and its value must be on one control statement.
- If specification of a keyword value is required, the keyword must be separated from its associated value by an equal sign. The equal sign must not be preceded by blanks, but can be followed by one or more blanks. The value must be separated from the next keyword by a blank, a comma, or a comma followed by one or more blanks.
- For keywords that do not have associated values, the keyword must be separated from the next keyword by a blank, a comma, or a comma followed by one or more blanks.
- In case of duplicate keywords, the last one coded is used.
- The value associated with the keyword can be specified in its entirety or only with its leading characters, so as to make it uniquely identifiable. For instance, in the case of the values YES and NO, Y and N are sufficient.

Syntax rules for FPXOPTS

FPXOPTS control statements and their associated keywords can be coded in free format in columns 1 through 72, subject to certain syntax rules.

- Control statements must start with a line that contains OPTIONS.
- A keyword that is not followed by an equal sign is regarded as a command name.
- Each keyword must have one value, they are connected by an equal sign. The equal sign must not be preceded or followed by blanks, but the value can be followed by one or more blanks.
- A control statement record can include one command and multiple keywords.
- Keywords must be delimited by at least one comma or one blank. The end of a line is treated as a blank. Any number of commas and blanks may be specified between adjacent keywords.
- Keywords and their associated values cannot span (or be continued on) multiple control statement records.
- Duplicate specification of a keyword or value is not allowed.
- A command and the first associated keyword are separated by at least one blank, any number of comment lines, or blank lines.
- A command and a keyword can start in any column.
- A comment must start with * in column 1.
- The control statements cannot be specified in lower case.

Syntax rules for FPXCTL and *areaname*

Control statements and their associated keywords are free form and are coded in columns 1 through 80, subject to certain syntax rules.

Control statements are 80-byte, fixed-length records.

- The full 80 characters can be used to define ODE statements, but optional sequence numbers (they must be numeric) are allowed in columns 73 - 80. This can be varied on a card-by-card basis.
- All control statements must start in column 1. Each control statement must have at least 1 keyword associated with it. Keywords and continuations of keywords can start in any column.
- Control statements and the first associated keyword are separated by at least 1 blank.
- Keywords are separated from their parameter values by equal (=) signs.

- Multiple keywords are separated by one of the following characters:
 - A comma
 - One or more blanks
 - A comma and one or more blanks
- All control statements can span (or be continued on) multiple control statement records. To continue a control statement onto the next record, the last non-blank character must be a comma (,).
- Keywords and their associated values CANNOT span (or be continued on) multiple control statement records.

Unless explicitly specified in the description of a particular control statement, none of the control statements or keywords are case-sensitive.

The image of all records that are read (including control statements, associated keywords, and comments) is written to the file identified by the SYSOUT DD statement.

Syntax rules for the OAE Rename program

The keywords and their associated parameter values for the OAE Rename program can be coded in a free format (columns 1-71), subject to certain syntax rules.

- Control statements are coded on 80-byte records.
- One control statement has only one keyword parameter.
- All control statement specifications must start in column 1.
- The keyword must be separated from its associated value by an equal sign. The equal sign must not be preceded by blanks.
- The keyword must have an associated value.
- '*' in column 1 is treated as a comment control statement.

FPO control statements

The following topics explain the FPO control statements.

ANALYZE control statement

The ANALYZE statement specifies information that control how OPC runs.

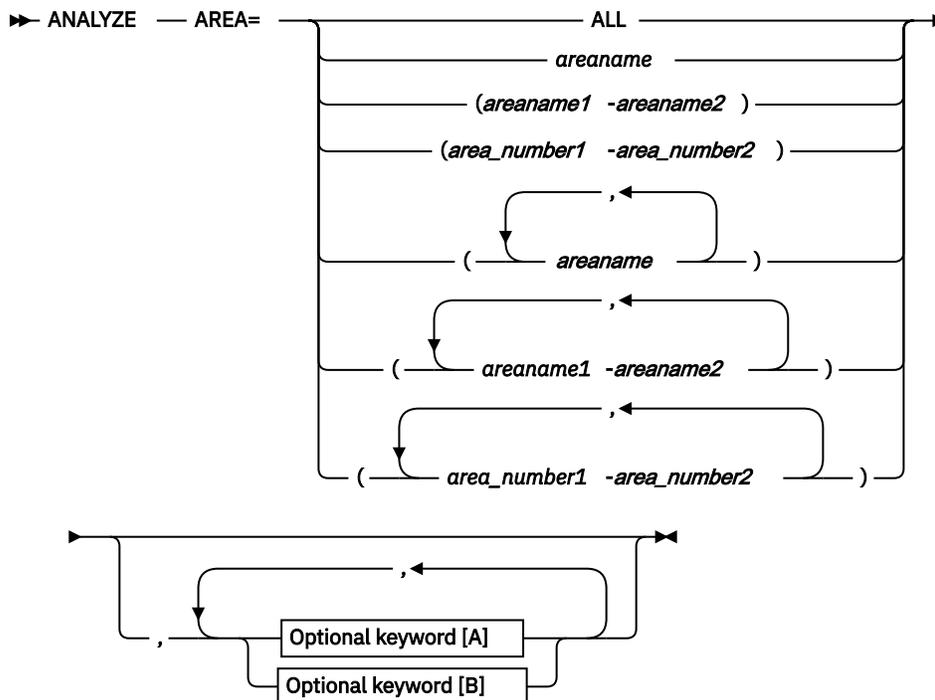
In FPXIN DD, the ANALYZE statement must start with the ANALYZE command, whereas in DASYSIN DD, the ANALYZE statement must not start with the ANALYZE command.

Related keywords:

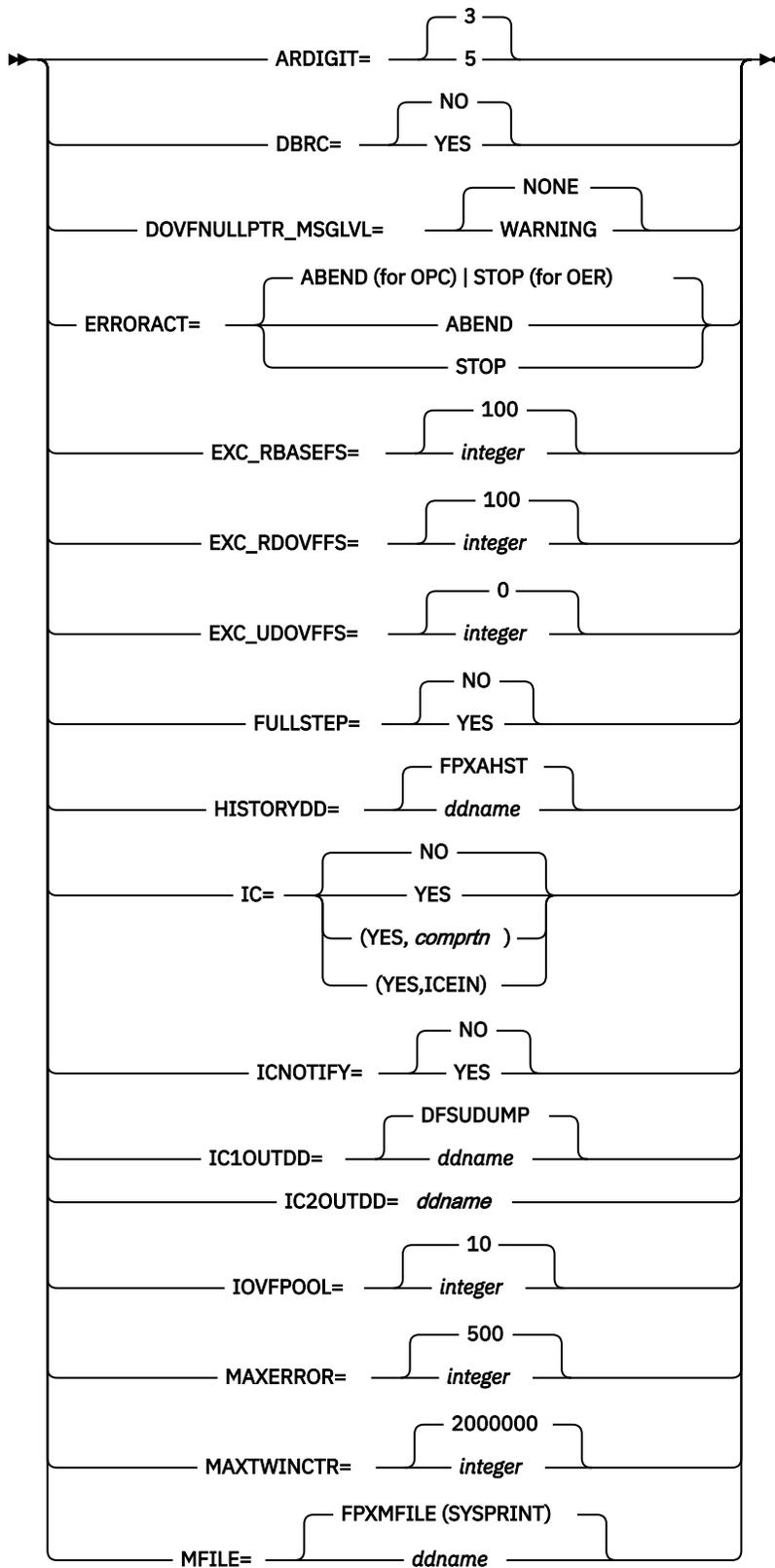
- [“ARDIGIT keyword” on page 863](#)
- [“AREA keyword” on page 864](#)
- [“DBRC keyword \(for GLOBAL\)” on page 868](#)
- [“DOVNULLPTR_MSGLVL keyword ” on page 869](#)
- [“ERRORACT keyword” on page 870](#)
- [“EXC_RBASEFS keyword” on page 872](#)
- [“EXC_RDOVFFS keyword” on page 873](#)
- [“EXC_UDOVFFS keyword” on page 873](#)
- [“FULLSTEP keyword” on page 882](#)
- [“HISTORYDD keyword” on page 882](#)
- [“IC keyword” on page 883](#)
- [“ICNOTIFY keyword” on page 885](#)

- [“IC1OUTDD keyword” on page 886](#)
- [“IC2OUTDD keyword” on page 886](#)
- [“IOVFPPOOL keyword” on page 888](#)
- [“MAXERROR keyword” on page 894](#)
- [“MAXTWINCTR keyword” on page 895](#)
- [“MFILE keyword” on page 896](#)
- [“PCHKONLY keyword” on page 901](#)
- [“PROC_AFT_ERROR keyword” on page 902](#)
- [“RCEXLOAD keyword” on page 904](#)
- [“RFILE keyword” on page 908](#)
- [“RGROUP keyword” on page 910](#)
- [“SDEP keyword” on page 912](#)
- [“SDEP_ORPHAN_MSGLVL keyword” on page 914](#)
- [“SDEP_PTRIOVF_MSGLVL keyword” on page 915](#)
- [“SENSOR keyword” on page 922](#)
- [“SENSOR_DBREC keyword” on page 923](#)
- [“SENSOR_PTRERR keyword” on page 924](#)
- [“SENSOR_UOW keyword” on page 925](#)
- [“SENSOR_UOW_REPORT keyword” on page 925](#)
- [“STAT keyword” on page 928](#)
- [“SYNTBL keyword” on page 932](#)
- [“TARDSNMASK keyword” on page 932](#)
- [“TARECGEN keyword” on page 933](#)
- [“TYPRUN keyword” on page 935](#)

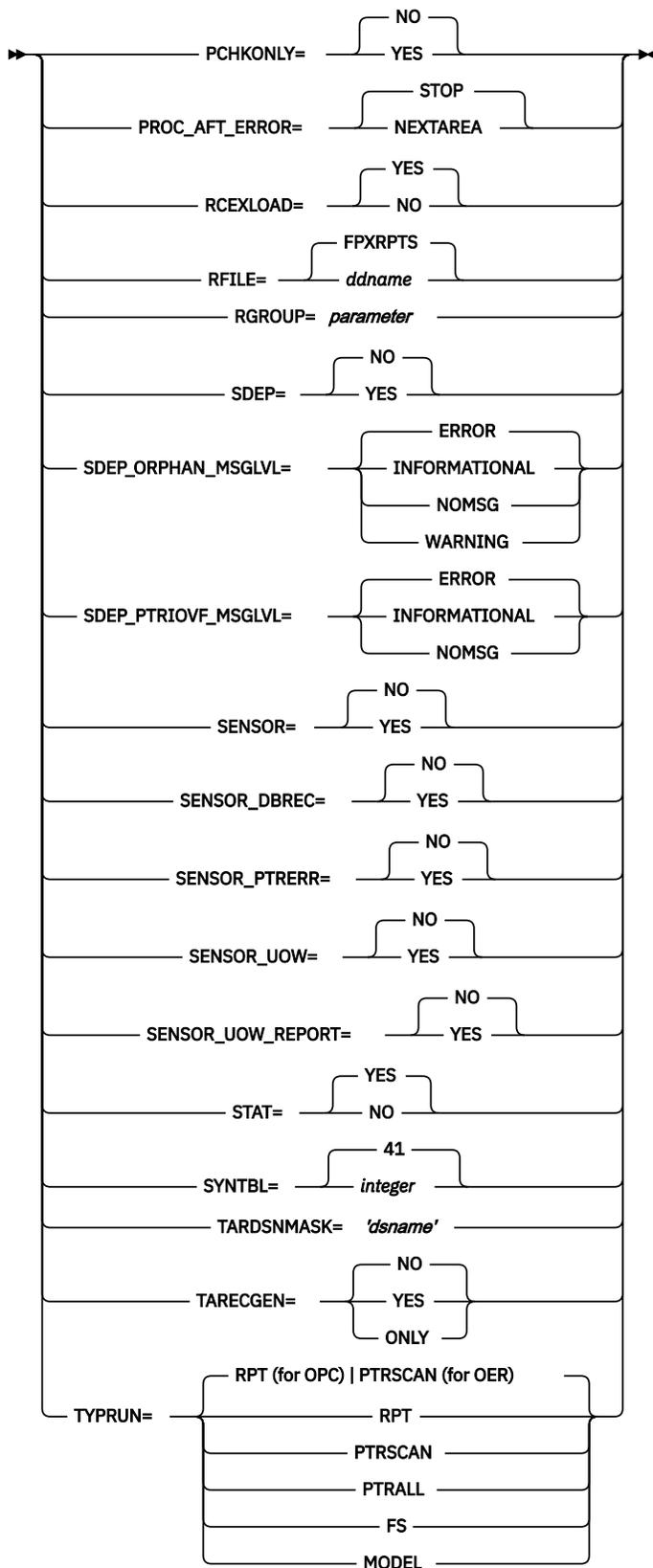
Syntax:



Optional keywords [A]:



Optional keywords [B]:



The keywords associated with the ANALYZE command are available for site default options.

Related information

[Site default options support for FPO](#)

The site default options are control statements that are specified in a member of a partitioned data set and that are used to override the system default option values, which can be changed without assembling and link-editing.

DMACPRT control statement

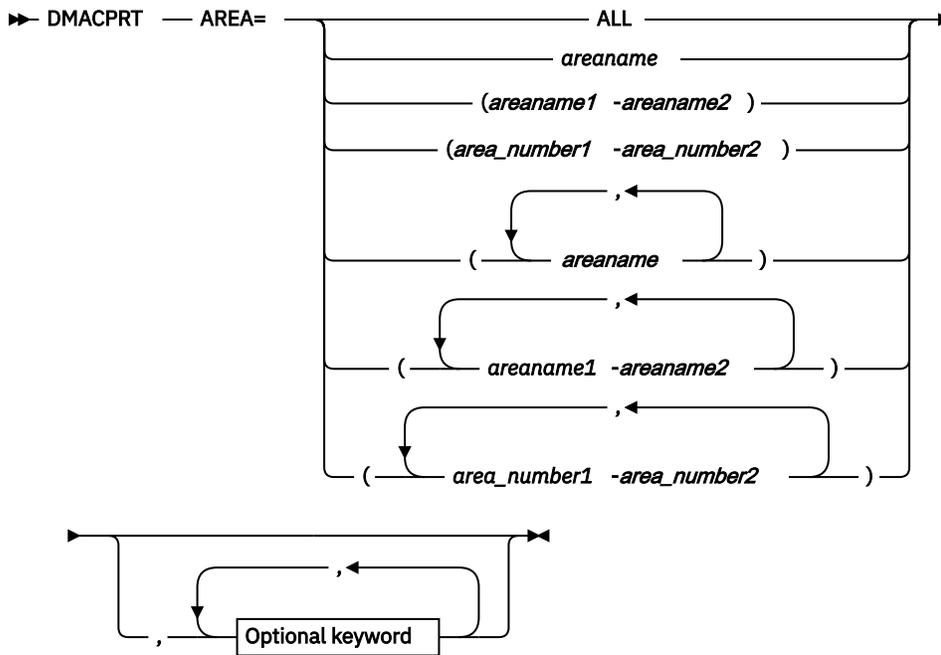
The DMACPRT statement specifies control information that affects how the ODM run is processed.

DMACPRT is a required statement for ODM.

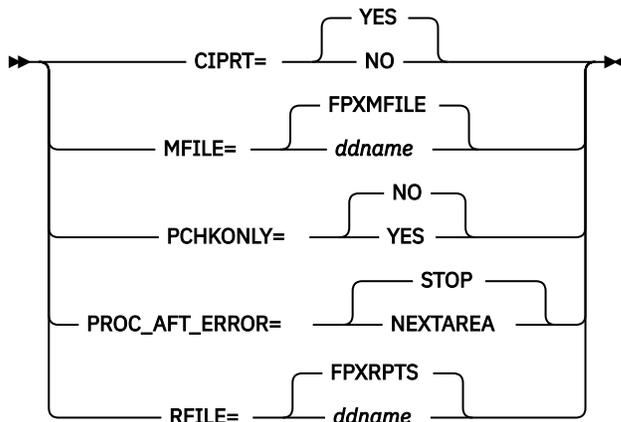
Related keywords:

- [“AREA keyword” on page 864](#)
- [“CIPRT keyword” on page 867](#)
- [“MFILE keyword” on page 896](#)
- [“PCHKONLY keyword” on page 901](#)
- [“PROC_AFT_ERROR keyword” on page 902](#)
- [“RFILE keyword” on page 908](#)

Syntax:



Optional keywords:



Alias:

PRINT_DMACH

The keywords associated with the DMACPRT statement are available for site default options.

Related information

Site default options support for FPO

The site default options are control statements that are specified in a member of a partitioned data set and that are used to override the system default option values, which can be changed without assembling and link-editing.

EXTEND control statement

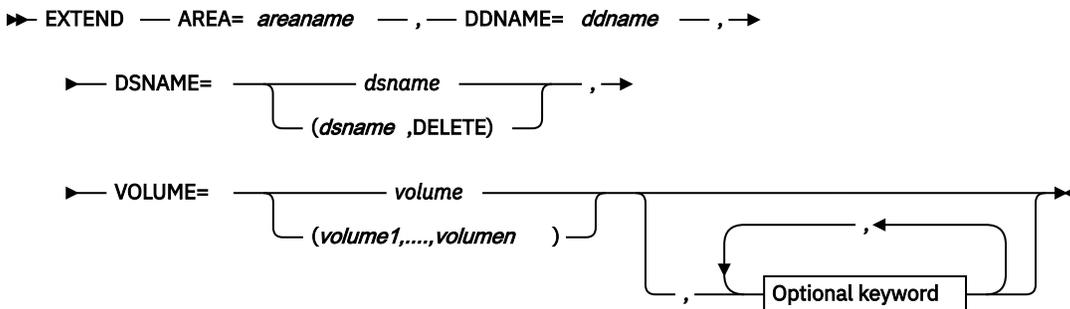
The EXTEND statement specifies information that controls how the OAE run is to be processed.

EXTEND is a required statement for OAE with original expansion.

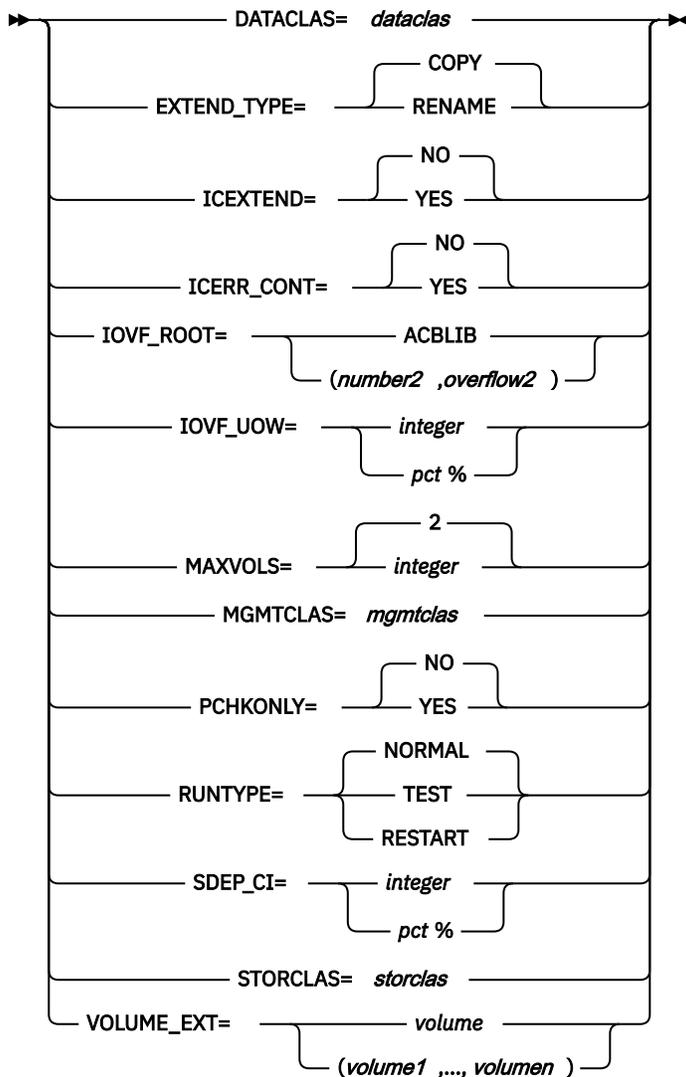
Related keywords:

- [“AREA keyword” on page 864](#)
- [“DATACLAS keyword” on page 867](#)
- [“DDNAME keyword” on page 869](#)
- [“DSNAME keyword” on page 870](#)
- [“EXTEND_TYPE keyword” on page 875](#)
- [“ICEXTEND keyword” on page 884](#)
- [“ICERR_CONT keyword” on page 884](#)
- [“IOVF_ROOT keyword” on page 889](#)
- [“IOVF_UOW keyword” on page 889](#)
- [“MAXVOLS keyword” on page 896](#)
- [“MGMTCLAS keyword” on page 897](#)
- [“PCHKONLY keyword” on page 901](#)
- [“RUNTYPE keyword” on page 911](#)
- [“SDEP_CI keyword” on page 913](#)
- [“STORCLAS keyword” on page 930](#)
- [“VOLUME keyword” on page 939](#)
- [“VOLUME_EXT keyword” on page 939](#)

Syntax:



Optional keywords:



The keywords associated with the `EXTEND` statement are not available for site default options.

EXTRACT control statement

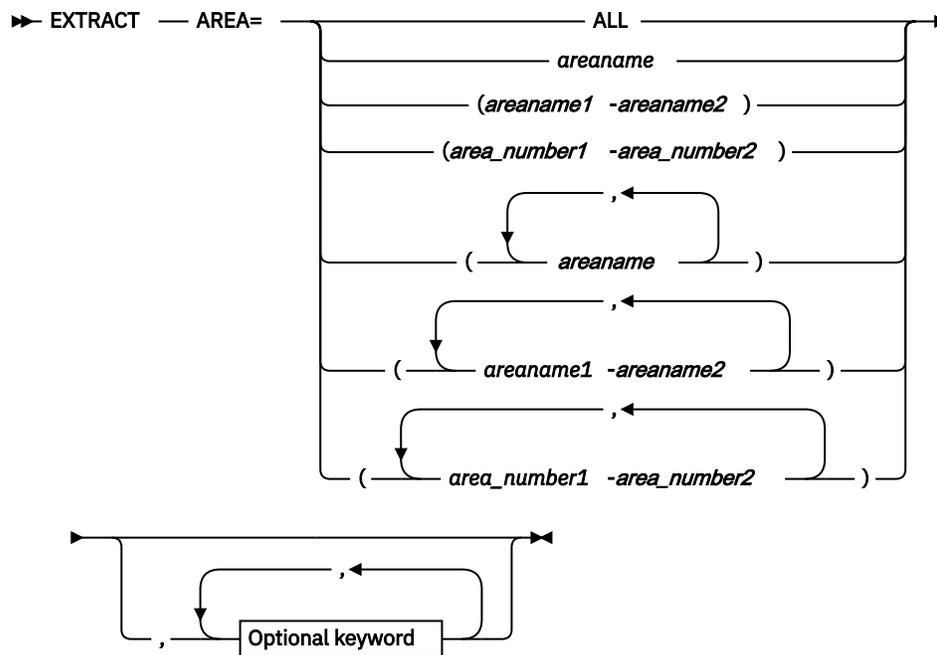
The `EXTRACT` statement specifies the areas to be extracted.

In `FPXIN DD`, the `EXTRACT` statement must start with the `EXTRACT` command, whereas in `FPXCTL` or `areaname DD`, the `EXTRACT` statement must not be specified.

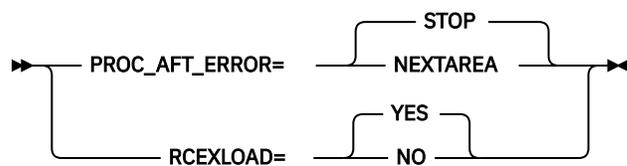
Related keywords:

- [“AREA keyword” on page 864](#)
- [“PROC_AFT_ERROR keyword” on page 902](#)
- [“RCEXLOAD keyword” on page 904](#)

Syntax:



Optional keywords:



Examples of EXTRACT control statement

In the following example, area DB23AR0 is to be extracted.

```
EXTRACT AREA=DB23AR0
```

In the following example, areas DB23AR0, DB23AR1, and DB23AR2 are to be extracted.

```
EXTRACT AREA=(DB23AR0,DB23AR1,  
DB23AR2)
```

In the following example, all areas that are defined in the DEDB are to be extracted.

```
EXTRACT AREA=ALL
```

FPXCTL control statement

The FPXCTL control statement specifies control information that influences how the ODE run is to be processed.

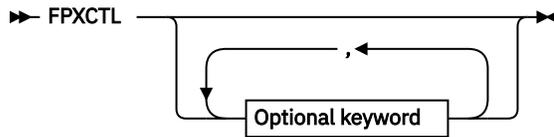
FPXCTL is an optional statement for ODE. If omitted, the indicated defaults are taken for each of the keywords.

Related keywords:

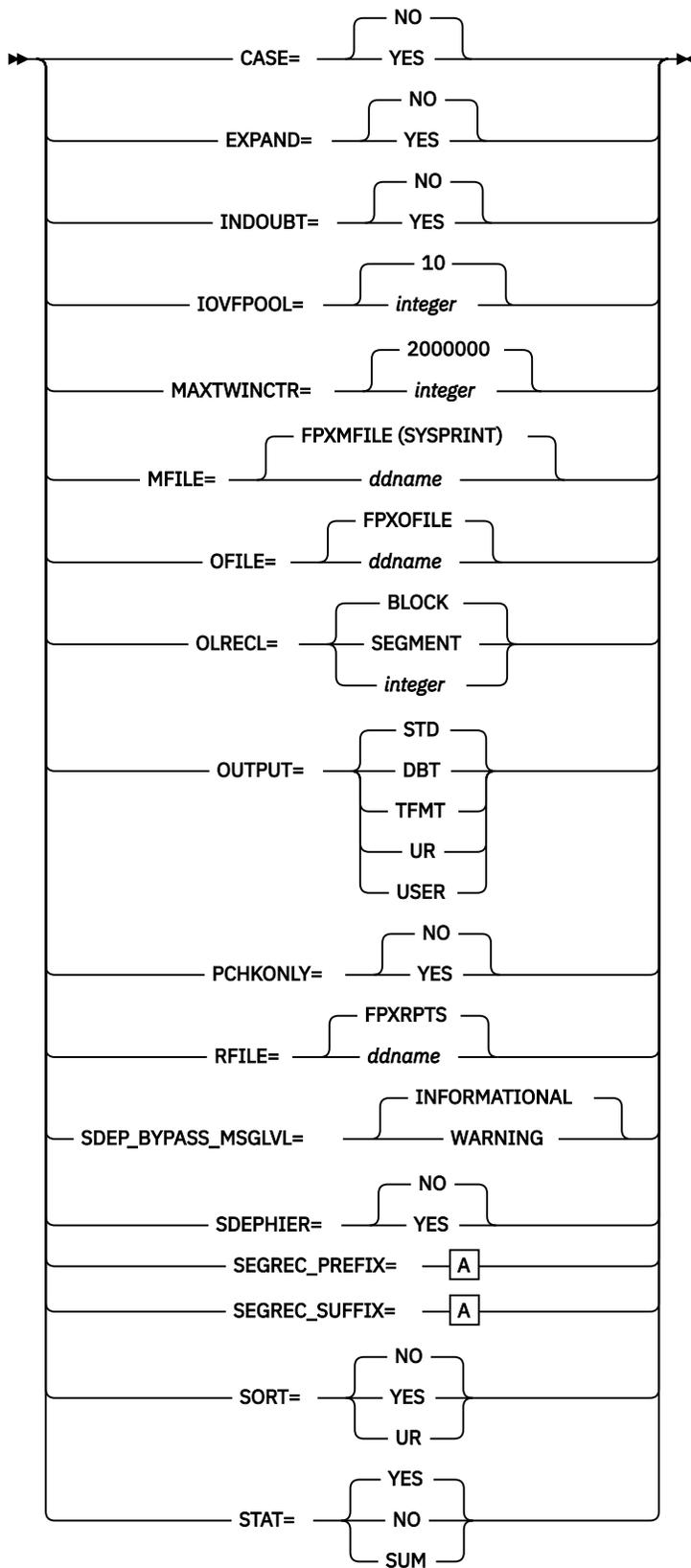
- [“CASE keyword” on page 866](#)
- [“EXPAND keyword” on page 874](#)

- [“INDOUBT keyword” on page 887](#)
- [“IOVFPOOL keyword” on page 888](#)
- [“MAXTWINCTR keyword” on page 895](#)
- [“MFILE keyword” on page 896](#)
- [“OFILE keyword” on page 898](#)
- [“OLRECL keyword” on page 899](#)
- [“OUTPUT keyword” on page 899](#)
- [“PCHKONLY keyword” on page 901](#)
- [“RFILE keyword” on page 908](#)
- [“SDEP_BYPASS_MSGLVL keyword” on page 912](#)
- [“SDEPHIER keyword” on page 913](#)
- [“SEGREC_PREFIX keyword” on page 919](#)
- [“SEGREC_SUFFIX keyword” on page 920](#)
- [“SORT keyword” on page 927](#)
- [“STAT keyword” on page 928](#)

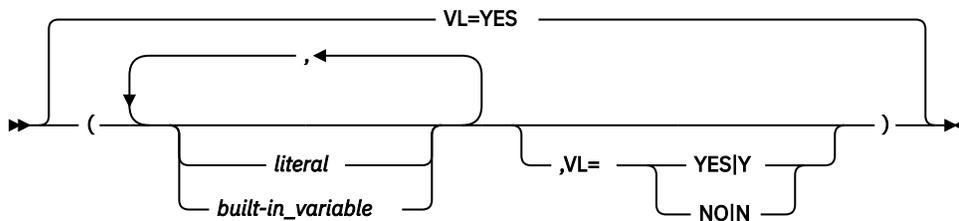
Syntax:



Optional keywords:



A:



The keywords associated with the FPXCTL statement are available for site default options except for SEGREC_PREFIX and SEGREC_SUFFIX.

Examples of FPXCTL control statement

In the following example, the standard output format is used. Defaults are taken for the SORT, EXPAND, IOVFPOOL, OFILE, MFILE, INDOUBT, STAT and CASE keywords.

```
FPXCTL OUTPUT=STD
```

In the following example, the standard output format is used. The records are sorted in ascending root key sequence. 50 IOVF buffers per buffer set are used. The output records are written to a file with the DD name AREA1FLE. The utility report is written to a file with the DD name AREARPT.

```
FPXCTL OUTPUT=STD,
SORT=YES,
EXPAND=NO,
IOVFPOOL=50,
OFILE=AREA1FLE,
MFILE=AREARPT
```

In the following example, the IMS DBT Unload/Reload output format is used. Compressed records are expanded. The output records are written to a file with the DD name FPXOFILE (the default). The utility report is written to a file with the DD name SYSPRINT (the default).

```
FPXCTL OUTPUT=DBT,
EXPAND=YES
```

Related information

Site default options support for FPO

The site default options are control statements that are specified in a member of a partitioned data set and that are used to override the system default option values, which can be changed without assembling and link-editing.

GLOBAL control statement

The GLOBAL control statement specifies the job step keyword values.

This control statement can be specified only in FPXIN DD. If the GLOBAL statement is specified, it must be specified as the first command in the input stream.

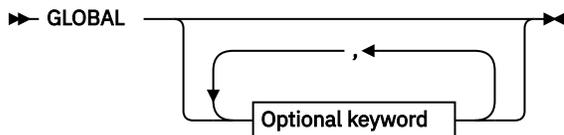
GLOBAL is an optional statement for the following utilities:

- OPC
- ODE
- OAE
- OER
- ODM

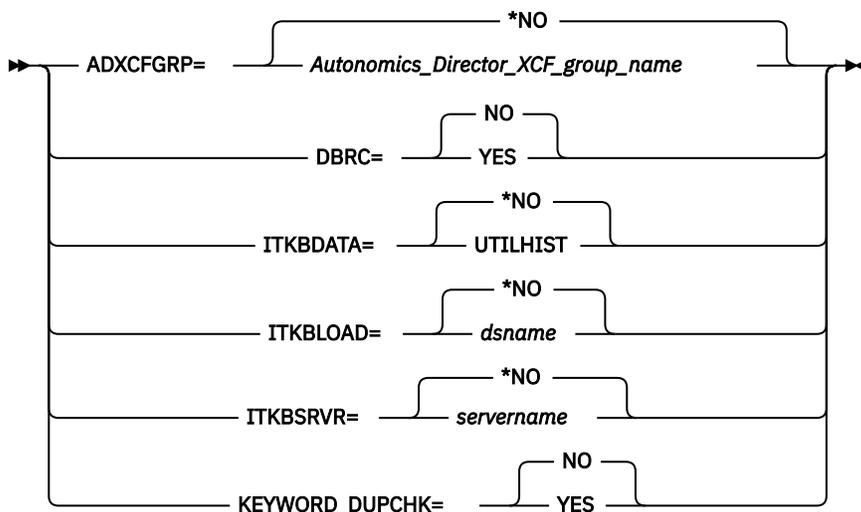
Related keywords:

- [“ADXCFCGRP keyword” on page 861](#)
- [“DBRC keyword \(for GLOBAL\)” on page 868](#)
- [“ITKBDATA keyword” on page 890](#)
- [“ITKBLOAD keyword” on page 890](#)
- [“ITKBSRVR keyword” on page 891](#)
- [“KEYWORD_DUPCHK keyword” on page 892](#)

Syntax:



Optional keywords:



The keywords associated with the GLOBAL command are available for site default options.

Related information

[Site default options support for FPO](#)

The site default options are control statements that are specified in a member of a partitioned data set and that are used to override the system default option values, which can be changed without assembling and link-editing.

OUTPUT control statement

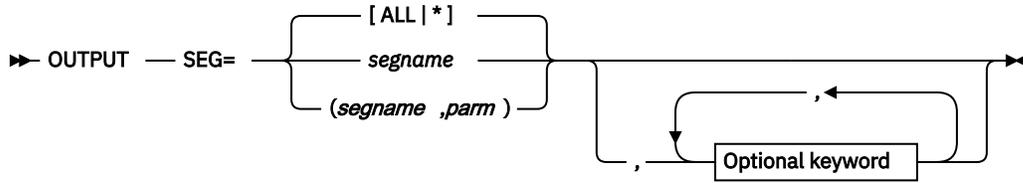
The OUTPUT statement defines what segments are to be included in the output file (OFILE).

The OUTPUT statement is a required statement for ODE if output is required. If specific keywords are omitted, the indicated defaults are taken for each of the keywords.

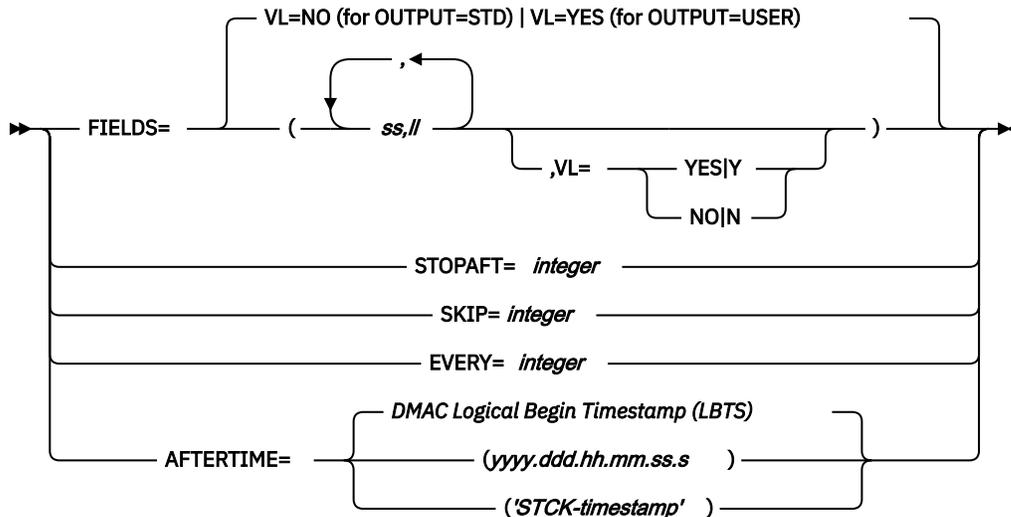
Related keywords:

- [“SEG keyword” on page 916](#)
- [“FIELDS keyword \(for OUTPUT\)” on page 876](#)
- [“STOPAFT keyword” on page 929](#)
- [“SKIP keyword” on page 926](#)
- [“EVERY keyword” on page 871](#)
- [“AFTERTIME keyword” on page 862](#)

Syntax:



Optional keywords:



The keywords associated with the OUTPUT statement are not available for site default options.

Examples of OUTPUT control statement

In the following example, every segment from the database will be included in the output. Every field within each segment will be included.

```
OUTPUT SEG=ALL
```

In the following example, only segments SEGA, SEGB, and SEGC will be included in the output. Data beginning at offset 1 for a length of 10, and offset 11 for a length of 20 will be included from SEGA. The entire segment will be included for SEGB. Data beginning at offset 5 for a length of 200 will be included from SEGC.

```
OUTPUT SEG=SEGA, FIELDS=(1, 10, 11, 20)
OUTPUT SEG=SEGB
OUTPUT SEG=SEGC, FIELDS=(5, 200)
```

In the following example, OUTPUT control criteria will be applied only to SEGA. Processing will stop after 10 occurrences of SEGA are matched under each parent of SEGA.

```
OUTPUT SEG=SEGA, STOPAFT=10
```

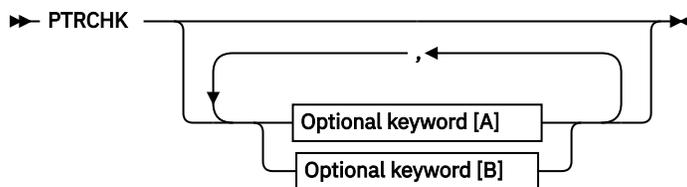
In the following example, OUTPUT control criteria will be applied only to SEGA. Five matching occurrences of SEGA under each parent of SEGA will be bypassed before processing. As in the prior example, processing will stop after 10 occurrences of SEGA are processed under each parent of SEGA.

```
OUTPUT SEG=SEGA, STOPAFT=10, SKIP=5
```

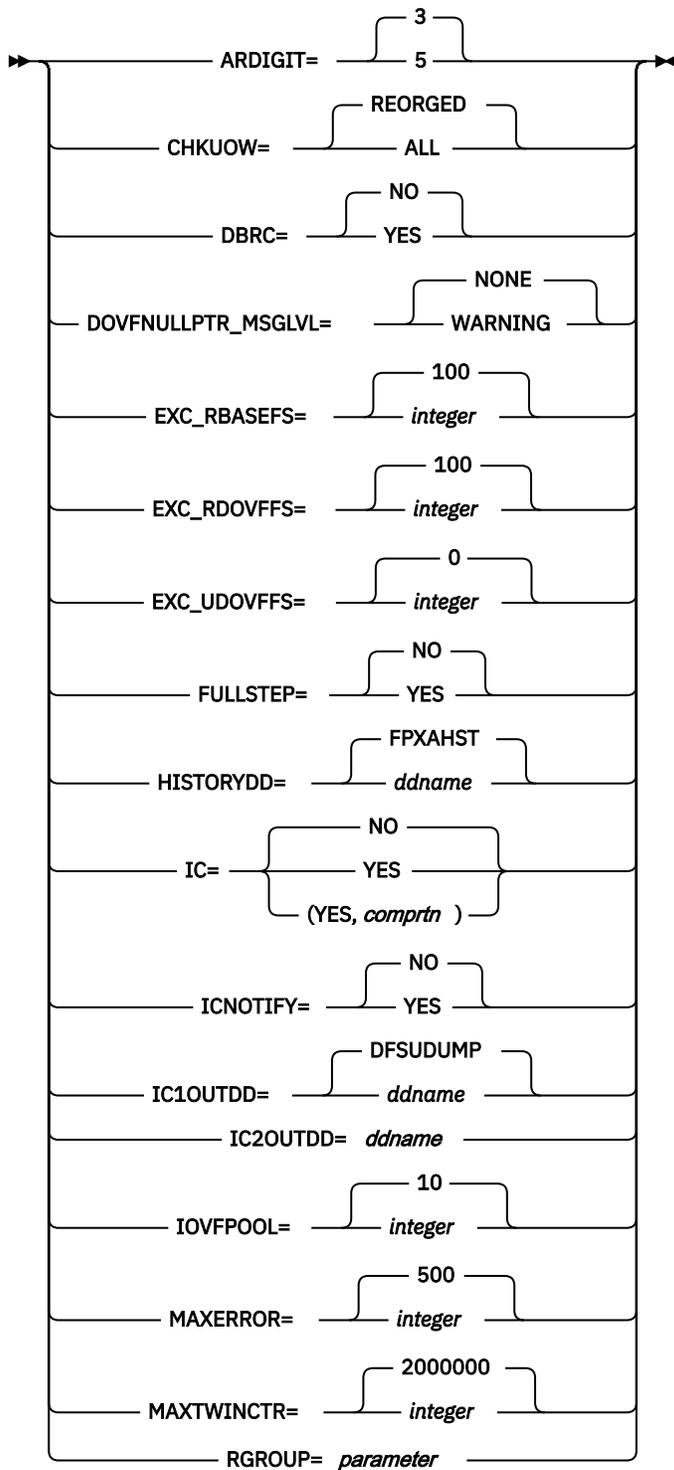
In the following example, OUTPUT control criteria will be applied only to SEGA. Before processing, 5 matching occurrences of SEGA will be skipped under each parent. During subsequent processing, every

- [“EXC_RDOVFFS keyword” on page 873](#)
- [“EXC_UDOVFFS keyword” on page 873](#)
- [“FULLSTEP keyword” on page 882](#)
- [“HISTORYDD keyword” on page 882](#)
- [“IC keyword” on page 883](#)
- [“ICNOTIFY keyword” on page 885](#)
- [“IC1OUTDD keyword” on page 886](#)
- [“IC2OUTDD keyword” on page 886](#)
- [“IOVFPOOL keyword” on page 888](#)
- [“MAXERROR keyword” on page 894](#)
- [“MAXTWINCTR keyword” on page 895](#)
- [“RGROUP keyword” on page 910](#)
- [“SDEP keyword” on page 912](#)
- [“SDEP_ORPHAN_MSGLVL keyword” on page 914](#)
- [“SDEP_PTRIOVF_MSGLVL keyword” on page 915](#)
- [“SENSOR keyword” on page 922](#)
- [“SENSOR_DBREC keyword” on page 923](#)
- [“SENSOR_PTRERR keyword” on page 924](#)
- [“SENSOR_UOW keyword” on page 925](#)
- [“SENSOR_UOW_REPORT keyword” on page 925](#)
- [“STAT keyword” on page 928](#)
- [“SYNTBL keyword” on page 932](#)
- [“TARDSNMASK keyword” on page 932](#)
- [“TARECGEN keyword” on page 933](#)
- [“TYPRUN keyword” on page 935](#)

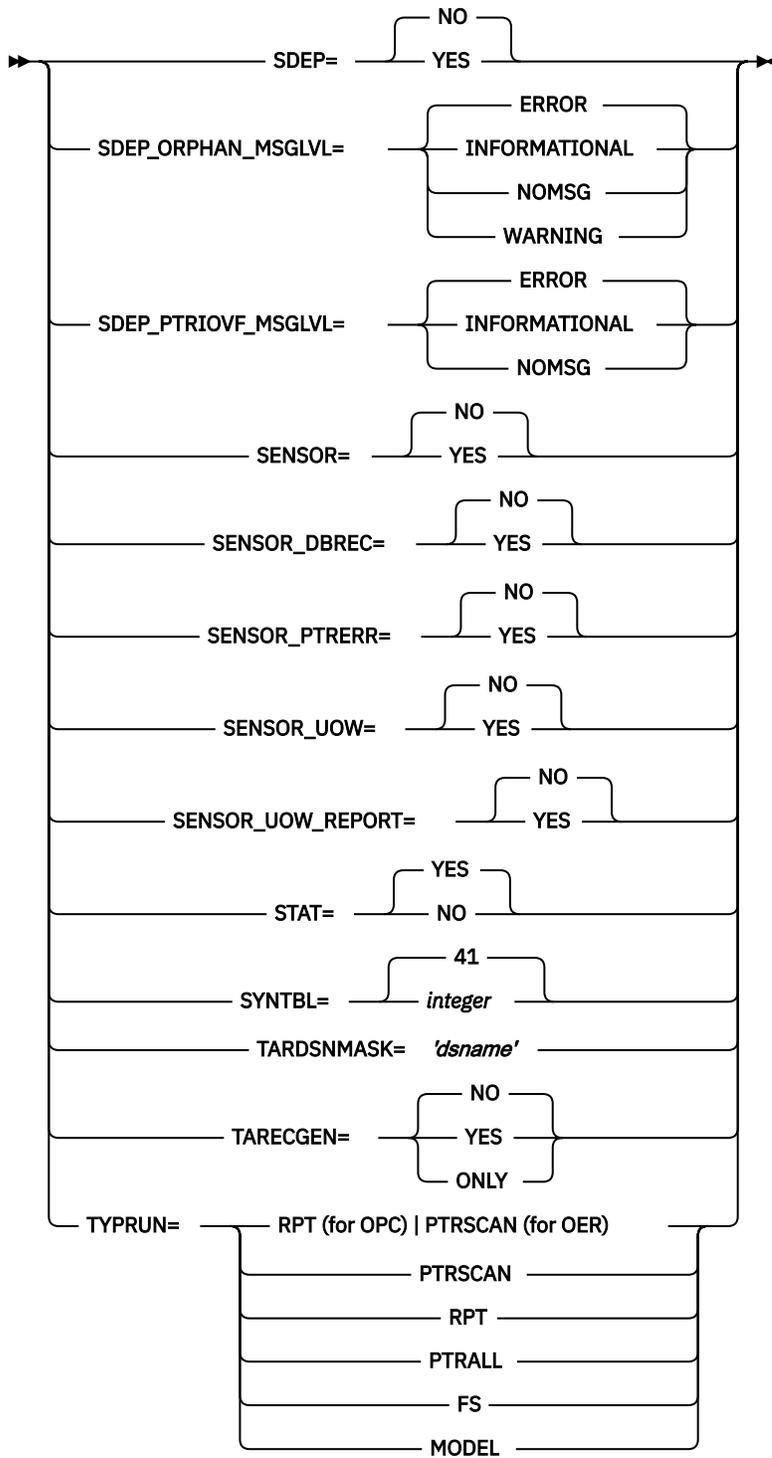
Syntax:



Optional keywords [A]:



Optional keywords [B]:



The keywords associated with the PTRCHK statement are available for site default options.

Related information

[Site default options support for FPO](#)

The site default options are control statements that are specified in a member of a partitioned data set and that are used to override the system default option values, which can be changed without assembling and link-editing.

REORG control statement

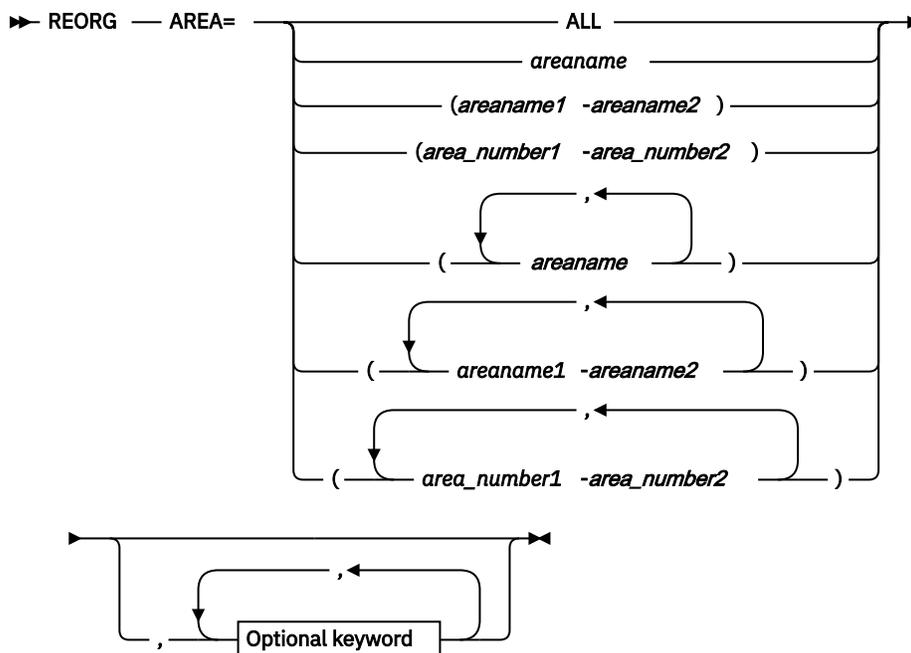
The REORG statement specifies information that controls how the OER run is to be processed.

REORG is a required statement for OER.

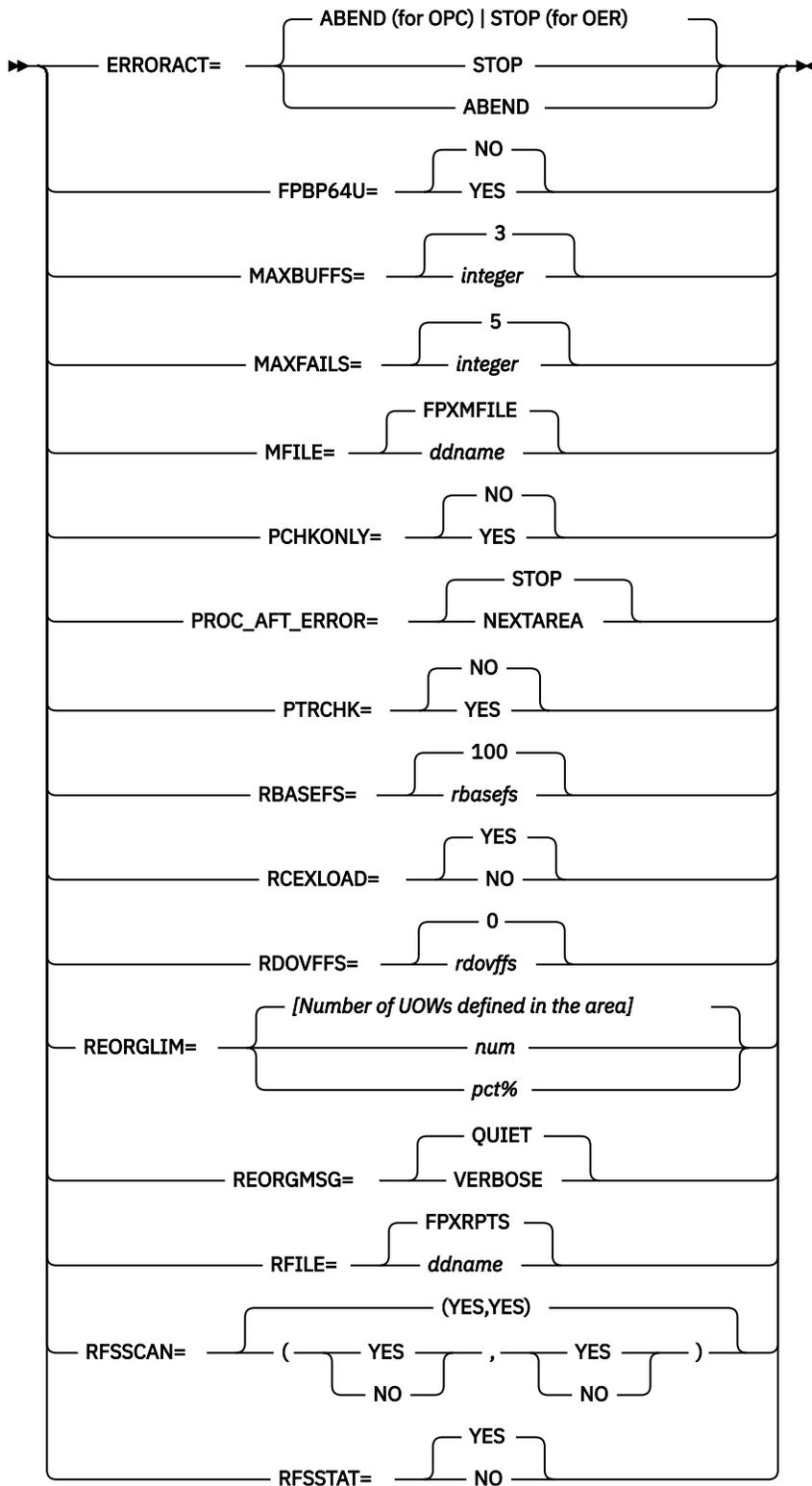
Related keywords:

- [“AREA keyword” on page 864](#)
- [“ERRORACT keyword” on page 870](#)
- [“FPBP64U keyword” on page 879](#)
- [“MAXBUFFS keyword” on page 894](#)
- [“MAXFAILS keyword” on page 895](#)
- [“MFILE keyword” on page 896](#)
- [“PCHKONLY keyword” on page 901](#)
- [“PROC_AFT_ERROR keyword” on page 902](#)
- [“PTRCHK keyword” on page 902](#)
- [“RBASEFS keyword” on page 904](#)
- [“RCEXLOAD keyword” on page 904](#)
- [“RDOVFFS keyword” on page 905](#)
- [“REORGLIM keyword” on page 906](#)
- [“REORGMSG keyword” on page 907](#)
- [“RFILE keyword” on page 908](#)
- [“RFSSCAN keyword” on page 909](#)
- [“RFSSTAT keyword” on page 909](#)

Syntax:



Optional keywords:



The keywords associated with the REORG statement are available for site default options.

Related information

[Site default options support for FPO](#)

The site default options are control statements that are specified in a member of a partitioned data set and that are used to override the system default option values, which can be changed without assembling and link-editing.

REPORT control statement

The REPORT control statement specifies the parameter values for the analysis reports.

This control statement is effective only when the FULLSTEP=YES keyword is specified with the TYPRUN=RPT or the TYPRUN=FS keyword.

The keywords that can be specified on the REPORT statement are similar to those of the Analyze function of FPA.

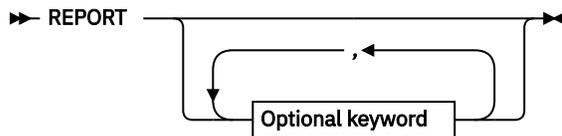
REPORT is an optional statement for the following utilities:

- OPC
- OER

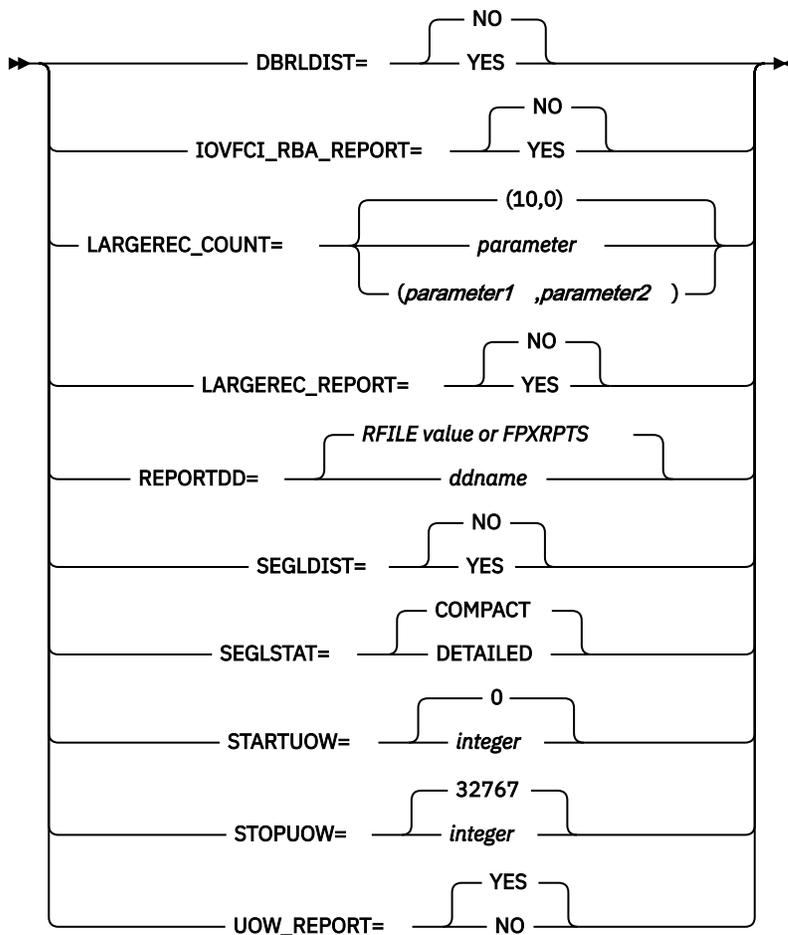
Related keywords:

- [“DBRLDIST keyword” on page 868](#)
- [“IOVFCI_RBA_REPORT keyword” on page 887](#)
- [“LARGEREC_COUNT keyword” on page 893](#)
- [“LARGEREC_REPORT keyword” on page 893](#)
- [“REPORTDD keyword” on page 907](#)
- [“SEGLDIST keyword” on page 918](#)
- [“SEGLSTAT keyword” on page 918](#)
- [“STARTUOW keyword” on page 928](#)
- [“STOPUOW keyword” on page 930](#)
- [“UOW_REPORT keyword” on page 939](#)

Syntax:



Optional keywords:



Restriction:

The REPORT control statement can be specified only in FPXIN DD.

The keywords associated with the REPORT statement are available for site default options.

SELECT control statement

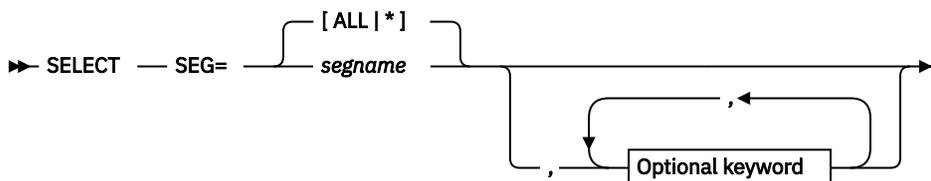
The SELECT statement specifies the segment name and the selection criteria for the segment.

SELECT is an optional statement for ODE. If omitted, the indicated defaults are taken for each of the keywords if the corresponding OUTPUT statement is coded for the segment. If both SELECT and OUTPUT are omitted for a specific segment, then no selection or extraction criteria is generated for it.

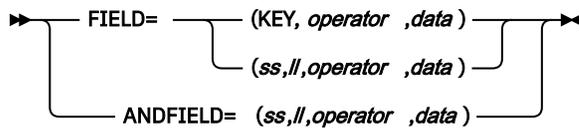
Related keywords:

- [“SEG keyword” on page 916](#)
- [“FIELD keyword” on page 875](#)
- [“ANDFIELD keyword” on page 862](#)

Syntax:



Optional keywords:



The keywords associated with the SELECT statement are not available for site default options.

Segment search and the hierarchy

The search criteria for a segment is applied hierarchically. If there is a SELECT statement for both a parent and child segment, the parent segment must be a match or the child segment will not be selected. However, if there is *no* SELECT statement for the parent but there is one for the child, all matching children will be returned.

SDEP segments cannot be extracted when the output format is TFMT or UR. If a TFMT or UR output format is selected, SDEPs are ignored if they are selected by using SEG=ALL, SEG=*, or an error message is generated if the SDEP segname is explicitly selected.

For the DBT output format, SDEP segments can only be extracted hierarchically when SDEPHIER=YES. When SDEPHIER=NO, SDEPs are ignored if they are selected by using SEG=ALL, SEG=*, or an error message is generated if the SDEP segname is explicitly selected.

For standard output, SDEP segments can be processed physically or hierarchically. If SDEPs are processed physically, they are processed in a fashion similar to the processing done by the SDEP Scan utility DBFUMSCO. With SDEP segments, the parent root does not have to match the SELECT criteria (if specified) for the SDEP segment(s) of the parent root to be extracted. SDEP segments are written to the output file *after* the root addressable area is processed. However, if SDEPs are processed hierarchically, SDEP segments are extracted only when the parent root matches the SELECT criteria (if specified), and they are written to the output file just after the root segment. Timestamps are stripped from SDEP segments. SELECT offsets that point into the timestamp are not considered a match. (The keyword AFTERTIME lets you specify SDEP segments by timestamp.)

For the DBT, the TFMT, or the DBT output format, the full hierarchical path from the lowest point of selection up to the root must be selected.

OUTPUT statement offsets cannot reference the SDEP timestamp; the timestamp is treated as beyond the segment length.

Examples of SELECT control statement

In the following example, every occurrence of all segments will be processed.

```
SELECT SEG=ALL
```

In the following example, segments with the segment name DEPTNAME will be retrieved when:

- The character string 'Engineering' is found at position 1 for a length of 13
- The number at position 15 for a length of 5 is greater than the number '10001216' in packed decimal

Note that if the provided data value has fewer characters than the explicit length specified by the ll value (as in the example where "Engineering" has 11 characters), then ODE will left align the provided data value in a field equal to the ll length and append low values to the right of the provided data value, up to the length specified by the ll value. This facilitates "partial value" matching, similar to partial key specification in IMS, provided the appropriate relational operator is utilized.

```
SELECT SEG=DEPTNAME,
       FIELD=(1,13,EQ,C'Engineering'),
       ANDFIELD=(15,5,GT,P'10001216')
```

In the following example, all areas will be searched for a segment with a match on a key field and a non-key field.

```
select seg=a1111111,  
       field=(key,eq,c'A060000390'),  
       andfield=(17,10,eq,c'FPEA060390')
```

In the following example, a search will be done for one segment type using four different search criteria.

```
FPXCTL SORT=YES  
SELECT SEG=DEPARTMT,  
       FIELD=(13,3,eq,C'J97')  
Comment  
SELECT SEG=DEPARTMT,  
       FIELD=(13,3,EQ,C'M81')  
Comment  
SELECT SEG=DEPARTMT,  
       FIELD=(13,3,Eq,C'J09')  
Comment  
SELECT SEG=DEPARTMT,  
       FIELD=(13,3,eQ,C'J03')  
Comment  
OUTPUT SEG=DEPARTMT
```

In the following example, a search will be done on segments from all areas using the LT key value and GT field value.

```
SELECT SEG=DEPARTMT, FIELD=(KEY, LT, C'R12200006'),  
       ANDFIELD=(03,12,GT,C'R12200000')
```

THRESHOLD control statement

The THRESHOLD control statement specifies the threshold parameter values, which cause warning messages to be generated when the actual values do not meet the specified criteria.

If you do not specify the TYPRUN=RPT keyword and the FULLSTEP=YES keyword, this control statement is ignored.

The THRESHOLD control statement is similar to that of the Analyze function of FPA.

THRESHOLD is an optional statement for the following utilities:

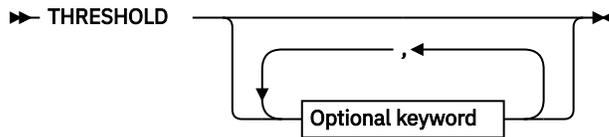
- OPC
- OER

Related keywords:

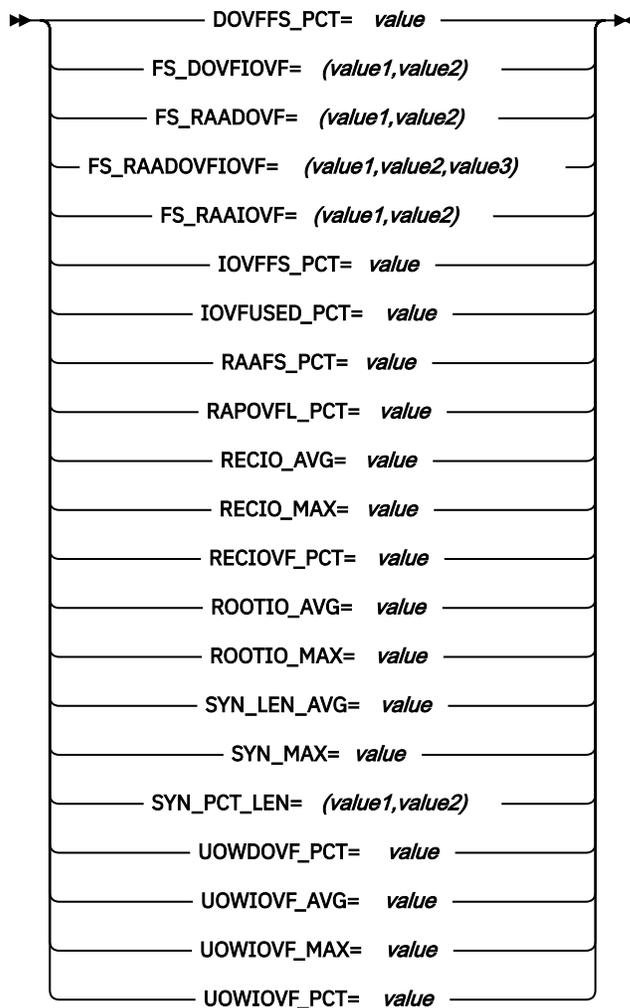
- [“DOVFFS_PCT keyword” on page 869](#)
- [“FS_DOVFIOVF keyword” on page 880](#)
- [“FS_RAADOVF keyword” on page 880](#)
- [“FS_RAADOVFIOVF keyword” on page 881](#)
- [“FS_RAAIOVF keyword” on page 881](#)
- [“IOVFFS_PCT keyword” on page 887](#)
- [“IOVFUSED_PCT keyword” on page 888](#)
- [“RAAFS_PCT keyword” on page 903](#)
- [“RAPOVFL_PCT keyword” on page 903](#)
- [“RECIO_AVG keyword” on page 905](#)
- [“RECIO_MAX keyword” on page 906](#)
- [“RECIOVF_PCT keyword” on page 906](#)
- [“ROOTIO_AVG keyword” on page 910](#)
- [“ROOTIO_MAX keyword” on page 911](#)

- [“SYN_LEN_AVG keyword” on page 930](#)
- [“SYN_MAX keyword” on page 931](#)
- [“SYN_PCT_LEN keyword” on page 931](#)
- [“UOWDOVF_PCT keyword” on page 937](#)
- [“UOWIOVF_AVG keyword” on page 937](#)
- [“UOWIOVF_MAX keyword” on page 938](#)
- [“UOWIOVF_PCT keyword” on page 938](#)

Syntax:



Optional keywords:



Restriction:

The THRESHOLD control statement can be specified only in FPXIN DD.

The keywords associated with the THRESHOLD statement are not available for site default options.

USER_RECORD control statement

The USER_RECORD statement specifies the user-defined header and the trailer records that are to be written to an extract file.

USER_RECORD is an optional statement for ODE. If this statement is omitted, no user-defined header and trailer records are written.

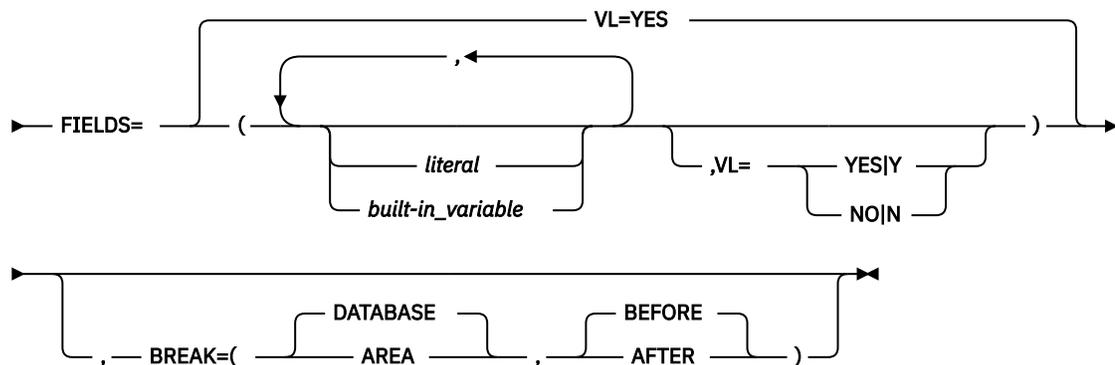
The USER_RECORD control statement can be specified only in FPXIN DD.

Related keywords:

- “BREAK keyword” on page 865
- “FIELD keyword” on page 875

Syntax:

►► USER_RECORD ►►



The keywords associated with the USER_RECORD statement are not available for site default options.

FPO keywords

The following topics explain the FPO keywords.

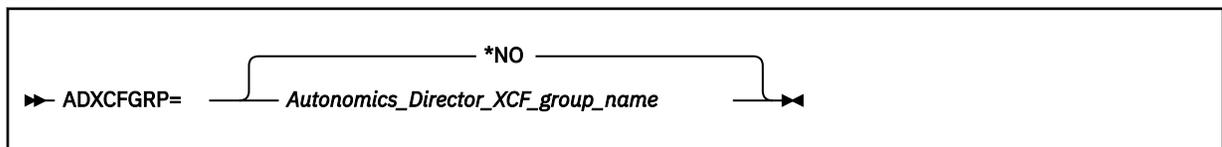
ADXCFCGRP keyword

The ADCFCGRP keyword specifies the Autonomics Director XCF group name. The name is used to send a sensor data notification to Autonomics Director when the sensor data is stored in the Sensor Data repository of IMS Tools KB.

ADXCFCGRP is an optional keyword for the GLOBAL command.

The sensor data notification is needed for Autonomics Director to schedule a follow-on action of evaluating sensor data. Consider specifying the Autonomics Director XCF group name if you want Autonomics Director to schedule the follow-on action.

Format:



*NO

Do not send a sensor data notification to Autonomics Director.

Autonomics_Director_XCF_group_name

Send the sensor data notification to Autonomics Director that has the specified XCF group name.

If you specify the Autonomics Director XCF group, you must specify DBRC=YES.

Default:

*NO, if the ADXCFGRP keyword is not specified.

Site default support:

Supported.

Restrictions:

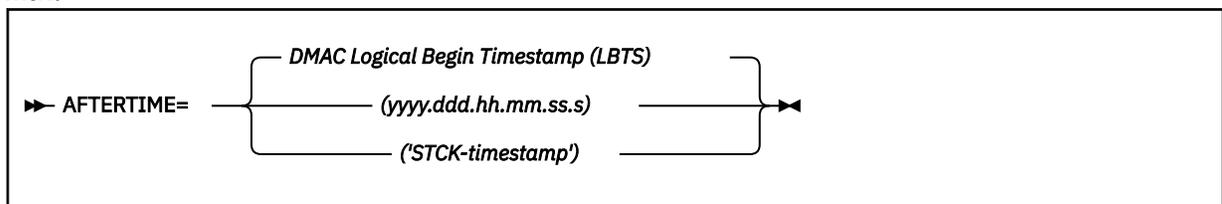
- The ADXCFGRP keyword can be specified only in FPXIN.
- The ADXCFGRP keyword is used only when SENSOR=YES is specified.

AFTERTIME keyword

The AFTERTIME keyword specifies the starting timestamp value for SDEP selections.

This keyword is only valid with SDEPHIER=NO.

AFTERTIME is an optional keyword for the OUTPUT statement of ODE. If this keyword is specified, it must be assigned a value.

Format:**yyyy.ddd.hh.mm.ss.s**

19 byte punctuated DBRC format date/timestamp.

STCK-timestamp

16 valid hexadecimal characters that are packed into an 8-byte hexadecimal (binary) time-of-day value.

Default:

DMAC Logical Begin Timestamp (LBTS), if the AFTERTIME keyword is not specified.

Site default support:

Not supported.

Restrictions:

- This keyword must be preceded by a valid OUTPUT SEG= specification.
- This keyword is only valid for SDEP segments.
- No segment older than the specified timestamp is extracted. If AFTERTIME is coded below the DMAC LBTS, a warning message is issued, but all selected SDEP segments are extracted. With the SDEPHIER=NO option, this technique lets you extract logically deleted SDEP segments in CIs that have not been overwritten. Logically deleted SDEP segments cannot be extracted when the SDEPHIER=YES option is specified.
- The two timestamp formats allow for different levels of granularity. If granularity smaller than a tenth of a second is required, it is suggested that you exploit the STCK format of the timestamp.

Both the DBRC and STCK formats of the timestamp, when used with the AFTERTIME keyword, are subject to MVS restrictions in terms of their scope, which currently spans January 1, 1900 through September 17, 2042.

ANDFIELD keyword

The ANDFIELD keyword specifies additional selection criteria (used in conjunction with the FIELD keyword criteria) for selecting segments for extraction.

ANDFIELD is an optional keyword for the SELECT statement of ODE. If this keyword is specified, it must be assigned appropriate values. For SDEP segments, this keyword is only valid with SDEPHIER=NO.

Format:

```
➤ ANDFIELD= — (ss,ll,operator,data ) ➤
```

ss

Integer specifying the position of the field to be compared relative to 1. The value of *ss* cannot be greater than the maximum segment length.

ll

Integer specifying the length of the field to be compared to the data value, in bytes. The value of *ll* cannot be greater than the maximum segment length.

operator

Relational operator. Valid values are EQ, NE, GE, GT, LT, and LE.

data

Value on which to match, in one of the following forms:

- X'xxxxxx' (hexadecimal)

Must be an even number of hexadecimal characters. Must specify a minimum of two hexadecimal characters (representing 1 byte) and a maximum of 56 hexadecimal characters (representing 28 bytes).

When *ll* is longer than the length of the data value, the data is filled with x'00', and when *ll* is shorter than the length of the data value, the length of the field to be compared is equal to *ll*.

- P'xxxxxx' (packed decimal)

Assumed positive. Must contain only numeric digits. Must specify a minimum of 1 and maximum of 31 digits (16 bytes).

The specified field is compared with the data value using CP.

- C'xxxxxx' (character)

Case-sensitive character data. Must specify a minimum of 1 and maximum of 28 characters (28 bytes).

When *ll* is longer than the length of the data value, the data is filled with x'00', and when *ll* is shorter than the length of the data value, the length of the field to be compared is equal to *ll*.

Default:

None.

Site default support:

Not supported.

Restrictions:

- The ANDFIELD keyword must be preceded by the FIELD keyword.
- Any combination of *ss+ll-1* that exceeds the maximum segment length is rejected.
- Up to 8 occurrences of a combination of ANDFIELD and FIELD can be specified per SEG keyword.
- If the target data field in the segment is negative, the compare fails as a NO MATCH.

ARDIGIT keyword

The ARDIGIT keyword specifies the digit of the area number field for the UOW group definition records that are defined in the FPXAUOWC file.

If the FPXAUOWC file is not specified in the JCL statement, this keyword is ignored.

ARDIGIT is an optional keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement of OER

You can also specify the range of areas by area names. *areaname1* and *areaname2* must be specified in alphabetical order. At least one area must exist in the specified range.

ALL

All areas are processed.

Note: ALL is a name reserved for the AREA keyword. If you use ALL as a DBD-defined area name, the utility will assume that you specified ALL to mean *process all areas*.

For example, the following areas are specified in the DBD:

Area number	Area name
1	AABBCC
2	CBA
3	ABCD
4	C
5	A
6	TEST1ALL
7	ALLTEST1
8	AAA
9	AA
10	AREA1234

- If you specify AREA=%B*, areas CBA and ABCD are selected.
- If you specify AREA=A-AAA, areas A, AA, and AAA are selected.
- If you specify AREA=6-7, areas TEST1ALL and ALLTEST1 are selected.

Multiple areas can be specified only with the following statements:

- ANALYZE
- EXTRACT
- REORG
- DMACPRT

Default:

None.

Site default support:

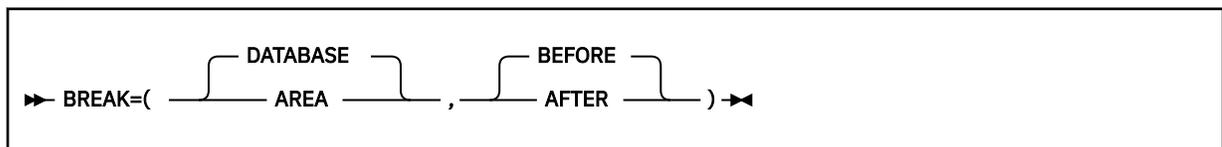
Not supported.

BREAK keyword

The BREAK keyword specifies the level and the timing for the trigger event for user-defined header and trailer of an extract file.

BREAK is an optional keyword for the USER_RECORD statement of ODE.

Format:



The first parameter specifies the level:

DATABASE

The trigger event occurs at the database processing level. This is the default value.

AREA

The trigger event occurs at the area processing level.

The second parameter specifies when to trigger the user record:

BEFORE

The user record is triggered when processing begins for the specified level. This is the default value.

AFTER

The user record is triggered after processing is complete for the specified level.

Default:

BREAK=(DATABASE,BEFORE)

Site default support:

Not supported.

Restriction:

The BREAK keyword can be specified only in FPXIN.

CASE keyword

The CASE keyword allows case sensitivity with character specification in the FIELD= and ANDFIELD= keywords.

CASE is an optional keyword for the FPXCTL statement of ODE.

Format:**YES (or Y)**

Support case sensitivity in values specified on the FIELD= and ANDFIELD= keywords.

NO (or N)

Do not support case sensitivity with respect to the FIELD= and ANDFIELD= keywords.

For FPXCTL DD and *areaname* DD, you can specify this keyword without a value. CASE or CASE= is regarded as CASE=YES.

Default:

NO, if the CASE keyword is not specified.

Site default support:

Supported.

CHKUOW keyword

The CHKUOW keyword specifies whether to do the pointer checking for all UOWs in the area or for the UOWs that have been reorganized.

CHKUOW is an optional keyword for the PTRCHK statement of OER.

Format:**REORGED (or R)**

OPC runs against only the UOWs that were reorganized.

ALL (or A)

OPC runs against all UOWs in the area.

CHKUOW=REORGED can be specified only when TYPRUN=PTRSCAN, SENSOR=NO, and SDEP=NO are specified. If CHKUOW=REORGED is specified, IC=YES cannot be specified.

Default:

REORGED, if the CHKUOW keyword is not specified.

Site default support:

Supported.

CIPRT keyword

The CIPRT keyword specifies whether the second CI of available ADS(s) is printed or not.

CIPRT is an optional keyword for the DMACPRT statement of ODM.

Format:**YES (or Y)**

Print the second CI of available ADS(s).

NO (or N)

Do not print the second CI of available ADS(s).

Default:

YES, if the CIPRT keyword is not specified.

Site default support:

Supported.

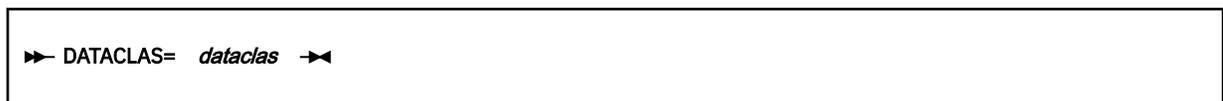
DATACLAS keyword

The DATACLAS keyword specifies the name of the SMS data class for the temporary area data set.

This parameter is used for the preparation of the temporary ADS.

For more information about SMS data class, see the *MVS JCL Reference*.

DATACLAS is an optional keyword for the EXTEND statement of OAE.

Format:

DATACLAS can be specified as DATACLASS.

dataclas

Specify the name in 1- to 8-characters.

Default:

None.

Site default support:

Not supported.

DBRC keyword (for GLOBAL)

The DBRC keyword specifies whether to store reports that are associated with a RECON to the IMS Tools KB repository.

DBRC is an optional keyword for the following statements:

- GLOBAL statement of OPC
- GLOBAL statement of OAE
- GLOBAL statement of OER
- GLOBAL statement of ODM

Format:



NO (or N)

Store the reports that are not associated with a RECON to the IMS Tools KB repository.

YES (or Y)

Stores the reports that are associated with a RECON to the IMS Tools KB repository.

Default:

NO, if the DBRC keyword is not specified.

Site default support:

Supported.

Restriction:

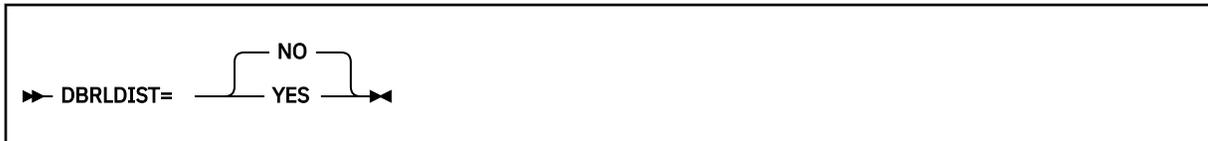
The DBRC keyword can be specified only in FPXIN.

DBRLDIST keyword

The DBRLDIST keyword specifies whether to generate the DB Record Length Distribution report.

DBRLDIST is an optional keyword for the REPORT statement.

Format:



YES (or Y)

Print the DB Record Length Distribution report.

DBRLDIST=YES is effective only when TYPRUN=RPT and FULLSTEP=YES are specified. For OER, PTRCHK=YES and CHKUOW=ALL are also required.

NO (or N)

Do not print the DB Record Length Distribution report.

Default:

DBRLDIST=NO

Site default support:

Supported.

DDNAME keyword

The DDNAME keyword specifies the DDNAME of the temporary area data set where the area will be copied.

This DDNAME should not be registered in DBRC. The total number of preexisting and new ADS(s) cannot exceed 7 because during utility execution, both the preexisting and the temporary ADS can be opened concurrently. This parameter is used for the preparation of the temporary ADS.

DDNAME is a required keyword for the EXTEND statement of OAE.

Format:

```
►► DDNAME= ddname ◄◄
```

ddname

Specifies the DD name of the temporary area data set in which the area will be copied.

Default:

None.

Site default support:

Not supported.

DOVFFS_PCT keyword

The DOVFFS_PCT keyword specifies a threshold setting for the percentage of free space in DOVF.

If the percentage of free space in DOVF is less than the value that you specified, a warning message is issued.

DOVFFS_PCT is an optional keyword for the following statements:

- THRESHOLD statement of OPC
- THRESHOLD statement of OER

Format:

```
►► DOVFFS_PCT= value ◄◄
```

value

An integer value in the range of 0 - 100.

Default:

None.

Site default support:

Not supported.

Restriction:

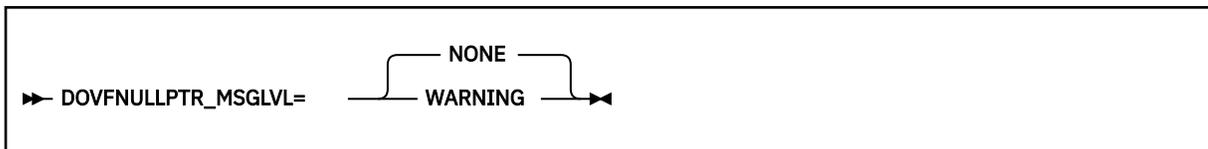
The DOVFFS_PCT keyword can be specified only in FPXIN.

DOVNULLPTR_MSGLVL keyword

The DOVNULLPTR_MSGLVL keyword specifies that the utility issues a warning message if the first DOVF CI in any UOW contains a null pointer for the current overflow CI.

DOVNULLPTR_MSGLVL is an optional keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement of OER

Format:**NONE (or N)**

Does not issue an alert message when the utility detects a null pointer for the current overflow CI in the first DOVF CI of a UOW.

WARNING (or W)

Issues a warning message when the utility detects a null pointer for the current overflow CI in the first DOVF CI of a UOW. The utility prints a warning message in the Processing Messages report, sets the return code to 4, and continues processing.

Default:

NONE, if the DOVFNULLPTR_MSGLVL keyword is not specified.

Site default support:

Supported.

Restriction:

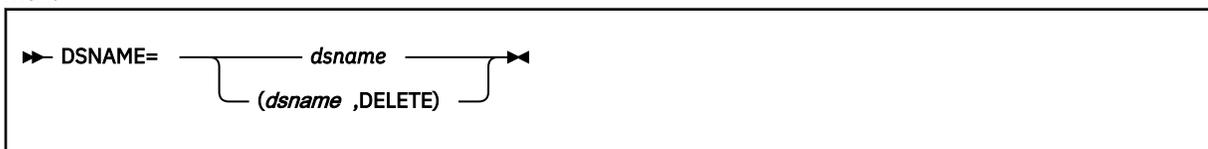
The DOVFNULLPTR_MSGLVL keyword can be specified only in FPXIN.

DSNAME keyword

The DSNAME keyword specifies the data set name of the temporary area data set where the area will be copied.

This data set name should not be registered in DBRC. When the DELETE option is specified, the data set is deleted and defined. This parameter is used for the preparation of the temporary ADS.

DSNAME is a required keyword for the EXTEND statement of OAE.

Format:***dsname***

Specifies the data set name of the temporary area data set in which the area will be copied.

Default:

None.

Site default support:

Not supported.

Note: When you run with the EXTEND_TYPE=RENAME option, the high-level qualifiers of the data set name must be the same as the ones of the original data set name. If the original data set name has no qualifier, the temporary data set must be cataloged to the VSAM catalog file to which the original one is cataloged to.

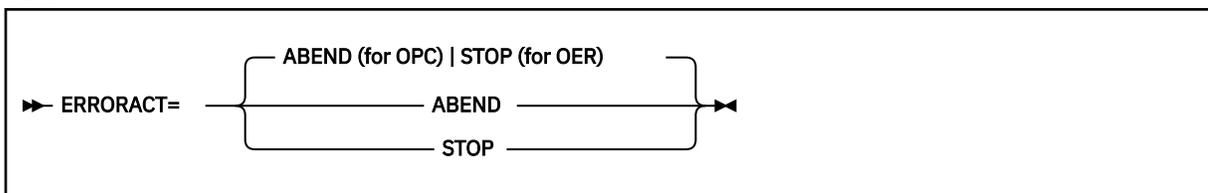
ERRORACT keyword

The ERRORACT keyword specifies the action for the utility to take at certain error conditions.

ERRORACT is an optional keyword for the following statements:

- ANALYZE statement of OPC
- REORG statement of OER

If you specify this keyword, you must specify a variable.

Format:**ABEND (or A)**

For OPC, specifies that when the number of errors encountered exceeds the value specified in the MAXERROR parameter, processing ends with a user abend code 3510.

For OER, specifies that when one of the following conditions is met, processing ends with the following user abend code:

U3500

Error-1: An error that occurs while the Area Sensor is running

U3523

Error-2: The number of errors that occur during the UOW reorganization has exceeded the value specified in the MAXFAILS keyword of the REORG control statement

U3510

Error-3: The number of errors that occurred in OPC has reached the value specified in the MAXERROR keyword of the PTRCHK control statement

STOP (or S)

Specifies that processing ends without a dump, but with a return code of 8.

For OER, if RFSSCAN=(NO,YES) or RFSSCAN=(NO,NO) is specified, ERRORACT=ABEND is not effective for the Error-2 error case. In this case, the processing always stops with a return code of 8 regardless of the ERRORACT parameter.

Default:

- For OPC, if the ERRORACT keyword is not specified, the default is ABEND.
- For OER, if the ERRORACT keyword is not specified, the default is STOP.

Site default support:

Supported.

Restriction:

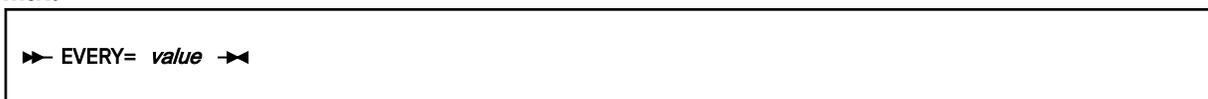
Specification of ERRORACT=ABEND is mutually exclusive with PROC_AFT_ERROR=NEXTAREA.

EVERY keyword

The EVERY keyword lets you select every Nth occurrence of a segment starting from the first segment.

For DDEP segments, every Nth occurrence of a segment under a parent, and for root segments, every Nth occurrence of a root segment in a synonym chain under a RAP. If the SKIP keyword is specified, the first segment is the next segment after the skipped number of segments. For SDEP segments, this keyword is only valid with SDEPHIER=NO, and the keyword lets you select every Nth segment starting from the first CI of the SDEP part.

EVERY is an optional keyword for the OUTPUT statement of ODE. If this keyword is specified, it must be assigned a value.

Format:**value**

Select segments at the indicated interval. The value you specify must be in the range of 1 - 9999999.

Default:

None.

Site default support:

Not supported.

Restrictions:

- This keyword must be preceded by a valid OUTPUT SEG= specification.
- Dependents of parents bypassed due to EVERY are not selected.
- You cannot have multiple values for the EVERY keyword specified for the same segment name. If you do specify multiple values, only the last valid occurrence of the EVERY keyword will be in effect.
- If more than one of the STOPAFT, SKIP, or EVERY keywords is specified, SKIP is applied first, then EVERY, and then STOPAFT.
- SELECT statement criteria are evaluated before EVERY. Only matched segments are counted. Counts are maintained for segment occurrences under a single parent and are reset when parentage changes.
- If DEPENDENTS is specified in the preceding SEG keyword, the EVERY keyword cannot be specified.

The start position of the selected segment occurrence is determined by the value of the SKIP keyword.

For example (there are 10 segment occurrences)

```
OUTPUT SEG=seg_name,EVERY=2,SKIP=1,STOPAFT=3
```

1. The first segment is skipped by SKIP=1.
2. The second, fourth, and sixth segments are included in the output by EVERY=2.
3. The eighths and tenth segments are not included in the output by STOPAFT=3.

In the following example, the start position is decided by starting from the first segment. For example, (There are 10 segment occurrences)

```
OUTPUT SEG=seg_name,EVERY=2,STOPAFT=3
```

1. The first, third, and fifth segments are included in the output by EVERY=2.
2. The seventh and ninth segments are not included in the output by STOPAFT=3.

EXC_RBSEFS keyword

The EXC_RBSEFS keyword specifies a criterion. If a RAP CI contains free space that is greater than the percentage specified, and the RAP CI also uses overflow CIs, a reorganization control statement is generated for the UOW, and the UOW is printed in the UOW Exception report.

EXC_RBSEFS is an optional keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement of OER

Format:**integer**

An integer value in the range of 0 - 100. If EXC_RBSEFS=100 is specified, the reorganization control statements are not generated.

Default:

100, if the EXC_RBSEFS keyword is not specified.

Site default support:

Supported.

Restrictions:

- The EXC_RBASEFS keyword can be specified only in FPXIN.
- The EXC_RBASEFS keyword is effective only when both the TYPRUN=RPT keyword and the FULLSTEP=YES keyword are specified. Otherwise, it is ignored.

EXC_RDOVFFS keyword

The EXC_RDOVFFS keyword specifies a criterion. If the DOVF part of a UOW contains free space that is greater than the percentage specified, and the UOW also uses IOVF CIs, a reorganization control statement is generated for the UOW and the UOW is printed in the UOW Exception report.

EXC_RDOVFFS is an optional keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement of OER

Format:**integer**

An integer value in the range of 0 - 100. If EXC_RDOVFFS=100 is specified, the reorganization control statements are not generated.

Default:

100, if the EXC_RDOVFFS keyword is not specified.

Site default support:

Supported.

Restrictions:

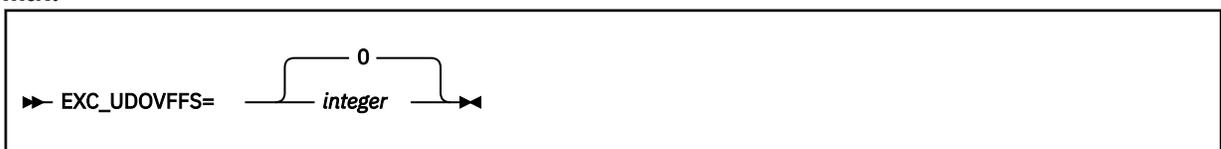
- The EXC_RDOVFFS keyword can be specified only in FPXIN.
- The EXC_RDOVFFS keyword is effective only when both the TYPRUN=RPT keyword and the FULLSTEP=YES keyword are specified. Otherwise, it is ignored.

EXC_UDOVFFS keyword

The EXC_UDOVFFS keyword specifies a criterion. If the DOVF part of a UOW contains free space that is less than the percentage specified, the UOW is printed in the UOW Exception report.

EXC_UDOVFFS is an optional keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement of OER

Format:**integer**

An integer value in the range of 0 - 100. If EXC_UDOVFFS=0 is specified, the UOW is not included in the UOW Exception report.

Default:

0, if the EXC_UDOVFFS keyword is not specified.

Site default support:

Supported.

Restrictions:

- The EXC_UDOVFFS keyword can be specified only in FPXIN.
- The EXC_UDOVFFS keyword is effective only when both the TYPRUN=RPT keyword and the FULLSTEP=YES keyword are specified. Otherwise, it is ignored.

EXPAND keyword

The EXPAND keyword specifies whether compressed data is to be expanded during extraction.

EXPAND is an optional keyword for the FPXCTL statement of ODE.

Format:**NO (or N)**

Do NOT expand compressed data during extraction.

There is no expansion of compressed segments. If it is desired to select compressed segments, the user should supply compressed data as SELECT arguments.

YES (or Y)

Expand compressed data during extraction.

Data is expanded as necessary for comparison of SELECT operators and for output processing. Only segments considered for selection are expanded. For example, if a dependent segment is compressed, but its parent segment type is not selected and never matched, the dependent segment is never expanded.

The compression exit must be loadable via STEPLIB if it is required. Loading is done during initialization, but processing continues even if the exit cannot be loaded, since it is not certain that it will, in fact, be required. If an exit is found to be absent when required, ODE ends with an abend at that point.

For FPXCTL DD and *areaname* DD, you can specify this keyword without a value. EXPAND or EXPAND= is regarded as EXPAND=YES.

Default:

NO, if the EXPAND keyword is not specified.

Site default support:

Supported.

Restrictions:

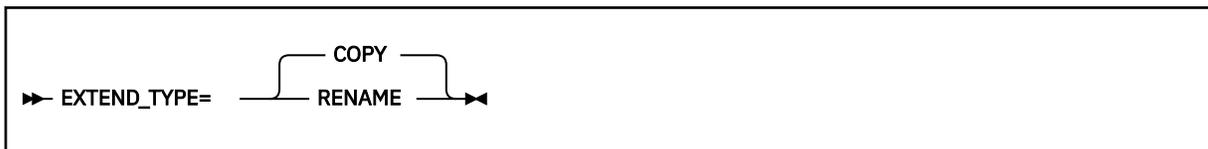
- A compression routine must exist.
- The segment edit/compression routine interface environment that is created by ODE conforms to the published interface as defined *IMS Exit Routines* with one exception. When the segment edit/compression routine is invoked by IMS, register 1 contains the address of EPST. However, when ODE calls a segment edit/compression routine, register 1 is set to 0, which indicates that it was not called by IMS.

EXTEND_TYPE keyword

The EXTEND_TYPE keyword specifies the extend type.

EXTEND_TYPE is an optional keyword for the EXTEND statement of OAE.

Format:



COPY

Specifies to copy back from the temporary ADS that contains extended image to the original one, which will be once deleted and redefined to extend the space.

RENAME

Specifies to rename the temporary ADS that contains extended image to the original ADS in the subsequent step with the OAE Rename program.

Default:

COPY, if the EXTEND_TYPE keyword is not specified.

Site default support:

Not supported.

Related information

[OAE Rename program](#)

To complete the OAE process for the original expansion with EXTEND_TYPE=RENAME, rename will be done by the OAE Rename program, which should be the step that immediately succeeds the OAE program.

FIELD keyword

The FIELD keyword specifies the criteria used to select segments for extraction.

The selection criteria can be based either on the segment's key (as defined in the SEGM statement in the DBD) or on a positional offset into the segment.

FIELD is an optional keyword for the SELECT statement of ODE. If this keyword is specified, it must be assigned appropriate values. For SDEP segments, this keyword is only valid with SDEPHIER=NO.

Format:



KEY

The segment key. When KEY is specified, the length of the data value (see below) must be equal to or less than the key sequence field specified in the DBD.

ss

Integer specifying the position of the field to be compared relative to 1. The value of ss cannot be greater than the maximum segment length.

ll

Integer specifying the length of the field to be compared, in bytes. The value of ll cannot be greater than the maximum segment length.

operator

Relational operator. Valid values are EQ, NE, GE, GT, LT, and LE.

data

Value on which to match, in one of the following forms:

- X'xxxxxx' (hexadecimal)

Must be an even number of hexadecimal characters. Must specify a minimum of two hexadecimal characters (representing 1 byte) and a maximum of 56 hexadecimal characters (representing 28 bytes).

For FIELD=(KEY,operator,data), when the key length is longer than the length of the data value, the data is filled with x'00', and when the key length is shorter than the length of the data value, error message FPX0043E is issued.

For FIELD=(ss,ll,operator,data), when ll is longer than the length of the data value, the data is filled with x'00', and when ll is shorter than the length of the data value, the length of the field to be compared is equal to ll.

- P'xxxxxx' (packed decimal)

Assumed positive. Must contain only numeric digits. Must specify a minimum of 1 and maximum of 31 digits (16 bytes).

For FIELD=(KEY,operator,data), when the key length is longer than the length of the data value, the length of the field to be compared is equal to the key length, and when the key length is shorter than the length of the data value, error message FPX0043E is issued.

For FIELD=(ss,ll,operator,data), the specified field is compared with the data value using CP.

- C'xxxxxx' (character)

Case-sensitive character data. Must specify a minimum of 1 and maximum of 28 characters (28 bytes).

For FIELD=(KEY,operator,data), when the key length is longer than the length of the data value, the data is filled with x'00', and when the key length is shorter than the length of the data value, error message FPX0043E is issued.

For FIELD=(ss,ll,operator,data), when ll is longer than the length of the data value, the data is filled with x'00', and when ll is shorter than the length of the data value, the length of the field to be compared is equal to ll.

If the selection criteria specifies the root segment, and the specification is an equal compare on the entire key field, either with FIELD=(KEY,operator,data) or with FIELD=(ss,ll,operator,data), ODE invokes the randomizer and determines which UOWs must be read to satisfy the selection criteria. For other selections, ODE scans UOWs sequentially to search for the valid segments.

Default:

None.

Site default support:

Not supported.

Restrictions:

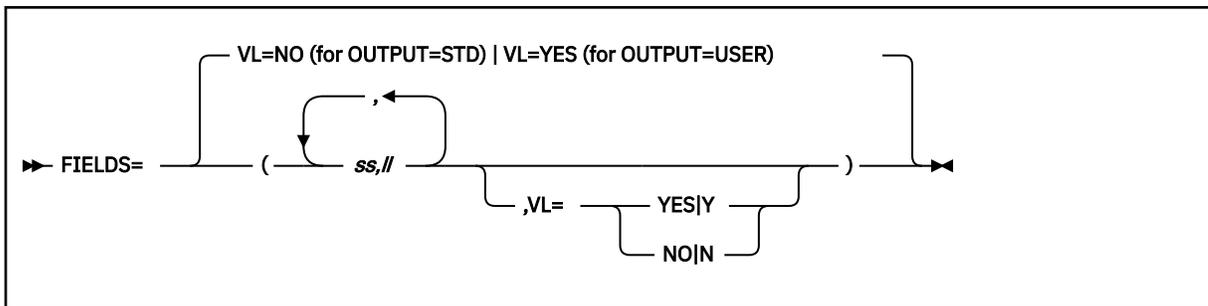
- The FIELD keyword must be preceded by a valid SELECT SEG= specification.
- Any combination of ss+ll-1 that exceeds the maximum segment length is rejected.
- Up to 8 occurrences of a combination of ANDFIELD and FIELD can be specified per SEG keyword.
- If the target data field in the segment is negative, the compare fails as a NO MATCH.

FIELDS keyword (for OUTPUT)

The FIELDS keyword specifies what data from a selected segment is to be written to the output file.

If you want a complete segment, do not use the FIELDS keyword. However, if you want to write partial data from a segment, use the FIELDS keyword to specify the positional offset and length of the data to be written to the output file (in this sense, FIELDS is similar to the FIELD/ANDFIELD selection criteria).

FIELDS is an optional keyword for the OUTPUT statement of ODE. If this keyword is specified, it must be assigned appropriate values. For SDEP segments, this keyword is only valid with SDEPHIER=NO.

Format:**ss**

Integer specifying the position of the field to be compared relative to 1. The value of *ss* cannot be greater than the maximum segment length.

ll

Integer specifying the length of the field to be compared, in bytes. The value of *ll* cannot be greater than the maximum segment length.

VL=YES

Request a length prefix.

VL=NO

Suppress a length prefix.

To access more than one area from the selected segment, you can specify multiple pairs of *ss* and *ll*.

Default:

VL=NO when OUTPUT=STD. VL=YES when OUTPUT=USER.

Site default support:

Not supported.

Restrictions:

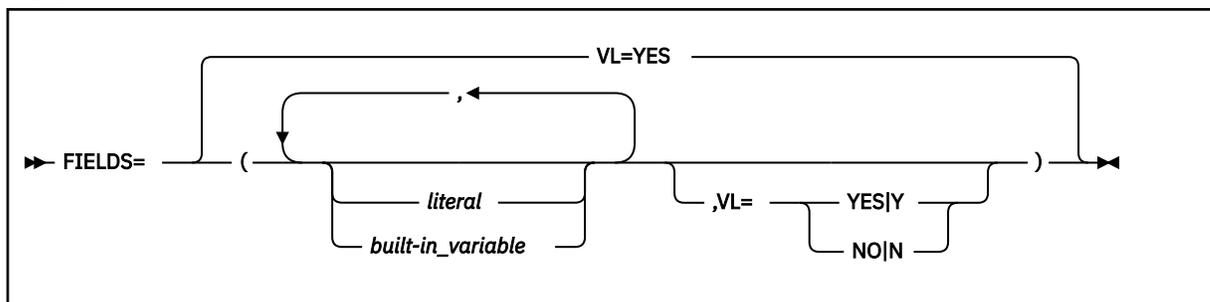
- This keyword must be preceded by a valid OUTPUT SEG= specification.
- The integer you specify cannot exceed the maximum length of the segment.
- Any combination of *ss+ll-1* that exceeds the maximum segment length is rejected.
- FIELDS cannot be specified if OUTPUT=DBT|TFMT|UR is specified in the FPXCTL control statement; this is because when DBT|TFMT|UR is specified, the entire segment must be written.
- If DEPENDENTS is specified in the preceding SEG keyword, the FIELDS keyword cannot be specified.
- The maximum number of combinations of *ss/ll* for one segment is 8.
- The maximum number of *ss/ll* for a FIELDS keyword is 8.

FIELDS keyword (for USER_RECORD)

The FIELDS keyword specifies the user-defined header records, or the user-defined trailer records that are written to the output file.

The extracted fields are in the output record in the order that is specified by the FIELDS parameter.

FIELDS is a required keyword for the USER_RECORD statement of ODE.

Format:**literal**

Specify one or more alphabetic or numeric characters that are contained within apostrophes (' ') or within quotation marks (" "). You can specify an optional data-type declaration as follows:

- C'xxxxxx' (character): Case-sensitive character data. You must specify a minimum of 1 and maximum of 28 characters (28 bytes).
- F'xxxxxx' (signed binary): Must contain only numeric digits. You must specify a minimum of 1 and maximum of 19 digits (4 bytes).
- P'xxxxxx' (packed decimal): Assumed positive. Must contain only numeric digits. You must specify a minimum of 1 and maximum of 31 digits (16 bytes).
- X'xxxxxx' (unsigned binary): Must be an even number of hexadecimal characters. You must specify a minimum of two hexadecimal characters (representing 1 byte) and a maximum of 56 hexadecimal characters (representing 28 bytes).

You can also declare the length, by specifying it before the data-type declaration.

The parenthesis cannot be specified within apostrophes.

Example:

```
8C 'CUSDB'
2P '01'
```

Built-in_variable

The following built-in variables can be specified.

Built-in variable name	Data type	Contents
AREA_COUNT	2X	<ul style="list-style-type: none"> • Number of areas that were attempted to be written to an extract file. (BREAK=(DATABASE, BEFORE)) • Number of areas written to an extract file. (BREAK=(DATABASE, AFTER))
AREA_NAME	8C	Area name BREAK=(AREA,when) only
AREA_NUMBER	1X	Area number (1 byte) BREAK=(AREA,when) only
AREA_NUMBER2	2X	Area number (2 bytes) BREAK=(AREA,when) only
DBD_AREAS	2F	Number of areas in DBD
DBD_LEVELS	2F	Maximum hierarchic level in DBD
DBD_NAME	8C	DBD name
DBD_SEGMENTS	2F	Number of segment types in DBD

Built-in variable name	Data type	Contents
RUNDATE	4X	Date when ODE started
RUNDATETIME	12X	Date and time when ODE started
RUNTIME	4X	Time when ODE started
SEGMENT_COUNT_DETAIL	4X x 127	Number of extracted segments for each segment type
SEGMENT_COUNT_DETAIL _SEG	4X x <i>n</i>	Number of extracted segments for each segment type defined in DBD
SEGMENT_COUNT_TOTAL	4X	Number of extracted segments
TODAY	4X	Date when ODE started

VL=YES

Request a length prefix.

VL=NO

Suppress a length prefix.

Default:

VL=YES

Site default support:

Not supported.

Restriction:

The FIELDS keyword can be specified only in FPXIN.

FPBP64U keyword

The FPBP64U keyword specifies whether to enable the Fast Path 64-bit buffer manager for the IMS High-Speed DEDB Direct Reorganization utility (DBFUHDR0).

FPBP64U is an optional keyword for the REORG statement of OER.

Format:



NO

Do not enable the Fast Path 64-bit buffer manager for the IMS High-Speed DEDB Direct Reorganization utility.

YES

Enable the Fast Path 64-bit buffer manager for the IMS High-Speed DEDB Direct Reorganization utility.

Default:

No, if the FPBP64U keyword is not specified.

Site default support:

Supported.

Restriction:

FPBP64U=YES is applicable only for IMS 14 and later.

FS_DOVFIOVF keyword

The FS_DOVFIOVF keyword specifies a threshold setting for the free space in DOVF and in IOVF.

If the free space in DOVF is greater than the value that you specified, and if the free space in IOVF is less than the value that you specified, a warning message is issued.

FS_DOVFIOVF is an optional keyword for the following statements:

- THRESHOLD statement of OPC
- THRESHOLD statement of OER

Format:

```
►► FS_DOVFIOVF= (value1,value2) ◄◄
```

value1,value2

An integer value in the range of 0 - 100. You must specify both values.

Default:

None.

Site default support:

Not supported.

Restriction:

The FS_DOVFIOVF keyword can be specified only in FPXIN.

FS_RAADOVF keyword

The FS_RAADOVF keyword specifies a threshold setting for the free space in RAA BASE and in DOVF.

If the free space in RAA BASE is greater than the value that you specified, and if the free space in DOVF is less than the value that you specified, a warning message is issued.

FS_RAADOVF is an optional keyword for the following statements:

- THRESHOLD statement of OPC
- THRESHOLD statement of OER

Format:

```
►► FS_RAADOVF= (value1,value2) ◄◄
```

value1,value2

An integer value in the range of 0 - 100. You must specify both values.

Default:

None.

Site default support:

Not supported.

Restrictions:

- This keyword is mutually exclusive with the FS_RAADOVFIOVF keyword.
- The FS_RAADOVF keyword can be specified only in FPXIN.

FS_RAADOVFIOVF keyword

The FS_RAADOVFIOVF keyword specifies a threshold setting for the free space in RAA BASE, DOVF, and IOVF.

If the free space in RAA BASE is greater than the value that you specified, and if the free space in either DOVF or IOVF is less than the value that you specified, a warning message is issued.

FS_RAADOVFIOVF is an optional keyword for the following statements:

- THRESHOLD statement of OPC
- THRESHOLD statement of OER

Format:

```
▶▶ FS_RAADOVFIOVF= (value1,value2,value3)  ◀◀
```

value1,value2,value3

An integer value in the range of 0 - 100. You must specify all values.

Default:

None.

Site default support:

Not supported.

Restrictions:

- This keyword is mutually exclusive with keywords FS_RAADOVF and FS_RAAIOVF.
- The FS_RAADOVFIOVF keyword can be specified only in FPXIN.

FS_RAAIOVF keyword

The FS_RAAIOVF keyword specifies a threshold setting for the free space in RAA BASE and in IOVF.

If the free space in RAA BASE is greater than the value that you specified, and if the free space in IOVF is less than the value that you specified, a warning message is issued.

FS_RAAIOVF is an optional keyword for the following statements:

- THRESHOLD statement of OPC
- THRESHOLD statement of OER

Format:

```
▶▶ FS_RAAIOVF= (value1,value2)  ◀◀
```

value1,value2

An integer value in the range of 0 - 100. You must specify both values.

Default:

None.

Site default support:

Not supported.

Restrictions:

- This keyword is mutually exclusive with the FS_RAADOVFIOVF keyword.
- The FS_RAAIOVF keyword can be specified only in FPXIN.

FULLSTEP keyword

The FULLSTEP keyword specifies whether OPC continues to process the pointer verification data collected from an online DEDB area.

FULLSTEP is an optional keyword that is used together with the TYPRUN keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement of OER

Format:



NO (or N)

OPC provides a fast scan of database integrity and collects the pointer verification data.

YES (or Y)

Depends on the TYPRUN keyword parameter as follows:

- When TYPRUN=PTRALL, OPC checks cross-reference of pointers.
- When TYPRUN=RPT, OPC performs all the functions executed in FULLSTEP=YES and TYPRUN=PTRALL mode, and produces complete analysis reports.
- When TYPRUN=FS, OPC produces free space analysis reports.

Default:

NO, if the FULLSTEP keyword is not specified.

Site default support:

Supported.

Restrictions:

- The FULLSTEP keyword can be specified only in FPXIN.
- A specification of YES is mutually exclusive with TYPRUN=PTRSCAN or TYPRUN=MODEL.

For the relationship among the TYPRUN keyword, the FULLSTEP keyword, and the required or the optional input and output DD, see [“Selecting the operating modes” on page 679](#).

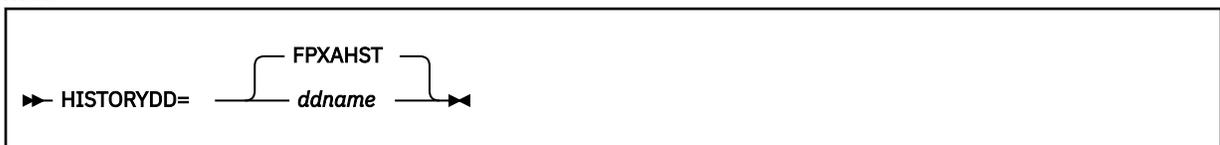
HISTORYDD keyword

The HISTORYDD keyword specifies the ddname of the DD statement for the output History file data sets that are specified in the JCL.

HISTORYDD is an optional keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement or OER

Format:



ddname

Specifies the DD name of the output data set or output stream to which History data is to be written.

If the DD statement for the DD name, determined explicitly or implicitly by the HISTORYDD keyword, is not coded in the JCL stream, the DD is allocated dynamically to the system output stream with SYSOUT=*.

Default:

FPXAHST, if the HISTORYDD keyword is not specified.

Site default support:

Supported.

Restrictions:

- z/OS DD name conventions.
- The HISTORYDD keyword can be specified only in FPXIN.
- The HISTORYDD keyword is effective only when both the TYPRUN=RPT keyword and the FULLSTEP=YES keyword are specified. Otherwise, it is ignored.

IC keyword

The IC keyword specifies whether to take an image copy while scanning the database area.

IC is an optional keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement of OER

The parameter *comprtn* and ICEIN are allowed only when IMS HP Image Copy is installed. If the keyword *comprtn* is omitted when IC=(YES,*comprtn*) is specified, Image Copy records are not compressed.

Format:



NO (or N)

Specifies that no image copy is to be taken.

YES (or Y)

Specifies that an image copy is to be taken.

comprtn

Specifies the name of the compression routines that are supported by IMS HP Image Copy.

Requirement: The load library of IMS HP Image Copy must be concatenated to the JOBLIB or the STEPLIB library.

ICEIN

Specifies that IMS HP Image Copy takes an image copy of the area.

Requirement: The load library of IMS HP Image Copy must be concatenated to the JOBLIB or the STEPLIB library.

For DASYSIN DD, you can specify this keyword without a value. IC or IC= is regarded as IC=YES.

Default:

NO, if the IC keyword is not specified.

Site default support:

Supported.

Note: In general, when using image copy (which produces identical results to IBM CIC (Concurrent Image Copy)), the following parameters are suggested:

- MAXERROR=1
- ERRACTION=ABEND

If there are any pointer errors, these parameters will cause the job to abend and not to register Image Copy with DBRC. Further analysis can then be undertaken.

Restrictions:

- IC=(YES,ICEIN) can be specified only in FPXIN DD.
- IC=YES or IC=(YES,*comprtn*) is not supported in multiple area processing.

ICEXTEND keyword

The ICEXTEND keyword activates the image copy extension option. When this option is activated, OAE extends the image copies during DEDB area extension. The extended image copy can be used to recover the DEDB area when necessary.

ICEXTEND is an optional keyword for the EXTEND statement of OAE original expansion.

Format:



NO (or N)

Do not extend the image copy data sets.

Important: Image copy data sets that were created before this OAE run cannot be used to recover the extended DEDB area. Create an image copy of the extended DEDB area when the OAE job ends.

YES (or Y)

Extend the latest image copy data set that is registered in the RECON data sets. The extended image copy data set can be used to recover the extended DEDB area.

Default:

NO, if the ICEXTEND keyword is not specified.

Site default support:

Not supported.

Restrictions:

- You can specify ICEXTEND=YES when one of the following conditions is met:
 - When you extend only the SDEP portion.
 - When you extend the IOVF portion and you are on IMS with APAR PI43928 or on IMS 15 or later.
- When the ICEXTEND=YES keyword is specified in FPXIN DD, you must include either or both of the FPXBIC0x DD statements in the JCL stream.

ICERR_CONT keyword

The ICERR_CONT keyword specifies whether to continue extending the DEDB area when image copy extension fails.

ICERR_CONT is an optional keyword for the EXTEND statement of OAE original expansion. This keyword is valid only when ICEXTEND is specified.

Format:



NO (or N)

Stop extending the DEDB area when image copy extension fails.

YES (or Y)

Continue to extend the DEDB area even when image copy extension fails.

Default:

NO, if the ICERR_CONT keyword is not specified.

Site default support:

Not supported.

ICNOTIFY keyword

The ICNOTIFY keyword specifies that image copies be registered to DBRC. DBRC registration of the area is required in order to DBRC register an image copy.

ICNOTIFY is an optional keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement of OER

Format:



NO (or N)

Specifies that no DBRC registration is to be performed.

YES (or Y)

Specifies that an image copy is to be registered to DBRC. If pointer errors are detected, the image copy is not registered.

For DASYSIN DD, you can specify this keyword without a value. ICNOTIFY or ICNOTIFY= is regarded as ICNOTIFY=YES.

Default:

NO, if the ICNOTIFY keyword is not specified.

Site default support:

Supported.

Note: The keyword that is shown in the processing messages FPX0203I and FPX0208I depends on the specification as follows:

- When neither the ICNOTIFY keyword nor the DBRC keyword is specified, ICNOTIFY is shown in the keyword column in the FPX0203I message and the FPX0208I message.
- When only the ICNOTIFY keyword is specified, ICNOTIFY is shown in the keyword column in the FPX0203I message and the FPX0208I message.
- When only the DBRC keyword is specified, ICNOTIFY (DBRC) is shown in the keyword column in the FPX0203I message and the FPX0208I message.
- When both the ICNOTIFY keyword and the DBRC keyword are specified, whether ICNOTIFY or ICNOTIFY (DBRC) is shown in the FPX0203I and the FPX0208I message is determined by the latter specification.

IC1OUTDD keyword

The IC1OUTDD keyword specifies the DD name of the output file to which the primary image copy data set is to be written.

IC1OUTDD is an optional keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement of OER

If this keyword is specified, it must be assigned a file name.

Format:



ddname

Specifies the DD name of the output file to which the primary image copy is to be written.

Default:

DFSUDUMP, if the IC1OUTDD keyword is not specified.

Site default support:

Supported.

Restriction:

- z/OS DD name conventions.
- IC1OUTDD and IC2OUTDD, if specified, must refer to different data sets.
- This keyword is mutually exclusive with IC=(YES,ICEIN).

IC2OUTDD keyword

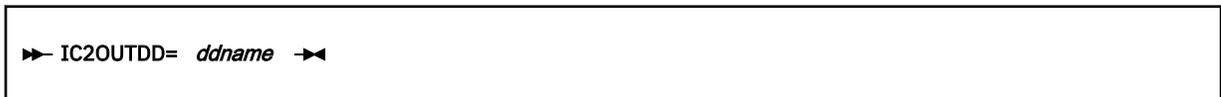
The IC2OUTDD keyword specifies the DD name of the output file to which the secondary image copy data set is to be written.

IC2OUTDD is an optional keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement of OER

If this keyword is omitted, the secondary image copy data set is not written to an output file. If the IC2OUTDD keyword is specified, it must be assigned a file name.

Format:



ddname

Specifies the DD name of the output file to which the secondary image copy is to be written.

Default:

None. If IC2OUTDD is not specified, no file for the secondary image copy data set is created.

Site default support:

Supported.

Restriction:

- z/OS DD name conventions.
- IC1OUTDD and IC2OUTDD, if specified, must refer to different data sets.
- This keyword is mutually exclusive with IC=(YES,ICEIN).

INDOUBT keyword

The INDOUBT keyword specifies whether in-doubt SDEP segments are to be selected.

The INDOUBT keyword pertains to in-doubt work units in a CICS®/DBCTL or ODBA environment in which the connection between CICS and DBCTL broke down between phase 1 and phase 2 commits.

INDOUBT is an optional keyword for the FPXCTL statement of ODE.

Format:



NO (or N)

Do not select in-doubt SDEP segments.

YES (or Y)

Select in-doubt SDEP segments.

For FPXCTL DD and *areaname* DD, you can specify this keyword without a value. INDOUBT or INDOUBT= is regarded as INDOUBT=YES.

Default:

NO, if the INDOUBT keyword is not specified.

Site default support:

Supported.

IOVFCI_RBA_REPORT keyword

The IOVFCI_RBA_REPORT keyword specifies whether to generate the RBA of Used IOVF CI report.

IOVFCI_RBA_REPORT is an optional keyword for the following statements:

- REPORT statement of OPC
- REPORT statement of OER

Format:



YES

Generate the RBA of Used IOVF CI report.

NO

Do not generate the RBA of Used IOVF CI report.

Default:

NO, if the IOVFCI_RBA_REPORT is not specified.

Site default support:

Supported.

Restriction:

The IOVFCI_RBA_REPORT keyword can be specified only in FPXIN.

IOVFFS_PCT keyword

The IOVFFS_PCT keyword specifies a threshold setting for the percentage of free space in IOVF.

If the percentage is less than the user-specified value, a warning message is issued.

IOVFFS_PCT is an optional keyword for the following statements:

- THRESHOLD statement of OPC
- THRESHOLD statement of OER

Format:

```
▶▶ IOVFFS_PCT= value ◀◀
```

value

An integer value in the range of 0 - 100.

Default:

None.

Site default support:

Not supported.

Restriction:

The IOVFFS_PCT keyword can be specified only in FPXIN.

IOVFPOOL keyword

The IOVFPOOL keyword specifies the number of IOVF buffers per buffer set.

In an OPC job, to allow for overlapped read/extract I/O, twice this number of buffers are actually obtained. The keyword should be required only for severely disorganized areas.

IOVFPOOL is an optional keyword for the following statements:

- ANALYZE statement of OPC
- FPXCTL statement of ODE
- PTRCHK statement of OER

If this keyword is specified, it must be assigned an integer value.

Format:

```
▶▶ IOVFPOOL= 10  
integer ◀◀
```

integer

Specifies the number of IOVF buffers per buffer set. The number you specify must be an integer value in the range of 5 - 50.

Default:

10, if the IOVFPOOL keyword is not specified.

Site default support:

Supported.

Restriction:

For ODE running with the run-level control, the IOVFPOOL value is used for all areas that are specified to be processed, regardless of the actual IOVF size of each area.

IOVFUSED_PCT keyword

The IOVFUSED_PCT keyword specifies a threshold setting for the percentage of IOVF CIs that are used.

If the percentage is greater than the value that you specified, a warning message is issued.

IOVFUSED_PCT is an optional keyword for the following statements:

- THRESHOLD statement of OPC
- THRESHOLD statement of OER

Format:

```
▶▶ IOVFUSED_PCT= value ▶▶
```

value

An integer value in the range of 0 - 100.

Default:

None.

Site default support:

Not supported.

Restriction:

The IOVFUSED_PCT keyword can be specified only in FPXIN.

IOVF_ROOT keyword

The IOVF_ROOT keyword specifies which expansion information is to be used when the IOVF portion is extended.

IOVF_ROOT is an optional keyword for the EXTEND statement of OAE. Either the IOVF_ROOT, the IOVF_UOW, or the SDEP_CI keyword must be specified.

Format:

This keyword syntax is the same as the ROOT parameter of the IMS DBD definition.

```
▶▶ IOVF_ROOT= ACBLIB (number2 ,overflow2 ) ▶▶
```

ACBLIB

Specifies to use the ROOT parameter in the database descriptors (DBDs) in the ACBLIB for the area(s) that is to be extended.

(number2,overflow2)

Specifies to change the ROOT parameter in the database descriptors (DBDs) for the area(s) that is to be extended, and uses the changed parameter.

Default:

None.

Site default support:

Not supported.

Restriction:

The IOVF_ROOT keyword is mutually exclusive with the IOVF_UOW keyword.

IOVF_UOW keyword

The IOVF_UOW keyword specifies the number of IOVF UOWs to expand when extending the IOVF portion.

The number of IOVF UOWs can be specified with a specific number or by a percentage of UOWs within the total number of UOWs defined in the IOVF portion.

IOVF_UOW is an optional keyword for the EXTEND statement of OAE. Either the IOVF_UOW, the IOVF_ROOT, or the SDEP_CI keyword must be specified.

Format:**integer**

An integer value in the range of 1 - 32766.

pct%

An integer value in the range of 1 - 100, followed by a % sign.

Default:

None.

Site default support:

Not supported.

Restriction:

The IOVF_UOW keyword is mutually exclusive with the IOVF_ROOT keyword.

ITKBDATA keyword

The ITKBDATA keyword specifies the type of additional data to store in the IMS Tools Base IMS Tools Knowledge Base repository.

ITKBDATA is an optional keyword for the GLOBAL statement of OER.

Format:**UTILHIST**

Store the utility history member for the OER job in the IMS Tools KB Sensor Data repository. The stored utility history member contains the reorganization timestamp, which indicates the time when the DEDB areas were reorganized.

***NO**

Do not store the utility history member.

To learn more about utility history, see the topic "Utility history environment" in the *Tools Base IMS Tools Knowledge Base User's Guide*.

Default:

*NO, if the ITKBDATA keyword is not specified.

Site default support:

Supported.

Restriction:

The ITKBDATA keyword can be specified only in FPXIN.

ITKBLOAD keyword

The ITKBLOAD keyword specifies the name of the load module library for IMS Tools KB.

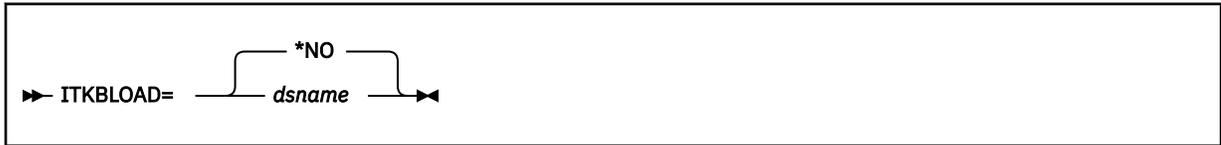
This keyword is used only when ITKBSRVR=*servername* is specified. If you do not explicitly specify this keyword, ITKBLOAD=*NO will be used and FPO attempts to load an IMS Tools KB module from the STEPLIB libraries.

ITKBLOAD is an optional keyword for the following statements:

- GLOBAL statement of OPC

- GLOBAL statement of OAE
- GLOBAL statement of OER
- GLOBAL statement of ODM

Format:



***NO**

The IMS Tools KB modules are loaded from the private library or the system library of the job.

dsname

Specifies the name of the load module library for IMS Tools KB.

Default:

*NO, if the ITKBLOAD keyword is not specified.

Site default support:

Supported.

Restrictions:

- The ITKBLOAD keyword can be specified only in FPXIN.
- ITKBLOAD=dsname is mutually exclusive with SENSOR=YES. When SENSOR=YES is specified, you must specify the IMS Tools Base load module library on the STEPLIB.

Related reference

FPO reports stored in IMS Tools KB

When ITKBSRVR=servername is specified, reports and messages of FPO can be stored in the output repository of IMS Tools KB.

ITKBSRVR keyword

The ITKBSRVR keyword specifies the name of the IMS Tools KB server XCF group.

ITKBSRVR is an optional keyword for the following statements:

- GLOBAL statement of OPC
- GLOBAL statement of OAE
- GLOBAL statement of OER
- GLOBAL statement of ODM

Format:



***NO**

FPO does not store reports in the IMS Tools KB repository.

servername

FPO stores reports in the IMS Tools KB Output repository of the specified server.

Default:

*NO, if the ITKBSRVR keyword is not specified.

Site default support:

Supported.

Restrictions:

- The ITKBSRVR keyword can be specified only in FPXIN.
- The name of the IMS Tools KB server XCF group must be specified when SENSOR=YES is specified.

Related reference

FPO reports stored in IMS Tools KB

When ITKBSRVR=*servername* is specified, reports and messages of FPO can be stored in the output repository of IMS Tools KB.

KEYWORD_DUPCHK keyword

The KEYWORD_DUPCHK keyword specifies to end the job with an error message when a duplicate keyword is found.

KEYWORD_DUPCHK is an optional keyword for the following statements:

- GLOBAL statement of OPC
- GLOBAL statement of ODE
- GLOBAL statement of OER
- GLOBAL statement of ODM

Format:**NO (or N)**

When a duplicate keyword is found, applies only the last instance of the keyword and continues the job.

YES (or Y)

When a duplicate keyword is found, ends the job with an error message.

Default:

NO, if the KEYWORD_DUPCHK keyword is not specified.

Site default support:

Supported.

Restrictions:

- The KEYWORD_DUPCHK keyword can be specified only in FPXIN.
- If any of the following keywords are specified more than once, the job ends with an error regardless of the KEYWORD_DUPCHK keyword parameter.
 - For OPC, ODE, OER and ODM, keywords that are associated with the OVERRIDE control statement.
 - For ODE:
 - The SEGREC_PREFIX keyword and the SEGREC_SUFFIX keyword of the FPXCTL control statement.
 - The SEG keyword of the OUTPUT control statement.
 - The SEG keyword and the FIELD keyword of the SELECT control statement.
 - The BREAK keyword and the FIELDS keyword of the USER_RECORD control statement. For the BREAK keyword, the job ends with an error only when the same parameters are specified.

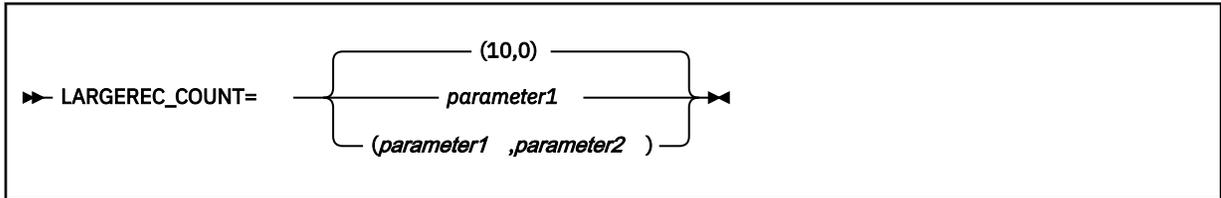
LARGEREC_COUNT keyword

The LARGEREC_COUNT keyword specifies the number of reporting entries in the Largest Database Records report and the Consolidated Largest Database Records report.

LARGEREC_COUNT is an optional keyword for the following statements:

- REPORT statement of OPC
- REPORT statement of OER

Format:



parameter1

Specify the number of reporting entries in the Largest Database Records report. The value is in the range of 0 - 9999. If 0 is specified, the Largest Database Records report is not generated.

parameter2

Specify the number of reporting entries in the Consolidated Largest Database Records report. The value is in the range of 0 - 9999. If 0 is specified, the Consolidated Largest Database Records report is not generated.

Default:

(10,0), if the LARGEREC_COUNT keyword is not specified.

Site default support:

Supported.

Restrictions:

- The LARGEREC_COUNT keyword can be specified only in FPXIN.
- LARGEREC_COUNT=(0,0) is mutually exclusive with LARGEREC_REPORT=YES.

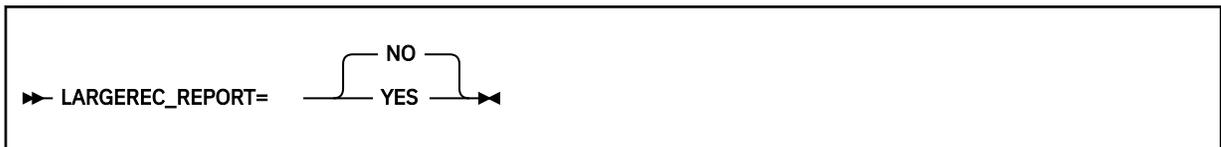
LARGEREC_REPORT keyword

The LARGEREC_REPORT keyword specifies whether to generate the Largest Database Records report and the Consolidated Largest Database Records report.

LARGEREC_REPORT is an optional keyword for the following statements:

- REPORT statement of OPC
- REPORT statement of OER

Format:



YES (or Y)

Generate the Largest Database Records report and the Consolidated Largest Database Records report.

NO (or N)

Do not generate the Largest Database Records report and the Consolidated Largest Database Records report.

Default:

NO, if the LARGEREC_REPORT keyword is not specified.

Site default support:

Supported.

Restriction:

The LARGEREC_REPORT keyword can be specified only in FPXIN DD.

MAXBUFFS keyword

The MAXBUFFS keyword specifies the number of buffer sets that are used in the reorganization.

The meaning of MAXBUFFS is the same as that of the BUFNO command for the High-Speed DEDB Direct Reorganization Utility (DBFUHDR0).

MAXBUFFS is an optional keyword for the REORG statement of OER.

Format:**integer**

An integer value in the range of 3 - 999.

Default:

3, if the MAXBUFFS keyword is not specified.

Site default support:

Supported.

MAXERROR keyword

The MAXERROR keyword specifies the maximum number of pointer errors that might be encountered before processing ends abnormally.

The number of errors might be related to:

- True pointer problems
- False problems due to concurrent online update activity

The ERRORACT keyword specifies the type of termination to take place when the MAXERROR threshold is reached. If ERRORACT is ABEND and the MAXERROR value is reached, the program ends with user abend 3510.

If you reach the MAXERROR threshold, determine the reason for the number of pointer problems. Either increase the MAXERROR threshold or take the area offline for further analysis.

When multiple areas are processed, the MAXERROR threshold is applied for each area.

MAXERROR is an optional keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement of OER

If this keyword is specified, it must be assigned an integer value.

Format:

integer

Specifies the maximum number of pointer errors that might be encountered before OPC ends. The number you specify must be an integer value in the range of 0 - 999. If 0 is specified, there is no maximum number of errors.

Default:

500, if the MAXERROR keyword is not specified.

Site default support:

Supported.

Restriction:

If OPC processing ends prematurely due to reaching the MAXERROR threshold, the output files DAR120, DAR13SO and DAR3CSO can be incomplete.

MAXFAILS keyword

The MAXFAILS keyword specifies the maximum allowable number of UOWs that fail to be reorganized, before the utility ends abnormally.

If the number of failures has exceeded the number specified in the MAXFAILS keyword, the reorganization processing is stopped and OER stops its processing or abends depending on the value specified for the ERRORACT keyword.

If PTRCHK=YES is specified with CHKUOW=REORGED for the PTRCHK control statement, and the MAXFAILS number has not been reached, only those UOWs that have been actually reorganized are checked by OPC. UOWs that caused errors are excluded from the list of UOWs that are to be checked by OPC.

MAXFAILS is an optional keyword for the REORG statement of OER.

Format:**integer**

An integer value in the range of 0 - 999.

If MAXFAILS is specified as 0, then no limit is set and the reorganization continues the attempt to reorganize the next UOW until all UOW in the area has been processed or the reorganization limit, if it is specified, is reached.

The failed UOWs will be reported in the SYSPRINT or the MFILE stream. The failed UOWs are also marked in the UOW Space Statistics report.

Default:

5, if the MAXFAILS keyword is not specified.

Site default support:

Supported.

Related reference

[FPXRPTS DD data set](#)

The Reorganization Processing Summary report, the UOW Space Statistics report, and the Process Summary report are produced in the data set or written to the output stream that is determined by the RFILE keyword of the REORG control statement.

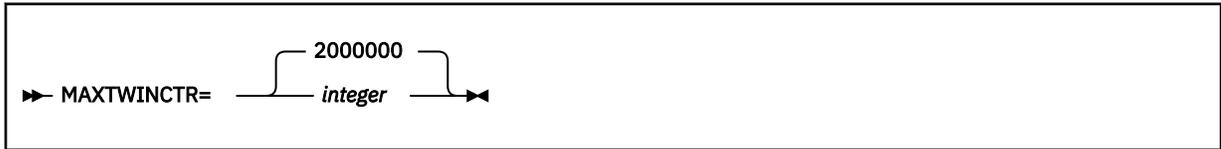
MAXTWINCTR keyword

The MAXTWINCTR keyword specifies the threshold value for the number of twin segments.

MAXTWINCTR is an optional keyword for the following statements:

- ANALYZE statement of OPC
- FPXCTL statement of ODE
- PTRCHK statement of OER

Format:



integer

Specifies the threshold value for the number of twin segments. The number you specify must be an integer value in the range of 2000000 - 9999999.

Default:

2000000, if the MAXTWINCTR keyword is not specified.

Site default support:

Supported.

MAXVOLS keyword

The MAXVOLS keyword specifies the maximum number of volumes for the extended original ADS. This keyword is effective only in SMS environments.

MAXVOLS is an optional keyword for the EXTEND statement of OAE.

Format:



integer

An integer value in the range of 1 - 59.

Default:

2, if both the MAXVOLS keyword and the VOLUME_EXT keyword are not specified.

Site default support:

Not supported.

Restrictions:

- The MAXVOLS keyword can be specified only in FPXIN.
- This keyword is mutually exclusive with the VOLUME_EXT keyword.

MFILE keyword

The MFILE keyword specifies the DD name of the output file to which utility messages are to be written.

The output can go to a data set or to a printer, routed through the output stream.

MFILE is an optional keyword for the following statements:

- ANALYZE statement of OPC
- FPXCTL statement of ODE
- REORG statement of OER
- DMAPRT statement of ODM

The OFILE keyword is optional for both types of processing. If neither FPXOFILE DD nor ddname DD specified by the OFILE keyword is present, a terminating error condition results. If both are present, the OFILE keyword supersedes the FPXOFILE specification.

Site default support:

Supported.

Restriction:

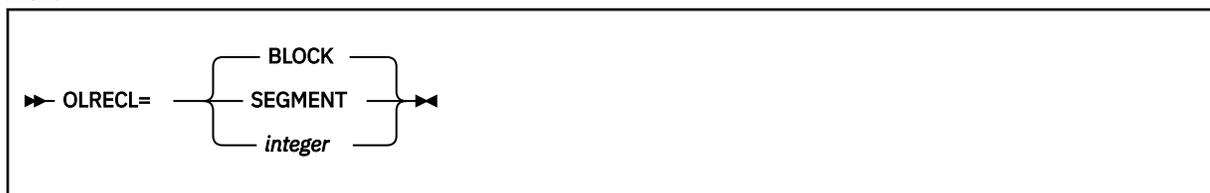
z/OS DD name conventions.

OLRECL keyword

The OLRECL keyword specifies the LRECL of FPXOFILE.

OLRECL is an optional keyword for the FPXCTL statement of ODE. If this keyword is specified, it must be assigned a value.

Format:



BLOCK

The LRECL of FPXOFILE is determined as BLKSIZE-4. The BLKSIZE is determined according to the device type of the data set.

SEGMENT

The LRECL of FPXOFILE is determined based on the following factors:

- The maximum segment length that is defined in the DBD.
- The SEG keyword on the OUTPUT statement, the FIELDS keyword on the OUTPUT statement, or both.

integer

The LRECL of FPXOFILE is determined by the specified integer if the value is available. The integer you specify must be in the range of 5 through 32760.

Default:

BLOCK, if the OLRECL keyword is not specified.

Site default support:

Supported.

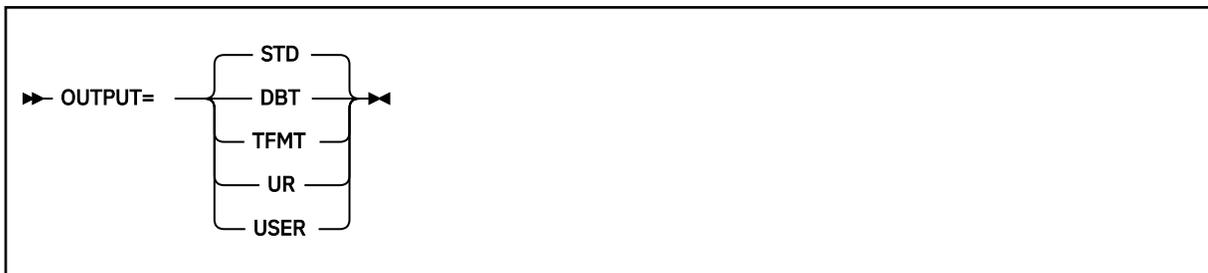
Restrictions:

- You cannot have multiple values for the OLRECL keyword. If you specify multiple values, only the last valid occurrence of the OLRECL keyword is used.
- OLRECL=SEGMENT is internally changed to OLRECL=BLOCK when OUTPUT=TFMT, OUTPUT=UR, or OUTPUT=USER is specified.
- When the integer you specified is unavailable, that is, larger than BLKSIZE-4, ODE uses the default value BLKSIZE-4.

OUTPUT keyword

The OUTPUT keyword specifies the format of the output file.

The OUTPUT keyword is an optional keyword for the FPXCTL statement of ODE. If this keyword is specified, it must be assigned an appropriate value.

Format:**STD (or S)**

Standard extract format is to be used.

DBT (or D)

FPB Unload/Reload DBT format is to be used.

TFMT (or T)

FPB Unload/Reload TFMT format is to be used. The format of the unloaded segment records is enhanced, which means that the prefix part of the record is generated based on the maximum number of segment levels that are defined in DBD.

UR (or U)

TRIMAR unload/reload format is to be used.

USER

Allows more granular control of the output file. This format should be used when an existing application program requires extracted data in a specific format.

Note: If ODE is run by FPXIN DD, a DBD Control Information record is created for each output file. If ODE is run by SYSIN DD, the number of DBD Control Information records that are written to the output file is the same as the number of areas that are written to the file, and the records are written at the beginning of the output data of each area.

Default:

STD, if the OUTPUT keyword is not specified.

Site default support:

Supported.

Restrictions:

- A specification of TFMT or UR is mutually exclusive with SORT=YES.
- If OUTPUT=USER and SORT=YES are specified, SORT=YES is internally changed to SORT=NO.
- If OUTPUT=DBT/TFMT/UR is specified, all segments in the hierarchic path, which includes the selected segment and all of its parent segments, must be selected for output.
- The extraction of SDEP segments depends on the OUTPUT, SORT, and SDEPHIER specifications. The following table shows the SDEP segments extraction.

Table 111. SDEP segments extraction

OUTPUT	SORT	SDEPHIER	SDEP segment extraction
STD	YES	YES	If ODESORT DD is specified, SDEP segments are extracted hierarchically. If ODESORT DD is not specified, ODE issues message FPX0002E and ends with an error.
		NO	SDEP segments cannot be extracted.
NO	NO	YES	SDEP segments are extracted hierarchically.
		NO	SDEP segments are extracted physically.

Table 111. SDEP segments extraction (continued)

OUTPUT	SORT	SDEPHIER	SDEP segment extraction
DBT	YES	YES	If ODESORT DD is specified, SDEP segments are extracted hierarchically. If ODESORT DD is not specified, ODE issues message FPX0002E and ends with an error.
		NO	SDEP segments cannot be extracted.
	NO	YES	SDEP segments are extracted hierarchically.
		NO	SDEP segments cannot be extracted.
TFMT	YES	-	Keyword error.
	NO	YES	Keyword error.
		NO	SDEP segments cannot be extracted.
UR	YES	-	Keyword error.
	NO	YES	Keyword error.
		NO	SDEP segments cannot be extracted.
USER	Yes	-	Internally changed to SORT=NO.
	NO	YES	SDEP segments are extracted hierarchically.
		NO	SDEP segments are extracted physically.

Related reference

[Extracted records layout](#)

Extracted records are created by the FPA Extract function and the FPO ODE utility.

PCHKONLY keyword

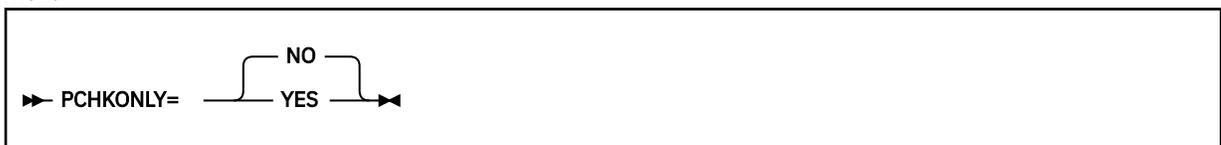
The PCHKONLY keyword specifies whether to evaluate only the control statement keywords or to run the functions that are designated by the control statements and their parameters.

For OPC and ODE, regardless of PCHKONLY specification, the utility gets connection to each area to obtain information about the area. For OAE, OER, and ODM, if PCHKONLY=YES is specified, no connection to any area will be made.

PCHKONLY is an optional keyword for the following statements:

- ANALYZE statement of OPC
- FPXCTL statement of ODE
- EXTEND statement of OAE
- REORG statement of OER
- DMAPRT statement of ODM

Format:



NO (or N)

Processing continues after evaluating the site default options and the utility control statements.

YES (or Y)

Processing ends after evaluating the site default options and the utility control statements.

For DASYSIN DD, FPXCTL DD, and *areaname* DD, you can specify this keyword without a value. PCHKONLY or PCHKONLY= is regarded as PCHKONLY=YES.

Default:

NO, if the PCHKONLY keyword is not specified.

Site default support:

Supported for OPC, ODE, OER and OER. Not supported for OAE.

PROC_AFT_ERROR keyword

The PROC_AFT_ERROR keyword specifies whether to end the job step when an area-specific error is encountered or to continue processing subsequent areas.

PROC_AFT_ERROR is an optional keyword for the following statements:

- ANALYZE statement of OPC
- EXTRACT statement of ODE
- REORG statement of OER
- DMACPRT statement of ODM

If you specify this keyword, you must specify a variable.

Format:



STOP (or S)

Specifies that processing ends when an area-specific error is encountered.

NEXTAREA (or N)

Specifies that processing continues to process the subsequent areas even when an area-specific error is encountered.

Default:

STOP, if PROC_AFT_ERROR keyword is not specified.

Site default support:

Supported.

Note: After you apply APAR PM07840, the job step stops when an error is encountered because the default value for PROC_AFT_ERROR is STOP. If you want ODM to work as before—that is, to have ODM continue processing subsequent areas even when an area-specific error is encountered—you must specify PROC_AFT_ERROR=NEXTAREA.

Restrictions:

- The PROC_AFT_ERROR keyword can be specified only in FPXIN DD.
- The PROC_AFT_ERROR keyword is ignored when only one area is specified in the AREA keyword.
- Specification of PROC_AFT_ERROR=NEXTAREA is mutually exclusive with ERRORACT=ABEND.
- PROC_AFT_ERROR=NEXTAREA is effective only for the errors that are specific to an area. For other errors, even if PROC_AFT_ERROR=NEXTAREA is specified, the processing ends with errors or ends abnormally without processing the subsequent areas.

PTRCHK keyword

The PTRCHK keyword specifies whether to run OPC after the area is reorganized.

The PTRCHK keyword is an optional keyword for the REORG statement of OER.

Format:**YES (or Y)**

Specifies that OPC be called.

NO (or N)

Specifies that OPC not be called.

Default:

NO, if the PTRCHK keyword is not specified.

Site default support:

Supported.

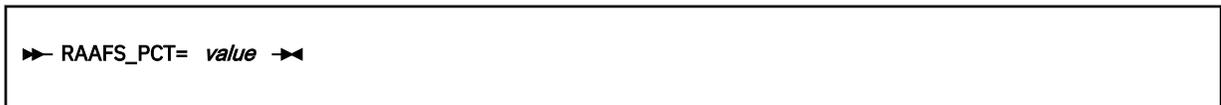
RAAFS_PCT keyword

The RAAFS_PCT keyword specifies a threshold setting for the percentage of free space in RAA BASE.

If the percentage is less than the value that you specified, a warning message is issued.

RAAFS_PCT is an optional keyword for the following statements:

- THRESHOLD statement of OPC
- THRESHOLD statement of OER

Format:**value**

An integer value in the range of 0 - 100.

Default:

None.

Site default support:

Not supported.

Restriction:

The RAAFS_PCT keyword can be specified only in FPXIN.

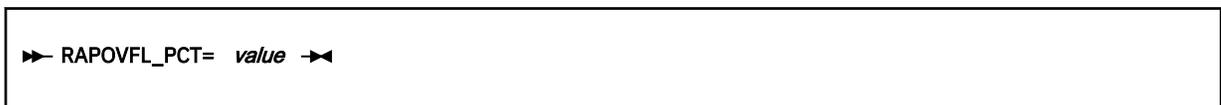
RAPOVFL_PCT keyword

The RAPOVFL_PCT keyword specifies a threshold setting for the percentage of RAA BASE CIs that use either DOVF or IOVF.

If the percentage is greater than the value that you specified, a warning message is issued.

RAPOVFL_PCT is an optional keyword for the following statements:

- THRESHOLD statement of OPC
- THRESHOLD statement of OER

Format:**value**

An integer value in the range of 0 - 100.

Default:

None.

Site default support:

Not supported.

Restriction:

The RAPOVFL_PCT keyword can be specified only in FPXIN.

RBASEFS keyword

The RBASEFS keyword specifies a criterion to select UOWs to reorganize.

RBASEFS is an optional keyword for the REORG statement of OER.

Format:***rbasefs***

Specifies the percentage of free space in each RAP CI. If a RAP CI contains free space that is greater than the specified percentage, and the RAP CI also uses overflow CIs, the UOW is selected as a candidate for the reorganization.

Specify an integer value 0 - 100, both inclusive. The maximum value 100 has a special meaning; if RBASEFS=100 is specified, the RBASEFS condition is not used to select an initial set of UOWs.

For details of how the number of UOWs to be reorganized is determined, see [“The number of UOWs to be reorganized”](#) on page 781.

Default:

100, if the RBASEFS keyword is not specified.

Site default support:

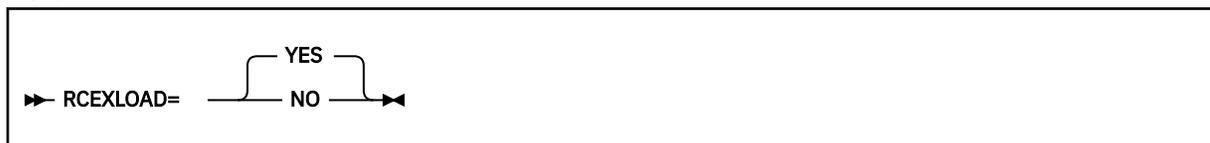
Supported.

RCEXLOAD keyword

The RCEXLOAD keyword specifies whether to call the return code exit routine FPXRCEX0 from the STEPLIB or JOBLIB data set concatenation.

The RCEXLOAD keyword is an optional keyword for the following statements:

- ANALYZE statement of OPC
- EXTRACT statement of ODE
- REORG statement of OER

Format:**YES (or Y)**

FPXRCEX0 is called if it is found in the STEPLIB or JOBLIB data set concatenation.

NO (or N)

FPXRCEX0 is not called.

Default:

YES, if the RCEXLOAD keyword is not specified.

Site default support:

Supported.

Restriction:

The RCEXLOAD keyword can be specified only in FPXIN.

RDOVFFS keyword

The RDOVFFS keyword specifies a criterion to select UOWs to reorganize.

RDOVFFS is an optional keyword for the REORG statement of OER.

Format:***rdovffs***

Specifies the percentage of free space in the dependent overflow area in each UOW. If the free space percentage in the dependent overflow area in a UOW is greater than the specified percentage, and the UOW also uses IOVF CIs, the UOW is selected as a candidate for the reorganization.

Specify an integer value in the range of 0 - 100. The maximum value 100 has a special meaning; if RDOVFFS=100 is specified, the RDOVFFS condition is not used to select an initial set of UOWs.

For details of how the number of UOWs to be reorganized is determined, see [“The number of UOWs to be reorganized”](#) on page 781.

Default:

0, if the RDOVFFS keyword is not specified.

Site default support:

Supported.

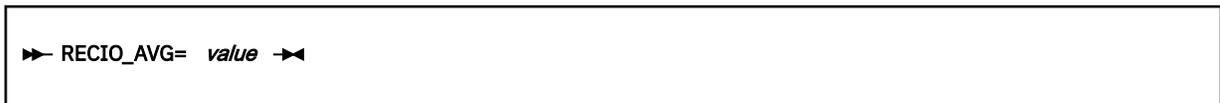
RECIO_AVG keyword

The RECIO_AVG keyword specifies a threshold setting for the average number of I/Os that can be done to access a database record.

If the number is greater than the value that you specified, a warning message is issued.

RECIO_AVG is an optional keyword for the following statements:

- THRESHOLD statement of OPC
- THRESHOLD statement of OER

Format:***value***

An integer value in the range of 0 - 32767.

Default:

None.

Site default support:

Not supported.

Restriction:

The RECIO_AVG keyword can be specified only in FPXIN.

RECIO_MAX keyword

The RECIO_MAX keyword specifies a threshold setting for the maximum number of I/Os that can be done to access a database record.

If the number is greater than the value that you specified, a warning message is issued.

RECIO_MAX is an optional keyword for the following statements:

- THRESHOLD statement of OPC
- THRESHOLD statement of OER

Format:

```
►► RECIO_MAX= value ◄◄
```

value

An integer value in the range of 0 - 32767.

Default:

None.

Site default support:

Not supported.

Restriction:

The RECIO_MAX keyword can be specified only in FPXIN.

RECIOVF_PCT keyword

The RECIOVF_PCT keyword specifies a threshold setting for the percentage of records that use IOVF CIs.

If the percentage is greater than the value that you specified, a warning message is issued.

RECIOVF_PCT is an optional keyword for the following statements:

- THRESHOLD statement of OPC
- THRESHOLD statement of OER

Format:

```
►► RECIOVF_PCT= value ◄◄
```

value

An integer value in the range of 0 - 100.

Default:

None.

Site default support:

Not supported.

Restriction:

The RECIOVF_PCT keyword can be specified only in FPXIN.

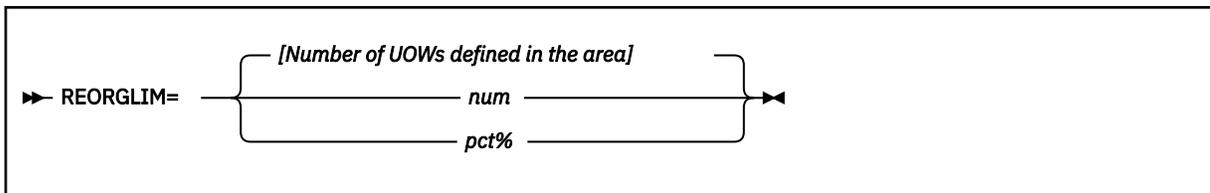
REORGLIM keyword

The REORGLIM keyword specifies the reorganization limit for the amount of UOWs that can be reorganized in the job.

The limit can be specified in the form of a maximum number of UOWs or a maximum percentage of the number of UOWs within the total number of UOWs defined in the area.

REORGLIM is an optional keyword for the REORG statement of OER.

Format:



num

A non-negative integer value less than 32,767.

pct%

A non-negative integer value less than or equal to 100 that is followed by a % sign.

If a non-negative integer value is specified, this keyword specifies the maximum number of UOWs that can be reorganized. If the number exceeds the total number of UOWs that is defined in the area, the total number of UOWs is used as the reorganization limit. If a non-negative integer value followed by a % sign is specified, this keyword specifies the maximum percentage of the number of UOWs within the total number of UOWs that is defined in the area. The number must be less than or equal to 100.

To print the UOW Space Statistics report without reorganizing a UOW, specify REORGLIM=0.

If RFSSCAN=(NO,NO) or RFSSCAN=(NO,YES) is specified, the REORGLIM keyword cannot be specified.

Default:

If the REORGLIM keyword is not specified, the number of UOWs defined in the area.

Site default support:

Supported.

REORGMSG keyword

The REORGMSG keyword specifies whether to generate detailed reorganization messages.

REORGMSG is an optional keyword for the REORG statement of OER.

Format:



QUIET

Suppress detailed reorganization messages.

VERBOSE

Generate reorganization messages.

Default:

QUIET, if the REORGMSG keyword is not specified.

Site default support:

Supported.

REPORTDD keyword

The REPORTDD keyword specifies the DD name of the output data set or output stream, to which OPC writes several reports.

The reports that are written by OPC or OER are:

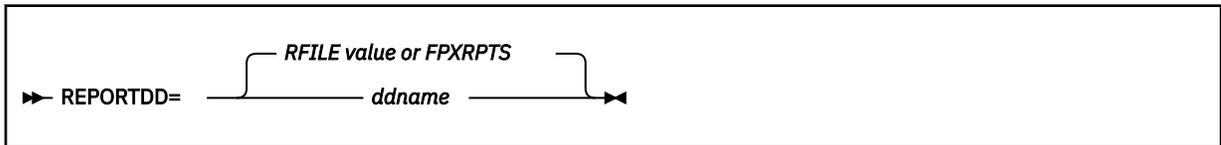
- DEDB Area Analysis reports
- UOW report
- UOW Exception report

- Largest Database Records report
- Consolidated Largest Database Records report
- Segment Length Distribution report
- DB Record Length Distribution report
- RBA of Used IOVF CI report

REPORTDD is an optional keyword for the following statements:

- REPORT statement of OPC
- REPORT statement of OER

Format:



ddname

Specifies the DD name of the output data set or the output stream, to which OPC writes the reports.

Default:

The DD name that is specified by the RFILE keyword. If the RFILE keyword is not specified, the default value is FPXRPTS.

Site default support:

Supported.

Restrictions:

- z/OS DD name conventions.
- The REPORTDD keyword can be specified only in FPXIN.

RFILE keyword

The RFILE keyword specifies the DD name of the output file to which the utility reports are to be written.

The output can go to a data set or to a printer, routed through the output stream.

RFILE is an optional keyword for the following statements:

- ANALYZE statement of OPC
- FPXCTL statement of ODE
- REORG statement of OER
- DMACPRT statement of ODM

If this keyword is specified, it must be assigned a DD name.

Format:



ddname

Specifies the DD name of the output data set or output stream to which the utility reports are to be written.

For FPXIN DD, if the DD statement for the DD name, determined explicitly or implicitly by the RFILE keyword, is not coded in the JCL stream, the DD is allocated dynamically to the system output stream with SYSOUT=*

Default:

FPXRPTS, if the RFILE keyword is not specified.

Site default support:

Supported.

Restrictions:

- z/OS DD name conventions.
- The RFILE keyword can be specified only in FPXIN.

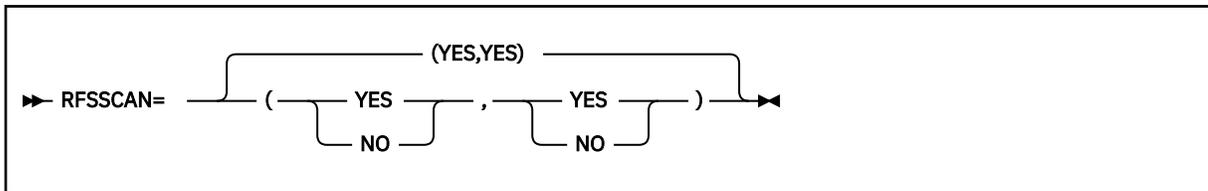
RFSSCAN keyword

The RFSSCAN keyword specifies whether to gather the statistics before and after reorganization by using Area Sensor.

The RFSSCAN keyword takes two parameters that are enclosed by a pair of parenthesis and separated by a comma; the first parameter specifies whether the statistics are gathered to evaluate UOW and the second parameter specifies whether the statistics are gathered for UOWs that have been reorganized.

The UOW Evaluation and UOW Ranking functions of OER are enabled only when the first parameter is YES. If the first parameter is NO, these functions are not enabled and all UOWs become candidates for the reorganization and the reorganization is attempted from the first UOW through the last UOW in the ascending order of the UOW number.

RFSSCAN is an optional keyword for the REORG statement of OER.

Format:

You can specify Y for YES, and N for NO.

Default:

RFSSCAN=(YES,YES), if the RFSSCAN keyword is not specified.

Site default support:

Supported.

RFSSTAT keyword

The RFSSTAT keyword specifies whether to generate the UOW Space Statistics report before and after reorganization.

The statistics report is written to a data set or a SYSOUT stream identified by the RFILE keyword parameter.

RFSSTAT is an optional keyword for the REORG statement of OER.

Format:**YES (or Y)**

Specifies that the statistics be generated.

NO (or N)

Specifies that the statistics not be generated.

In order for RFSSTAT=YES to be effective, at least one of the first and second parameters of the RFSSCAN keyword must be YES. If both the first and second parameters of RFSSCAN are NO, RFSSTAT=YES is ignored.

If RFSSTAT=YES is effective, and both the first and second parameters of the RFSSCAN keyword are YES, the statistics before and after the reorganization are reported. If the first or second parameter of RFSSCAN is not YES, the statistics are reported for the phase for which YES is specified. There is an exception. When REORGLIM=0 or REORGLIM=0% is specified and RFSSCAN=(YES,YES) or RFSSCAN=(YES,NO) is specified, the space statistics is printed for those UOWs that satisfy either the RBASEFS or the RDOVFFS condition.

Default:

YES, if the RFSSTAT keyword is not specified.

Site default support:

Supported.

Related reference

[FPXRPTS DD data set](#)

The Reorganization Processing Summary report, the UOW Space Statistics report, and the Process Summary report are produced in the data set or written to the output stream that is determined by the RFILE keyword of the REORG control statement.

RGROUP keyword

The RGROUP keyword sets the group identifier for the statistics that are collected and stored in the FPXAHST2 file or in a sensor data record set. For the FPXAHST2 file, the keyword value is set in the Repository Group field, and for a sensor data record set, the value is set for the DB_SENSOR_DATA_GROUP_ID data element.

RGROUP is an optional keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement of OER

Format:

```
▶▶ RGROUP= parameter ◀◀
```

parameter

Specify a 1- to 4-character repository group name.

Default:

None. If this keyword is not specified, FPO sets the group identifier as follows:

- The Repository Group field in the History2 file contains 4 bytes of blank characters.
- The DB_SENSOR_DATA_GROUP_ID data element is not included in the sensor data record set.

Site default support:

Supported.

Restriction:

The RGROUP keyword can be specified only in FPXIN.

ROOTIO_AVG keyword

The ROOTIO_AVG keyword specifies a threshold setting for the average number of I/Os to get access to a root segment.

If the number is greater than the value that you specified, a warning message is issued.

ROOTIO_AVG is an optional keyword for the following statements:

- THRESHOLD statement of OPC

- THRESHOLD statement of OER

Format:

```
▶▶ ROOTIO_AVG= value ▶▶
```

value

An integer value in the range of 0 - 32767.

Default:

None.

Site default support:

Not supported.

Restriction:

The ROOTIO_AVG keyword can be specified only in FPXIN.

ROOTIO_MAX keyword

The ROOTIO_MAX keyword specifies a threshold setting for the maximum number of I/Os that can be done to get access to a root segment.

If the number is greater than the value that you specified, a warning message is issued.

ROOTIO_MAX is an optional keyword for the following statements:

- THRESHOLD statement of OPC
- THRESHOLD statement of OER

Format:

```
▶▶ ROOTIO_MAX= value ▶▶
```

value

An integer value in the range of 0 - 32767.

Default:

None.

Site default support:

Not supported.

Restriction:

The ROOTIO_MAX keyword can be specified only in FPXIN.

RUNTYPE keyword

The RUNTYPE keyword specifies the run mode type.

RUNTYPE is an optional keyword for the EXTEND statement of OAE.

Format:

```
▶▶ RUNTYPE= {
  NORMAL
  TEST
  RESTART
} ▶▶
```

NORMAL

Specifies to do a processing run.

TEST

Specifies to do a trial run. All the input parameters are only verified for correctness. Output statistics are generated showing the structure of the new area without actually performing the expansion.

RESTART

Specifies to do a restart run.

For a discussion of when you would want to specify the RESTART keyword, see [“Recovering and restarting OAE” on page 758](#).

Default:

NORMAL, if the RUNTYPE keyword is not specified.

Site default support:

Not supported.

SDEP keyword

The SDEP keyword lets you specify whether or not run totals and SDEP pointer integrity is to be verified and any errors reported.

The keyword is ignored if the database area does not contain SDEPs.

SDEP is an optional keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement of OER

If this keyword is omitted, the SDEP pointer integrity is not verified.

Format:**NO (or N)**

Do not perform SDEP pointer integrity checking.

YES (or Y)

Perform SDEP pointer integrity checking and report any errors.

For DASYSIN DD, you can specify this keyword without a value. SDEP or SDEP= is regarded as SDEP=YES.

Default:

NO, if the SDEP keyword is not specified.

Site default support:

Supported.

Related reference

[Determining how to process SDEP segments](#)

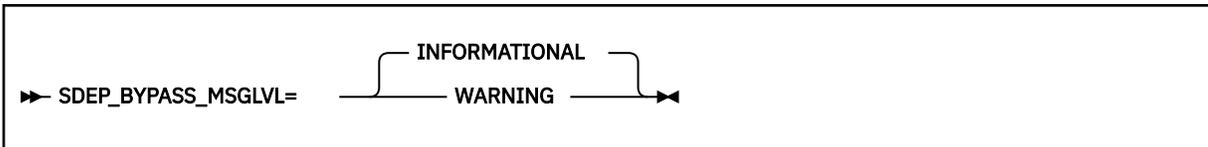
IMS HP Fast Path Utilities provides several utilities to process SDEP segments. You can use these utilities to analyze and maintain SDEP segments.

SDEP_BYPASS_MSGLVL keyword

The SDEP_BYPASS_MSGLVL keyword specifies the severity level of the message that the utility issues when the SDEP is defined but SDEP segment data is not extracted.

SDEP_BYPASS_MSGLVL is an optional keyword for the FPXCTL statement of ODE.

Format:



INFORMATIONAL

Issues informational message FPX0356I when the SDEP is defined but SDEP segment data is not extracted.

WARNING

Issues warning message FPX0357W when the SDEP is defined but SDEP segment data is not extracted. The utility sets the return code to 4 and continues processing.

Default:

INFORMATIONAL, if the SDEP_BYPASS_MSGLVL keyword is not specified.

Site default support:

Supported.

SDEP_CI keyword

The SDEP_CI keyword specifies the number of SDEP CIs to expand when extending the SDEP portion.

The number of SDEP CIs can be specified with a specific number or by a percentage of CIs within the total number of CIs defined in the SDEP portion.

SDEP_CI is an optional keyword for the EXTEND statement of OAE. Either the SDEP_CI, the IOVF_UOW, or the IOVF_ROOT keyword must be specified.

Format:



integer

An integer value in the range of 1 - 8388608.

pct%

An integer value in the range of 1 - 100, followed by a % sign.

Default:

None.

Site default support:

Not supported.

Related reference

[Determining how to process SDEP segments](#)

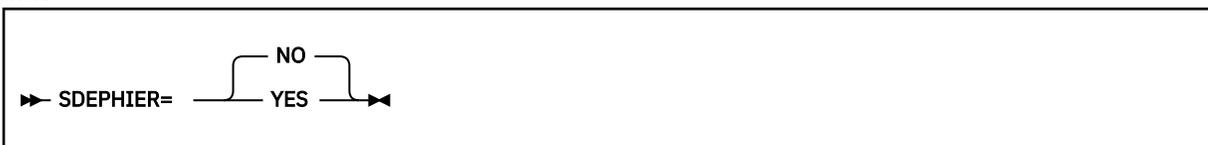
IMS HP Fast Path Utilities provides several utilities to process SDEP segments. You can use these utilities to analyze and maintain SDEP segments.

SDEPHIER keyword

The SDEPHIER keyword specifies whether SDEP segment records should be extracted hierarchically.

SDEPHIER is an optional keyword for the FPXCTL statement of ODE.

Format:



- Integrity verification message that identifies the RBA of the segment whose SPCF or SPTF pointer points to the IOVF portion in the Online Pointer Scan report

NOMSG (or N)

Set the return code to 0 and generate no messages about the pointer.

Default:

ERROR, if the SDEP_PTRIOVF_MSGLVL keyword is not specified.

Site default support:

Supported.

Related reference

Determining how to process SDEP segments

IMS HP Fast Path Utilities provides several utilities to process SDEP segments. You can use these utilities to analyze and maintain SDEP segments.

SEG keyword

For the SELECT statement, the SEG keyword specifies what segments are to be selected. For the OUTPUT statement, the SEG keyword specifies which segments are to be included in the output.

SEG is a required keyword for the following statements:

- SELECT statement of ODE
- OUTPUT statement of ODE

If the keyword is specified, it must be assigned a segment name.

Format:



segname

For the SELECT statement, selects the specified segment (segname is the name defined in the DBD). The same segment can be specified more than once, in which case the selection criteria are applied in an OR fashion; this makes a range of matches possible.

For the OUTPUT statement, includes the specified segment in the output (segname is the name in the DBD). Only one specification per segment name is allowed. If no corresponding SELECT statement is specified, an associated SELECT SEG=*segname* without any keywords is assumed.

parm

This parameter is valid only for the OUTPUT statement.

ONLY or O

Has the same meaning as *segname*.

Note: (*segname*) and (*segname,*) will be processed as if you specified (*segname*,ONLY).

DEPENDENTS or D

Include the dependent segments of the specified segment, and not the segment itself, in the output. Only one specification per segment name is allowed.

BOTH or B

Include both the specified segment and its dependent segments in the output. Only one specification per segment name is allowed.

TREE or T

Include the specified segment, its parent segments, and dependent segments in the output. Only one specification per segment name is allowed.

ALL or *

For the SELECT statement, selects every segment defined in the DBD.

For the OUTPUT statement, includes in the output every segment defined in the DBD.

Note: Note that ALL is a name reserved for the SEG keyword. If you have used ALL as a DBD-defined segment name, ODE will assume that ALL is your specification (rather than the segname).

Default:

ALL, if the statement is not specified.

Site default support:

Not supported.

Alias:

SEGMENT

Restrictions:**For the SELECT statement:**

- SEG must specify a valid segment name unless ALL or * is specified. If ALL or * is specified:
 - All other instances of the SEG keyword are ignored.
 - The FIELD and ANDFIELD keywords are ignored.
- The values of SEG=*segname* and SEG=ALL or SEG=* are mutually exclusive, and concurrent use results in a terminating error condition.

For the OUTPUT statement:

- SEG must specify a valid segment name unless ALL or * is specified.
- The values of SEG=*segname* and SEG=ALL or SEG=* are mutually exclusive, and concurrent use results in a terminating error condition.
- If the STOPAFT, SKIP, or EVERY keywords is specified (see below), only 1 specification per segment name is allowed.
- The FIELDS, STOPAFT, SKIP, EVERY, AFTERTIME keywords are effective only for the specified segment. As for the associated dependent segments and parent segments, in case of DEPENDENTS, BOTH, or TREE, the default values are applied.
- The specification of DEPENDENTS or BOTH are not permitted if the specified segment is an SDEP segment.
- If DEPENDENTS is specified, the FIELDS, STOPAFT, SKIP, and EVERY keywords are not permitted.
- The *parm* keyword must be ONLY (O), DEPENDENTS (D), BOTH (B), or TREE (T).
- DEPENDENTS and BOTH cannot be specified for segments that have no dependent segments.
- If both DEPENDENTS and BOTH are specified for the same segment, BOTH will be in effect. If both TREE and BOTH are specified for the same segment, TREE will be in effect.
- If DEPENDENTS, BOTH, or TREE is specified for a root segment, the logically deleted SDEP segments cannot be extracted.
- If DEPENDENTS is specified, DBT, TFMT, and UR parameters cannot be specified for the OUTPUT keyword. Because when DBT, TFMT, or UR is specified on the OUTPUT keyword, all segments in the hierarchic path, which includes the selected segment and all of its parent segments, must be selected for output.

Effects of interplay between SELECT SEG= and OUTPUT SEG= keywords

Specifying SEG=ALL (or SEG=*) for the SELECT control statement puts this specification into effect for the OUTPUT control statement as well. Conversely, specifying SEG=ALL (or SEG=*) for the OUTPUT control statement puts this specification into effect for the SELECT control statement.

Specifying SELECT SEG=ALL (or SEG=*) is the only specification that produces output if no corresponding OUTPUT control statement is processed. Specifying OUTPUT SEG=ALL (or SEG=*) without a SELECT statement has the same effect.

If SELECT SEG=segname is used, an associated OUTPUT statement *must* be specified to produce output. However, if OUTPUT SEG=segname is specified without a corresponding SELECT SEG=segname, a SELECT statement is assumed to exist and output is produced.

SEGLDIST keyword

The SEGLDIST keyword specifies whether to generate the Segment Length Distribution report.

SEGLDIST is an optional keyword for the following statements:

- REPORT statement of OPC
- REPORT statement of OER

Format:



YES (or Y)

Print the Segment Length Distribution report.

SEGLDIST=YES is effective only when TYPRUN=RPT and FULLSTEP=YES are specified. For OER, PTRCHK=YES and CHKUOW=ALL are also required.

NO (or N)

Do not print the Segment Length Distribution report.

Default:

NO, if the SEGLDIST keyword is not specified.

Site default support:

Supported.

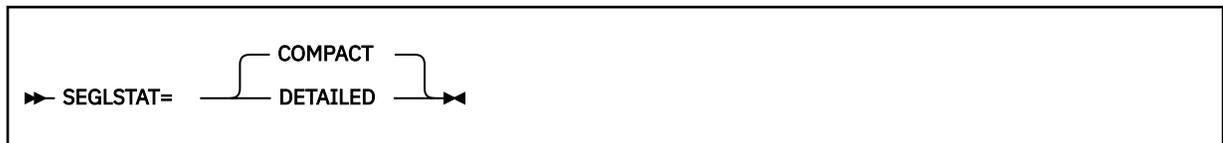
SEGLSTAT keyword

The SEGLSTAT keyword specifies the format of the DB Record Profile Analysis report, which is one of the DEDB Area Analysis reports.

SEGLSTAT is an optional keyword for the following statements:

- REPORT statement of OPC
- REPORT statement of OER

Format:



COMPACT

The DB Record Profile Analysis report provides analysis information of the database records. It does not include details such as prefix length and data flags.

DETAILED

The DB Record Profile Analysis report provides prefix length and data flags in addition to the default contents. This additional information is helpful in database tuning.

Default:

COMPACT, if the SEGLSTAT keyword is not specified.

Site default support:

Supported.

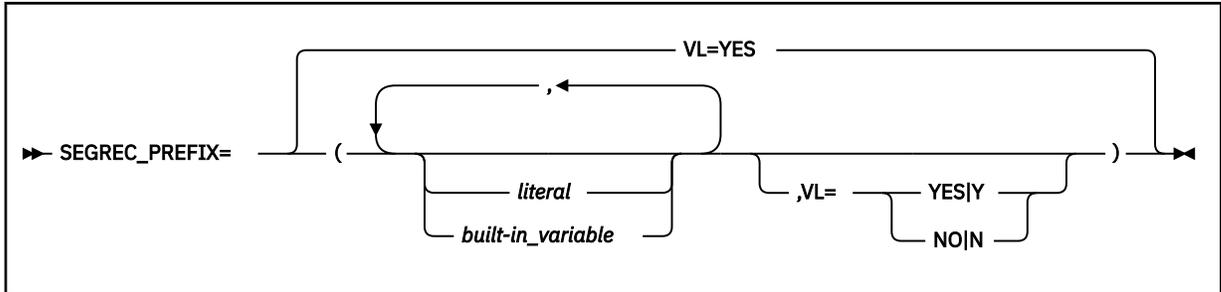
SEGREC_PREFIX keyword

The SEGREC_PREFIX keyword specifies the contents of the fields within the prefix portion of the segment records, which are to be written to an extract file.

This keyword is valid only when OUTPUT=USER is specified.

SEGREC_PREFIX is an optional keyword for the FPXCTL statement of ODE.

Format:



literal

Specify one or more alphabetic or numeric characters that are contained within apostrophes (' ') or within quotation marks (" "). You can specify an optional data-type declaration as follows:

- C'xxxxxx' (character): Case-sensitive character data. You must specify a minimum of 1 and maximum of 28 characters (28 bytes).
- F'xxxxxx' (signed binary): Must contain only numeric digits. You must specify a minimum of 1 and maximum of 19 digits (4 bytes).
- P'xxxxxx' (packed decimal): Assumed positive. Must contain only numeric digits. You must specify a minimum of 1 and maximum of 31 digits (16 bytes).
- X'xxxxxx' (unsigned binary): Must be an even number of hexadecimal characters. You must specify a minimum of two hexadecimal characters (representing 1 byte) and a maximum of 56 hexadecimal characters (representing 28 bytes).

You can also declare the length, by specifying it before the data-type declaration.

The parenthesis cannot be specified within apostrophes.

Example:

```
8C 'CUSDB '  
2P '01 '
```

Built-in_variable

The following built-in variables can be specified.

Built-in variable name	Data type	Contents
AREA_NAME	8C	Area name
AREA_NUMBER	1X	Area number (1 byte)
AREA_NUMBER2	2X	Area number (2 bytes)
DBD_AREAS	2F	Number of areas in DBD
DBD_LEVELS	2F	Maximum hierarchic level in DBD
DBD_NAME	8C	DBD name
DBD_SEGMENTS	2F	Number of segment types in DBD
KEY or SEGMENT_KEY	nC	Key for current segment
ROOT_KEY	nC	Key for root segment (Not valid for SDEP segments when SDEPHIER=NO is specified)

Built-in variable name	Data type	Contents
ROOT_KEY_LENGTH	2F	Length of root segment key
RUNDATE	4X	Date when ODE started
RUNDATETIME	12X	Date and time when ODE started
RUNTIME	4X	Time when ODE started
SEGMENT_CKEY	nC	Concatenated key for current segment (Not valid for SDEP segments when SDEPHIER=NO is specified)
SEGMENT_CKEY_LENGTH	2F	Length of concatenated key for current segment
SEGMENT_CODE	1X	Segment code
SEGMENT_KEY_LENGTH	2F	Length of current segment key
SEGMENT_LENGTH	2F	Length of segment. For compressed segment: <ul style="list-style-type: none"> • If EXPAND=YES is specified, length of expanded segment. • If EXPAND=NO is specified, length of compressed segment.
SEGMENT_LEVEL	1X	Segment level
SEGMENT_NAME	8C	Segment name
TODAY	4X	Date when ODE started

VL=YES

Request a length prefix.

VL=NO

Suppress a length prefix.

Default:

VL=YES

Site default support:

Not supported.

Alias:

PREFIX

Restrictions:

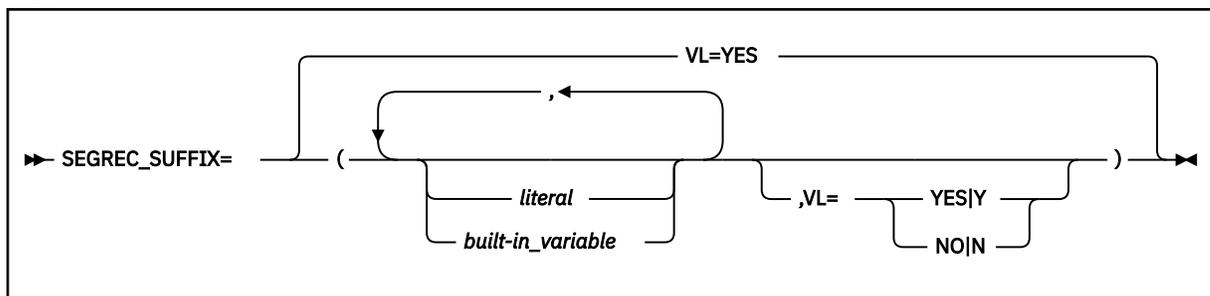
- The SEGREC_PREFIX keyword can be specified only in FPXIN DD.
- The SEGREC_PREFIX keyword is mutually exclusive with OUTPUT=STD/DBT/TFMT/UR.

SEGREC_SUFFIX keyword

The SEGREC_SUFFIX keyword specifies the contents of the fields within the suffix portion of the segment records, which are to be written to an extract file.

This keyword is valid only when OUTPUT=USER is specified.

SEGREC_SUFFIX is an optional keyword for the FPXCTL statement of ODE.

Format:**literal**

Specify one or more alphabetic or numeric characters that are contained within apostrophes (' ') or within quotation marks (" "). You can specify an optional data-type declaration as follows:

- C'xxxxxx' (character): Case-sensitive character data. You must specify a minimum of 1 and maximum of 28 characters (28 bytes).
- F'xxxxxx' (signed binary): Must contain only numeric digits. You must specify a minimum of 1 and maximum of 19 digits (4 bytes).
- P'xxxxxx' (packed decimal): Assumed positive. Must contain only numeric digits. You must specify a minimum of 1 and maximum of 31 digits (16 bytes).
- X'xxxxxx' (unsigned binary): Must be an even number of hexadecimal characters. You must specify a minimum of two hexadecimal characters (representing 1 byte) and a maximum of 56 hexadecimal characters (representing 28 bytes).

You can also declare the length, by specifying it before the data-type declaration.

The parenthesis cannot be specified within apostrophes.

Example:

```
8C 'CUSDB'
2P '01'
```

Built-in_variable

The following built-in variables can be specified.

Built-in variable name	Data type	Contents
AREA_NAME	8C	Area name
AREA_NUMBER	1X	Area number (1 byte)
AREA_NUMBER2	2X	Area number (2 bytes)
DBD_AREAS	2F	Number of areas in DBD
DBD_LEVELS	2F	Maximum hierarchic level in DBD
DBD_NAME	8C	DBD name
DBD_SEGMENTS	2F	Number of segment types in DBD
KEY or SEGMENT_KEY	nC	Key for current segment
ROOT_KEY	nC	Key for root segment (Not valid for SDEP segments when SDEPHIER=NO is specified)
ROOT_KEY_LENGTH	2F	Length of root segment key
RUNDATE	4X	Date when ODE started
RUNDATETIME	12X	Date and time when ODE started
RUNTIME	4X	Time when ODE started

Built-in variable name	Data type	Contents
SEGMENT_CKEY	nC	Concatenated key for current segment (Not valid for SDEP segments when SDEPHIER=NO is specified)
SEGMENT_CKEY_LENGTH	2F	Length of concatenated key for current segment
SEGMENT_CODE	1X	Segment code
SEGMENT_KEY_LENGTH	2F	Length of current segment key
SEGMENT_LENGTH	2F	Length of segment. For compressed segment: <ul style="list-style-type: none"> If EXPAND=YES is specified, length of expanded segment. If EXPAND=NO is specified, length of compressed segment.
SEGMENT_LEVEL	1X	Segment level
SEGMENT_NAME	8C	Segment name
TODAY	4X	Date when ODE started

VL=YES

Request a length prefix.

VL=NO

Suppress a length prefix.

Default:

VL=YES

Site default support:

Not supported.

Alias:

SUFFIX

Restrictions:

- The SEGREC_SUFFIX keyword can be specified only in FPXIN DD.
- The SEGREC_SUFFIX keyword is mutually exclusive with OUTPUT=STD/DBT/TFMT/UR.

SENSOR keyword

The SENSOR keyword specifies whether to store the sensor data in the Sensor Data repository of IMS Tools KB.

SENSOR is an optional keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement of OER

Format:



NO

Do not store the sensor data.

YES

Store the sensor data in the Sensor Data repository of IMS Tools KB.

Sensor data from the following areas is not stored:

- When SENSOR_PTRERR=YES is not specified, areas that have pointer errors
- When SDEP_ORPHAN_MSGLVL=ERROR is specified, areas that have orphan SDEP segments

Default:

NO, if the SENSOR keyword is not specified.

Site default support:

Supported.

Restrictions:

- The SENSOR keyword can be specified only in FPXIN.
- When SENSOR=YES is specified, you must specify the name of the IMS Tools KB server.
- SENSOR=YES is mutually exclusive with ITKBLOAD=dsname. When SENSOR=YES is specified, you must specify the IMS Tools Base load module library on the STEPLIB.
- When SENSOR=NO is specified, or when the SENSOR keyword is not specified, you cannot specify SENSOR_DBREC=YES or SENSOR_UOW=YES.
- SENSOR=YES is mutually exclusive with TYPRUN=MODEL.
- SENSOR=YES is mutually exclusive with CHKUOW=REORGED.

For a list of the data elements that are collected by IMS HP Fast Path Utilities, see the topics about data elements in the *Tools Base Policy Services User's Guide*.

SENSOR_DBREC keyword

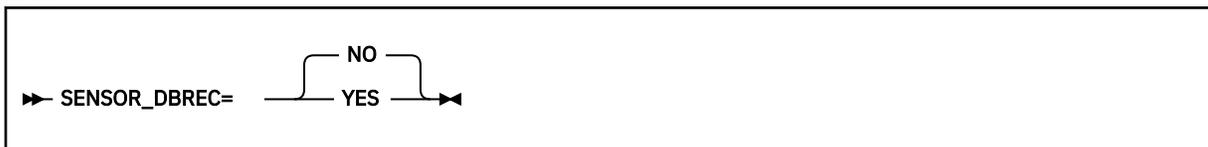
The SENSOR_DBREC keyword specifies whether to store the data elements that are related to database records and root segment distribution in the Sensor Data repository of IMS Tools KB.

The following data elements are related to database records and root segment distribution:

- DB_MAX_DBREC_LENGTH
- DB_MIN_DBREC_LENGTH
- DB_PCT_NUM_DBREC_IOVF
- DB_AVG_LEN_SYNONYM_CHAIN
- DB_MAX_LEN_SYNONYM_CHAIN
- DB_AVG_DBREC_IO
- DB_MAX_DBREC_IO
- DB_AVG_ROOT_IO
- DB_MAX_ROOT_IO

SENSOR_DBREC is an optional keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement of OER

Format:**NO**

Do not store the sensor data for the data elements that are related to database records and root segment distribution.

YES

Store the sensor data for the data elements that are related to database records and root segment distribution in the Sensor Data repository of IMS Tools KB.

Considerations for **SENSOR_DBREC** and the policy evaluation process:

If **SENSOR_DBREC=NO** is specified, Policy Services and Autonomics Director cannot evaluate the data elements that are collected when **SENSOR_DBREC=YES** is specified, and they might not provide adequate information about the state of the area and the actions that you must take. Consider specifying **SENSOR_DBREC=YES** unless you have performance concerns.

When you specify **SENSOR_DBREC=YES**, the CPU time and the elapsed time increase compared to when **SENSOR_DBREC=NO** is specified.

Even if you specify **SENSOR_DBREC=NO**, rough estimations of the number of I/Os that are required to read database records and root segments are calculated and stored in the IMS Tools KB repository. These values are stored as **DB_ESTIMATED_DB_REC_IO** and **DB_ESTIMATED_ROOT_IO** data elements, and are collected regardless of the **SENSOR_DBREC** keyword specification.

Default:

NO, if the **SENSOR_DBREC** keyword is not specified.

Site default support:

Supported.

Restrictions and requirements:

- The **SENSOR_DBREC** keyword can be specified only in **FPXIN**.
- When you specify **SENSOR_DBREC=YES**, you must also specify **SENSOR=YES**, **TYPRUN=RPT**, and **FULLSTEP=YES**.

For a list of the data elements that are collected by IMS HP Fast Path Utilities, see the topics about data elements in the *Tools Base Policy Services User's Guide*.

SENSOR_PTRERR keyword

The **SENSOR_PTRERR** keyword specifies to store the sensor data in the Sensor Data repository of IMS Tools KB even when pointer errors are detected in the area.

SENSOR_PTRERR is an optional keyword for the following statements:

- **ANALYZE** statement of **OPC**
- **PTRCHK** statement of **OER**

Format:



NO

Do not store the sensor data in the Sensor Data repository of IMS Tools KB when pointer errors are detected in the area.

YES

Store the sensor data in the Sensor Data repository of IMS Tools KB even when pointer errors are detected in the area.

Default:

NO, if the **SENSOR_PTRERR** keyword is not specified.

Site default support:

Supported.

Restriction:

The **SENSOR_PTRERR** keyword can be specified only in **FPXIN**.

SENSOR_UOW keyword

The SENSOR_UOW keyword specifies whether to store the data elements that are related to UOW in the IMS Tools KB repository.

The data elements that will be stored differ by the FPXAUOWC DD specification as follows:

- If you specify the FPXAUOWC DD statement, data elements are collected at the UOW group level (data elements that begin with DEBUG.)
- If you do not specify the FPXAUOWC DD statement, data elements are collected at the UOW level (data elements that begin with DBU.)

SENSOR_UOW is an optional keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement of OER

Format:



NO

Do not store the sensor data for the data elements that are related to UOW.

YES

Store the sensor data for the data elements that are related to UOW in the Sensor Data repository of IMS Tools KB.

Considerations for SENSOR_UOW and the policy evaluation process:

If you specify SENSOR_UOW=YES, data elements for each UOW or UOW group are collected. When you process many UOWs or UOW groups, be aware that the number of data elements that will be stored in the IMS Tools KB repository can be large. Processing a large number of data elements might increase elapsed time and require more storage space in the repository.

Default:

NO, if the SENSOR_UOW keyword is not specified.

Site default support:

Supported.

Restrictions and requirements:

- The SENSOR_UOW keyword can be specified only in FPXIN.
- When you specify SENSOR_UOW=YES, you must also specify SENSOR=YES, SENSOR_DBREC=YES, TYPRUN=RPT, and FULLSTEP=YES.

For a list of the data elements that are collected by IMS HP Fast Path Utilities, see the topics about data elements in the *Tools Base Policy Services User's Guide*.

SENSOR_UOW_REPORT keyword

The SENSOR_UOW_REPORT keyword specifies whether to print the data elements that are related to UOW in the Sensor Data Statistics report.

SENSOR_UOW is an optional keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement of OER

Format:**NO**

Do not print the data elements that are related to UOW in the Sensor Data Statistics report.

YES

Print the data elements that are related to UOW in the Sensor Data Statistics report.

Considerations for SENSOR_UOW_REPORT:

When you specify the FPXAUOWC DD statement, the SENSOR_UOW=YES keyword, and the SENSOR_UOW_REPORT=YES keyword, the data elements that are related to UOW are collected at the UOW group level and are printed in the Sensor Data Statistics report.

When you specify the SENSOR_UOW=YES keyword and the SENSOR_UOW_REPORT=YES keyword without an FPXAUOWC DD statement, the data elements that are related to UOW are collected at the UOW level and are printed in the Sensor Data Statistics report; however, the amount of data can be large and might require additional time to print the Sensor Data Statistics report.

Default:

NO, if the SENSOR_UOW_REPORT keyword is not specified.

Site default support:

Supported.

Restrictions and requirements:

- The SENSOR_UOW_REPORT keyword can be specified only in FPXIN.
- When you specify SENSOR_UOW_REPORT=YES, you must also specify SENSOR_UOW=YES.

SKIP keyword

The SKIP keyword lets you skip a number of segment occurrences.

For DDEP segments, a specific number of segment occurrences under a parent, and for root segments, a specific number of root segment occurrences in a synonym chain under a RAP. For SDEP segments, this keyword is only valid with SDEPHIER=NO, and the keyword lets you skip a specific number of segments starting from the first CI of the SDEP part.

SKIP is an optional keyword for the OUTPUT statement of ODE. If this keyword is specified, it must be assigned a value.

Format:***integer***

Bypass the indicated number of segments before processing data. The integer you specify must be in the range of 1 through 9999999.

Default:

None.

Site default support:

Not supported.

Restrictions:

- This keyword must be preceded by a valid OUTPUT SEG= specification.
- Dependents of parents bypassed due to SKIP are not selected.

- You cannot have multiple values for the SKIP keyword specified for the same segment name. If you do specify multiple values, only the last valid occurrence of the SKIP keyword will be in effect.
- If more than one of the STOPAFT, SKIP, or EVERY keywords is specified, SKIP is applied first, then EVERY, and then STOPAFT.
- SELECT statement criteria are evaluated before SKIP. Only matched segments are counted. Counts are maintained for segment occurrences under a single parent and are reset when parentage changes.
- If DEPENDENTS is specified in the preceding SEG keyword, the SKIP keyword cannot be specified.

SORT keyword

The SORT keyword specifies how extracted information is to be sorted for output.

SORT is an optional keyword for the FPXCTL statement of ODE.

Format:



NO (or N)

Write the extract in RAP (physical) sequence.

YES (or Y)

If ODESORT DD statement is specified in the JCL, ODE generates the sort control statements that will write the extract in ascending root key sequence. The sort control statements are written to // ODESORT for an offline sort. If no ODESORT DD statement is specified in the JCL, extracted records are internally sorted for output instead of generating the sort control statements.

When SORT=YES is specified, the following DD statement can be provided. The DCB parameter is not needed. ODE internally sets necessary DCB attributes.

```
//ODESORT DD DISP=SHR,DSN=IMSESA.SORTCARD
```

UR (or U)

Same as OUTPUT=UR.

If OUTPUT=STD|DBT|TFMT, SORT=UR is specified, it can be identified as OUTPUT=UR.

If SORT=UR, OUTPUT=STD|DBT|TFMT is specified, it can be identified as OUTPUT=STD|DBT|TFMT.

For FPXCTL DD and *areaname* DD, you can specify this keyword without a value. SORT or SORT= is regarded as SORT=YES.

Default:

NO, if the SORT keyword is not specified.

Site default support:

Supported.

Restriction:

- The specification of YES is mutually exclusive with OUTPUT=TFMT and OUTPUT=UR.
- If OUTPUT=USER and SORT=YES are specified, SORT=YES is internally changed to SORT=NO.
- If you specify SORT=YES when OUTPUT=STD(DBT) is specified, the SDEP segment is not extracted unless SDEPHIER=YES is specified.
- The function of internal sort is effective when ODE is run by using the run-level control and SDEPHIER=NO is specified. If no DD statement is found for ODESORT, and ODE is run by using the

area-level control or if SDEPHIER=YES is specified, ODE issues message FPX0002E and ends with an error.

- SORT=UR cannot be specified in FPXIN DD.

STARTUOW keyword

The STARTUOW keyword specifies the number that defines the first UOW to be reported in the UOW report and the UOW Exception report.

STARTUOW is an optional keyword for the following statements:

- REPORT statement of OPC
- REPORT statement of OER

Format:



integer

Specify a value in the range of 0 - 32767. This value must be less than or equal to the value specified for the STOPUOW keyword.

Default:

0, if the STARTUOW keyword is not specified.

Site default support:

Supported.

Restriction:

The STARTUOW keyword can be specified only in FPXIN.

STAT keyword

The STAT keyword lets you specify whether or not statistics are to be generated.

For OPC, these statistics are written to the message data set and the report data set, and for ODE, these are written to the message data set.

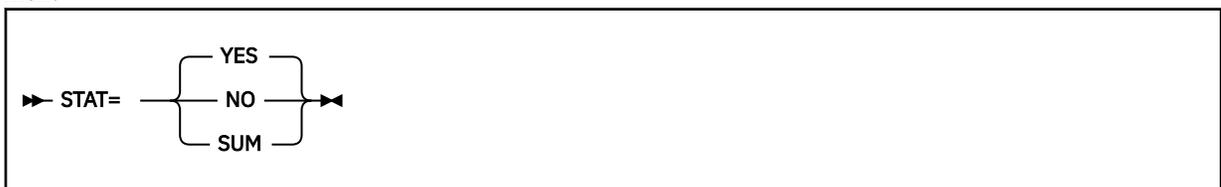
For ODE, if you specify to generate the statistics, you have the option of having them specified per area and DEDB, or only per DEDB. The extraction statistics by ODE illustrate the records processed by the ODE job.

STAT is an optional keyword for the following statements:

- ANALYZE statement of OPC
- FPXCTL statement of ODE
- PTRCHK statement of OER

If this keyword is omitted, statistics are generated.

Format:



YES (or Y)

Generate statistics (For ODE, for each area and DEDB).

NO (or N)

Do not generate statistics.

SUM (or S)

Generate statistics for the DEDB. This parameter can be specified only for FPXCTL statement.

For FPXCTL DD and *areaname* DD, you can specify this keyword without a value. STAT or STAT= is regarded as STAT=YES.

Default:

YES, if the STAT keyword is not specified.

Site default support:

Supported.

Restriction:

In the area-level control of ODE, the option specified for the last area is applicable for the DEDB statistics. For example, if STAT=NO is specified for the last area, the DEDB statistics report is not generated even though STAT=YES or SUM is specified for the preceding areas.

STOPAFT keyword

The STOPAFT keyword lets you stop selection of segments.

For DDEP segments, after a specific number of segment occurrences under a parent segment, and for root segments, after a specific number of root segments in a synonym chain under a RAP. The counter is reset every time parentage changes. For SDEP segments, this keyword is only valid with SDEPHIER=NO, and the keyword lets you stop selection of segments after a specific number of segments starting from the first CI of SDEP part.

STOPAFT is an optional keyword for the OUTPUT statement of ODE. If this keyword is specified, it must be assigned a value.

Format:

►► STOPAFT= *integer* ◄◄

integer

End processing after processing the indicated number of segments. The integer you specify must be in the range of 1 through 9999999.

Default:

None.

Site default support:

Not supported.

Restrictions:

- This keyword must be preceded by a valid OUTPUT SEG= specification.
- Dependents of parents bypassed due to STOPAFT are not selected.
- You cannot have multiple values for the STOPAFT keyword specified for the same segment name. If you do specify multiple values, only the last valid occurrence of the STOPAFT keyword will be in effect.
- If more than one of the STOPAFT, SKIP, or EVERY keywords is specified, SKIP is applied first, then EVERY, and then STOPAFT.
- SELECT statement criteria are evaluated before STOPAFT. Only matched segments are counted. Counts are maintained for segment occurrences under a single parent and are reset when parentage changes.
- If DEPENDENTS is specified in the preceding SEG keyword, the STOPAFT keyword cannot be specified.

STOPUOW keyword

The STOPUOW keyword specifies the number that defines the last UOW to be reported in the UOW report and the UOW Exception report.

STOPUOW is an optional keyword for the following statements:

- REPORT statement of OPC
- REPORT statement of OER

Format:



integer

Specify a value in the range of 0 - 32767. This value must be greater than or equal to the value specified for the STARTUOW keyword.

Default:

32767, if the STOPUOW keyword is not specified.

Site default support:

Supported.

Restriction:

The STOPUOW keyword can be specified only in FPXIN.

STORCLAS keyword

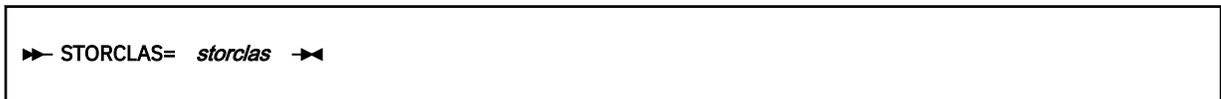
The STORCLAS keyword specifies the name of the SMS storage class for the temporary area data set.

This parameter is used for the preparation of the temporary ADS.

For more information about SMS storage class, see the *MVS JCL Reference*.

STORCLAS is an optional keyword for the EXTEND statement of OAE.

Format:



STORCLAS can be specified as STORAGECLASS.

storclas

Specify the name in 1- to 8-characters.

Default:

None.

Site default support:

Not supported.

SYN_LEN_AVG keyword

The SYN_LEN_AVG keyword specifies a threshold setting for the average synonym chain length.

If the length is greater than the value that you specified, a warning message is issued.

SYN_LEN_AVG is an optional keyword for the following statements:

- THRESHOLD statement of OPC
- THRESHOLD statement of OER

Format:

```
►► SYN_LEN_AVG= value ◄◄
```

value

An integer value in the range of 0 - 32767.

Default:

None.

Site default support:

Not supported.

Restriction:

The SYN_LEN_AVG keyword can be specified only in FPXIN.

SYN_MAX keyword

The SYN_MAX keyword specifies a threshold setting for the maximum synonym chain length.

If the length is greater than the value that you specified, a warning message is issued.

SYN_MAX is an optional keyword for the following statements:

- THRESHOLD statement of OPC
- THRESHOLD statement of OER

Format:

```
►► SYN_MAX= value ◄◄
```

value

An integer value in the range of 0 - 32767.

Default:

None.

Site default support:

Not supported.

Restriction:

The SYN_MAX keyword can be specified only in FPXIN.

SYN_PCT_LEN keyword

The SYN_PCT_LEN keyword specifies a threshold setting for the percentage of synonym chains and the synonym chain length.

If the percentage of the exceeding synonym chain length that you specified is greater than the value that you specified, a warning message is issued.

SYN_PCT_LEN is an optional keyword for the following statements:

- THRESHOLD statement of OPC
- THRESHOLD statement of OER

Format:

```
►► SYN_PCT_LEN= (value1,value2) ◄◄
```

value1

An integer value in the range of 0 - 100.

value2

An integer value. The value must be less than the value that is specified by SYNTBL in the HFPAPARM DD statement.

You must specify both values.

Default:

None.

Site default support:

Not supported.

Restriction:

The SYN_PCT_LEN keyword can be specified only in FPXIN.

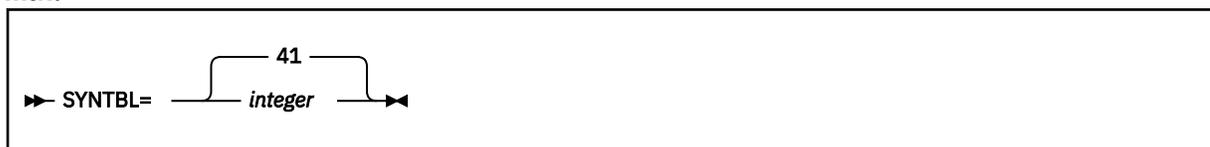
SYNTBL keyword

The SYNTBL keyword specifies the number of reporting entries in the Synonym Chain Analysis table.

All synonym chains longer than the specified value will be grouped and reported together.

SYNTBL is an optional keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement of OER

Format:**integer**

An integer value in the range of 12 - 99.

Default:

If the SYNTBL keyword is not specified, the default is 41 (that is, all synonym chains longer than 40 are grouped and reported together).

Site default support:

Supported.

Restrictions:

- The SYNTBL keyword can be specified only in FPXIN.
- The SYNTBL keyword is effective only when both the TYPRUN=RPT keyword and the FULLSTEP=YES keyword are specified. Otherwise, it is ignored.

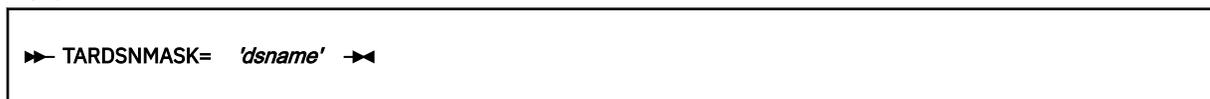
TARDSNMASK keyword

The TARDSNMASK keyword specifies the data set name or the mask for dynamically allocating cataloged pre-existing tuning aid record data sets.

You can make a relative reference to a GDG by specifying a relative generation number in the mask, a number enclosed in brackets such as in '*gdg-name*(-2)'. The number you specify in the brackets must be zero or smaller.

TARDSNMASK is an optional keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement of OER

Format:

dsname

Specify a data set name or a data set mask. Enclose the *dsname* in single quotation marks. Use the following variables, as necessary, to create the mask.

&DBD

DBD name

&AREA

Area name

&AREA#

3-digit relative area number. (If you specify an area number that is greater than 999, FPO does not convert the mask variable to the relative area number and issues error message FPX0334E.)

&AREAn#

n-digit relative area number. *n* is 1 - 5.

When the mask variable &AREA# is used for an area whose relative area number is greater than *n*-digit, for example, when &AREA2# is used for an area whose relative area number is greater than or equal to 100, FPO does not convert the mask variable to the relative area number and issues error message FPX0334E.

For example, if you specify TARDSNMASK='HFP.TEST.TAR.&AREA' and the area name is AREA123, dynamic allocation of tuning aid record data sets will dynamically allocate 'HFP.TEST.TAR.AREA123'. If you specify TARDSNMASK='HFP.TEST.TAR.HPFP&AREA4#' and the relative area number is 1500, the resulting data set name will be 'HFP.TEST.TAR.HPFP1500'.

Default:

None.

Site default support:

Not supported.

Restriction:

The TARDSNMASK keyword can be specified only in FPXIN DD.

Related conceptsTuning aid records

Tuning aid records are the sorted and resolved data analysis records. A tuning aid record data set contains the analysis records for an area. The Tune function uses the tuning aid records to simulate new database attributes.

TARECGEN keyword

The TARECGEN keyword specifies whether to generate the tuning aid records.

TARECGEN is an optional keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement of OER

Format:**YES**

Generate the tuning aid records.

You must specify the TARDSNMASK keyword or TARnnnnn DD statements to define the tuning aid record data sets.

NO

Do not generate the tuning aid records.

ONLY

Generate the tuning aid records and suppress generating the following reports:

- DEDB Area Analysis reports
- UOW report
- UOW Exception report
- Largest Database Records report
- Consolidated Largest Database Records report
- Segment Length Distribution report
- Process summary of analysis report

Default:

NO, if the TARECGEN keyword is not specified.

Site default support:

Supported.

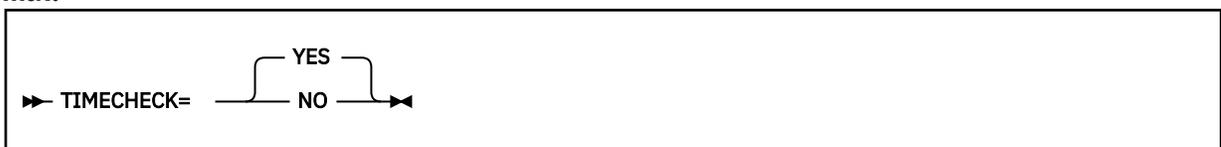
Restrictions:

- The TARECGEN keyword can be specified only in FPXIN DD.
- If TARECGEN=YES or TARECGEN=ONLY is specified in the ANALYZE statement, you must specify TYPRUN=RPT and FULLSTEP=YES.
- If TARECGEN=YES or TARECGEN=ONLY is specified in the PTRCHK statement, you must specify TYPRUN=RPT, CHKUOW=ALL, and FULLSTEP=YES.
- If TARECGEN=ONLY is specified in the ANALYZE or PTRCHK statement, the REPORT and THRESHOLD statements are ignored.

TIMECHECK keyword

The TIMECHECK keyword specifies whether to verify the timestamp of the interface information in the FPXBINTP DD file.

TIMECHECK is an optional keyword for the OAE Rename program.

Format:**YES**

Specifies to verify the timestamp of the interface information in the FPXBINTP DD file. If the time lag between the current time and the timestamp of the interface information is less than three minutes, the process continues. If not, the process abends and message FPX0151E is issued. The OAE Rename program is expected to run immediately after the OAE program with the EXTEND_TYPE=RENAME option.

NO

Specifies not to verify the timestamp of the interface information in the FPXBINTP DD file.

Default:

YES, if the TIMECHECK keyword is not specified.

Site default support:

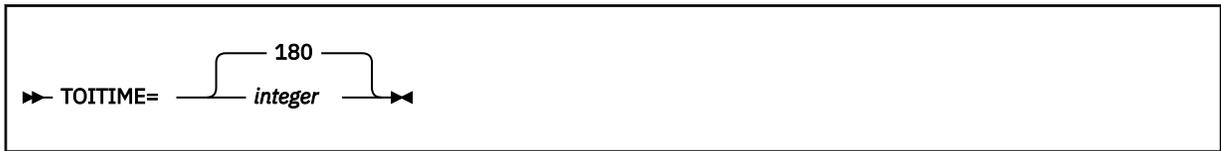
Not supported.

TOITIME keyword

The TOITIME keyword specifies the execution time value that is used for /DIS AREA requests.

TOITIME is an optional keyword for the OAE Rename program.

Format:



integer

Specifies, in seconds, the maximum allowable execution time for requests to process from /STOP AREA to /DIS AREA. If the program does not receive response from /DIS AREA within the specified time, FPX0191E message is issued and ABEND U3529 occurs. The number you specify must be an integer value in the range of 1 - 999.

Default:

180, if the TOITIME keyword is not specified.

Site default support:

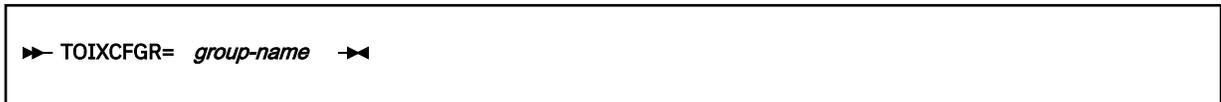
Not supported.

TOIXCFGR keyword

The TOIXCFGR keyword identifies the XCF group that the IMS Tools Online System Interface uses to interface with OAE.

TOIXCFGR is an optional keyword for the OAE Rename program.

Format:



group-name

Specify a 1- to 5-alphanumeric character to generate the XCF group name. The IMS Tools Online System Interface prefixes this string with characters "TOI" to generate the XCF group name.

Default:

None.

Site default support:

Not supported.

TYPRUN keyword

The TYPRUN keyword specifies the operating mode.

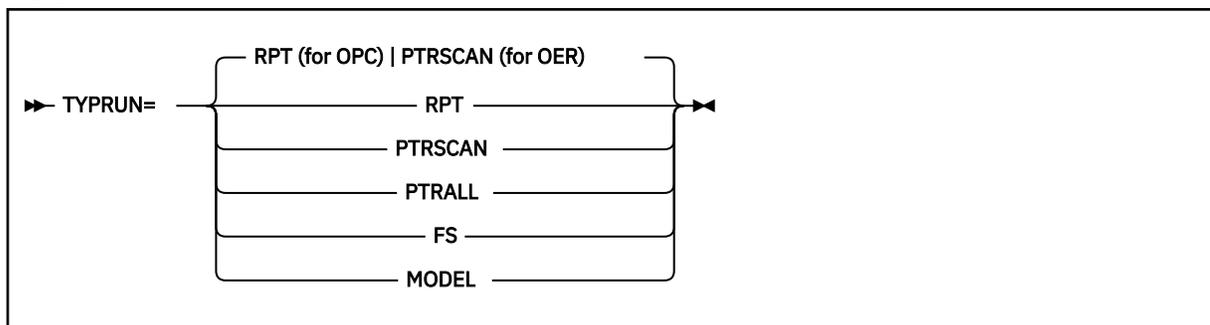
TYPRUN is an optional keyword for the following statements:

- ANALYZE statement of OPC
- PTRCHK statement of OER

If this keyword is specified, it must be assigned a variable.

When this keyword is used with the PTRCHK statement, you must specify CHKUOW=ALL to specify TYPRUN=RPT, TYPRUN=PTRALL, TYPRUN=FS, or TYPRUN=MODEL.

Format:



RPT (or R)

The behavior and the output from the job depends on the FULLSTEP keyword specification as follows:

- When FULLSTEP=YES is specified, OPC scans the database in FULL mode and generates complete analysis reports.
- When FULLSTEP=NO is specified, OPC scans the database in QUICK mode and generates output records that are required by DEDB Pointer Checker. DEDB Pointer Checker uses these records to run the integrity verification in FULL mode and to generate complete analysis reports.

PTRSCAN (or PTRS)

OPC scans the database in QUICK mode for pointer integrity errors without creating any output records for further analysis and reporting. Messages are generated if any pointer integrity errors are detected. The exact RBA of some errors cannot be reported in this mode.

PTRSCAN provides a very fast method of integrity verification, using minimal computer resources. Although unlikely, it is possible that some pointer errors are undetected in this mode. Use PTRALL or RPT when complete integrity verification is needed.

PTRALL (or PTRAL)

The behavior and the output from the job depends on the FULLSTEP keyword specification as follows:

- When FULLSTEP=YES is specified, OPC scans the database in FULL mode.
- When FULLSTEP=NO is specified, OPC scans the database in QUICK mode and generates output records that are required by DEDB Pointer Checker. DEDB Pointer Checker uses these records to run the integrity verification in FULL mode.

The exact RBA of the error is reported in this mode. Any output records associated with further analysis of the database are not generated in this mode.

FS (or F)

The behavior and the output from the job depends on the FULLSTEP keyword specification as follows:

- When FULLSTEP=YES is specified, OPC scans the database in QUICK mode and generates free space analysis reports.
- When FULLSTEP=NO is specified, OPC scans the database in QUICK mode and generates the output records that are required by DEDB Pointer Checker. DEDB Pointer Checker uses these records to analyze and report free space information.

MODEL (or M)

OPC scans the database in QUICK mode and generates output records that are required by DEDB Pointer Checker. DEDB Pointer Checker uses these records to run the integrity verification in FULL mode and to generate complete analysis reports. OPC also generates segment information records including the root key, which can be used as input for FPB DEDB Tuning Aid.

Default:

- For OPC, if the TYPRUN keyword is not specified, the default value is RPT.

- For OER, if the TYPRUN keyword is not specified, the default value is PTRSCAN.

Site default support:

Supported.

Restrictions:

- PTFSCAN or MODEL is mutually exclusive with FULLSTEP=YES.
- TYPRUN=MODEL can be specified only in FPXIN.
- TYPRUN=MODEL cannot be specified on the PTRCHK statement.
- Only the following operating modes are supported when processing multiple areas:
 - PTRSCAN
 - FS when FULLSTEP=YES is specified
 - PTRALL when FULLSTEP=YES is specified
 - RPT when FULLSTEP=YES is specified

For the relationship among the TYPRUN keyword, the FULLSTEP keyword, and the required or the optional input and output DDs, see [“Selecting the operating modes” on page 679](#).

UOWDOVF_PCT keyword

The UOWDOVF_PCT keyword specifies a threshold setting for the percentage of UOWs that use DOVF.

If the percentage is greater than the value that you specified, a warning message is issued.

UOWDOVF_PCT is an optional keyword for the following statements:

- THRESHOLD statement of OPC
- THRESHOLD statement of OER

Format:

```
▶▶ UOWDOVF_PCT= value ◀◀
```

value

An integer value in the range of 0 - 100.

Default:

None.

Site default support:

Not supported.

Restriction:

The UOWDOVF_PCT keyword can be specified only in FPXIN.

UOWIOVF_AVG keyword

The UOWIOVF_AVG keyword specifies a threshold setting for the average number of IOVF CIs that are used by a UOW.

If the number is greater than the value that you specified, a warning message is issued.

UOWIOVF_AVG is an optional keyword for the following statements:

- THRESHOLD statement of OPC
- THRESHOLD statement of OER

Format:

```
▶▶ UOWIOVF_AVG= value ◀◀
```

value

An integer value in the range of 0 - 32767.

Default:

None.

Site default support:

Not supported.

Restriction:

The UOWIOVF_AVG keyword can be specified only in FPXIN.

UOWIOVF_MAX keyword

The UOWIOVF_MAX keyword specifies a threshold setting for the maximum number of IOVF CIs that are used by a UOW.

If the number is greater than the value that you specified, a warning message is issued.

UOWIOVF_MAX is an optional keyword for the following statements:

- THRESHOLD statement of OPC
- THRESHOLD statement of OER

Format:

```
▶▶ UOWIOVF_MAX= value ▶▶
```

value

An integer value in the range of 0 - 32767.

Default:

None.

Site default support:

Not supported.

Restriction:

The UOWIOVF_MAX keyword can be specified only in FPXIN.

UOWIOVF_PCT keyword

The UOWIOVF_PCT keyword specifies a threshold setting for the percentage of UOWs that use IOVF.

If the percentage is greater than the value that you specified, a warning message is issued.

UOWIOVF_PCT is an optional keyword for the following statements:

- THRESHOLD statement of OPC
- THRESHOLD statement of OER

Format:

```
▶▶ UOWIOVF_PCT= value ▶▶
```

value

An integer value in the range of 0 - 100.

Default:

None.

Site default support:

Not supported.

Restriction:

The UOWIOVF_PCT keyword can be specified only in FPXIN.

UOW_REPORT keyword

The UOW_REPORT keyword specifies whether to generate the UOW report.

UOW_REPORT is an optional keyword for the following statements:

- REPORT statement of OPC
- REPORT statement of OER

Format:



YES

Generate the UOW report.

NO

Do not generate the UOW report.

Default:

YES, if the UOW_REPORT keyword is not specified.

Site default support:

Supported.

Restriction:

The UOW_REPORT keyword can be specified only in FPXIN.

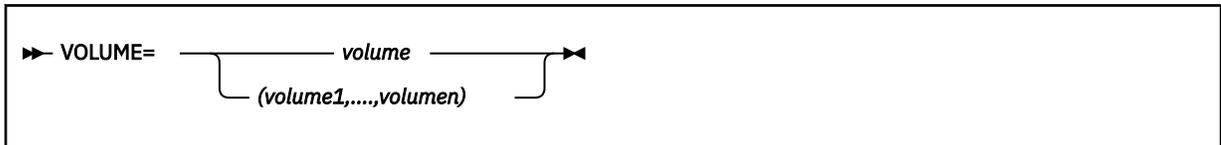
VOLUME keyword

The VOLUME keyword specifies one or more volume-serial identifiers for the temporary area data set.

This parameter is used for the preparation of the temporary ADS.

VOLUME is an optional keyword for the EXTEND statement of OAE. If non-SMS environment, this keyword is required.

Format:



volume

Specifies the volume-serial identifier for the temporary area data set.

(volume1, ..., volumen)

Specifies multiple volume-serial identifiers for the temporary area data set.

Default:

None.

Site default support:

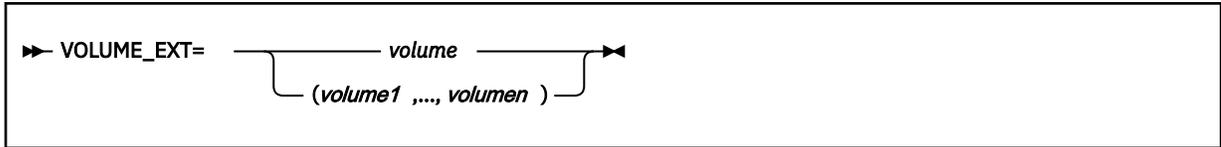
Not supported.

VOLUME_EXT keyword

The VOLUME_EXT keyword specifies one or more volume serial identifiers for the extended original ADS.

This parameter is used for the preparation of the extended original ADS.

VOLUME_EXT is an optional keyword for the EXTEND statement of OAE.

Format:***volume***

Specifies the volume serial identifier for the extended original ADS.

(*volume1, ..., volumen*)

Specifies multiple volume serial identifiers for the extended original ADS.

Default:

- Non-SMS environment: The volume serial identifiers of the original ADS (before expansion).
- SMS environment: None.

Site default support:

Not supported.

Restrictions:

- The VOLUME_EXT keyword can be specified only in FPXIN.
- This keyword cannot be specified when the original ADS is a MADS.
- This keyword is mutually exclusive with the MAXVOLS keyword.

Chapter 26. FPO reports stored in IMS Tools KB

When `ITKBSRVR=servername` is specified, reports and messages of FPO can be stored in the output repository of IMS Tools KB.

The following table summarizes the messages and reports generated by FPO. The table also shows whether each report is stored in the IMS Tools Knowledge Base Output repository when the `ITKBLOAD` and the `ITKBSRVR` keywords are specified.

Table 112. Messages and reports stored in IMS Tools KB

Utility	Generated messages and reports	Whether the report can be stored in the IMS Tools KB Output repository
OPC	Processing messages	Y
	Online Pointer Checker statistics	Y
	Processing Summary of Analysis report	Y
	DEDB Area Analysis reports	Y
	Largest Database Records report (<code>LARGEREC_REPORT=YES</code>)	Y
	Consolidated Largest Database Records report (<code>LARGEREC_REPORT=YES</code> and <code>LARGEREC_COUNT=(value1,value2)</code>)	Y
	DB Record Length Distribution report (<code>DBRLDIST=YES</code>)	Y
	Segment Length Distribution report (<code>SEGLDIST=YES</code>)	Y
	UOW report	Y
	UOW Exception report	Y
	Process Summary report	Y
	Image Copy NOTIFY report	Y
	Sensor Data Statistics report (<code>SENSOR=YES</code>)	Y
	RBA of Used IOVF CI report (<code>IOVFCI_RBA_REPORT=YES</code>)	Y
ODE	Processing messages	-
	Process Summary report	-
OAE	Processing messages	Y

Table 112. Messages and reports stored in IMS Tools KB (continued)

Utility	Generated messages and reports	Whether the report can be stored in the IMS Tools KB Output repository
OER	Processing messages	Y
	Reorganization Processing Summary report	Y
	UOW Space Statistics report	Y
	Process Summary report	Y
	Online Pointer Checker statistics	Y
	Processing Summary of Analysis report	Y
	DEDB Area Analysis reports	Y
	Largest Database Records report (LARGEREC_REPORT=YES)	Y
	Consolidated Largest Database Records report (LARGEREC_REPORT=YES and LARGEREC_COUNT=(value1,value2))	Y
	DB Record Length Distribution report (DBRLDIST=YES)	Y
	Segment Length Distribution report (SEGLDIST=YES)	Y
	UOW report	Y
	UOW Exception report	Y
	Image Copy NOTIFY report	Y
	Sensor Data Statistics report (SENSOR=YES)	Y
	RBA of Used IOVF CI report (IOVFCI_RBA_REPORT=YES)	Y
ODM	Processing messages	Y
	DMACPRT report	Y
	Process Summary report	Y

Related reference

ITKBLOAD keyword

The ITKBLOAD keyword specifies the name of the load module library for IMS Tools KB.

ITKBSRVR keyword

The ITKBSRVR keyword specifies the name of the IMS Tools KB server XCF group.

Chapter 27. Site default options support for FPO

The site default options are control statements that are specified in a member of a partitioned data set and that are used to override the system default option values, which can be changed without assembling and link-editing.

These reference topics introduce what the site default options are, how the site default options can be defined, and how they are used.

FPO supports the site default options. There are three kinds of site default options:

- Global site default options
- DB-level site default options
- Area-level site default options

The global site default options are always used if it is specified. The DB-level site default options are created for each DEDB and the area-level site default options are created for each area.

Topics:

- [“Supported keywords for the site default option” on page 943](#)
- [“How keyword values are determined” on page 943](#)
- [“Setting site default values for FPO” on page 944](#)

Supported keywords for the site default option

Most of the control statement keywords are available for the site default options. Though there are several exceptions.

The exceptions are:

For all tools

- The AREA keyword
- Keywords that are associated with the OVERRIDE control statement

OPC

- Keywords that are associated with the THRESHOLD control statement
- The TARDSNMASK keyword of the ANALYZE control statement

ODE

- Keywords that are associated with the SELECT, the OUTPUT, the USER_RECORD control statements
- The SEGREC_PREFIX and the SEGREC_SUFFIX keywords of the FPXCTL control statement

OAE

- Keywords that are associated with the EXTEND control statement
- Keywords for the OAE Rename program

OER

- Keywords that are associated with the THRESHOLD control statement
- The TARDSNMASK keyword of the PTRCHK control statement

How keyword values are determined

The value specified for each keyword has an order of priority.

A value in a higher position in the following list overrides one in a lower position:

1. The value specified in the JCL stream
2. The value specified in the AREA member in the parameter library
3. The value specified in the DB member in the parameter library
4. The global default value specified in the PROCLIB member
5. The system default value

Exceptions:

- AREA member is not used in the following cases:
 - OPC except when AREA=*areaname* is specified
 - OER except when AREA=*areaname* is specified
 - ODE running with the run-level control
 - ODM
- DB members and AREA members are not applied to the GLOBAL control statement.

Setting site default values for FPO

To use the site default options, you must create a configuration member FPXCFG00 and set the members to be used for the site default options.

Creating configuration member FPXCFG00

A configuration member determines the members that include the global site default options, and the parameter library that includes the DB-level site default options and the AREA-level site default options that are to be used.

About this task

To use the site default options, a configuration member FPXCFG00 must be created in a library that is specified on the PROCLIB DD statement, which is always coded in the FPUTIL cataloged procedure of the IMS system. If the configuration member FPXCFG00 is not created in PROCLIB, no site default options are applied.

The following keyword parameters can be specified in FPXCFG00:

- xxxSUF=*yy*
- xxxPARMLIB=*a_parameter_library_dsname*

where *xxx* can be GBL, OPC, ODE, OER, or ODM.

The global site default options are specified in member FPXxxx00, which must reside in a library that is concatenated to the PROCLIB DD statement. The default member FPXxxx00 can be changed to FPXxxxyy by specifying the 2-character suffix *yy* using the keyword xxxSUF=*yy*. If xxxSUF=*yy* is specified but the member FPXxxxyy does not exist, processing ends with an error. If the xxxSUF parameter is not specified in FPXCFG00, the default member FPXxxx00 is used. If the default member does not exist in PROCLIB, processing continues as if no global site default options are specified.

Keyword xxxPARMLIB=*a_parameter_library_dsname* specifies the name of the parameter library for utility *xxx*. The DB-level site default options are specified in a member with a DBD name (a DB member) which can be created for each DEDB, and the area-level site default options are specified in a member with an AREA name (an AREA member) which can be created for each area. The DB member is used for an area to be processed only when the corresponding AREA member does not exist.

When creating the configuration member, follow these rules:

- Each keyword parameter must be specified on an 80-byte line.
- A line must not contain more than one keyword parameter.
- Each keyword parameter must start in the first column.

- The keyword parameters can be specified in any order.
- If a same keyword parameter is specified more than once, the value specified in the last occurrence of the keyword parameter is used.

The following figure is an example of the content of FPXCFG00.

```
GBLSUF=01
OPCSUF=01
ODESUF=00
OERSUF=01
ODMSUF=02
OPCPARMLIB=HPFP.OPC.PARMLIB
ODEPARMLIB=HPFP.ODE.PARMLIB
OERPARMLIB=HPFP.OER.PARMLIB
ODMPARMLIB=HPFP.ODM.PARMLIB
```

Figure 371. Sample configuration member FPXCFG00

Statements in the sample FPXCFG00 means as follows:

- The global site default options of the GLOBAL control statement are specified in member FPXGBL01
- The global site default options of the OPC utility are specified in member FPXOPC01
- The global site default options of the ODE utility are specified in member FPXODE00
- The global site default options of the OER utility are specified in member FPXOER01
- The global site default options of the ODM utility are specified in member FPXODM02
- The DB-level site default options or the area-level site default options of the OPC utility are created in HPFP.OPC.PARMLIB
- The DB-level site default options or the area-level site default options of the ODE utility are created in HPFP.ODE.PARMLIB
- The DB-level site default options or the area-level site default options of the OER utility are created in HPFP.OER.PARMLIB
- The DB-level site default options of the ODM utility are created in HPFP.ODM.PARMLIB

Setting members for site default options

The FPXxxxxyy members that contain the global site default options for each utility should be created in a library that is concatenated to the PROCLIB DD statement. The DB members that contain the DB-level site default options and the AREA members that contain the area-level site default options should be created in the parameter library that is specified in FPXCFG00.

About this task

The statements specified in each of the following members must follow the control statement syntactical rules for each utility:

- A PROCLIB member for the global site default options
- A DB member in the parameter library
- An AREA member in the parameter library

If none of these members exist for the utility, no site default options are applied to the subject utility.

When FPXIN DD is specified in the JCL stream, the control statements of site default must start with a line that contains one of the following commands:

- GLOBAL
- ANALYZE
- DMACPRT
- EXTRACT
- REORG

For FPXIN syntax rules, see [“Syntax rules for FPXIN”](#) on page 835.

When SYSIN DD is specified in the JCL stream, the control statements of site default must not start with a line that contains these commands. For syntax rules, see the following topics:

- [“Syntax rules for DASYSIN”](#) on page 836
- [“Syntax rules for FPXCTL and areaname”](#) on page 836

Example

The following figure is a sample member of the site default options for OER.

Note that the AREA keyword, which must be specified in FPXIN DD in a JCL stream for an OER job step, is not specified in the REORG statement in the member for the site default options.

```
REORG
  RFSSCAN=(YES,YES)
  MAXFAILS=10
  MFILE=FPXMFIL
  PTRCHK=YES
  RBASEFS=35
  RDOVFFS=50
  REORGLIM=5
PTRCHK
  CHKUOW=ALL
```

Figure 372. Sample member for site default options

Chapter 28. Return code user exit routine (FPXRCEX0)

IMS Fast Path Online Tools supports a return code user exit routine that can be used to change the return codes of OPC, ODE, or OER.

Overview of the return code use exit routine

The following list provides the overview of the return code user exit routine:

- This user exit routine must be named FPXRCEX0, and should be included in STEPLIB or JOBLIB.
- If FPXRCEX0 is found by OPC, ODE, or OER, and RCEXLOAD=NO is not specified, it will be called no matter what the return code is.
- You can change the return codes according to the message and return code table.
- FPXRCEX0 must be specified with AMODE=31 and RMODE=ANY.
- FPO provides a sample JCL (FPXRCEXT) in *.SHFPSAMP. FPXRCEXT is composed of two steps: one to assemble FPXRCEX0, and another to link edit FPXRCEX0.
- For this release, the interface is available only for OPC, ODE, and OER.

Interface of the return code user exit routine

The following figure shows the input and output of the return code user exit routine.

Input

- R1**
Parameter list address
- R13**
Save area address
- R14**
Return address
- R15**
Entry point of FPXRCEX0

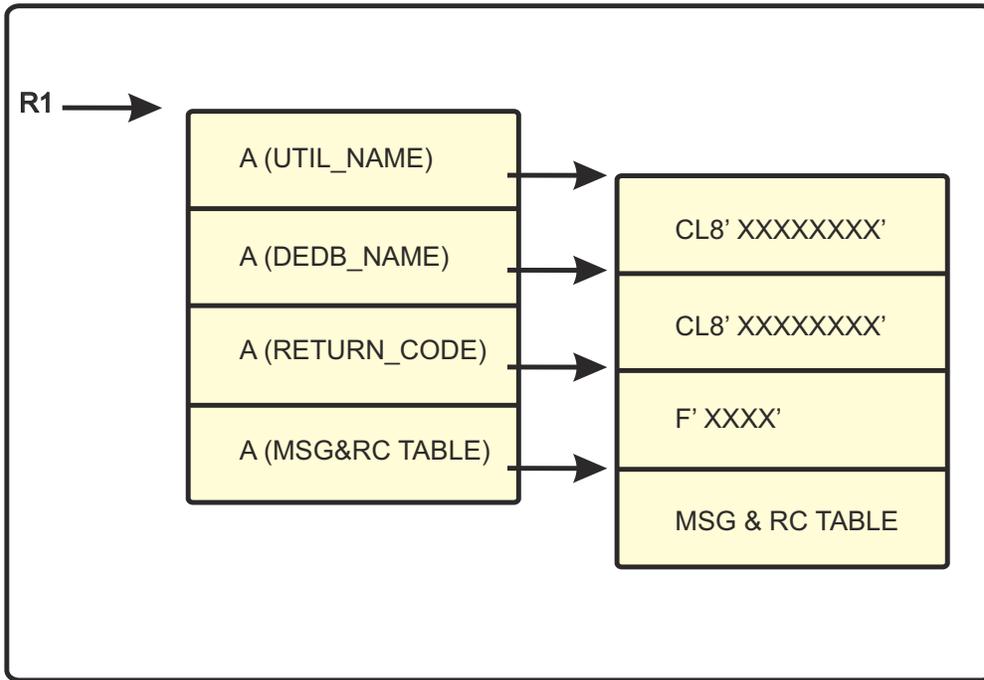


Figure 373. Parameter list

A(UTIL_NAME):

Address of the 8-byte utility name; either OPC, ODE, or OER.

A(DEDB_NAME):

Address of the 8-byte DEDB name.

A(RETURN_CODE):

Address of the full-word return code.

Input: The original return code.

Output: The changed return code.

A(MSG&RC TABLE):

The address of the message and return code table.

Every entry contains the following information:

- Message ID: 2-byte
- Message suffix: 1-byte
- Reserved area: 1-byte
- Return code: Full-word, currently this field is not used.

The following table shows an example of a message and return code table. If messages FPX0039I, FPX0116W, and FPX0332I are issued, the message and return code table would look like:

Table 113. Example of a message and return code table

Message ID	Message suffix	Reserved area	Reserved area
X'0003'	X'00'	X'00'	X'00000000'
X'0027'	X'C9'	X'00'	X'00000000'
X'0074'	X'E6'	X'00'	X'00000000'
X'014C'	X'C9'	X'00'	X'00000000'

Where X'0003' is the number of entries in the message and return code table.

The following messages are not listed in the message and return code table even if these messages are issued by FPO:

FPX0117I, FPX0118I, FPX0173I, FPX0174I, FPX0175E, FPX0181I, FPX0182W, FPX0183E,
FPX0184E, FPX0185W, FPX0323I, FPX0324I, FPX0325E, FPX0326W, FPX0327E

Output

The changed return code that is pointed to by address in A(RETURN_CODE).

Chapter 29. RACF considerations for FPO

If you are planning to use security validation, such as those provided by RACF, you need to activate the security function.

Before any FPO processing is run, RACF verifies that the user ID of the IFP utility is authorized to class IMSTxxx to run the required function, where xxx can be OPC, ODE, OAE, or ODM. For OER, if you are planning to use security validation when the Area Sensor or OPC is used, specify OPC for xxx. If validation fails, processing ends.

The following call is issued by the security module to validate that the user has sufficient authority to access the utility of FPO:

```
RACROUTE REQUEST=AUTH,APPL='FPX',ATTR=READ,
        CLASS='IMSTxxx',ENTITYX='imsid.dbname.areaname'
```

Note: The high-level qualifier of the ADS does not need to be the same as the ENTITYX parameter 'imsid.dbname.areaname'.

To activate security, you need to run two tasks: activate CLASS IMSTxxx and set up the appropriate RACF permit list.

If RACF is being used, a system IPL is required to activate the updated class descriptor table.

The following figure is an example of the class descriptor entries, which must be added to the class descriptor table maintained by the RACF ICHERCDE utility.

```
IMSTxxx ICHERCDE CLASS=IMSTxxx, +
        ID=128, +
        MEMBER=IMSMxxx, +
        MAXLNTH=128, +
        OTHER=ANY, +
        POSIT=25, +
        OPER=NO, +
        RACLIST=DISALLOWED, +
        GENLIST=DISALLOWED, +
        DFTUACC=NONE
IMSMxxx ICHERCDE CLASS=IMSMxxx, +
        ID=129, +
        GROUP=IMSTxxx, +
        MAXLNTH=128, +
        OTHER=ANY, +
        POSIT=25, +
        OPER=NO, +
        RACLIST=DISALLOWED, +
        GENLIST=DISALLOWED, +
        DFTUACC=NONE
```

Figure 374. Example of the class descriptor entries for an FPO utility

The following figure shows an example provided to help you in setting up your RACF permit list for an FPO utility.

Remember, if you update the class descriptor table, you must IPL the system to activate it.

```
RDEFINE IMSTxxx imsid.dbname.areaname0 UACC(NONE)
RDEFINE IMSTxxx imsid.dbname.areaname1 UACC(NONE)
RDEFINE IMSTxxx imsid.dbname.areaname2 UACC(NONE)

PERMIT imsid.dbname.areaname0 CLASS(IMSTxxx) ID(userid1) ACCESS(UPDATE)
PERMIT imsid.dbname.areaname0 CLASS(IMSTxxx) ID(userid2) ACCESS(UPDATE)
PERMIT imsid.dbname.areaname1 CLASS(IMSTxxx) ID(userid1) ACCESS(UPDATE)
PERMIT imsid.dbname.areaname1 CLASS(IMSTxxx) ID(userid2) ACCESS(UPDATE)
/*
```

Figure 375. Example to set up the FPO RACF permit list

Part 4. Online Space Management utility

The following topics introduce the Online Space Management (OSM) utility of IMS HP Fast Path Utilities and explain how to use the utility.

Topics:

- [Chapter 30, “Online Space Management utility overview,” on page 955](#)
- [Chapter 31, “Using the Online Space Management utility in stand-alone mode,” on page 959](#)
- [Chapter 32, “Using the Online Space Management utility in the active autonomies environment,” on page 963](#)
- [Chapter 33, “JCL requirements for the Online Space Management utility,” on page 971](#)
- [Chapter 34, “Utility control statements for the Online Space Management utility,” on page 977](#)
- [Chapter 35, “Online Space Management utility reports,” on page 1001](#)

Chapter 30. Online Space Management utility overview

The Online Space Management utility facilitates reorganization and area extension tasks for online IMS DEDBs. The utility provides advanced functions that help IMS database administrators manage the space utilization of DEDB areas without taking DEDB areas offline.

Topics:

- [“Online Space Management utility functions” on page 955](#)
- [“Online Space Management utility benefits” on page 956](#)
- [“Online Space Management utility input and output” on page 956](#)

Online Space Management utility functions

The Online Space Management utility facilitates reorganization and area extension tasks for online IMS DEDBs by integrating and automating these tasks.

The Online Space Management utility integrates the following functions of IMS Fast Path Online Tools (FPO) into one job step:

- *Reorganization support* to reorganize a set of UOWs in a DEDB area based on the ranked set of UOWs.
- *Area extension support* to extend the size of the IOVF portion, SDEP portion, or both of the DEDB area.
- *DEDB analysis and Integrated Database Sensor support* to verify the DEDB integrity, generate analysis reports, and collect sensor data for the DEDB area after the area is reorganized or extended.

The Online Space Management utility reads one or more actions, which are specified through the input control statements, to process the DEDB area. The actions that are supported are reorganization, IOVF extension, and SDEP extension. For each action, the utility internally calls the relevant FPO utility:

- For reorganization, the utility calls the Online Expert Reorganization utility (OER).
- For area extension, the utility calls the Online Area Extender utility (OAE).

Verification of the DEDB integrity and generation of the analysis reports are done by the Online Pointer Checker utility (OPC) process, which is always started after a reorganization or area extension. When IMS Tools KB server is specified in the JCL statements, sensor data is collected during the OPC utility process.

The processing options to control each utility process are read from the Online Space Management utility JCL stream. When IMS Tools KB server is specified in the JCL statements, the Online Space Management utility also reads sensor data from the Sensor Data repository of IMS Tools KB. For area reorganization, if certain data element values are stored in the repository, the utility determines the UOWs to reorganize based on the sensor data. Therefore, the control statement keywords for selecting UOWs (RBASEFS and RDOVFFS keywords of the OER utility) can be omitted in the Online Space Management utility JCL stream.

You can run the Online Space Management utility in stand-alone mode or automate the utility job in the *active autonomies environment*.

- In stand-alone mode, the utility runs the functions that you request through the control statements.
- The active autonomies environment is enabled by IMS HP Fast Path Utilities and Autonomics Director.

In this environment, Autonomics Director periodically collects and evaluates sensor data, makes recommendations, and starts Online Space Management utility jobs to perform the recommended actions. Therefore, the functions that are run by the utility are determined based on the state of the areas. For example, if Autonomics Director identifies that area AREA1 requires reorganization and area AREA2 requires SDEP extension, it schedules and submits an Online Space Management utility job to reorganize AREA1 and another Online Space Management utility job to extend the SDEP portion of AREA2.

The JCL stream for the Online Space Management utility is created on demand for each area from a skeletal JCL for the utility and actions that are recommended by Autonomics Director.

Restriction: An Online Space Management utility job can process only one area with one job step.

Related concepts

Using the Online Space Management utility in stand-alone mode

By running the Online Space Management utility in stand-alone mode, you can reorganize, extend, and collect sensor data from an online area in one job step.

Using the Online Space Management utility in the active autonomics environment

By using the Online Space Management utility with active autonomics features of Autonomics Director, you can automate database reorganization, IOVF extension, and SDEP extension tasks.

Online Space Management utility benefits

The Online Space Management utility provides advanced functions that help IMS database administrators manage the space utilization of DEDB areas without taking DEDB areas offline.

Reorganization and extension of DEDB areas typically involve complex analysis tasks. Generally, these time-consuming tasks require knowledge, expertise, and experience in space management of the IMS DEDBs. The Online Space Management utility simplifies reorganization and extension of DEDB areas by integrating and automating the database space management tasks.

The Online Space Management utility provides the following benefits:

- Provides an easy-to-use, flexible utility for database space management.
- Integrates reorganization support, area extension support, and DB Sensor support in one utility.
- Performs multiple database space management tasks in one job step.
- Reduces database space management costs significantly when the utility is used with active autonomics features of Autonomics Director. The database is reorganized or extended only when a database exception that is associated with the action is detected.

Online Space Management utility input and output

The Online Space Management utility provides area reorganization support, area extension support, and integrated DB Sensor support. The utility can be run as a stand-alone utility, or it can be used with active autonomics features of Autonomics Director.

The following figure shows the input and output for the Online Space Management utility. The OSM controller starts one or more utility processes by reading the actions that are specified in the input control statements.

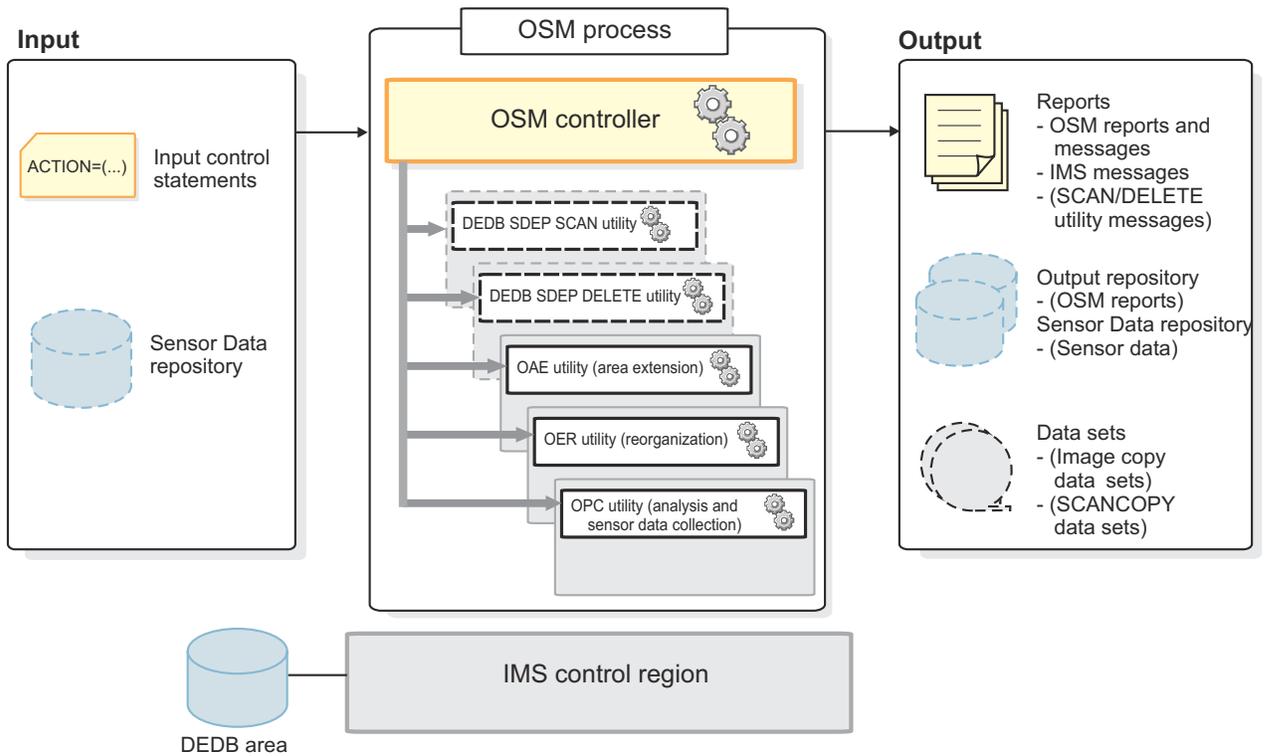


Figure 376. Input and output for the Online Space Management utility

Chapter 31. Using the Online Space Management utility in stand-alone mode

By running the Online Space Management utility in stand-alone mode, you can reorganize, extend, and collect sensor data from an online area in one job step.

Topics:

- [“Running the Online Space Management utility job” on page 959](#)
- [“Sample scenario for using the Online Space Management utility in stand-alone mode” on page 960](#)

Running the Online Space Management utility job

You can use the sample JCL member to create JCL for the Online Space Management utility. In stand-alone mode, you specify the functions to perform, which can be reorganization, area extension, or both, in the control statements.

Procedure

1. Create a JCL member for the Online Space Management utility.

You can use the sample JCL member, HFPOSM0J, in the SHFPSAMP data set. This member is the skeleton JCL for use with Autonomics Director so you must replace the symbolic variables (\$DBNAME\$, \$ARNAME\$, \$IMSSYS\$, and \$ACTION\$) with the names of database and area to process, IMS ID, and the actions to perform.

Modify the DD statements and control statements to meet your requirements:

- For DD statements, see [Chapter 33, “JCL requirements for the Online Space Management utility,” on page 971.](#)
- For control statements, see [Chapter 34, “Utility control statements for the Online Space Management utility,” on page 977.](#)

Tip: You can have the site default options that are defined for the OER utility and the OPC utility applied to the OER and OPC utility processes. For more information, see [Chapter 27, “Site default options support for FPO,” on page 943.](#)

2. Optional: Estimate the region size that is required for the job and, if necessary, specify the region size on the EXEC statement by coding the REGION parameter.

Although the Online Space Management utility operates as z/OS batch job, reorganization, area extension, and sensor data collection are performed by IMS IFP utilities.

The region size for an Online Space Management utility job depends on the size of a UOW and the keyword specifications. Because the main functions of the Online Space Management utility operate in a virtual storage region as IMS IFP utilities, the region size that is required to run the Online Space Management utility is larger than the following values:

- For the buffer sets portion for UOW processing, calculate the required storage by using the following formula:

```
Buffer sets: UOW1 * CISZ * 5
```

If the value of UOW1 is smaller than 120, 120 is used instead of UOW1.

- For the IOVFPPOOL-dependent portion, calculate the required storage by using the following formulas:

```
Work buffer: (UOW1 + IOVFPPOOL) * CISZ * 10
IOVF private buffer: IOVFPPOOL * CISZ * 5
```

Tip: The MAXBUFFS keyword of the SETREORGPARM subcommand specifies the number of buffer sets that are used in reorganization. The meaning of MAXBUFFS is the same as that of the BUFNO command for the High-Speed DEDB Direct Reorganization utility (DBFUHDRO).

3. Submit the job.
4. Check the job log and ensure that the job completed successfully.

Locate message HFPT0011I in the HFPPRINT DD. Message HFPT0011I indicates that the utility process for the area completed successfully. To view detailed information about each process in the Online Space Management utility job, see the following resources:

Resource	Purpose
Message FPX0099I in the SYSPRINT DD	To see the result of area extension.
Reorganization Processing Summary report in the FPXRPTS DD	To see the result of reorganization.
Sensor Data Statistics report in the FPXSPRT DD	To see the collected sensor data values.

If you cannot locate message HFPT0011I, locate the following messages:

HFPT0012W

This message indicates that the utility process completed for the area, but one or more warning conditions were encountered. Locate other warning or error messages to identify the cause.

HFPT0013E

This message indicates that the utility process ended with one or more errors. To identify the cause, locate the following messages:

- For the OAE, OER, or OPC utility process, locate the FPX error messages (FPXnnnnE).
- For the SDEP SCAN or SDEP DELETE process, locate the DFS error messages in the SYSPRINT DD.

Related reference

[Tips and best practices for the Online Space Management utility](#)

The Online Space Management utility (OSM) supports several keywords that simplify the management of DEDBs. The following sections describe how to use those keywords to improve the performance of OSM jobs and further simplify management of DEDBs.

Sample scenario for using the Online Space Management utility in stand-alone mode

This scenario describes a case where a DBA maintains a DEDB area that has insufficient space in the IOVF portion, and its segments often become scattered across many separate CIs or the free space in CIs become fragmented into small pieces.

About this scenario

The following steps describe how a DBA uses the Online Space Management utility in stand-alone mode to manage the space utilization of a DEDB area.

Scenario

1. The DBA periodically runs the OPC job to monitor the DEDB area.
After each run of the job, the DBA examines the reports that were generated by the OPC utility to check the state of the area.
2. The DBA detects an exceptional state in the area. From the OPC utility reports, the DBA discovers that the number of physical I/O is higher than the threshold, and the free space of the IOVF portion is lower

than the threshold. From the exceptional state, the DBA determines that expansion of the IOVF portion and reorganization are needed for the area.

3. The DBA submits the following JCL to start the Online Space Management utility job.

```
//OSM          EXEC PGM=HFPMAIN0
//STEPLIB      DD DISP=SHR,DSN=HPFP.SHFPLMD0
//            DD DISP=SHR,DSN=IMSVS.SDFSRESL
//            DD DISP=SHR,DSN=ITB.SHKTLOAD
//IMSDALIB     DD DISP=SHR,DSN=IMSVS.MDALIB
//PROCLIB      DD DISP=SHR,DSN=IMSVS.PROCLIB
//SYSPRINT     DD SYSOUT=*
//SYSOUT       DD SYSOUT=*
//HFPSYSIN     DD *
GLOBAL
  IMSID=IMS1
MANAGE
  DBD=DEDBJN22,
  IAREA=DB22AR0,
  ACTION=(REORG,EXTENDIOVF)
SETEXTENDPARM
  ACBLIB=IMSVS.ACBLIB,
  DATACLAS=DATA,
  MGMTCLAS=MGMT,
  STORCLAS=STOR,
  DSNAME='TEMP.OAE.DB22AR0.ADS',
  DDNAME=TEMPADS,
  IOVF_UOW=10%
SETREORGPARM
  RBASEFS=30,
  RDOVFFS=50
/*
```

Figure 377. JCL example: JCL for the Online Space Management utility

In this JCL, ACTION=(REORG,EXTENDIOVF) specifies that the OAE utility process is started to expand the IOVF portion of the area and the OER utility process is started to reorganize the area.

When the area is expanded and reorganized, the Online Space Management utility starts the OPC utility job to generate analysis reports.

4. The DBA checks the job log to verify that the job completed successfully.
5. The DBA examines the OPC utility reports that were generated during the Online Space Management utility job to check the state of the expanded and reorganized area. The DBA did not detect any exceptional state in the area.
6. The DBA brings online the new DEDB member in the ACBLIB because the IOVF portion of the area was extended by the OAE utility and the OAE utility requires post execution steps.

Chapter 32. Using the Online Space Management utility in the active autonomics environment

By using the Online Space Management utility with active autonomics features of Autonomics Director, you can automate database reorganization, IOVF extension, and SDEP extension tasks.

Topics:

- [“Online Space Management utility in the active autonomics environment” on page 963](#)
- [“Configuring the active autonomics environment” on page 964](#)
- [“Verifying successful completion of Online Space Management utility jobs” on page 968](#)

Online Space Management utility in the active autonomics environment

The Online Space Management utility integrates with active autonomics features of Autonomics Director. Together, they establish the *active autonomics environment* for IMS Fast Path DEDBs.

In the active autonomics environment, Autonomics Director periodically collects and evaluates sensor data against your defined policies and rules, recommends database reorganizations and area extensions as needed, and initiates the Online Space Management utility job to perform reorganizations and area extensions.

The active autonomics environment is implemented through Autonomics Director's group-managed reorganizations. You can define a group of areas that belong to a DEDB, have same characteristics, or are used by a group of applications, and have those areas managed with the same condition. Managing areas at a group level can help minimize the amount of time and manual intervention that is required to maintain and optimize the areas. The ability to initiate and control reorganization and area extensions by group adds efficiency and allows for these activities to be processed in parallel for multiple areas.

The group-managed reorganization structure supports four separate phases of scheduling:

1. Pre-group-managed reorganization
2. Reorganization (also includes IOVF and SDEP extensions)
3. Post reorganization
4. Post-group-managed reorganization

In the active autonomics environment, only the reorganization phase is required. The other phases are optional. The Online Space Management utility is activated in the reorganization phase.

Autonomics Director can schedule Online Space Management utility jobs for a subset or all areas in a database maintenance window or in non-peak hours of the application group. Factors that influence Autonomics Director's decision to schedule evaluation and Online Space Management utility jobs include the evaluation policy, evaluation interval, sensor data maximum age, application peak period definitions, and database maintenance window definitions.

To learn more about group-managed reorganization and an example of how Autonomics Director maintains DEDBs in the active autonomics environment, see the *Tools Base Autonomics Director User's Guide*.

The following figure shows the general process flow in the active autonomics environment.

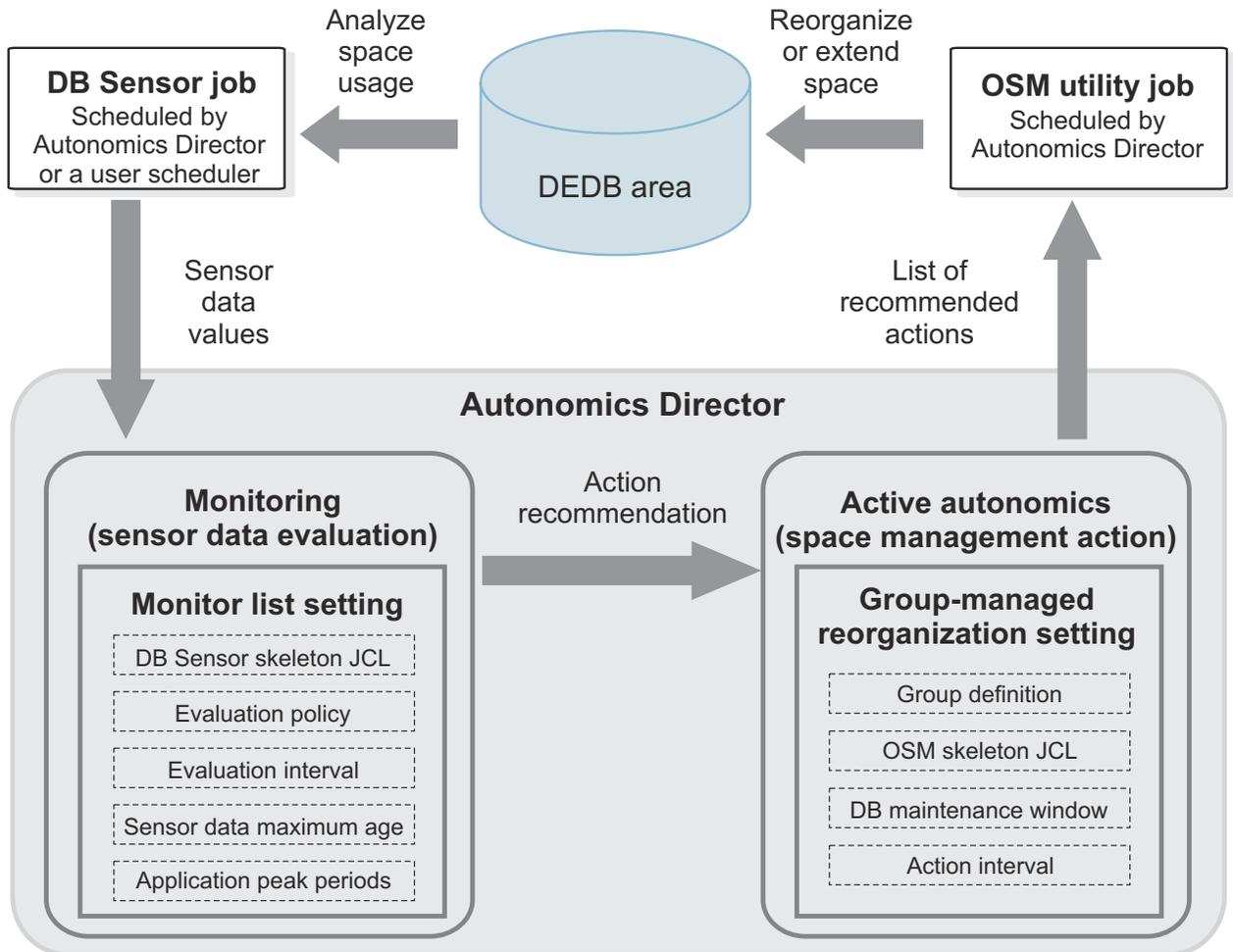


Figure 378. Process flow in the active autonomics environment

For more information about active autonomics features and how to use them, see the topic "Autonomics overview" in the *Tools Base Autonomics Director User's Guide*.

Related tasks

[Configuring the active autonomics environment](#)

To establish the active autonomics environment, you must configure Policy Services, prepare skeleton JCL for use with DB Sensor and the Online Space Management utility, and configure Autonomics Director.

Configuring the active autonomics environment

To establish the active autonomics environment, you must configure Policy Services, prepare skeleton JCL for use with DB Sensor and the Online Space Management utility, and configure Autonomics Director.

Before you begin

- Ensure that IMS Tools Knowledge Base and Policy Services are configured. See the *Tools Base Configuration Guide for IMS*.
- By listing the installed policies and rules (see the topic "Listing installed policies and rules" in the *Tools Base Configuration Guide for IMS*), ensure that the following rules are installed:
 - IBM.DEDB_FS.31
 - IBM.DEDB_FS.81
 - IBM.DEDB_RFS.11
 - IBM.DEDB_RFS.21

These rules are used for detecting exceptional states of DEDB areas. These rules also associate each exceptional state with a specific space management action that removes the exceptional state. If these rules are not installed, see the topic "Installing initial domain policies and rules from IBM maintenance data set" in the *Tools Base Configuration Guide for IMS* and install the rules.

About this task

You must first configure Policy Services so that appropriate space management action recommendations are generated by sensor data evaluation. Customize the IBM-provided policy by assigning an action to specific exceptions. When exceptions are detected, Autonomics Director initiates the Online Space Management utility to process the associated actions.

Then, you create sensor JCL for collecting sensor data. The sensor JCL is used for the automated collection of sensor data and the on-demand collection of sensor data. The collected sensor data is evaluated with the policy that you define for the area, and recommendations for reorganization, IOVF extension, and SDEP extension are made if they are needed.

Next, you create skeleton JCL for the Online Space Management utility, which controls one or more of the reorganization, IOVF extension, and SDEP extension processes that are run by the Online Space Management utility.

After you prepare the skeleton JCL, you define the maintenance window, add areas to the monitor list, and define a user group by using the Autonomics Director ISPF client. While you activate the user group, you specify the skeleton JCL for DB Sensor, the policy, the skeleton JCL for Online Space Management utility, and the maintenance window.

Procedure

1. Determine which actions you want to automate in the active autonomics environment.

You can automate one or more of the following actions:

REORG

Reorganizes the area if an exceptional state that can be resolved by free space reorganization is detected in the area.

EXTENDIOVF

Extends the IOVF portion of the area if an exceptional state that can be resolved by IOVF extension is detected in the area.

EXTENDSDEP

Extends the SDEP portion of the area if an exceptional state that can be resolved by SDEP extension is detected in the area.

2. Customize the IBM-provided policy SYS.DBDDTYPE.DEDB to assign actions for specific exceptions. Use the ISPF user interface of Policy Services to do so.

Based on this policy, Autonomics Director determines its action: it either submits an Online Space Management utility job or only sends a notification message.

The rules that you can add or remove are summarized in the following table. You can also create your own policy by modeling the policy SYS.DBDDTYPE.DEDB.

Tip: See the topic "Using Policy Services" in the *Tools Base Policy Services User's Guide* for information about customizing IBM-provided policies and rules, and creating your own policies. The user's guide also provides a complete reference for IBM-provided policies and rules.

Table 114. Rules for automating actions

Action to automate	Rule to add	Rule to remove	Description
REORG	IBM.DEDB_RFS.11	IBM.DEDB_RFS.10 (if already added)	<p>Both IBM.DEDB_RFS.10 and IBM.DEDB_RFS.11 are used to detect a condition that needs free space reorganization of an area. The condition is evaluated based on thresholds on the number of UOWs that match the RBASEFS or RDOVFFS condition. An exception is reported if a threshold is exceeded.</p> <ul style="list-style-type: none"> • If you want Autonomics Director to automatically submit an Online Space Management utility job that reorganizes the area, use IBM.DEDB_RFS.11. • If you do not want Autonomics Director to automatically submit an Online Space Management utility job but want to be notified of the exception, use IBM.DEDB_RFS.10. <p>Do not use both rules together in a single policy.</p> <p>You must modify the threshold values in these rules because dummy values are set as the default threshold values.</p>
REORG	IBM.DEDB_RFS.21	IBM.DEDB_RFS.20 (if already added)	<p>Both IBM.DEDB_RFS.20 and IBM.DEDB_RFS.21 are used to detect a condition that needs free space reorganization of an area. The condition is evaluated based on thresholds on the percentage of UOWs that match the RBASEFS or RDOVFFS condition. An exception is reported if a threshold is exceeded.</p> <ul style="list-style-type: none"> • If you want Autonomics Director to automatically submit an Online Space Management utility job that reorganizes the area, use IBM.DEDB_RFS.21. • If you do not want Autonomics Director to automatically submit an Online Space Management utility job but want to be notified of the exception, use IBM.DEDB_RFS.20. <p>Do not use both rules together in a single policy.</p> <p>You can modify the threshold values in these rules to meet your requirements.</p>
EXTENDIOVF	IBM.DEDB_FS.31	IBM.DEDB_FS.30	<p>Both IBM.DEDB_FS.30 and IBM.DEDB_FS.31 are used to detect a condition that needs extension of the IOVF portion of an area. The condition is evaluated based on thresholds on the percentage of free space in the IOVF portion of the area. An exception is issued if the percentage falls below a threshold.</p> <ul style="list-style-type: none"> • If you want Autonomics Director to automatically submit an Online Space Management utility job that extends the IOVF portion, use IBM.DEDB_FS.31. • If you do not want Autonomics Director to automatically submit an Online Space Management utility job but want to be notified of the exception, use IBM.DEDB_FS.30. <p>Do not use both rules together in a single policy.</p> <p>You can modify the threshold values in these rules to meet your requirements.</p>

Table 114. Rules for automating actions (continued)

Action to automate	Rule to add	Rule to remove	Description
EXTENDSDEP	IBM.DEDB_FS.81	IBM.DEDB_FS.80	<p>Both IBM.DEDB_FS.80 and IBM.DEDB_FS.81 are used to detect a condition that needs extension of the SDEP portion of an area. The condition check is based on thresholds on the percentage of free space in the SDEP portion of the area. An exception is issued if the percentage falls below a threshold.</p> <ul style="list-style-type: none"> • If you want Autonomics Director to automatically submit an Online Space Management utility job that extends the SDEP portion, use IBM.DEDB_FS.81. • If you do not want Autonomics Director to automatically submit an Online Space Management utility job but want to be notified of the exception, use IBM.DEDB_FS.80. <p>Do not use both rules together in a single policy.</p> <p>You can modify the threshold values in these rules to meet your requirements.</p>

3. If you made these changes in a maintenance environment, promote the environment into the operation environment so that the changes are applied to the operation environment.
4. Create skeleton JCL for collecting sensor data. For JCL examples, see the topic "Creating skeleton JCL for Data Sensors" in the *Tools Base Configuration Guide for IMS*.

The sensor JCL is used for the automated collection of sensor data and the on-demand collection of sensor data. The collected sensor data is evaluated with the policy that you define for the area, and recommendations for reorganization, IOVF extension, and SDEP extension are made if they are needed.

5. Create skeleton JCL for the Online Space Management utility, which controls one or more of the reorganization, IOVF extension, and SDEP extension processes. You can use the sample JCL member, HFPOS0J, in the SHFPSAMP data set.

See the topic "Creating skeleton JCL for Fast Path Online Space Management utility" in the *Tools Base Configuration Guide for IMS* to create skeleton JCL.

For DD statements and control statements for the Online Space Management utility, see [Chapter 33, "JCL requirements for the Online Space Management utility," on page 971](#) and [Chapter 34, "Utility control statements for the Online Space Management utility," on page 977](#).

6. By using the Autonomics Director ISPF client, configure Autonomics Director. For instructions, see the topic "Automating monitoring and maintenance tasks" in the *Tools Base Autonomics Director User's Guide*.

What to do next

The Autonomics Director scheduler initiates the job based on the schedule.

- If an exception is detected by policy evaluation and a notification list is specified for the policy, Autonomics Director sends a notification about the state of the areas. When you receive such a notification message, check the state of the areas.
- If the Online Space Management utility job is executed for an area as a result of policy evaluation, verify the completion of the Online Space Management utility job. If the IOVF portion of an area was extended (action EXTENDIOVF), you must perform post-processing on the area. For more information, see ["Verifying successful completion of Online Space Management utility jobs" on page 968](#).

Verifying successful completion of Online Space Management utility jobs

The Online Space Management utility job that is started by Autonomics Director includes one or more of the OAE, OER, and OPC utility processes, and optionally SDEP SCAN and SDEP DELETE utility processes. You can check whether each job completed successfully by checking the reorganization and extension history of user-defined groups through the Autonomics Director ISPF user interface.

Procedure

1. From the Autonomics Director Primary Option Menu panel, select option **5 Schedule a reorganization and area extend on demand**.

Reorganization and area extension history of the user-defined group is displayed.

Check the status of each job and ensure that the job completed successfully. You can also view the reorganization and area extension details.

For more information about viewing the reorganization and area extension history, see the topic "Viewing reorganization and area extension history of user-defined groups" in the *Tools Base Autonomics Director User's Guide*.

To view detailed information about each process in the Online Space Management utility job, see the following resources:

Resource	Purpose
Message HFPT0011I in the HFPPRINT DD	To confirm the completion of the job.
Message FPX0099I in the SYSPRINT DD	To see the result of area extension.
Reorganization Processing Summary report in the FPXRPTS DD	To see the result of reorganization.
Sensor Data Statistics report in the FPXSPRT DD	To see the collected sensor data values.

2. If any of the jobs failed, view the job log for that job.

- a) Locate the following messages:

HFPT0012W

This message indicates that the utility process completed for the area, but one or more warning conditions were encountered. Locate other warning or error messages to identify the cause.

HFPT0013E

This message indicates that the utility process ended with one or more errors. To identify the cause, locate the following messages:

- For the OAE, OER, or OPC utility process, locate the FPX error messages (FPXnnnnE).
- For the SDEP SCAN or SDEP DELETE process, locate the DFS error messages in the SYSPRINT DD.

- b) Locate WTO message HFPO1019I to verify that the result notification was sent to Autonomics Director.

If you cannot locate message HFPO1019I, locate messages HFPO1020W or HFPO1021W. These messages indicate that the result notification was not set to Autonomics Director. To identify the cause, see the messages in the HFPPRINT DD.

3. From the Autonomics Director Primary Option Menu panel, select option **2 View recommendation lists** to display the latest evaluation status of the areas.
4. If any areas are marked as action needed, examine the Database Diagnosis report for that area and plan for actions to resolve the state of the area.

If you need to tune the areas, you can use the Tune function of FPA. Also, see the *IMS Solution Packs IMS Database Space Tuning Guide* for guidance in tuning areas.

5. Identify the areas whose IOVF portion was extended. To do so, search for WTO message HFPO1022I containing the string EXTENDIOVF.

For these areas, you must plan IMS online change for each member in the ACB library. For more information about this step, see [“Step 3: Post execution steps for OAE” on page 757](#). The new DBD definitions for the extended areas can be found in the job logs.

Chapter 33. JCL requirements for the Online Space Management utility

DD statements for the Online Space Management utility determine the input and output data sets and specify how to run the utility.

Subsections:

- [“List of DD statements” on page 971](#)
- [“Input DD statements” on page 972](#)
- [“Output DD statements” on page 973](#)
- [“Input/Output DD statements” on page 976](#)

List of DD statements

The following table summarizes the DD statements for the Online Space Management utility.

Table 115. DD statements for the Online Space Management utility

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
JOBLIB or STEPLIB	Input	PDS	No	Required
PROCLIB	Input	PDS	No	Required
IMSDALIB	Input	PDS	No	Optional
DFSRESLB	Input	PDS	No	Optional
HFPSYSIN	Input	LRECL=80	No	Required
HFPPRINT	Output	SYSOUT	Yes	Optional
HFPRTS	Output	SYSOUT	Yes	Optional
SYSPRINT	Output	SYSOUT	No	Required
FPXMFIL	Output	SYSOUT	Yes	Optional
FPXRPTS	Output	SYSOUT	Yes	Optional
FPXSPRT	Output	SYSOUT	Yes	Optional
SYSOUT	Output	SYSOUT	Yes	Optional
RECON <i>n</i>	Input and output	KSDS	Yes	Required when DFSMDA members are not used
<i>i001WKnn</i>	Input and output		Yes	Optional
<i>i001SMMSG</i>	Output	LRECL=133	Yes	Optional
ICEIN	Input	LRECL=80	No	Optional
ICEPRINT	Output	LRECL=133	Yes	Optional
ICERPRT	Output	LRECL=133	No	Optional
<i>dataout1</i>	Output	PS	Yes	Optional

Table 115. DD statements for the Online Space Management utility (continued)

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
<i>dataout2</i>	Output	PS	Yes	Optional
FPXBINTP	Output	PS	No	Required when EXTEND_TYPE=RENAME
FPXBX RTP	Output	PS	Yes	Optional
SCANCOPY	Output	PS	No	Required when SCANDELT=YES
INDD	Input	LRECL=80	No	Optional

Input DD statements

The following input DD statements are used for the Online Space Management utility.

JOBLIB/STEPLIB DD

This DD statement is a required DD statement that defines the following input data sets:

- The OSM load module library (required).
- The IMS load module library (required).
- The IMS Tools Base library (required).
- The library that contains DFSMDA dynamic allocation members for the RECON data set. When the RECON n DD statement is omitted, this DD statement is required.
- The IMS HP Image Copy library, if you specify the ICOUT=YES option.

The data set that contains the DBRC SCI registration exit routine (DSPSCIX0) must be concatenated when using the RECON data set in which an IMSplex name is set.

The Online Space Management utility must be run as an APF-authorized program. All of the data sets that are listed within the JOBLIB or the STEPLIB concatenation must also be APF-authorized. If you use DFSMDA members and the DFSMDA members are in a non-APF authorized library, you can specify the DFSMDA library on the IMSDALIB DD statement.

PROCLIB DD

This DD statement is a required DD statement. If configuration member FPXCFG00 is included in this library, the Online Space Management utility applies the site default options to the OER and OPC utility processes.

IMSDALIB DD

This DD statement is an optional DD statement that defines the IMSDALIB data set. This DD statement is useful when the data set for dynamic allocation members is not APF-authorized.

DFSRESLB DD

This DD statement is an optional DD statement that points to the authorized library that contains the IMS SVC modules.

HFPSYSIN DD

This DD statement is a required DD statement that defines the input control statements to control the behavior of Online Space Management utility processes.

The HFPSYSIN DD statement can be coded as a standard SYSIN file, a sequential data set, or a PDS member. LRECL=80 is required for the DCB of this data set.

Online Space Management utility analyzes the command syntax in columns 1 - 72. Columns 73 - 80 are regarded as comment.

For more information about coding the HFPSYSIN DD statement, see [Chapter 34, “Utility control statements for the Online Space Management utility,”](#) on page 977.

ICEIN DD

This DD statement is an optional DD statement. This statement specifies the input data set that contains the control statements for IMS HP Image Copy. The data set can reside on a tape, a direct-access volume, or it can be routed through the input stream (DD * or DD DATA).

This data set is used only when you specify ICOUT=YES. If the DD statement is not coded and if ICOUT=YES is specified, the default control statements are used. For details about the ICEIN control statements, see *IMS High Performance Image Copy User's Guide*.

INDD DD

Defines the segment shunting input. You can specify which segments to shunt using this data set. This data set is similar to the INDD data set for the IMS DEDB High-Speed Direct Reorganization utility (HSR).

This data set must be a fixed block with a logical record length (LRECL) of 80, a block length of 80, and one input card per record. You can also specify NULLFILE or DUMMY, or an empty data set. The input card must be left-aligned or the card is ignored. This data set is only required for those interested in the segment shunting function.

The only segments that can be shunted are direct dependent segments. Root segments and SDEPs cannot be shunted.

The INDD data set consists of three input card types:

ERRACTN=TERM

An optional input card type. The card specifies that if any cards in the INDD data set contain invalid data, the job ends with return code 8 and reorganization will not occur. If you include this card, it must be the first card in the INDD data set.

TERM is the default value for this card. Even if ERRACTN=TERM is omitted or a value that is supported by HSR is coded (ERRACTN=CONT or ERRACTN=EXIT), the OSM utility always applies ERRACTN=TERM.

AREA=areaname

Starts the input stream for a single area. The input stream for this area terminates at the next AREA card, or at the end of the file. All of the following cards in the input stream are considered segments to shunt for the associated AREA card.

You can specify multiple AREA cards. The areas you specify with INDD AREA cards must also be specified with the IAREA keyword of HFPSYSIN DD.

segmentname

Contains the segment names associated with the prior AREA card.

Output DD statements

The following output DD statements are used for the Online Space Management utility.

HFPPRINT DD

This DD statement is an optional DD statement that defines the processing log output data set, which contains the HFP processing messages. If coded, HFPPRINT DD must point to SYSOUT.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, the Online Space Management utility dynamically allocates the data set by using SYSOUT=*

HFPRPTS DD

This DD statement is an optional DD statement that defines the report output data set. The data set contains the reports that can be used to verify the processes that are run by the Online Space Management utility. If coded, HFPRPTS DD must point to SYSOUT.

The following reports are generated in this data set:

- Audit report

- Processing report

Note on dynamic allocation:

When this statement is not specified in the JCL statements, the Online Space Management utility dynamically allocates the data set by using SYSOUT=*.

SYSPRINT DD

This DD statement is a required DD statement that defines the system message output data set. SYSPRINT DD must point to SYSOUT.

FPXMFIL DD

This DD statement is an optional DD statement that defines the processing log output data set, which contains the FPX processing messages. If coded, SYSPRINT DD must point to SYSOUT.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, the Online Space Management utility dynamically allocates the data set by using SYSOUT=*.

FPXRPTS DD

This DD statement is an optional DD statement that defines the report output data set, which contains the reports that are created by OPC and OER. If coded, FPXRPTS DD must point to SYSOUT.

The following reports are generated in this data set:

- Reports by OER:
 - Reorganization Processing Summary report
 - UOW Space Statistics report
 - Process Summary report
- Reports by OPC:
 - Online Pointer Checker Statistics
 - Online Pointer Scan report
 - Process Summary report
 - Resolve Pointers report
 - Processing Summary of Analysis report
 - DEDB Area Analysis reports
 - UOW report
 - UOW exception report
 - Image Copy NOTIFY report

OER reports are generated only when the area is reorganized.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, the Online Space Management utility dynamically allocates the data set by using SYSOUT=*.

FPXSPRT DD

This DD statement is an optional DD statement that defines the report output data set. This output data set contains the Sensor Data Statistics report that can be used to verify which sensor data is stored in the Sensor Data repository of IMS Tools KB. If coded, FPXSPRT DD must point to SYSOUT.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, the Online Space Management utility dynamically allocates the data set by using SYSOUT=*.

SYSOUT DD

This DD statement is an optional DD statement that is used by DFSORT for the message function. If coded, SYSOUT DD must point to SYSOUT.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, the Online Space Management utility dynamically allocates the data set by using SYSOUT=*.

i001SMMSG DD

This DD statement is an optional DD statement that defines an output data set for storing DFSORT messages. For the condition for running DFSORT and the naming rules for the DD names, see the description of *i001WKnn* DD.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, the Online Space Management utility dynamically allocates the data set by using SYSOUT=*.

ICEPRINT DD

This DD statement is an optional DD statement. This statement specifies the output data set that contains the reports that are produced by IMS HP Image Copy. The data set can reside on a tape, a direct-access volume, or a printer, or it can be routed through the output stream (SYSOUT).

This data set is used only when you specify ICOUT=YES. If the DD statement is not coded and ICOUT=YES is specified, a SYSOUT stream is dynamically allocated for the DD.

ICERPRT DD

This DD statement is an optional DD statement. This statement specifies the output data set that contains the Image Copy Dump Process report, which is produced by IMS HP Image Copy. It can reside on a tape, a direct-access volume, or a printer, or it can be routed through the output stream (SYSOUT).

This data set is used only when you specify ICOUT=YES.

dataout1 DD

This DD statement is an optional DD statement. This statement defines the primary image copy data set. One DD statement is required for each data set that is to be dumped. The DD name can be any 1- to 8-character string. The data set specified by the DD statement must be on either a direct-access device or a tape device.

This data set is used only when you specify ICOUT=YES. For details of the format and the characteristics of the image copy data set, see *IMS High Performance Image Copy User's Guide*. You can request IMS HP Image Copy to allocate the image copy data set dynamically by omitting this DD statement and specifying the allocation parameters that are associated with the data set in the ICEIN data set.

dataout2 DD

This DD statement is an optional DD statement. This statement defines the secondary image copy data set. This statement is required only if the associated ICEIN control statement requests the secondary image copy. The data set specified by the DD statement must be on either a direct-access device or a tape device.

This data set is used only when you specify ICOUT=YES. For details of the format and the characteristics of the image copy data set, see *IMS High Performance Image Copy User's Guide*.

FPXBINTP DD

This DD statement is a required DD statement for the original expansion process with the EXTEND_TYPE=RENAME option. It contains the interface information for the subsequent step, which is used in the rename process. Do not specify DCB parameters except RECFM=VB. Do not specify DISP=MOD for this DD statement.

FPXBX RTP DD

This DD statement is an optional DD statement that defines the SYSPRINT data set for the DSPURXRT module.

If this DD statement is not specified, the OSM utility allocates this DD data set dynamically.

Do not specify DUMMY or NULLFILE for this DD statement. Do not code the DCB information in the JCL.

SCANCOPY DD

This DD statement is a required DD statement when SCANDELT=YES is specified. It is the scan output data set that contains sequential dependent segments in variable-length, blocked records.

If you specify NULLFILE or DUMMY for this data set, no sequential dependent segments are written.

Restriction: Do not specify SYSOUT for this DD statement.

Input/Output DD statements

The following input/output DD statements are used for the Online Space Management utility.

RECON1/RECON2/RECON3 DD

These DD statements define the DBRC RECON data sets. The DD statements must be specified if DFSMDA members are not used. The RECON n data set must be the same data set that is used in the IMS control region.

Note on dynamic allocation:

If these statements are not specified in the JCL statements when the data sets are necessary, the data sets are allocated dynamically by using DFSMDA members.

i001WK nn DD

Optional DD statements. These are intermediate storage data sets that are used by DFSORT. For more information about coding the SORTWK nn DD statements, see the *DFSORT Application Programming Guide*. The adequate size of storage data sets is 1.5 to 2.5 times the input file size.

Allocating twice the space used by the SORTIN data set is usually adequate for each work data set.

Do not specify DISP=MOD for these DD statements.

In the analyze step, the size of each input file is as follows:

i:

A: Always used

$30 \times$ the number of pointers in the area

D: If SDEP=YES is specified

$30 \times$ the number of SDEP segments in the area

J: Always used

$X \times$ (the number of non-SDEP segments in the areas + the number of free space elements in the areas + the number of references to other CIs)

Where X is:

- When LARGEREC_REPORT=NO is specified, X is calculated as follows:

$34 + (\text{the maximum segment level} \times 4)$

However, if X is smaller than 48, 48 is used for X .

- When LARGEREC_REPORT=YES is specified, X is calculated as follows:

$34 + (\text{the maximum segment level} \times 4) + \text{root key length}$

However, if X is smaller than 48, 48 is used for X .

Q: Always used

$34 \times$ the number of segments in the area

Note on dynamic allocation:

If this statement is not specified in the JCL statement when these data sets are necessary, Online Space Management utility allocates them dynamically.

Chapter 34. Utility control statements for the Online Space Management utility

The utility control statements for the Online Space Management utility define the user description of the GLOBAL command, the MANAGE command, and two optional subcommands, SETEXTENDPARM and SETREORGPARM, all of which control the Online Space Management utility processes.

The utility control statements must be specified in the HFPSYSIN data set. This control statement data set usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte, fixed-length records. Block size, if coded, must be a multiple of 80.

The following figure shows an example of the HFPSYSIN data set.

```
//HFPSYSIN DD *
GLOBAL
  ADXCFGRP=Autonomics_Director_XCF_group_name,
  ITKBSRVR=servername,
  IMSID=imsid
MANAGE
  DBD=dbdname,
  IAREA=areaname,
  ACTION=(REORG)
/*
```

Figure 379. HFPSYSIN data set for the Online Space Management utility

The Online Space Management utility supports the site default options that are defined for the OER utility and the OPC utility. For more information about the site default options of these utilities, see [Chapter 27, “Site default options support for FPO,”](#) on page 943.

Subsections:

- [“GLOBAL command keywords”](#) on page 977
- [“MANAGE command keywords”](#) on page 978
- [“SETEXTENDPARM subcommand keywords”](#) on page 978
- [“SETREORGPARM subcommand keywords”](#) on page 979

GLOBAL command keywords

For the GLOBAL command, you can specify the keywords that are listed in the following table. For the format of the GLOBAL command, see [“GLOBAL command keywords”](#) on page 980.

Table 116. GLOBAL command keywords

Keyword	Required or optional	Description	Topic
ADXCFGRP	Optional	Specifies the name of the Autonomics Director XCF group.	“ADXCFGRP keyword” on page 980
ICNOTIFY	Optional	Specifies whether to issue the NOTIFY.IC command to DBRC.	“ICNOTIFY keyword” on page 981
ICOUT	Optional	Specifies whether to generate one or more image copies.	“ICOUT keyword” on page 981
IMSID	Required	Specifies the IMS system ID.	“IMSID keyword” on page 982
ITKBDATA	Optional	Specify the type of additional data to store in the IMS Tools Base IMS Tools Knowledge Base repository.	“ITKBDATA keyword” on page 982

Table 116. GLOBAL command keywords (continued)

Keyword	Required or optional	Description	Topic
ITKBSRVR	Optional	Specifies the name of the IMS Tools KB server XCF group.	“ITKBSRVR keyword” on page 983
SCAN	Optional	Specifies to verify the syntax of command, keywords, and parameters in command sets without running the functions of the Online Space Management utility. The Scan function also allocates some (but not all) internal resources for running the process function.	“SCAN keyword” on page 983

MANAGE command keywords

The MANAGE command specifies the action (function) and the area to process. It also supports keywords for process control.

For the MANAGE command, you can specify the keywords that are listed in the following table. For the format of the MANAGE command, see [“MANAGE command keywords” on page 984](#).

Table 117. MANAGE command keywords

Keyword	Required or optional	Description	Topic
ACTION	Required	Specifies the functions to run.	“ACTION keyword” on page 984
DBD	Required	Specifies the database description (DBD) that contains the area to be processed.	“DBD keyword” on page 985
IAREA	Required	Specifies the input area for the Online Space Management utility process.	“IAREA keyword” on page 985
REORGATIOVFEXT	Optional	Specifies to suppress reorganization of the area after the area is extended. This keyword is effective only when both the REORG and EXTENDIOVF parameters are specified for the ACTION keyword.	“REORGATIOVFEXT keyword” on page 986
SCANDELT	Optional	Specifies to process SDEPs by using the SDEP SCAN and DELETE utilities before area extension.	“SCANDELT keyword” on page 986

SETEXTENDPARM subcommand keywords

The SETEXTENDPARM subcommand controls the area extension process, which is run by OAE.

For the SETEXTENDPARM subcommand, you can specify the keywords that are listed in the following table. The keywords of the SETEXTENDPARM subcommand are the same as the keywords for the FPO OAE utility. For the format of the SETEXTENDPARM subcommand, see [“SETEXTENDPARM subcommand keywords” on page 987](#).

Table 118. SETEXTENDPARM subcommand keywords

Keyword	Required or optional	Description	Topic
ACBLIB	Optional	Specifies the ACBLIB to use.	“ACBLIB keyword” on page 987
DATACLAS	Optional	Specifies the name of the SMS data class for the temporary area data set.	“DATACLAS keyword” on page 988
DDNAME	Required	Specifies the DDNAME of the temporary area data set where the area will be copied.	“DDNAME keyword” on page 988
DSNAME	Required	Specifies the data set name of the temporary area data set where the area will be copied.	“DSNAME keyword” on page 989
EXTEND_TYPE	Optional	Specifies the extend type.	“EXTEND_TYPE keyword” on page 989

Table 118. SETEXTENDPARM subcommand keywords (continued)

Keyword	Required or optional	Description	Topic
IOVF_UOW	Optional	Specifies the number of IOVF UOWs to extend for IOVF extension. Either the IOVF_UOW or SDEP_CI keyword must be specified.	“IOVF_UOW keyword” on page 990
MAXVOLS	Optional	Specifies the maximum number of volumes for the extended original area data set. This keyword is effective only in SMS environments.	“MAXVOLS keyword” on page 990
MGMTCLAS	Optional	Specifies the name of the SMS management class for the temporary area data set.	“MGMTCLAS keyword” on page 990
SDEP_CI	Optional	Specifies the number of SDEP CIs to extend for SDEP extension. Either the SDEP_CI or IOVF_UOW keyword must be specified.	“SDEP_CI keyword” on page 991
STORCLAS	Optional	Specifies the name of the SMS storage class for the temporary area data set.	“STORCLAS keyword” on page 991
VOLUME	Required	Specifies for non-SMS environments, one or more volume-serial identifiers for the temporary area data set.	“VOLUME keyword” on page 992
VOLUME_EXT	Optional	Specifies one or more volume serial identifiers for the extended original area data set.	“VOLUME_EXT keyword” on page 992

SETREORGPARM subcommand keywords

The SETREORGPARM subcommand controls the reorganization process, which is run by OER.

For the SETREORGPARM subcommand, you can specify the keywords that are listed in the following table. The keywords of the SETREORGPARM subcommand are the same as the keywords for the FPO OER utility. For the format of the SETREORGPARM command, see [“SETREORGPARM subcommand keywords” on page 992](#).

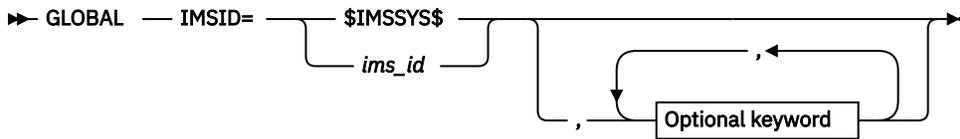
Table 119. SETREORGPARM subcommand keywords

Keyword	Required or optional	Description	Topic
ERRORACT	Optional	Specifies the action that OER takes when one of the following errors occurs: <ul style="list-style-type: none"> An error that occurs while the Area Sensor is running The number of errors that occur during the UOW reorganization exceeds the value that is specified on the MAXFAILS keyword 	“ERRORACT keyword” on page 993
FPBP64U	Optional	Enables the Fast Path 64-bit buffer manager for the IMS High-Speed DEDB Direct Reorganization utility (DBFUHDRO).	“FPBP64U keyword” on page 994
MAXBUFFS	Optional	Specifies the number of buffer sets that are used in the reorganization.	“MAXBUFFS keyword” on page 994
MAXFAILS	Optional	Specifies the maximum allowable number of UOWs that fail to be reorganized, before the utility ends abnormally.	“MAXFAILS keyword” on page 995
RBASEFS	Optional	Specifies a criterion to select UOWs to reorganize.	“RBASEFS keyword” on page 995
RDOVFFS	Optional	Specifies a criterion to select UOWs to reorganize.	“RDOVFFS keyword” on page 996
REORGLIM	Optional	Specifies the reorganization limit for the number of UOWs that can be reorganized in the job.	“REORGLIM keyword” on page 997
REORGMSG	Optional	Specifies whether to generate detailed reorganization messages.	“REORGMSG keyword” on page 997
RFSSCAN	Optional	Specifies whether to gather the statistics before and after reorganization by using Area Sensor.	“RFSSCAN keyword” on page 998
RFSSTAT	Optional	Specifies whether to generate the UOW Space Statistics report before and after reorganization.	“RFSSTAT keyword” on page 998

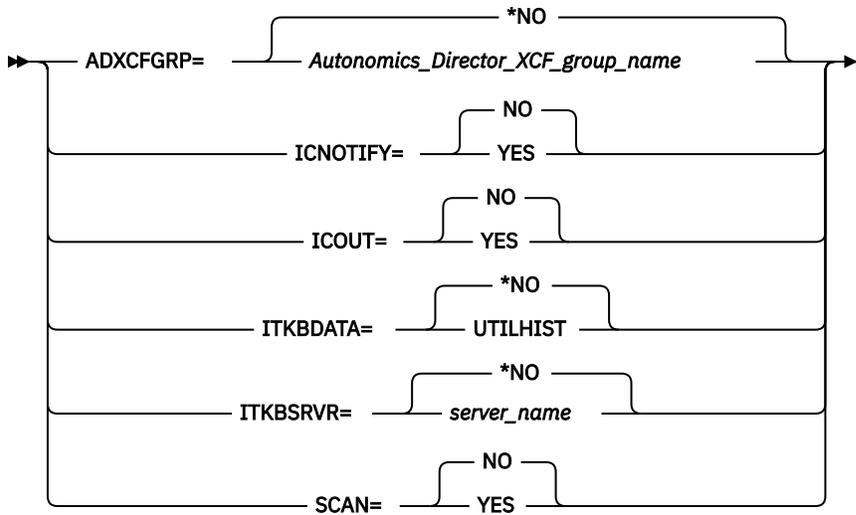
GLOBAL command keywords

Use the GLOBAL command to establish implicit job-step keyword values. GLOBAL is a required command for the HFPSYSIN DD statement. It must be the first command in the input stream.

You can specify the following keywords for the GLOBAL command.



Optional keywords:



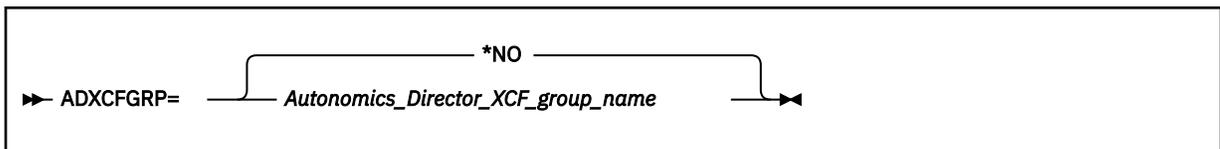
ADXCFCGRP keyword

Use the ADXCFCGRP keyword to specify the Autonomics Director XCF group name. The name is used to send a sensor data notification to Autonomics Director when the sensor data is stored in the Sensor Data repository of IMS Tools KB. It is also used to send a result notification to Autonomics Director before the Online Space Management utility job ends.

ADXCFCGRP is an optional keyword for the GLOBAL command.

The sensor data notification is needed for Autonomics Director to schedule a follow-on action of evaluating sensor data. The result notification is needed for Autonomics Director to determine whether the Online Space Management utility job succeeded or failed.

Format:



Autonomics_Director_XCF_group_name

Send a sensor data notification and a result notification to Autonomics Director of the specified XCF group name.

The utility does not send the result notification to Autonomics Director when one or more of the following conditions are met:

- SCAN=YES is specified on the GLOBAL command and command analysis processing ends with an error.
- The job ends abnormally before it sends the notification.

- The Autonomics Director XCF group name that is specified on the ADXCFGRP keyword of the GLOBAL command is invalid.
- The RECON data sets are not registered to IMS Tools KB.

***NO**

Do not send notifications to Autonomics Director.

Default:

ADXCFGRP=*NO

Site default support:

Not supported.

ICNOTIFY keyword

Use the ICNOTIFY keyword to issue NOTIFY.IC commands to DBRC.

If a pointer error is detected in the area, the NOTIFY IC command is not issued.

If you do not specify ICOUT=YES, this keyword is ignored.

ICNOTIFY is an optional keyword for the GLOBAL command.

Related keywords:

[“ICOUT keyword” on page 981](#)

Format:



YES

Issue the NOTIFY.IC commands if the return code of the preceding process is 4 or lower. If the return code of the preceding process is 8 or higher, the NOTIFY.IC commands are not issued.

NO

Do not issue NOTIFY.IC commands.

Default:

ICNOTIFY=NO

Site default support:

Not supported.

ICOUT keyword

Use the ICOUT keyword to generate one or more image copies.

ICOUT is an optional keyword for the GLOBAL command.

Related keywords:

[“ICNOTIFY keyword” on page 981](#)

Format:



YES

Generate one or more image copies.

NO

Do not generate any image copies.

Default:

ITKBDATA=*NO

Site default support:

Supported.

ITKBSRVR keyword

Use the ITKBSRVR keyword to specify the name of the IMS Tools KB server XCF group.

ITKBSRVR is an optional keyword for the GLOBAL command.

Format:**server_name**

Store reports and sensor data in the repositories of the specified IMS Tools KB server. Also, if the RBASEFS keyword or the RDOVFFS keyword of the SETREORGPARM subcommand is not specified, obtain these values from the specified IMS Tools KB server.

***NO**

Do not store reports and sensor data in IMS Tools KB repositories, also do not read sensor data from the Sensor Data repository of IMS Tools KB.

Default:

ITKBSRVR=*NO

Site default support:

Not supported.

SCAN keyword

Use the SCAN keyword to verify the syntax of command, keywords, and parameters in command sets without running the Online Space Management utility.

The Scan function also allocates some (but not all) internal resources for running the process function. Information concerning the command are routed to the data set that is pointed to by the HFPRPTS DD statement.

SCAN is an optional keyword for the GLOBAL command.

Format:**YES**

Verify the syntax and parameters of the command, but do not run the functions of Online Space Management utility.

NO

Verify the syntax and parameters of the command and then run the Online Space Management utility.

Default:

SCAN=NO

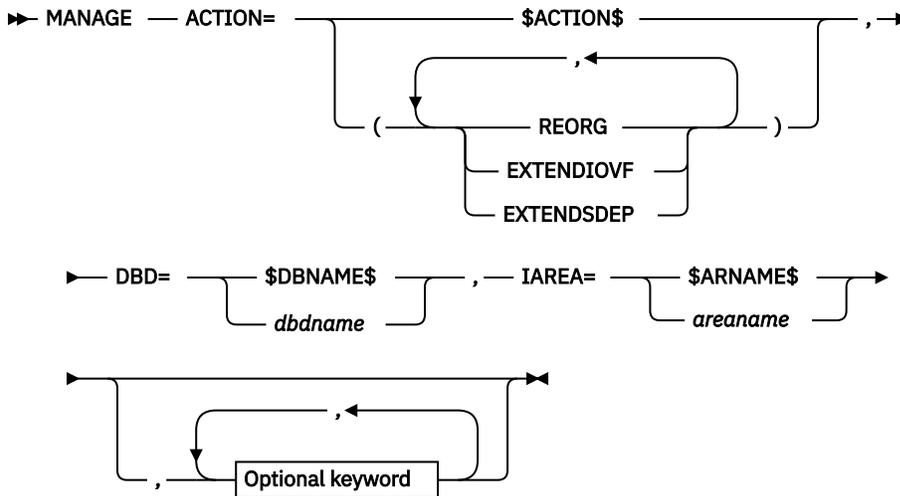
Site default support:

Not supported.

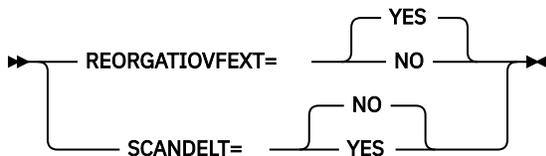
MANAGE command keywords

Use the MANAGE command to manage space utilization of the specified area. MANAGE is a required command for the HFPSYSIN DD statement.

You can specify the following keywords for the MANAGE command.



Optional keywords:

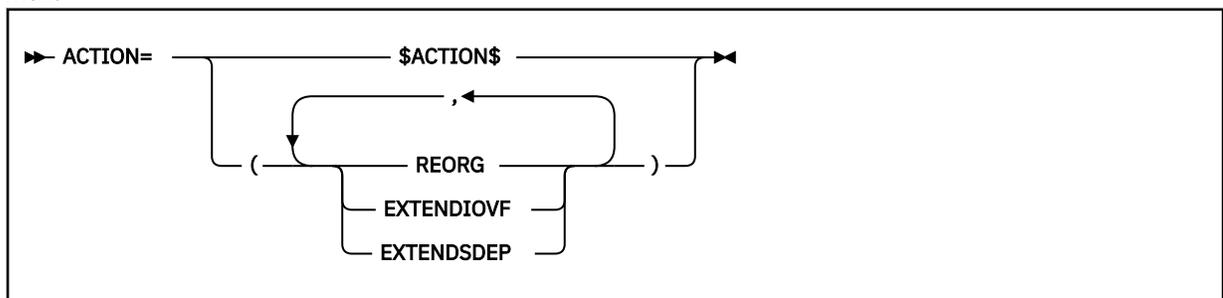


ACTION keyword

Use the ACTION keyword to specify an action or a list of actions to perform.

ACTION is a required keyword for the MANAGE command.

Format:



\$ACTION\$

Specify this value when you use the Online Space Management utility with active autonomies features of Autonomics Director.

When Policy Services detects exceptions that are associated with the action of area reorganization or area extension, Autonomics Director internally replaces the symbolic variable \$ACTION\$ with the associated actions:

- REORG
- EXTENDIOVF
- EXTENDSDEP

REORG

Start the Online Expert Reorganization process to reorganize the area. Specify this parameter only when you run the Online Space Management utility in stand-alone mode.

EXTENDIOVF

Start the Online Area Extender utility process to extend the IOVF portion of the area. Specify this parameter only when you run the Online Space Management utility in stand-alone mode.

EXTENDSDEP

Start the Online Area Extender utility process to extend the SDEP portion of the area. Specify this parameter only when you run the Online Space Management utility in stand-alone mode.

For stand-alone execution of the Online Space Management utility, you can specify multiple parameters separated by a comma. For example, ACTION=(REORG,EXTENDSDEP).

When both area reorganization and area extension are requested, area extension occurs first.

Default:

None.

Site default support:

Not supported.

DBD keyword

Use the DBD keyword to identify the database description (DBD) that contains the area to be processed.

DBD is a required keyword for the MANAGE command.

Related keywords:

[“IAREA keyword” on page 985](#)

Format:



\$DBNAME\$

Specify this value when you use the Online Space Management utility with active autonomics features of Autonomics Director. Autonomics Director internally replaces the symbolic variable \$DBNAME\$ with the name of the DBD to process at run time.

dbdname

Specify a 1- to 8-character database name.

Default:

None.

Site default support:

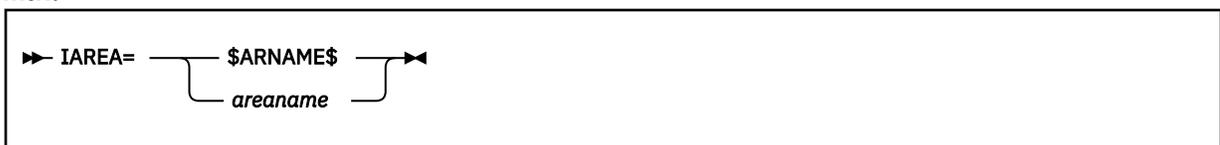
Not supported.

IAREA keyword

Use the IAREA keyword to specify the input area.

IAREA is a required keyword for the MANAGE command.

Format:



\$ARNAME\$

Specify this value when you use the Online Space Management utility with active autonomies features of Autonomics Director. Autonomics Director internally replaces the symbolic variable \$ARNAME\$ with the name of the area to process at run time.

areaname

Specify a 1- to 8-character area name.

Default:

None.

Site default support:

Not supported.

REORGATIOVFEXT keyword

Use the REORGATIOVFEXT keyword to suppress reorganization of the area after the area is extended. This keyword is effective only when both the REORG and EXTENDIOVF parameters are specified for the ACTION keyword.

Generally, this keyword is used in the active autonomies environment. If you specify this keyword and Autonomics Director sets both REORG and EXTENDIOVF parameters for the ACTION keyword, the Online Space Management utility job extends the area but it does not reorganize the area. This keyword reduces the elapsed time that is required for the Online Space Management utility job by suppressing the reorganization process when the IOVF portion is extended.

REORGATIOVFEXT is an optional keyword for the MANAGE command.

Format:



YES

Reorganize the area after IOVF extension.

NO

Do not reorganize the area after IOVF extension.

Default:

REORGATIOVFEXT=YES

Site default support:

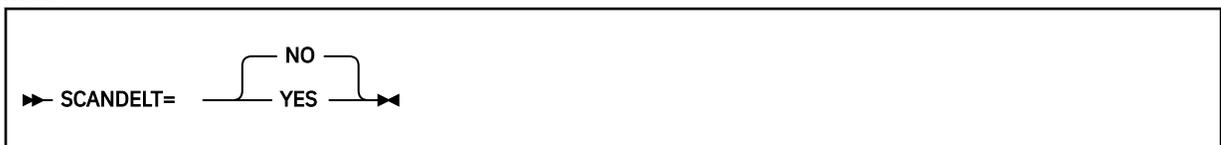
Not supported.

SCANDELT keyword

Use the SCANDELT keyword to process SDEPs by using the SCAN and DELETE utilities before area extension.

SCANDELT is an optional keyword for the MANAGE command.

Format:



YES

If EXTENDIOVF, EXTENDSDEP, or both are specified on the ACTION keyword, process the SDEPs by using the SCAN and DELETE utilities before area extension.

NO

Do not process SDEPs with the SCAN and DELETE utilities.

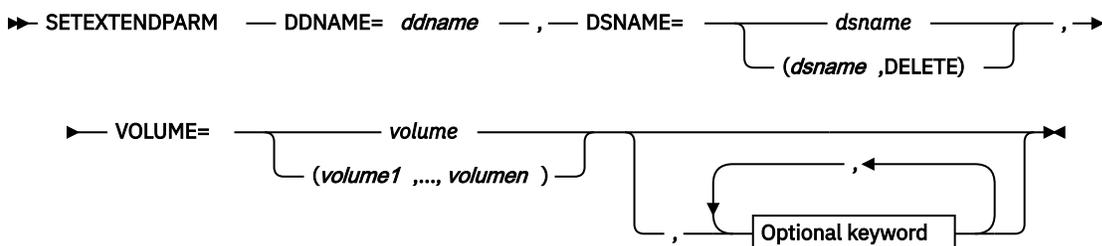
Default:
SCANDELT=NO

Site default support:
Not supported.

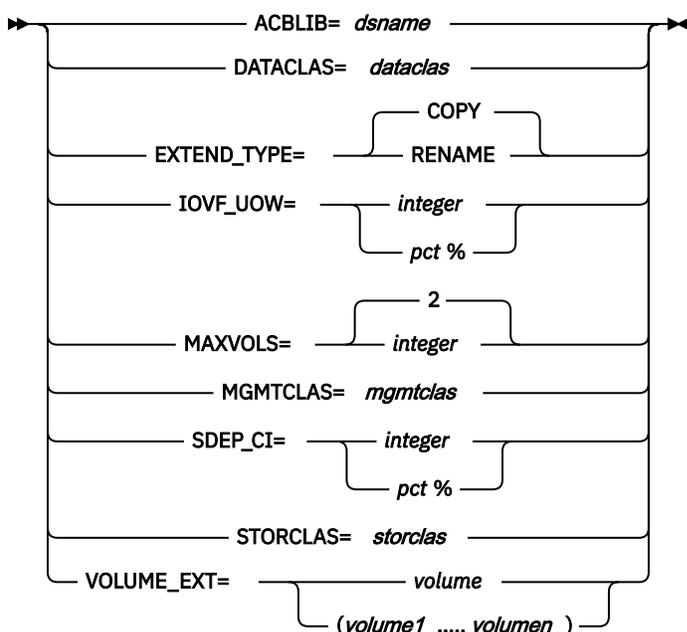
SETEXTENDPARM subcommand keywords

Use the SETEXTENDPARM subcommand to control the area extension process (OAE utility process). SETEXTENDPARM is an optional subcommand for the MANAGE command. If EXTENDIOVF or EXTENDSDEP is specified for the ACTION keyword of the MANAGE command, you must specify the SETEXTENDPARM subcommand.

You can specify the following keywords for the SETEXTENDPARM subcommand.



Optional keywords:



ACBLIB keyword

Use the ACBLIB keyword to specify the ACBLIB to use.

ACBLIB is an optional keyword for the SETEXTENDPARM subcommand, however, this keyword must be specified for IOVF extension.

Format:

```
➤ ACBLIB= dsname ➤
```

dsname

Specify the name of the ACBLIB library.

Default:

None.

Site default support:

Not supported.

DATACLAS keyword

Use the DATACLAS keyword to specify the name of the SMS data class for the temporary area data set.

This parameter is used for the preparation of the temporary ADS.

For more information about SMS data class, see the *MVS JCL Reference*.

DATACLAS is an optional keyword for the SETEXTENDPARM subcommand.

Related keywords:

[“DSNAME keyword” on page 989](#)

Format:

```
▶▶ DATACLAS= dataclas ◀◀
```

dataclas

Specify a 1- to 8-character data class name.

Default:

None.

Site default support:

Not supported.

DDNAME keyword

Use the DDNAME keyword to specify the DDNAME of the temporary area data set where the area will be copied.

You can specify the temporary data set by replacing the keyword value with a unique DD name. The DD name must be unique because the name will be registered to DBRC. For skeleton JCL for use in the active autonomies environment, if you specify DDNAME=\$ARNAME\$T, the name that is used for the temporary data set is the name of the area with a T added to the end. Therefore, when the area name has 8 characters, you must specify a unique DD name on the DDNAME keyword.

This DDNAME must not be registered in DBRC. The total number of preexisting and new ADSs cannot exceed 7 because during utility execution, both the preexisting and the temporary ADS can be opened concurrently. This parameter is used for the preparation of the temporary ADS.

DDNAME is a required keyword for the SETEXTENDPARM subcommand.

Related keywords:

[“DSNAME keyword” on page 989](#)

Format:

```
▶▶ DDNAME= ddname ◀◀
```

ddname

Specify the DD name of the temporary area data set to which the area will be copied.

Default:

None.

Site default support:

Not supported.

DSNAME keyword

Use the DSNAME keyword to specify the data set name of the temporary area data set to which the area will be copied.

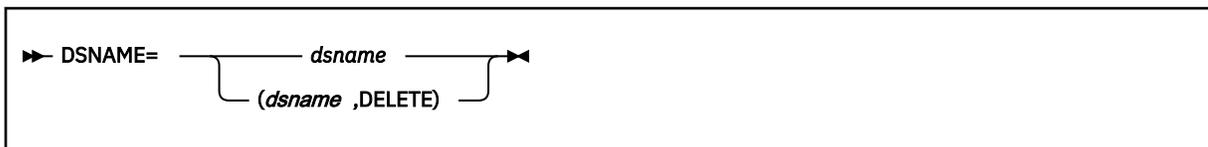
This data set name must not be registered in DBRC. When the DELETE option is specified, the data set is deleted and defined. This parameter is used for the preparation of the temporary ADS.

DSNAME is a required keyword for the SETEXTENDPARM subcommand.

Related keywords:

[“DDNAME keyword” on page 988](#)

Format:



dsname

Specifies the data set name of the temporary area data set to which the area will be copied.

Default:

None.

Site default support:

Not supported.

Note: When you run with the EXTEND_TYPE=RENAME option, the high-level qualifiers of the data set name must be the same as the ones of the original data set name. If the original data set name has no qualifier, the temporary data set must be cataloged to the VSAM catalog file to which the original one is cataloged to.

EXTEND_TYPE keyword

Use the EXTEND_TYPE keyword to specify the extend type.

EXTEND_TYPE is an optional keyword for the SETEXTENDPARM subcommand.

Format:



COPY

Specifies to copy back from the temporary ADS that contains extended image to the original one, which will be once deleted and redefined to extend the space.

RENAME

Specifies to rename the temporary ADS that contains extended image to the original ADS in the subsequent step with the OAE Rename program.

If you specify EXTEND_TYPE=RENAME, you must manually run an additional step, the OAE Rename program, after the Online Space Management utility job. During the rename process, the area is temporarily stopped in all sharing IMS systems.

Default:

EXTEND_TYPE=COPY

Site default support:

Not supported.

IOVF_UOW keyword

Use the IOVF_UOW keyword to specify the number of IOVF UOWs to extend for IOVF extension.

The number of IOVF UOWs can be specified with a specific number or by a percentage of UOWs within the total number of UOWs that are defined in the IOVF portion.

IOVF_UOW is an optional keyword for the SETEXTENDPARM subcommand. However, this keyword is required when the EXTENDIOVF parameter is specified for the ACTION keyword.

Format:



integer

Specify an integer value in the range of 1 - 32766.

pct%

Specify an integer value in the range of 1 - 100, followed by a percent sign (%).

Default:

None.

Site default support:

Not supported.

MAXVOLS keyword

The MAXVOLS keyword specifies the maximum number of volumes for the extended original ADS. This keyword is effective only in SMS environments.

MAXVOLS is an optional keyword for the SETEXTENDPARM subcommand.

Format:



integer

An integer value in the range of 1 - 59.

Default:

2, if the MAXVOLS keyword is not specified.

Site default support:

Not supported.

Restriction:

This keyword is mutually exclusive with the VOLUME_EXT keyword.

MGMTCLAS keyword

Use the MGMTCLAS keyword to specify the name of the SMS management class for the temporary area data set.

This parameter is used for the preparation of the temporary ADS.

For more information about SMS management class, see the *MVS JCL Reference*.

MGMTCLAS is an optional keyword for the SETEXTENDPARM subcommand.

VOLUME keyword

Use the VOLUME keyword to specify one or more volume-serial identifiers for the temporary area data set.

This parameter is used for the preparation of the temporary ADS.

VOLUME is an optional keyword for the SETEXTENDPARM subcommand. However, this keyword is required in non-SMS environments.

Format:



volume

Specifies the volume-serial identifier for the temporary area data set.

(volume1,...,volumen)

Specifies multiple volume-serial identifiers for the temporary area data set.

Default:

None.

Site default support:

Not supported.

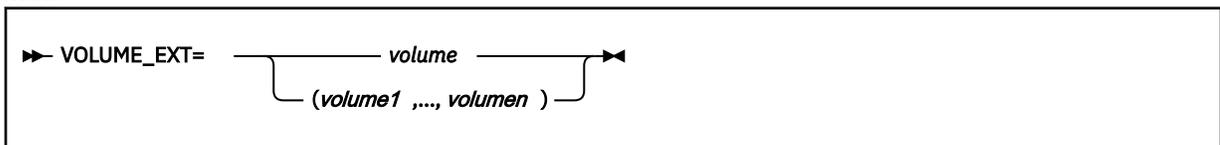
VOLUME_EXT keyword

The VOLUME_EXT keyword specifies one or more volume serial identifiers for the extended original ADS.

This parameter is used for the preparation of the extended original ADS.

VOLUME_EXT is an optional keyword for the SETEXTENDPARM subcommand.

Format:



volume

Specifies the volume serial identifier for the extended original ADS.

(volume1,...,volumen)

Specifies multiple volume serial identifiers for the extended original ADS.

Default:

The volume serial identifiers of the original ADS before expansion.

Site default support:

Not supported.

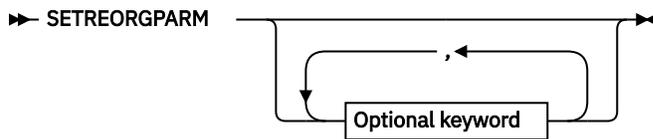
Restrictions:

- This keyword cannot be specified when the original ADS is a MADS.
- This keyword is mutually exclusive with the MAXVOLS keyword.

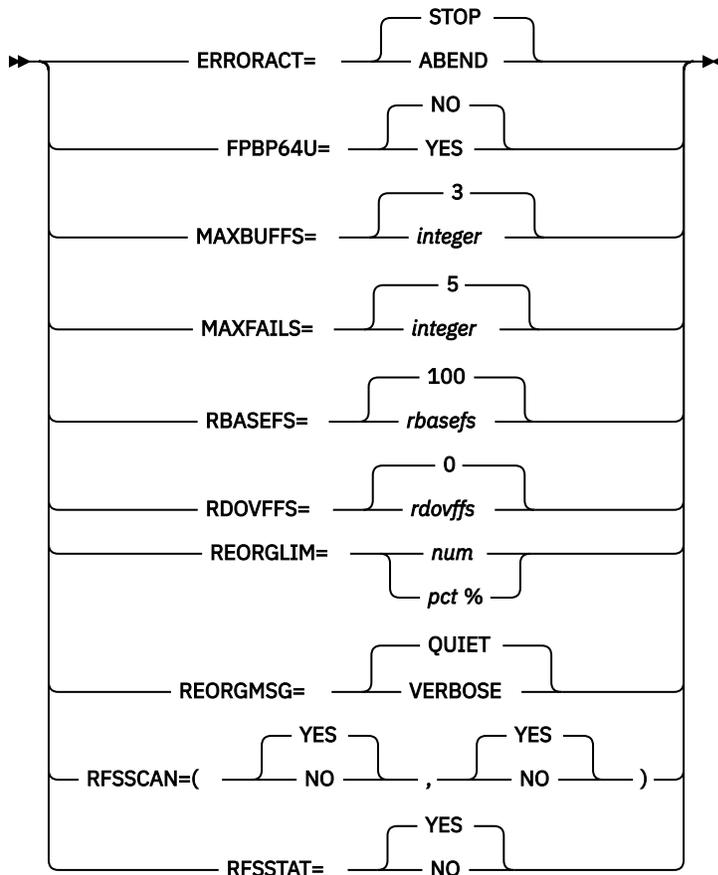
SETREORGPARM subcommand keywords

Use the SETREORGPARM subcommand to control the reorganization process (OER utility process). SETREORGPARM is an optional subcommand for the MANAGE command.

You can specify the following keywords for the SETREORGPARM subcommand.



Optional keywords:



ERRORACT keyword

Use the ERRORACT keyword to specify the action for the OER utility to take at certain error conditions. ERRORACT is an optional keyword for the SETREORGPARM subcommand.

Format:



ABEND

When one of the following conditions is met, processing ends with the following user abend code:

U3500

Error-1: An error that occurs while the Area Sensor is running

U3523

Error-2: The number of errors that occur during the UOW reorganization exceeds the value that is specified on the MAXFAILS keyword.

If RFSSCAN=(NO,YES) or RFSSCAN=(NO,NO) is specified, ERRORACT=ABEND is not effective for the Error-2 error case. In this case, the processing always stops with a return code of 8 regardless of the ERRORACT parameter.

STOP

Specifies that processing ends without a dump, but with a return code of 8.

Default:

ERRORACT=STOP

Site default support:

Supported.

FPBP64U keyword

Use the FPBP64U keyword to enable the Fast Path 64-bit buffer manager for the IMS High-Speed DEDB Direct Reorganization utility (DBFUHDR0).

FPBP64U is an optional keyword for the SETREORGPARM subcommand.

Format:**NO**

Do not enable the Fast Path 64-bit buffer manager for the IMS High-Speed DEDB Direct Reorganization utility.

YES

Enable the Fast Path 64-bit buffer manager for the IMS High-Speed DEDB Direct Reorganization utility.

Default:

FPBP64U=NO

Site default support:

Supported.

Restriction:

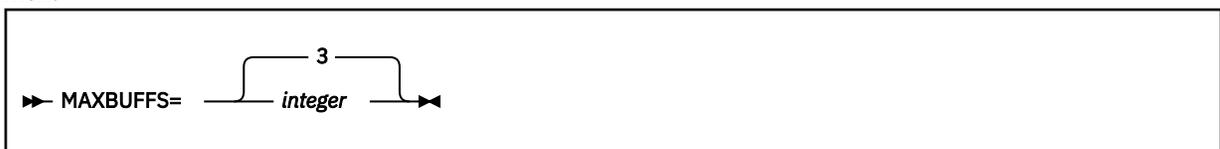
FPBP64U=YES is applicable only for IMS 14 and later.

MAXBUFFS keyword

Use the MAXBUFFS keyword to specify the number of buffer sets that are used in the reorganization.

Tip: The meaning of MAXBUFFS is the same as that of the BUFNO command for the High-Speed DEDB Direct Reorganization Utility (DBFUHDR0).

MAXBUFFS is an optional keyword for the SETREORGPARM subcommand.

Format:**integer**

Specify an integer value in the range of 3 - 999.

Default:

MAXBUFFS=3

Site default support:

Supported.

MAXFAILS keyword

Use the MAXBUFFS keyword to specify the maximum allowable number of UOWs that fail to be reorganized, before the utility is ends abnormally.

If the number of failures exceeds the number that is specified on the MAXFAILS keyword, the reorganization processing is stopped and the OER utility stops its processing or abends depending on the value that is specified for the ERRORACT keyword.

MAXFAILS is an optional keyword for the SETREORGPARM subcommand.

Format:



integer

Specify an integer value in the range of 0 - 999.

If you specify MAXFAILS=0, then no limit is set and the reorganization continues the attempt to reorganize the next UOW until all UOWs in the area are processed or the reorganization limit, if it is specified, is reached.

The failed UOWs are reported in the SYSPRINT or the FPXMFILE stream. The failed UOWs are also marked in the UOW Space Statistics report.

Default:

MAXFAILS=5

Site default support:

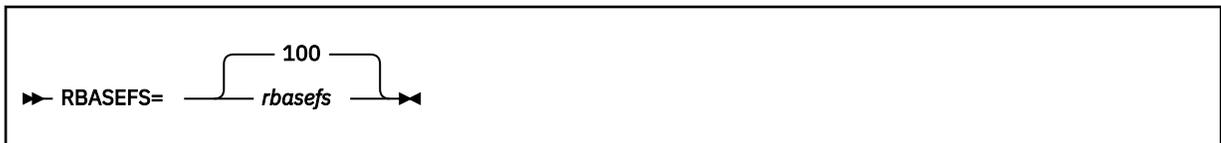
Supported.

RBASEFS keyword

Use the RBASEFS keyword to specify a criterion to select UOWs to reorganize.

RBASEFS is an optional keyword for the SETREORGPARM subcommand.

Format:



rbasefs

Specifies the percentage of free space in each RAP CI. If a RAP CI contains free space that is greater than the specified percentage, and the RAP CI also uses overflow CIs, the UOW is selected as a candidate for the reorganization.

Specify an integer value 0 - 100, both inclusive. The maximum value 100 has a special meaning; if RBASEFS=100 is specified, the RBASEFS condition is not used to select an initial set of UOWs.

For details of how the number of UOWs to be reorganized is determined, see [“The number of UOWs to be reorganized”](#) on page 781.

Default:

100, if the RBASEFS keyword is not specified.

Site default support:

Not supported.

Important: If the name of the IMS Tools KB server XCF group is specified on the ITKBSRVR keyword, and if the RBASEFS keyword is not specified, the utility reads the value of data element

DB_THRESHOLD_RBASEFS from the Sensor Data repository of the specified IMS Tools KB server and uses it as the value of the RBASEFS keyword.

To store this value in the IMS Tools KB repository with DB Sensor, add the following keyword in the JCL for use with DB Sensor:

- For the OPC utility with DB Sensor, specify the EXC_RBASEFS keyword in the FPXIN DD. For more information, see [“EXC_RBASEFS keyword” on page 872](#).
- For stand-alone DB Sensor, specify the RBASEFS keyword in the HFPAPARM DD. For more information, see the *IMS Solution Packs Data Sensor User's Guide*.
- For FPA Analyze with DB Sensor, specify the RBASEFS keyword in the HFPAPARM DD. For more information, see [“DD statements for the Analyze function” on page 203](#).

RDOVFFS keyword

Use the RDOVFFS keyword to specify a criterion to select UOWs to reorganize.

RDOVFFS is an optional keyword for the SETREORGPARM subcommand.

Format:



The variable field is defined as follows:

rdovffs

Specifies the percentage of free space in the dependent overflow area in each UOW. If the free space percentage in the dependent overflow area in a UOW is greater than the specified percentage, and the UOW also uses IOVF CIs, the UOW is selected as a candidate for the reorganization.

Specify an integer value in the range of 0 - 100. The maximum value 100 has a special meaning; if RDOVFFS=100 is specified, the RDOVFFS condition is not used to select an initial set of UOWs.

For details of how the number of UOWs to be reorganized is determined, see [“The number of UOWs to be reorganized” on page 781](#).

Default:

0, if the RDOVFFS keyword is not specified.

Site default support:

Not supported.

Important: If the name of the IMS Tools KB server XCF group is specified on the ITKBSRVR keyword, and if the RDOVFFS keyword is not specified, the utility reads the value of data element DB_THRESHOLD_RDOVFFS from the Sensor Data repository of the specified IMS Tools KB server and uses it as the value of the RDOVFFS keyword.

To store this value in the IMS Tools KB repository with DB Sensor, add the following keyword in the JCL for use with DB Sensor:

- For the OPC utility with DB Sensor, specify the EXC_RDOVFFS keyword in the FPXIN DD. For more information, see [“EXC_RDOVFFS keyword” on page 873](#).
- For stand-alone DB Sensor, specify the RDOVFFS keyword in the HFPAPARM DD. For more information, see the *IMS Solution Packs Data Sensor User's Guide*.
- For FPA Analyze with DB Sensor, specify the RDOVFFS keyword in the HFPAPARM DD. For more information, see [“DD statements for the Analyze function” on page 203](#).

REORGLIM keyword

Use the REORGLIM keyword to specify the reorganization limit for the number of UOWs that can be reorganized in the job. The limit can be specified in the form of a maximum number of UOWs or a maximum percentage of the number of UOWs within the total number of UOWs defined in the area.

REORGLIM is an optional keyword for the SETREORGPARM subcommand.

Format:



num

Specify a non-negative integer value less than 32767.

The value specifies the maximum number of UOWs that can be reorganized. If the number exceeds the total number of UOWs that is defined in the area, the total number of UOWs is used as the reorganization limit.

pct%

Specify a non-negative integer value less than or equal to 100, followed by a percent sign (%).

The value specifies the maximum percentage of the number of UOWs within the total number of UOWs that is defined in the area. The number must be less than or equal to 100.

Tip: If you want to obtain the UOW Space Statistics report without reorganizing UOWs, specify REORGLIM=0.

Default:

If the REORGLIM keyword is not specified, the number of UOWs defined in the area.

Site default support:

Supported.

Restriction:

If RFSSCAN=(NO,NO) or RFSSCAN=(NO,YES) is specified, the REORGLIM keyword cannot be specified.

REORGMMSG keyword

Use the REORGMMSG keyword to generate detailed reorganization messages.

REORGMMSG is an optional keyword for the SETREORGPARM subcommand.

Format:



QUIET

Suppress detailed reorganization messages.

VERBOSE

Generate reorganization messages.

Default:

REORGMMSG=QUIET

Site default support:

Supported.

RFSSCAN keyword

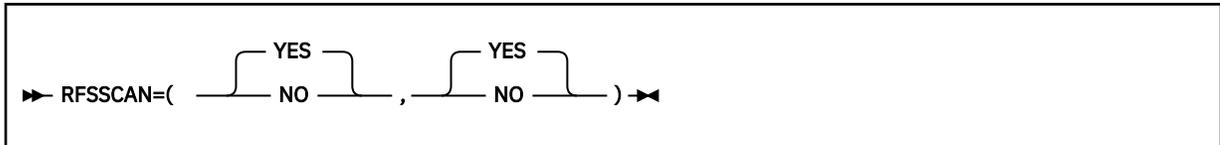
Use the RFSSCAN keyword to gather the statistics before and after reorganization by using Area Sensor.

The RFSSCAN keyword takes two parameters that are enclosed by a pair of parenthesis and separated by a comma; the first parameter specifies whether the statistics are gathered to evaluate UOW and the second parameter specifies whether the statistics are gathered for UOWs that have been reorganized.

The UOW Evaluation and UOW Ranking functions of the OER utility process are enabled only when the first parameter is YES. If the first parameter is NO, these functions are not enabled and all UOWs become candidates for the reorganization and the reorganization is attempted from the first UOW through the last UOW in the ascending order of the UOW number.

RFSSCAN is an optional keyword for the SETREORGPARM subcommand.

Format:



YES

Gather the statistics.

NO

Do not gather the statistics.

Default:

RFSSCAN=(YES,YES)

Site default support:

Supported.

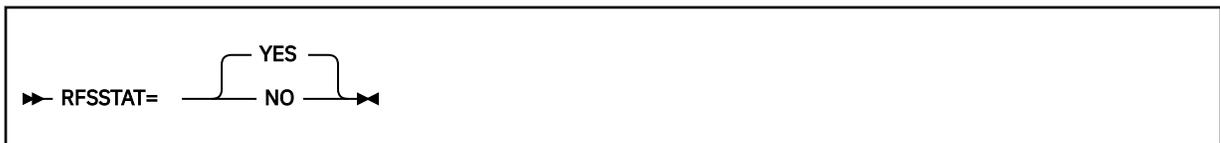
Tip: If you want to reorganize all UOWs and you do not want to generate the UOW Space Statistics report, specify RFSSCAN=(NO,NO)

RFSSTAT keyword

Use the RFSSTAT keyword to generate the UOW Space Statistics report before and after reorganization.

RFSSTAT is an optional keyword for the SETREORGPARM subcommand.

Format:



YES

Generate the UOW Space Statistics report before and after reorganization.

NO

Do not generate the UOW Space Statistics report before and after reorganization.

For RFSSTAT=YES to take effect, at least one of the first and second parameters of the RFSSCAN keyword must be YES. If both the first and second parameters of RFSSCAN are NO, RFSSTAT=YES is ignored.

If RFSSTAT=YES is effective and both the first and second parameters of the RFSSCAN keyword are YES, the statistics before and after the reorganization are reported. If the first or second parameter of RFSSCAN is not YES, the statistics are reported for the phase for which YES is specified.

Exception: When REORGLIM=0 or REORGLIM=0% is specified and RFSSCAN=(YES,YES) or RFSSCAN=(YES,NO) is specified, the space statistics is printed for those UOWs that satisfy either the RBASEFS or the RDOVFFS condition.

Default:

RFSSTAT=YES

Site default support:

Supported.

Chapter 35. Online Space Management utility reports

The following table summarizes the messages and reports that are generated by the Online Space Management utility. The table also shows whether each report is stored in the IMS Tools KB Output repository.

Table 120. Messages and reports stored in IMS Tools KB

Generated messages and reports	Whether the report can be stored in the IMS Tools KB Output repository
OAE processing messages	Yes
OER processing messages	Yes
OPC processing messages	Yes
Audit report	-
Processing report	-
Online Pointer Checker Statistics report	Yes
Online Pointer Scan report	-
Processing Summary of Analysis report	Yes
DEDB Area Analysis reports	Yes
UOW report	Yes
UOW Exception report	Yes
Process Summary report (OPC)	Yes
Image Copy NOTIFY report	Yes
Resolve Pointers report	-
Reorganization Processing Summary report	Yes
UOW Space Statistics report	Yes
Process Summary report (OER)	Yes
Sensor Data Statistics report	Yes

Topics:

- [“Reports in HFPRPTS DD” on page 1001](#)
- [“Reports in FPXRPTS DD” on page 1004](#)
- [“Reports in FPXSPRT DD” on page 1004](#)

Reports in HFPRPTS DD

The HFPRPTS DD data set contains the reports that are generated by the Online Space Management utility and that show processing options applied to the job.

The following reports are generated in this data set:

- [“Audit report” on page 1002](#)
- [“Processing report” on page 1002](#)

Audit report

The Audit report provides information about the HFPSYSIN statements.

The following figure is an example of the Audit report.

```
IMS HPFP UTILITIES                                "Audit report"                                PAGE: 1
5698-FPP V2R1                                     2019-03-01 20:21:17

HFPSYSIN
0.....1.....2.....3.....4.....5.....6.....7.....8
1234567890123456789012345678901234567890123456789012345678901234567890

- 1:  GLOBAL
- 2:  ADXCFGRP=ADSRV01,
- 3:  ITKBSVR=FPQSRV01,
- 4:  IMSID=IMS1,
- 5:  DBRC=YES
- 6:  MANAGE
- 7:  IAREA=DB22AR0,
- 8:  DBD=DEDBJN22,
- 9:  ACTION=(REORG,EXTENDIOVF,EXTENDSDEP)
- 10: SETREORGPARM
- 11: *
- 12: SETEXTENDPARM
- 13: SDEP_CI=10%,
- 14: IOVF_UOW=10%,
- 15: ACBLIB=IMSVS.ACBLIB,
- 16: DDNAME=DB22AR0T,
- 17: DSNAMES=(TEMP.OAE.DB22AR0.ADS,DELETE),
- 18: VOLUME=TEMPADS

PROCESSING INFORMATION

EXEC
- DBRCGRP           :
- IMSID             :
- IMSPLEX           :

GLOBAL
- ADXCFGRP          : ADSRV01
- CHKP              : 1000
- DBRC              : (YES)
- IMSID             : IMS1
- ITKBLOAD          : *NO
- ITKBSVR           : FPQSRV01
- LIUGOPT           : SYS1
- MSTIMESTAMP       : NO
- RESTART           : NO
- RESTART_COUNT     : 5
- SCAN              : NO
- TOINUM            : 10
- TOITIME           : 30
- TOIXCFGRP         : *NO
```

Figure 380. Audit report

HFPSYSIN

Echo of the control statements that you specified in the HFPSYSIN data set.

PROCESSING INFORMATION

Parameters on the EXEC statement and the GLOBAL command in the HFPSYSIN data set.

Processing report

The Processing report contains the keyword parameters that were applied to the Online Space Management utility process.

The following figure shows an example of the Processing report.

MANAGE STARTED DATE: 2019-03-01 TIME: 20:21:18 ENDED DATE: 2019-03-01 TIME: 20:21:33

MANAGE PROCESSING OPTION
 - ACTION : (REORG,EXTENDIOVF,EXTENDSDEP)
 - DBD : DEDBJN22
 - IAREA : (DB22AR0)
 - REORGATIOVFEXT : YES

MANAGE SUBCOMMAND PROCESSING OPTION
 SETEXTENDPARM
 - ACBLIB : IMSVS.ACBLIB
 - DATACLASS :
 - DDNAME : DB22AR0T
 - DSNNAME : (TEMP.OAE.DB22AR0.ADS,DELETE)
 - EXTEND_TYPE :
 - IOVF_UOW : 10%
 - MGMTCLAS :
 - SCANDELT :
 - SDEP_CI : 10%
 - STORCLAS :
 - VOLUME : (TEMPADS)

MANAGE SUBCOMMAND PROCESSING OPTION
 SETREORGPARM
 - ERRORACT :
 - MAXBUFFS :
 - MAXFAILS :
 - RBASEFS :
 - RDOVFFS :
 - REORGLIM :
 - REORGMSG :
 - RFSSCAN :
 - RFSSTAT :

PROCESSING INFORMATION
 - THE FOLLOWING FPO CONTROL STATEMENTS WERE GENERATED.

- EXTEND CONTROL STATEMENT
 GLOBAL
 DBRC=YES
 ITKBSRVR=FPQSRV01
 EXTEND
 AREA=DB22AR0
 DDNAME=DB22AR0T
 DSNNAME=(TEMP.OAE.DB22AR0.ADS,DELETE)
 IOVF_UOW=10%
 SDEP_CI=10%
 VOLUME=(TEMPADS)
 PCHKONLY=NO
 - REORG CONTROL STATEMENT
 GLOBAL

DBRC=YES
 ITKBSRVR=FPQARV01
 REORG
 AREA=DB22AR0
 MFILE=FPXMFIL
 RBASEFS=13
 RDOVFFS=13
 RFILE=FPXRPTS
 PTRCHK=NO
 PCHKONLY=NO
 - ANALYZE CONTROL STATEMENT
 GLOBAL
 DBRC=YES
 ADXCFGRP=ADSRV01
 ITKBSRVR=FPQSRV01
 ANALYZE
 AREA=DB22AR0
 FULLSTEP=YES
 MFILE=FPXMFIL
 EXC_RBASEFS=20
 EXC_RDOVFFS=30
 RFILE=FPXRPTS
 SENSOR=YES
 SENSOR_DBREC=YES
 SDEP=NO
 IC=NO
 PCHKONLY=NO
 - DBRC KEYWORD OF GLOBAL COMMAND IN HFPSYSIN DD IS IGNORED.

Figure 381. Processing report

command PROCESSING OPTION

subcommand SUBCOMMAND PROCESSING OPTION

Keyword parameters that are specified for the indicated command or subcommand in the HFPSYSIN data set.

PROCESSING INFORMATION

FPO control statements that are generated internally based on the values specified in the HFPSYSIN data set.

EXTEND CONTROL STATEMENT

Control statement values that were applied to the OAE utility process.

REORG CONTROL STATEMENT

Control statement values that were applied to the OER utility process.

ANALYZE CONTROL STATEMENT

Control statement values that were applied to the OPC utility process.

This field also shows the keywords that were ignored.

Reports in FPXRPTS DD

The FPXRPTS DD data set contains the reports that were generated by the OPC and OER utility processes.

The following reports are generated by the OPC utility process:

- Online Pointer Checker Statistics report
- Online Pointer Scan report, when pointer errors are detected
- Image Copy NOTIFY report, when ICOUT=YES is specified
- Resolve Pointers report
- Processing Summary of Analysis report
- DEDB Area Analysis reports
- UOW report
- UOW Exception report
- Process Summary report

The format and the content of these reports are the same as the reports that are generated by a stand-alone OPC utility job. For more information about these reports, see [“FPXRPTS DD data set” on page 702](#).

When an area is reorganized, the following reports are generated by the OER utility process:

- Reorganization Processing Summary report
- UOW Space Statistics report, when RFSSTAT=YES is specified
- Process Summary report

The format and the content of these reports are the same as the reports that are generated by a stand-alone OER utility job. For more information about these reports, see [“FPXRPTS DD data set” on page 803](#).

Reports in FPXSPRT DD

The FPXSPRT DD data set contains Sensor Data Statistics report.

The Sensor Data Statistics report contains the names and the values of the data elements that are stored in the Sensor Data repository of IMS Tools KB.

The format and the content of the report are the same as the report that is generated by a stand-alone OPC utility job. For more information about this report, see [“FPXSPRT DD data set” on page 710](#).

For a list of the data elements that are collected by IMS HP Fast Path Utilities, see the topics about data elements in the *Tools Base Policy Services User's Guide*.

Part 5. Online Structure Change utility

The following topics introduce the Online Structure Change (OSC) utility of IMS HP Fast Path Utilities and explain how to use the utility.

Note: To use the Online Structure Change utility, IMS 15 is required. This utility cannot be used with IMS 14.

Topics:

- [Chapter 36, “Online Structure Change utility overview,” on page 1007](#)
- [Chapter 37, “Restructuring an online DEDB,” on page 1019](#)
- [Chapter 38, “JCL requirements for the Online Structure Change utility,” on page 1027](#)
- [Chapter 39, “Utility control statements for the Online Structure Change utility,” on page 1031](#)
- [Chapter 40, “Online Structure Change utility reports,” on page 1075](#)
- [Chapter 41, “Examples for the Online Structure Change utility,” on page 1085](#)

Chapter 36. Online Structure Change utility overview

The Online Structure Change utility facilitates restructuring tasks for online IMS DEDBs. The utility provides advanced functions that help IMS database administrators change DBD descriptions, randomizing module, and segment edit/compression routine without taking DEDB areas offline.

Topics:

- [“Online Structure Change utility benefits” on page 1007](#)
- [“Online Structure Change utility functions” on page 1007](#)
- [“Online Structure Change utility internal jobs” on page 1010](#)
- [“Online Structure Change utility restrictions” on page 1013](#)
- [“Online Structure Change utility considerations” on page 1014](#)

Online Structure Change utility benefits

The Online Structure Change utility provides advanced functions that help IMS database administrators change the structure of DEDB areas without taking DEDB areas offline.

With the online restructuring capability of the Online Structure Change utility, you can change any or all of the following database specifications while keeping the database available:

- Number of database areas.
 - New areas can be added but they must be added to the end of the DEDB. Existing areas can be deleted but they must be deleted from the end of the DEDB.
- Randomizing module, including the following changes:
 - RMNAME parameter. For example, the name of the randomizing module.
 - Logic of the randomizing module.
- Segment edit/compression routines. The COMPRTN parameter can be changed. For example, the name of the segment edit/compression routine.
- UOW parameter values.
- ROOT parameter values.
- CI size.
- Size of the VSAM data set.
- Pointer options.
- Add new segments (New segments must be added to the lowest hierarchical level.)
- Change existing segment hierarchical structure within the same parent.
- Increase the maximum or the minimum length of the segment.

Online Structure Change utility functions

The Online Structure Change utility (OSC) facilitates restructuring tasks for online IMS DEDB. The utility provides advanced functions that help IMS database administrators change DBD descriptions, randomizing module or segment edit/compression routine of a DEDB while the DEDB is online and being read and updated.

The Online Structure Change utility can concurrently restructure all areas of a DEDB, it can also restructure specific areas without impacting other areas of the DEDB.

Subsections:

- [“OSC process phases” on page 1008](#)

- “Data flow” on page 1009
- “OSC jobs and operating modes” on page 1009

OSC process phases

As shown in the following figure, the restructuring process of the Online Structure Change utility is composed of seven phases. To reduce the unavailability time for DEDB or specific areas, the Online Structure Change utility copies original area data sets (ADSs) to shadow ADSs. Therefore, except for the outage while replacing the original ADSs with the restructured ADSs, areas are kept online and updatable during the OSC restructuring process.

OSC process phase

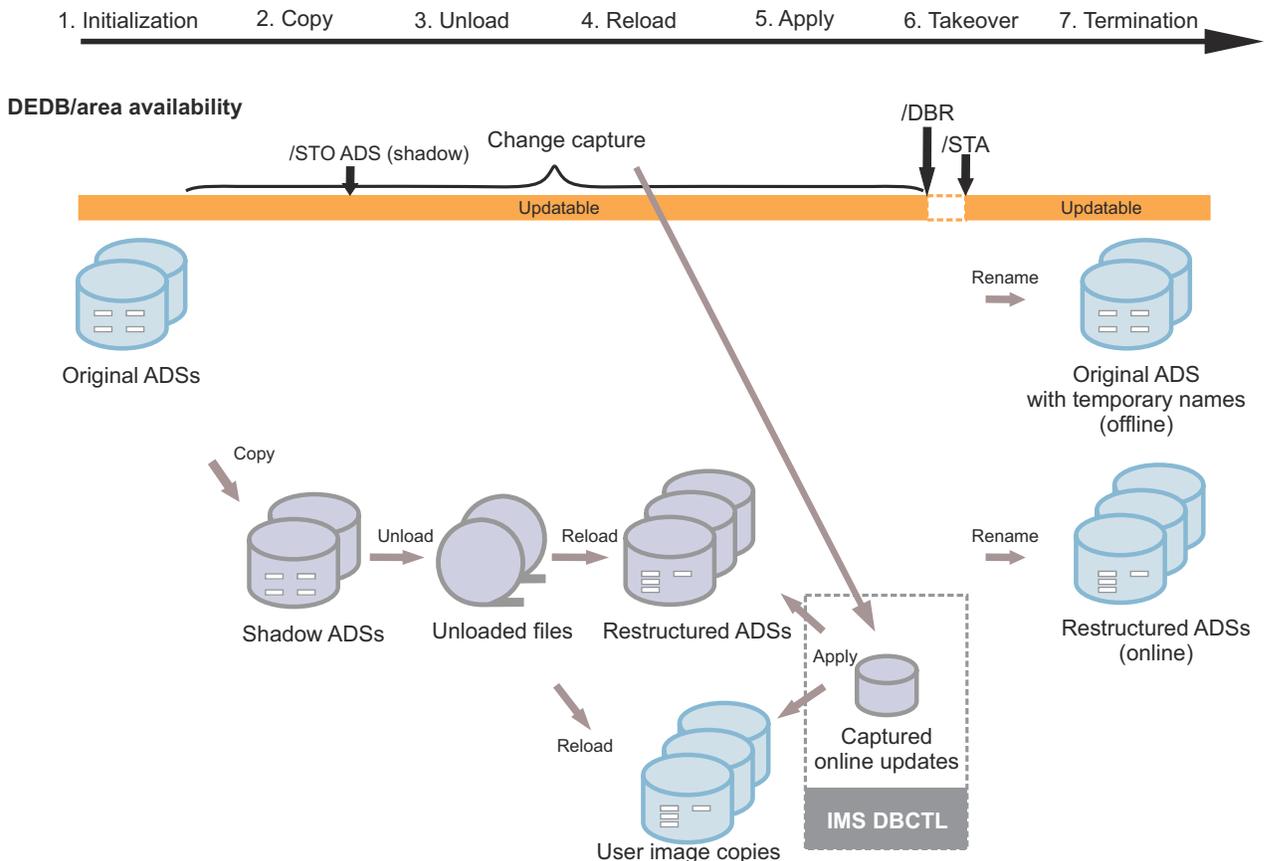


Figure 382. Online Structure Change utility process flow

1. Initialization phase

Performs initialization processing. Generates all the OSC internal jobs and stores them in a data set.

2. Copy phase

Invokes the change capture to capture any changes that are made to the DEDB during the OSC restructuring process, copies data from original ADSs to shadow ADSs, and stops shadow ADSs.

3. Unload phase

Unloads data from the shadow ADSs and generates unloaded files.

4. Reload phase

Deletes shadow ADSs, defines new ADSs (restructured ADSs) with new database specifications, and loads data into the new ADSs. Also loads data into user image copy (UIC) data sets and deletes the unloaded files.

5. Apply phase

Applies captured updates to the restructured ADSs and UICs.

6. Takeover phase

Issues the /DBR command to stop the DEDB and areas, performs online change, renames the restructured ADSs to original names, and issues the /STA command to start the DEDB and areas.

7. Termination phase

Performs termination processing. Collects all the internal job logs and stores them in a data set.

Data flow

The following figure shows the data flow of a typical Online Structure Change utility job. The utility dynamically allocates shadow ADSs for the copy process and unloaded files for the unload process, then deletes the shadow ADSs and defines restructured ADSs as well as user image copies with the new ACB for the reload process. The new ACB is the staging ACB that contains new database specifications for the restructured DEDB.

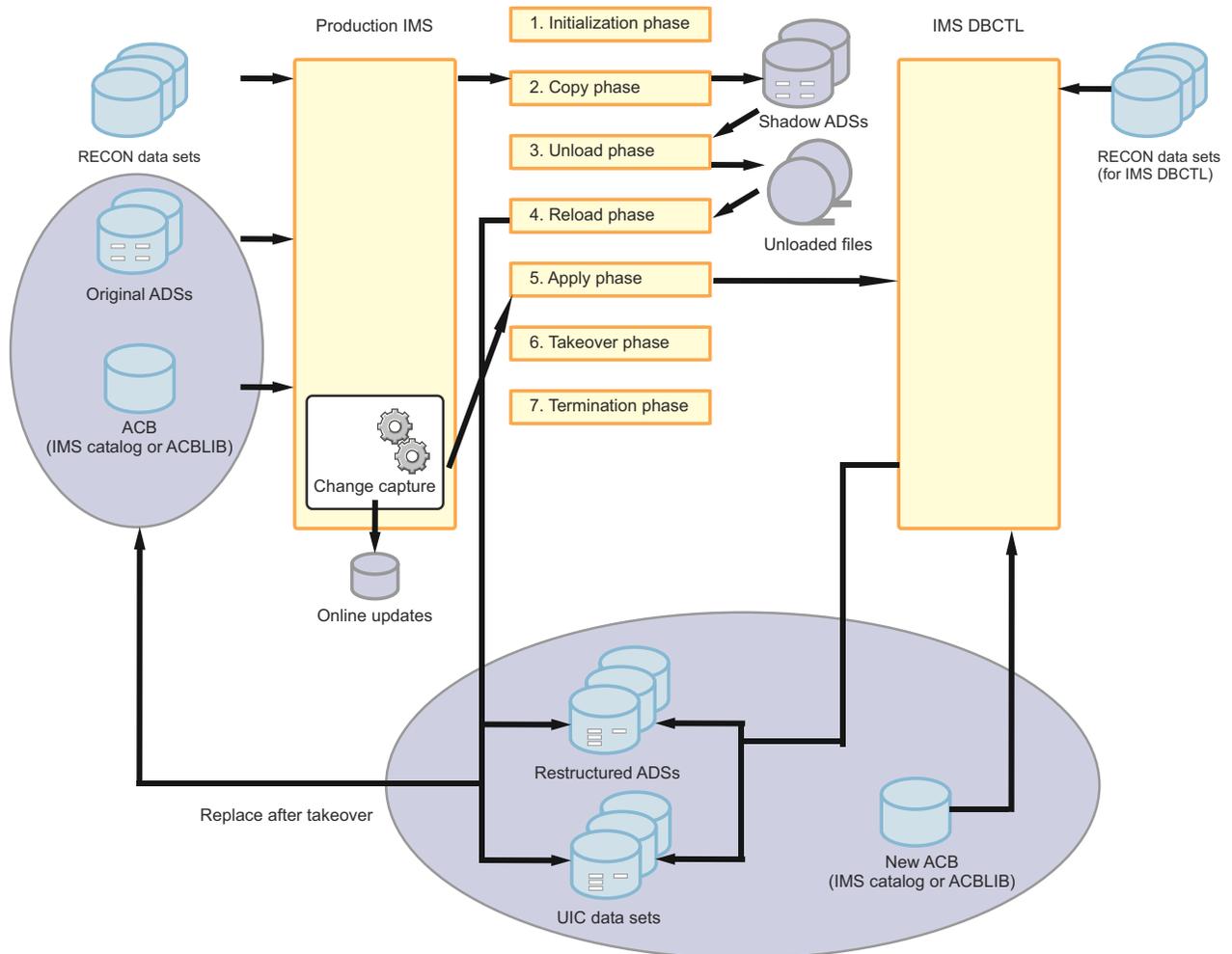


Figure 383. Online Structure Change utility data flow

OSC jobs and operating modes

The Online Structure Change utility has two types of jobs.

OSC driver job:

- An OSC driver job is coded and submitted by the user. The user must code the JCL statements and submit the job.
- One OSC driver job is required for each OSC execution.

- Supports the site default option. The user can define default values for utility control statement keywords in the site default module.

OSC internal job:

- OSC internal jobs are generated and submitted by OSC driver jobs.
- The user can specify the first six characters of OSC internal job names.
- Some of these jobs can be executed on a LPAR that is different from the one where the OSC driver job is executed.
- Options specified in OSC internal jobs are inherited from the OSC driver job.
- Multiple OSC internal jobs run with one OSC driver job execution.

The Online Structure Change utility supports the following operating modes. Operating mode is selected based on the RUNTYPE keyword value of the OSC driver job.

FULL mode (RUNTYPE=FULL)

Generates OSC internal jobs and submits them.

GENJCL mode (RUNTYPE=GENJCL)

Generates OSC internal jobs but does not submit them.

RUNJCL mode (RUNTYPE=RUNJCL)

Does not generate OSC internal jobs but submits the OSC internal jobs that were generated previously by the same OSC driver job that was executed in GENJCL mode.

You can generate OSC internal jobs in advance (GENJCL mode) and then execute the OSC internal jobs later (RUNJCL mode). In this case, only the RUNTYPE keyword parameter can be changed in the OSC driver job. Other portions of the JCL statements and control statements must remain the same.

Online Structure Change utility internal jobs

The Online Structure Change utility generates internal jobs (referred to as OSC internal jobs), submits them to restructure the DEDB or areas, and collects all OSC internal job logs in a data set.

OSC internal jobs

OSC internal jobs are generated in the data set that the JCLDSN keyword specifies. Job logs from OSC internal jobs are generated in the data set that the LOGDSN keyword specifies.

The following table summarizes OSC internal jobs. Each OSC internal job has a unique job ID so that the jobs can be easily identified. For the prefix portion of the OSC internal job names, the first six characters of the OSC driver job name are assigned by default. You can change the first six characters by using the JOBN_PREFIX keyword.

If you generate OSC internal jobs in GENJCL mode, you can review and customize the generated jobs before they are submitted in RUNJCL mode. When you customize OSC internal jobs, make sure you do not delete the START job step and the END job step in each OSC internal job. If deleted, the OSC driver job cannot determine if the OSC internal job ended successfully.

Table 121. OSC internal jobs

Phase	Job name	Job ID	Description
Initialization	xxxxxxE0	CHKRTN	Issues the CHKRAND or CHKCOMP command by using IMS Tools Online System Interface (TOSI) to check if the randomizer or the edit/compression routines are being used by other databases. This job is generated and submitted only when replacement of the randomizer or edit/compression routines is requested. That is, when keyword values differ between IMSRESLB_RND_DSN and NEWRESLB_RND_DSN keywords or between IMSRESLB_EDC_DSN and NEWRESLB_EDC_DSN keywords.
	xxxxxA0	INVAPPLY	Invokes the change apply job.
	xxxxxF0	DEFSADS	Defines shadow area data sets (ADSs) by allocating shadow ADSs and registering them to the production RECON data sets as multiple area data sets (MADSs).
	xxxxxIO	INITSADS	Initializes shadow ADSs.
Copy	xxxxxC0	COPYSADS	Copies data from original ADSs to shadow ADSs using the DEDB ADS Create utility.
	xxxxxP0	STOPSADS	Stops shadow ADSs and unregisters them from the production RECON data sets.
Unload	xxxxxU0	UNLOAD	Unloads data from shadow ADSs to unloaded files.
Reload	xxxxxR0	RELOAD	Deletes shadow ADSs and defines destination ADSs for reload, defines user image copy data sets (UIC), updates IMS DBCTL RECON data sets, reloads data into restructured ADSs and UICs, and deletes unloaded files.
	xxxxxK0	ANALYZE	Analyzes restructured ADSs. This job is generated and submitted only when ANALYZE=AFTRELOAD, which specifies to analyze the restructured ADSs after reload, is specified.
APPLY	xxxxxJ*	STAAPPLY	Issues the STASync command by using IMS Tools Online System Interface (TOSI) to start the DEDB and areas in the IMS DBCTL system. Asterisk (*) indicates the job sequence number starting from 0.

Table 121. OSC internal jobs (continued)

Phase	Job name	Job ID	Description
Takeover	xxxxxxD0	DBRPROD	Issues the /DBR command to stop the DEDB and areas in all sharing production IMS systems.
	xxxxxxH0	CKPTSTAT	Issues the /CHECKPOINT STATISTICS command to flush XCF buffers. This job might be submitted more than once.
	xxxxxxS0	DBRDBCTL	Issues the DBRSYNC command by using IMS Tools Online System Interface (TOSI) to stop the DEDB and areas in the IMS DBCTL system.
	xxxxxxB0	UNDDBCTL	Unregisters restructured ADSs and UICs from IMS DBCTL RECON data sets.
	xxxxxxM0	OLC	Issues the INITIATE OLC command to perform online change of the ACB library. This job is generated and submitted only when ACBLIB-managed ACBs are used.
	xxxxxxX0	IMPORT	Issues the IMPORT DEFN SOURCE(CATALOG) command. The command performs online change to activate the pending ACB in the IMS catalog. This job is generated and submitted only when IMS managed ACBs are used.
	xxxxxxY0	COPYRTN	Copies the new randomizing routine or new edit/compression routines from NEWRESLB to IMSRESLB. This job is generated and submitted only when the randomizing routine or compression routines are changed.
	xxxxxxQ0	REPLRAND	Issues the REPLRAND command by using IMS Tools Online System Interface (TOSI) to replace the randomizer. This randomizer replacement does not stop the entire DEDB but only certain areas. This job is generated and submitted only when REPLRAND_AREALVL=YES, which specifies to issue the TOSI REPLRAND command, is specified and certain conditions are met.
	xxxxxxO0	DELAREA	Unregisters deleted areas from production RECON data sets. This job is generated and submitted only when one or more areas need to be deleted.
	xxxxxxW0	ADDAREA	Registers new areas to production RECON data sets. This job is generated and submitted only when new areas are added.
xxxxxxN0	RENAME	Renames original ADSs to temporary data set names, renames restructured ADSs to original data set names, and issues the NOTIFY.UIC command to production RECON data sets.	
xxxxxxT0	STAPROD	Issues the /STA command to start the DEDB and areas.	

Table 121. OSC internal jobs (continued)

Phase	Job name	Job ID	Description
Termination	xxxxxxL0	TERMPROC	Performs termination processing, such as collecting all internal job logs and storing them in the data set that the LOGDSN keyword specifies.

Additional jobs generated by the OSC driver job

The OSC driver job also generates some jobs that are not intended to be submitted automatically during the OSC restructuring process. These additional jobs are generated to assist you in certain situations.

Table 122. Additional jobs generated by the OSC driver job

Job name	Job ID	Description
xxxxxxG0	SAVELOG	Collects all the logs (OSC driver job log and OSC internal job logs) and stores them in the data set that the LOGDSN keyword specifies. Submit this job if the OSC driver job ends abnormally and you need to collect all the logs to send to IBM Software Support.
xxxxxxV0	STAORIG	Issues the /STA command to start the original DEDB and areas. Submit this job if the OSC driver job ends abnormally and you need to bring the original DEDB and areas back online.

Internal SYSIN files

The OSC driver job generates some internal SYSIN files in the ENQ data set that the ENQDSN keyword specifies. These SYSIN files are used by OSC internal jobs. The ENQ data set is also used for serializing certain OSC internal jobs of other OSC driver jobs when multiple OSC driver jobs are executed simultaneously in an IMSplex.

Table 123. SYSIN files generated in the ENQ data set

Member name	Description
xxxxxxM1	This SYSIN file is used by the batch SPOC utility invoked by the xxxxxxM0 internal job. It is used to issue the INIT OLC PHASE(PREPARE) command. This member is generated only when ACBLIB-managed ACBs are used.
xxxxxxM2	This SYSIN file is used by the batch SPOC utility invoked by the xxxxxxM0 internal job. It is used to issue the INIT OLC PHASE(COMMIT) command. This member is generated only when ACBLIB-managed ACBs are used.
xxxxxxX1	This SYSIN file is used by the batch SPOC utility invoked by the xxxxxxX0 internal job. It is used to issue the IMPORT DEFN SOURCE(CATALOG) command. This member is generated only when IMS managed ABCs are used.

Online Structure Change utility restrictions

Before using the Online Structure Change utility, you need to understand the restrictions of the utility.

- Multiple OSC jobs that restructure the same DEDB cannot run concurrently.
- The Online Structure Change utility does not support DEDBs that have subset pointers.

- During an OSC job, path calls and FLD calls cannot be issued.

Unsupported DEDB structure changes

The following DEDB structure changes are not supported:

- Changing the names of existing segments.
- Deleting existing segments.
- Reducing the maximum lengths of existing segments.
- Changing fixed-length segments to variable-length segments or variable-length segments to fixed-length segments.

The Online Structure Change utility can restructure a DEDB that has Fast Path secondary indexes (FPSIs) but it cannot create or rebuild FPSIs. Therefore, the following structure changes, which affect FPSIs, are not supported:

- Adding new FPSI databases to the DEDB.
- Changing the DBD description of the source segments, which is used to generate pointer segments in FPSIs. For example, changing the length of existing segment field (BYTES parameter in DBD), or changing the partition selection exit routine.

The Online Structure Change utility cannot restructure the DEDB if the randomizer defined for that DEDB is a randomizer consisting of multiple modules generated with IMS Sequential Randomizer Generator (SRTYPE=MULTI option).

Unsupported randomizer changes

The Online Structure Change utility does not support replacing a randomizer that is being used by other databases.

Unsupported edit/compression exit routine changes

The following edit/compression exit routine changes are not supported:

- Replacing an edit/compression exit routine that is being used by other databases.
- Replacing with an edit/compression exit routine that has identical definitions.

Online Structure Change utility considerations

This topic describes what you need to consider before using the Online Structure Change utility.

Subsections:

- [“IMS DBCTL system” on page 1014](#)
- [“IMS online change \(when ACBLIB-managed ACBs are used\)” on page 1015](#)
- [“IMS online change \(when IMS managed ACBs are used\)” on page 1015](#)
- [“Status of the DBD and ACB” on page 1015](#)
- [“XCF groups” on page 1015](#)
- [“Multiple area data set \(MADS\)” on page 1016](#)
- [“RECON information” on page 1016](#)
- [“VSAM cluster names for area data sets \(ADSs\) and user image copies \(UICs\)” on page 1016](#)
- [“Replacing randomizer and segment edit/compression exit routines” on page 1016](#)

IMS DBCTL system

The Online Structure Change utility uses an IMS DBCTL system to apply the application updates that it captured while the OSC processes are running.

- The IMS DBCTL system must be configured in a separate IMS production environment. The IMS DBCTL system must not be a member of the data sharing group that the target DEDB belongs to.
- The IMS DBCTL system must be started before an OSC job is submitted and stopped after the OSC job is complete. The IMS DBCTL system must be active during the OSC job execution.
- The DEDB to restructure must be registered to the IMS DBCTL system. However, the DEDB should not be registered to the IMS DBCTL RECON data sets before the OSC job.
- The DEDB that the utility restructures must be specified on the PCB statement of the PSB definition. The utility obtains the PSB name from the PSBNAME_APPLY keyword and uses it during the apply phase.
- One IMS DBCTL system can be shared by multiple OSC jobs. However, an IMS DBCTL system cannot be shared by two or more OSC jobs that concurrently process physically different DEDBs but have the same DBD name.
- The ACB used by the IMS DBCTL system must be the new ACB that has the new database specifications.
- When the utility processes DEDBs that have Fast Path secondary indexes, it adds the Set Index Maintenance Off (SETI) control statement to the DFSCAL DD statement to suppress index maintenance in the IMS DBCTL system.
- Information about the IMS DBCTL system that the Online Structure Change utility requires to apply updates must be specified in the HPSYSIN data set of the OSC driver job.
- Instead of using an IMS DBCTL system, you can use an IMS DB/DC system.

IMS online change (when ACBLIB-managed ACBs are used)

During the takeover phase, the Online Structure Change utility issues IMS global online change commands so that the IMS online change function changes the ACBLIB members. IMS online change supports both area-level and database-level changes.

Area-level change

An area-level change involves increasing or decreasing the size of an area (IOVF, DOVF, CI). Only the affected areas are stopped.

Database-level change

A database-level change affects the structure of the DEDB, and includes such changes as adding or deleting an area, adding a segment type, or changing the segment edit/compression routines. All areas of the DEDB are stopped.

If the DBD change accompanies changes to the PSB, you must run the IMS online change function to replace the PSB after the Online Structure Change utility job completes.

IMS online change (when IMS managed ACBs are used)

The Online Structure Change utility issues the IMPORT DEFN SOURCE(CATALOG) command. By this command, the IMS online change function activates the pending ACBs that are stored in the staging data set of the IMS catalog.

If the DBD change accompanies changes to the PSB, you must run the IMS online change function to replace the PSB after the Online Structure Change utility job completes.

Status of the DBD and ACB

Before an OSC job is submitted, the DBD and ACB that IMS uses must match the structure of the current DEDB area. If you extended the IOVF portion of the area by using the FPO OAE utility, you must use the online change function to bring the new ACB member in the ACB library online, or use the IMPORT DEFN SOURCE(CATALOG) command to activate the pending ACB in the IMS catalog.

XCF groups

The Online Structure Change utility uses the following XCF groups:

- XCF group used for XCF communications among IMS control regions, OSC driver job, and OSC internal apply job (xxxxxxA0). The name of this XCF group is OSCXCF. You can change the XCF group name using the XCFGROUP keyword of the base configuration module.
- XCF group used for the IMS Tools Online System Interface. The IMS Tools Online System Interface is used in the production IMS systems and in the IMS DBCTL system.

The name of the XCF group used by the IMS Tools Online System Interface in the production IMS systems is specified by the TOIXCFGRP keyword.

The name of the XCF group used by the IMS Tools Online System Interface in the IMS DBCTL system is specified by the TOIXCFGRP_APPLY keyword.

Multiple area data set (MADS)

If the original ADS is MADS, the Online Structure Change utility changes it to SADS.

A new shadow ADS is allocated for each area. The Online Structure Change utility copies data from the original ADSs to the shadow ADSs by using the DEDB ADS Create utility. Before you run the OSC driver job, you must ensure that the number of copies registered to the RECON data sets is less than seven. If seven copies are already registered to the RECON data sets, the OSC driver job fails.

RECON information

During the takeover phase, the Online Structure Change utility issues the NOTIFY.UIC command to register user image copies to the production RECON data sets. To issue this command, the utility requires that the attribute of the corresponding DBDS record be NOREUSE. If the utility detects a REUSE attribute, it changes it to NOREUSE.

VSAM cluster names for area data sets (ADSs) and user image copies (UICs)

The VSAM cluster name and the data component name must conform to the following rules:

Original area data sets

To have the Online Structure Change utility rename the original ADSs instead of deleting them, the total length of the cluster name and the data component name must be 42 characters or less.

Shadow area data sets and user image copy data sets

If you specify the DATACOMP_SUFFIX keyword, the cluster names of the shadow ADSs and UICs must be "44 - the length of suffix - 1" or less. Cluster names for shadow ADSs and UICs are determined by the DSNMASK_SHADOW keyword and the DSNMASK_UIC keyword.

Restructured area data sets

If you do not specify the DATA parameter in the IDCAMS control statements for the restructured ADS, the following rules apply:

- If DATACOMP_SUFFIX=DATA, the cluster name for the restructured ADS must be 38 characters or less.
- If DATACOMP_SUFFIX=D, the cluster name of for restructured ADS must be 39 to 42 characters.

Replacing randomizer and segment edit/compression exit routines

The following considerations apply when replacing a randomizer or segment edit/compression exit routines.

Randomizers and segment edit/compression exit routines used by other databases

When you replace a randomizer or segment edit/compression exit routines used in production IMS systems, you must ensure that the randomizer or the segment edit/compression exit routines to replace are not being used by any other databases.

Generally, when they are replaced, an IMS restart is required to refresh these resources. Therefore, during the OSC initialization phase, the Online Structure Change utility calls IMS Tools Online System Interface to issue the CHKRAND or CHKCOMP command to check whether the randomizer or the

routines to be replaced are not being used by any other databases. If they are used by other databases, the OSC driver job ends abnormally without replacing them.



Attention: The CHKCOMP command, issued through IMS Tools Online System Interface, checks whether segment edit/compression exit routines to be replaced are used by other databases. For full-function databases, it checks only the DMBs found in the DMB pool. This means that it cannot determine whether a full-function database is using the segment edit/compression exit routines if the full-function database is stopped. So it is possible that when a full-function database is started after replacing segment edit/compression exit routines, the full-function database is using unexpected segment edit/compression exit routines. It is strongly recommended that, before you run an OSC driver job, you ensure that the edit/compression exit routines to be replaced are not being used by any other databases, especially by full-function databases that are in stopped status.

Note: The randomizer and segment/edit compression exit routine considerations discussed in this subsection do not apply if you are only changing DBD definitions without replacing the randomizer or segment/edit compression exit routines. In this case,

- The randomizer or segment edit/compression exit routines can be used by multiple databases.
- The new randomizer or segment edit/compression exit routines must exist in the production IMS systems.

For more information about replacing a randomizer or segment edit/compression exit routines, see [“NEWRESLB_RND_DSN keyword” on page 1065](#) or [“NEWRESLB_EDC_DSN keyword” on page 1064](#).

Area-level randomizer replacement

Generally, all areas of the DEDB must be stopped when replacing the randomizer. However, when all of the following conditions are met, area-level randomizer replacement will take place, therefore, only the affected areas are stopped:

- The replacement of the randomizer affects only certain areas.
- All areas affected by the replacement are specified with the IAREA keyword.
- The REPLRAND_AREALVL=YES keyword is specified.



Attention: Ensure that all areas affected by the replacement are specified with the IAREA keyword. Otherwise, area-level randomizer replacement might destroy the database.

Chapter 37. Restructuring an online DEDB

The following topics provide instructions to restructure a DEDB with the Online Structure Change utility.

Topics:

- [“Running the Online Structure Change utility” on page 1019](#)
- [“Region size requirements for the Online Structure Change utility” on page 1021](#)
- [“Setting site default values for the Online Structure Change utility” on page 1022](#)
- [“Troubleshooting Online Structure Change utility problems” on page 1023](#)

Running the Online Structure Change utility

To restructure an online DEDB with the Online Structure Change utility, complete the following steps.

Before you begin

Before you can run the Online Structure Change utility, you must ensure that your environment is ready to run an OSC job:

- Ensure that the Online Structure Change utility environment is configured correctly. For configuration steps, see the topic "Setting up the Online Structure Change utility" in *IMS Fast Path Solution Pack: Overview and Customization*.
- Ensure that, for each ADS, the number of copies is less than seven. For more information, see [“Multiple area data set \(MADS\)” on page 1016](#).
- Ensure that all IMS systems in the IMSplex where you run the OSC job have the executable load library of IMS HP Fast Path Utilities (SHFPLMD0) in the STEPLIB concatenation.
- If you want to replace the randomizer or segment edit/compression exit routines, ensure that the randomizer or the segment edit/compression exit routines are not being used by other databases.

Tip: If you want to change default values of the utility control statements, create a site default table. For more information, see [“Setting site default values for the Online Structure Change utility” on page 1022](#).

Procedure

1. Create an OSC driver job.
2. Optional: Estimate the region size that is required to run the job. For more information, see [“Region size requirements for the Online Structure Change utility” on page 1021](#).
3. Code DD statements to define input and output data sets.

For a list of DD statements, see [Chapter 38, “JCL requirements for the Online Structure Change utility,” on page 1027](#).

- If you created a base configuration module, you must add the module to the STEPLIB concatenation.
 - If you created a site default table, you must add the library that contains the site default member (HFPOCTL0) to the STEPLIB or JOBLIB concatenation.
4. Code the utility control statements for the HFPSYSIN DD statement.

For a list of control statement keywords, see [Chapter 39, “Utility control statements for the Online Structure Change utility,” on page 1031](#). GLOBAL command, RESTRUCTURE command, ALLOCATE subcommand, and JCLPARM subcommand are required. LOADCTL subcommand is optional.

The following sub-steps describe keywords that require allocation of data sets:

- a) Define and allocate an OSC internal jobs data set. If the data set exists, ensure that it has enough space. Specify this data set for the JCLDSN keyword of the JCLPARM subcommand.

The Online Structure Change utility generates OSC internal jobs in this data set.

- This data set, if used by multiple OSC driver jobs, must be a PDSE.
- The name of the data set must match the SDN parameter that is specified in the internal reader facility procedure.

- b) Define and allocate an OSC internal job log data set. If the data set exists, ensure that it has enough space. Specify the prefix of this data set for the LOGDSN keyword of the JCLPARM subcommand.

The Online Structure Change utility stores job logs from the OSC internal jobs in this data set.

Tip: If the OSC driver job ends abnormally, you can submit the xxxxxxG0 job (generated by the OSC driver job) to collect all the job logs in this data set.

- c) Create a partitioned data set and code IDCAMS control statements for each area. Specify these data sets to the ATTRIBUTE keyword of the ALLOCATE subcommand.

For each area, you must create a member that contains IDCAMS control statements for allocating the restructured area. For more information, see [“Example: Specifying the ALLOCATE subcommand”](#) on page 1087.

- d) Define and allocate an ENQ data set. The Online Structure Change utility uses this data set in the online change process. If the data set exists, ensure that it has enough space. Specify this data set for the ENQDSN keyword of the JCLPARM subcommand.

The Online Structure Change utility generates some internal SYSIN files in the ENQ data set. For more information about SYSIN files generated in ENQ data sets, see [“Internal SYSIN files”](#) on page 1013.

- This data set, if used by multiple OSC driver jobs, must be a PDSE.
- If you plan to execute multiple OSC driver jobs simultaneously in an IMSplex, all the SYSIN files must be generated in one ENQ data set. Ensure that the ENQDSN keywords in all OSC driver jobs specify the same ENQ data set.
- The name of the ENQ data set must not be the same as the name of the OSC internal jobs data set.

For the following keyword values, consult your system administrators for the values they used to configure the Online Structure Change utility environment:

CONFIGLIB_DSN keyword (JCLPARM subcommand)

Specify the base configuration module.

INTRDR keyword (JCLPARM subcommand)

Specify the internal reader facility procedure.

TOIXCFGRP keyword (GLOBAL command)

Specify the XCFGROUP name that the IMS Tools Online System Interface uses in the production IMS systems.

TOIXCFGRP_APPLY keyword (JCLPARM subcommand)

Specify the XCF group name that the IMS Tools Online System Interface uses in the IMS DBCTL system.

5. Optional: Supply the ERRORACT keyword. The ERRORACT keyword enables you to specify the action for the utility to take when one of the OSC internal job fails. This keyword helps you troubleshoot Online Structure Change utility problems.

If you specify ERRORACT=WTOR, the utility issues WTOR message HFPT0512A to prompt you to select the next action to take. If you specify ERRORACT=ABEND, the OSC driver job terminates when one of the OSC internal jobs fails.

6. If you specify ORIGADS_PROC=RENAME to rename the original ADSs, delete or rename the VSAM data set whose name is `original_ADS_name + .T`.
7. Ensure that the DEDB is registered to the IMS DBCTL system and that the IMS DBCTL system is started with the new ACB definition.

8. Activate the OSC logger exit routine.

The OSC logger exit routine is activated when the IMS control region is started. Ensure that the following messages are issued:

```
HFP01000I  LOGGER  EXIT  INITIALIZING
HFP01503I  JOINED  XCF  GROUP xxxxxxxx AS MEMBER yyyy
HFP01000I  LOGGER  EXIT  INITIALIZED
```

9. Submit the OSC driver job.

Ensure that the job ends without errors. If you receive error messages, look up the messages in [Part 7, “Troubleshooting,” on page 1209](#). Also, see [“Troubleshooting Online Structure Change utility problems” on page 1023](#).

If an OSC internal job ends with a JCL error, cancel the OSC driver job. Review the JCL for the OSC driver job, specify the correct job statements using the JOBCARDx keywords, and rerun the OSC driver job.

10. Stop the IMS DBCTL system.

Region size requirements for the Online Structure Change utility

The Online Structure Change utility operates in a virtual storage region as z/OS batch job steps, consisting of a driver job and multiple internal jobs.

Typically, database maintenance is a relatively important function. To increase the performance of the restructure program, it would be advisable to run it in a region that has a relatively high dispatching priority (that is, at least equal to the dispatching priority of a message processing region).

Estimating the region size for an OSC driver job

An OSC driver job (program HFPOSC00) runs in 31-bit addressing mode. The region size required for an OSC driver job is a function of the number of areas being restructured, the ITASKCTL and OTASKCTL specifications.

In most cases, the specification on the EXEC statement for HFPOSC00 of a region that is too small, is detected during the initialization processing of the main task. This causes processing to be terminated at that point. Under certain conditions, the region size shortfall does not become evident until the actual restructure has begun. The job must be completely rerun. Thus, the region size calculations must always be performed prior to initiating restructure processing.

The following worksheet is for estimating the region size that is required for a typical OSC driver job. The actual region size depends on the keywords that are specified for the job. Therefore, when you specify the region size in your JCL, use a higher value than the value calculated from this worksheet.

OSC driver programs, internal tables, control blocks	8,500,000
Number of areas defined in input DBD * 1,300	-----
Number of areas defined in output DBD * 1,300	-----
Total of input areas and output area * 4,400	-----
Total of output areas and output area * 4,400	-----
Total bytes of change storage	-----

Estimating the region size for the OSC internal APPLY job (xxxxxxA0)

An OSC internal APPLY job (program HFPOSC10) runs in 31-bit addressing mode. The region size required for an OSC internal APPLY job is a function of the number of unkey segments and the amount of captured online updates.

JOB statements for the OSC internal jobs, including the specification for the region size, can be specified with the JOBCARDx keyword of the OSC driver job. In most cases, the specification that is too small, is detected during the initialization processing of the main task. This causes processing to be terminated at that point. Under certain conditions, the region size shortfall does not become evident until the actual

apply has begun. The OSC driver job must be completely rerun. Thus, the region size calculations must always be performed prior to initiating restructure processing.

The following worksheet is for estimating the region size that is required for a typical OSC internal APPLY job. The actual region size depends on the actual online updates that are captured during the restructure processing. Therefore, when you specify the region size in your JCL, use a higher value than the value calculated from this worksheet.

OSC internal APPLY programs, internal tables, control blocks	270,000,000
Number of areas defined in input DBD * 1,300	-----
Number of areas defined in output DBD * 1,300	-----
Total of input areas and output area * 4,400	-----
Total of output areas and output area * 4,400	-----
Total bytes of change storage	-----

Use the following information to estimate the size of data space for the OSC internal APPLY job.

RBA table initial size 2G, extending with 2G if not enough
Message buffer 256M

Estimating the region size for other OSC internal jobs

- xxxxxxIO: See [“Region size requirements for the Initialize function”](#) on page 369.
- xxxxxxJx: See [“Region size requirements for the Analyze function”](#) on page 200.
- xxxxxxUO: See [“Region size requirements for the Unload function”](#) on page 55.
- xxxxxxRO: See [“Region size requirements for the Reload function”](#) on page 93.

Setting site default values for the Online Structure Change utility

The Online Structure Change utility supports site default options to override the system default values. By creating a site default module HFPOCTL0 and enabling the site default options, you can change the defaults for keywords and omit specific keywords in the HFPSYSIN control statement.

Procedure

1. Copy member HFPOCTL0 in the sample JCL library to your JCL library.
2. Add keywords and their values to the HFPCRE.HFPSYSIN DD statement.

The following figure shows example JCL statements to create a site default module HFPOCTL0.

```

//SITEMOD PROC
//*
//HFPCRE EXEC PGM=HFPOSC00
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//HFPPRINT DD SYSOUT=*
//HFPRPTS DD SYSOUT=*
//SYSPUNCH DD DISP=(NEW,PASS,DELETE),
//          SPACE=(TRK,(1,1)),UNIT=SYSDA,DSN=&&SOURCE
//*
//ASM EXEC PGM=ASMA90,PARM='OBJECT,NODECK',COND=(4,LT,HFPCRE)
//SYSLIB DD DISP=SHR,DSN=SYS1.MACLIB
//SYSLIN DD DSN=&&OBJLIB,DISP=(NEW,PASS,DELETE),
//          UNIT=SYSDA,SPACE=(TRK,(1,1))
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD UNIT=SYSDA,SPACE=(TRK,(1,1))
//SYSIN DD DISP=(OLD,DELETE,DELETE),DSN=&&SOURCE
//*
//LKED EXEC PGM=IEWL,COND=(4,LE,ASM)
//SYSPRINT DD SYSOUT=*
//SYSLIN DD DSN=&&OBJLIB,DISP=(OLD,DELETE,DELETE)
//SYSUT1 DD UNIT=SYSDA,SPACE=(TRK,(1,1))
//SYSLMOD DD DISP=OLD,DSN=HPFP.USERLIB(HFPOCTL0)
//*
// PEND
//*
//STEP01 EXEC SITEMOD
//HFPCRE.HFPPRINT DD *
RESTRUCTURE
ERRORACT=WTOR,
ORIGADS_PROC=DELETE
/*
//*

```

Figure 384. JCL for creating a site default module (HFPOCTL0)

3. Submit the job.
4. Concatenate the library that includes HFPOCTL0 to the STEPLIB DD or JOBLIB DD concatenation of the OSC driver job.

Troubleshooting Online Structure Change utility problems

If you receive error messages, you must identify the cause of the error, correct it, and, if necessary, restart the OSC job.

Before you start troubleshooting an OSC problem, ensure that your environment is correctly set up for the Online Structure Change utility. If any of the following conditions is not satisfied, the OSC driver job ends with errors:

- Make sure that, for each ADS, the number of copies is less than seven. The Online Structure Change utility registers the shadow ADS as a copy during the job, and MADS (Multiple Area Data Sets) can only maintain up to seven copies of each area.
- Make sure that all IMS systems in the same IMSplex where you run OSC have the executable load library of IMS HP Fast Path Utilities (SHFPLMD0) in the STEPLIB concatenation.
- Make sure that the IMS DBCTL system is started with the new ACB definition.
- Define a new OSC internal job data set if it does not exist. If it already exists, make sure that it has enough space. Also make sure that this data set is specified on the JCLDSN keyword and that the data set contains the OSC internal jobs that were generated and submitted during the OSC process.
- Define a new ENQ data set if it does not exist. If it already exists, make sure that it has enough space. Also make sure that this data set is specified on the ENQDSN keyword and that the data set contains the members that were generated during the OSC process.
- Define a new OSC internal job log data set if it does not exist. If it already exists, make sure that it has enough space. Also make sure that the prefix of this data set is specified on the LOGDSN keyword and that the data set contains the OSC internal job logs collected during the OSC process.
- If ORIGADS_PROC=RENAME is specified, delete or rename the VSAM data set whose name is "original ADS + .T".

For details, see [“Running the Online Structure Change utility” on page 1019](#).

The following sections explain how to troubleshoot and restart Online Structure Change utility jobs.

- [“When message HFPT0512A is issued” on page 1024](#)
- [“OSC driver job ended abnormally when RUNTYPE=GENJCL” on page 1024](#)
- [“OSC driver job ended abnormally before the takeover phase” on page 1025](#)
- [“OSC driver job ended abnormally during the takeover phase” on page 1025](#)

When message HFPT0512A is issued

When the ERRORACT=WTOR option is enabled and one of the OSC internal job fails, the utility issues WTOR message HFPT0512A. When this message is displayed, follow these steps to restart the OSC driver job:

1. From the HFPT0512A message, identify the OSC internal job that failed.
2. Locate error messages in the corresponding job log and identify the cause of the error.
3. Correct the error. If the error occurred in one of the following OSC internal jobs, follow the instructions and correct the failed OSC internal job in the internal job data set (specified by the JCLDSN keyword).

If xxxxxxCO (COPYSADS) failed

- a. Check the status of the shadow ADS in the production DBRC.
 - If the shadow ADS is in AVAIL status, separate the shadow ADS from the production IMS online system by issuing the /STOP ADS command, then unregister the shadow ADS from the production DBRC.
 - If the shadow ADS is in UNAVAIL status, unregister the shadow ADS from the production DBRC.
- b. Submit the xxxxxxFO (DEFSADS) internal job to define the shadow ADS.
- c. Submit the xxxxxxIO (INITSADS) internal job to initialize the shadow ADS.

If xxxxxxR0 (RELOAD) failed

Check the restructured ADS and UIC in the DBCTL DBRC. If the restructured ADS and UIC are registered in the DBCTL DBRC, submit the xxxxxxBO (UNDDBCTL) internal job.

If xxxxxxMO (OLC) failed

Issue the TERMINALTE OLC command to terminate the global online change.

If xxxxxxNO (RENAME) failed

Check whether the original ADSs exist.

- If ORIGADS_PROC=DELETE and the original ADSs do not exist, remove the DELETE ORIGINAL ADS job step from the xxxxxxNO internal job.
 - If ORIGADS_PROC=RENAME and the original ADSs do not exist, remove the RENAME ORIGINAL ADS job step from the xxxxxxNO internal job.
4. Rerun the failed OSC internal job and check the corresponding job log.
 - If the error is resolved, reply to the HFPT0512A message with C. The OSC driver job restarts.
 - If the OSC internal job fails, repeat steps [“3” on page 1024](#) and [“4” on page 1024](#).

OSC driver job ended abnormally when RUNTYPE=GENJCL

If the OSC driver job was submitted with the RUNTYPE=GENJCL option and the job failed, follow these steps:

1. Review the OSC driver job log to obtain information about error.
2. Correct the error conditions.
3. Submit the OSC driver job.

OSC driver job ended abnormally before the takeover phase

Follow these steps if the OSC driver job was submitted with the RUNTYPE=FULL or RUNTYPE=RUNJCL option and ended abnormally before the takeover phase. The following steps are valid if the xxxxxxDO (DBRPROD) internal job was not submitted.

1. Submit the xxxxxxG0 (SAVELOG) job, which is generated in the internal job data set that the JCLDSN keyword specifies, to collect the job logs of the OSC driver job and OSC internal jobs.
2. Check the job logs in the job log data set that the LOGDSN keyword specifies to obtain information about the error.
3. Correct the error conditions.
4. Check the status of the shadow ADS in the production DBRC.
 - If the shadow ADS is in AVAIL status, separate the shadow ADS from the production IMS online system by issuing the /STOP ADS command, then unregister the shadow ADS from the production DBRC.
 - If the shadow ADS is in UNAVAIL status, unregister the shadow ADS from the production DBRC.
5. Check the restructured ADS and UIC in the DBCTL DBRC. If the restructured ADS and UIC are registered in the DBCTL DBRC, unregister them.
6. Issue the /START PGM ALL command to the DBCTL region.
7. Submit the OSC driver job.

OSC driver job ended abnormally during the takeover phase

Follow these steps if the OSC driver job was submitted with the RUNTYPE=FULL or RUNTYPE=RUNJCL option and ended abnormally during the takeover phase. The following steps are valid if the xxxxxxDO (DBRPROD) internal job was submitted.

1. Submit the xxxxxxG0 (SAVELOG) job, which is generated in the internal job data set that the JCLDSN keyword specifies, to collect the job logs of the OSC driver job and OSC internal jobs.
2. Check the job logs in the job log data set that the LOGDSN keyword specifies to obtain information about the error.
3. Correct the error conditions.
4. Submit the following OSC internal jobs.
 - If xxxxxxM0 (issues the INITIATE OLC command) or xxxxxxX0 (issues the IMPORT command) has ended normally, submit the following OSC internal jobs:
 - xxxxxxO0 if this internal job is generated and was not submitted during the previous run
 - xxxxxxW0 if this internal job is generated and was not submitted during the previous run
 - xxxxxxN0 if this internal job is generated and was not submitted during the previous run
 - xxxxxxT0 if this internal job was not submitted during the previous run
 - xxxxxxL0 if this internal job was not submitted during the previous run
 - If xxxxxxM0 (issues the INITIATE OLC command) or xxxxxxX0 (issues the IMPORT command) was not submitted or ended with an error, complete the following steps:
 - a. If xxxxxxM0 ended with an error during or after the INITIATE OLC PHASE(PREPARE) command was issued and before the INITIATE OLC PHASE(COMMIT) command was issued, issue the TERMINATE OLC command to abort IMSplex-wide global online change that was initiated by the INITIATE OLC PHASE(PREPARE) command.
 - b. If xxxxxxY0 ended successfully, restore the old randomizing routine or the old edit/compression routines to the IMSRESLB data set.
 - c. Submit the xxxxxxV0 (STAORIG) job to bring the input DEDB and areas back online.
 - d. If xxxxxxA0 ended abnormally, issue the /START PGM ALL command to the DBCTL region.

Chapter 38. JCL requirements for the Online Structure Change utility

DD statements for the Online Structure Change utility determine the input and output data sets and specify how to run the utility.

Subsections:

- [“List of DD statements” on page 1027](#)
- [“Input DD statements” on page 1027](#)
- [“Output DD statements” on page 1028](#)

List of DD statements

The following table summarizes the DD statements for the OSC driver job of the Online Structure Change utility.

Table 124. DD statements for the Online Structure Change utility

DDNAME	Use	Format	Can be dynamically allocated?	Required or optional
JOBLIB or STEPLIB	Input	PDS	No	Required
IMSACB	Input	PDS	No	Required when IMSCATHLQ=*NO
NEWACB	Input	PDS	No	Required when IMSCATHLQ=*NO
HFPYSIN	Input	LRECL=80	No	Required
HFPPRINT	Output	SYSOUT	Yes	Optional
HFPRPTS	Output	LRECL=133	Yes	Optional
HFP3XMSG	Output	LRECL=133	Yes	Optional
HFP3XTRA	Output	LRECL=133	Yes	Optional

Note: Do not specify RECONx, IMSDALIB, HFPDDCHK, HFP1JCLD, and HFP1ENQ DD statements.

Input DD statements

The following input DD statements are used for the Online Structure Change utility.

JOBLIB DD STEPLIB DD

This DD statement is a required DD statement that defines the following input data sets:

- OSC load module library (required)
- Library that contains the site default member (HFPOCTL0) (optional)
- Library that contains the base configuration module HFP#*imsid* or HFPSETOP (optional)
- IMS load module library (required)
- IMS Tools Base library SFOILOAD (required)
- IMS Tools Base library SGLXLOAD (required when IMSCATHLQ is specified)

- IMS Tools Base library SHKTLOAD (required when ITKBSRVR is specified)

The Online Structure Change utility must be run as an APF-authorized program. All the data sets in the JOBLIB or the STEPLIB concatenation must also be APF-authorized.

IMSACB DD

This DD statement is a required DD statement. It defines the library that contains the active ACB member for the database.

If you specify the IMSCATHLQ keyword, IMSACB DD statements are not used. The IMS directory is used instead of the ACB library.

NEWACB DD

This DD statement is a required DD statement. It defines the library that contains the staging ACB member for the database.

If you are using a DFSMDA member with TYPE=IMSACB to specify the ACBLIB staging library, the NEWACB DD statement of the OSC driver job must specify the same ACBLIB library.

If you specify the IMSCATHLQ keyword, NEWACB DD statement is not used. The IMS directory is used instead of the ACB library.

HFPSYSIN DD

This DD statement is a required DD statement that defines the input control statement for the Online Structure Change utility.

The HFPSYSIN DD statement can be coded as a standard SYSIN file, a sequential data set, or a PDS member. LRECL=80 is required for the DCB of this data set.

The Online Structure Change utility analyzes the command syntax in columns 1 - 72. Columns 73 - 80 are regarded as comment.

For details about coding the HFPSYSIN DD statement, see [Chapter 39, "Utility control statements for the Online Structure Change utility," on page 1031.](#)

Output DD statements

The following output DD statements are used for the Online Structure Change utility.

HFPPRINT DD

This DD statement is an optional DD statement that defines the processing log output data set to which the Online Structure Change utility writes processing messages.

The following DCB attributes of the file are set by the OSC main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set. You cannot specify DUMMY.

The HFPPRINT data set contains 133-byte, fixed-length records. It is recommended that you code the DD statement as follows:

```
//HFPPRINT DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, the Online Structure Change utility dynamically allocates the data set by using SYSOUT=*.

HFPRPTS DD

This DD statement is an optional DD statement that defines the report output data set.

The following Online Structure Change utility reports, which you can use to verify the processes that are run by the Online Structure Change utility, are written to this data set:

- Audit report
- Processing report
- DBD Definition reports for both active DMB obtained from IMSACB and staging DMB obtained from NEWACB
- Base Configuration report
- OSC Internal Jobs Scenario report

The following DCB attributes of the file are set by the OSC main task:

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

Do not code the DCB information in your JCL statement. The output data set must be a standard SYSOUT or a sequential data set.

The HFPRPTS data set contains 133-byte, fixed-length records. It is recommended that you code the DD statement as follows:

```
//HFPRPTS DD SYSOUT=A
```

Note on dynamic allocation:

When this statement is not specified in the JCL statements, the Online Structure Change utility dynamically allocates the data set by using SYSOUT=*.

HFP3XMSG DD

This DD statement is an optional DD statement that defines the processing log output data set to which the XCF CONNECT task writes processing messages.

The output data set must be a standard SYSOUT or a sequential data set.

The HFP3XMSG data set contains 133-byte, fixed-length records.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, the Online Structure Change utility dynamically allocates the data set by using SYSOUT=*.

HFP3XTRA DD

This DD statement is an optional DD statement that defines output data set that the Online Structure Change utility uses to write diagnostic trace records. This data set might be required for problem diagnosis.

The HFP3XTRA data set contains 133-byte, fixed-length records.

Note on dynamic allocation:

When this statement is not specified in the JCL statements, the Online Structure Change utility dynamically allocates the data set by using SYSOUT=*.

Output destination of other reports

- Apply report is generated in the data set that the HFPRPTS DD statement of the internal apply job (xxxxxxA0) specifies.
- Other reports, such as unload and reload reports, are generated in the report data set that is specified in each OSC internal job.

Chapter 39. Utility control statements for the Online Structure Change utility

Utility control statements for the Online Structure Change utility define the user description of the GLOBAL command, RESTRUCTURE command, ALLOCATE subcommand, JCLPARM subcommand, and optional LOADCTL subcommand, all of which control the OSC process.

Utility control statements must be specified in the HFPSYSIN data set. This control statement data set usually resides in the input stream. However, it can also be defined as a sequential data set or as a member of a partitioned data set. It must contain 80-byte, fixed-length records. Block size, if coded, must be a multiple of 80.

You can change the default values of utility control statement keywords by setting the values in the site default module. For more information, see [“Setting site default values for the Online Structure Change utility” on page 1022](#).

Subsections:

- [“GLOBAL command keywords” on page 1031](#)
- [“RESTRUCTURE command keywords” on page 1031](#)
- [“ALLOCATE subcommand keywords” on page 1032](#)
- [“JCLPARM subcommand keywords” on page 1033](#)
- [“LOADCTL subcommand keywords” on page 1035](#)

GLOBAL command keywords

The GLOBAL command is a required command. Use the GLOBAL command to establish implicit job step keyword values.

For the GLOBAL command, you can specify the keywords that are listed in the following table. For the format of the GLOBAL command, see [“GLOBAL command keywords” on page 1036](#).

Table 125. GLOBAL command keywords

Keyword	Required or optional	Description	Topic
IMSCATHLQ	Optional	Specifies the high-level qualifier of the bootstrap data set of the IMS directory.	“IMSCATHLQ keyword” on page 1036
IMSID	Required	Specifies the production IMS system ID.	“IMSID keyword” on page 1037
ITKBSRVR	Optional	Specifies the name of the IMS Tools KB server XCF group.	“ITKBSRVR keyword” on page 1037
MSTIMESTAMP	Optional	Specifies whether to include a timestamp in the messages.	“MSTIMESTAMP keyword” on page 1037
SCAN	Optional	Specifies whether to verify the syntax of command, keywords, and parameters without running OSC functions.	“SCAN keyword” on page 1038
TOIXCFGRP	Required	Specifies the XCF group that IMS Tools Online System Interface uses to interface with the Online Structure Change utility.	“TOIXCFGRP keyword” on page 1038

RESTRUCTURE command keywords

The RESTRUCTURE command is a required command. Use the RESTRUCTURE command to specify how you want to change the structure of DEDB areas.

For the RESTRUCTURE command, you can specify the keywords that are listed in the following table. For the format of the RESTRUCTURE command, see [“RESTRUCTURE command keywords” on page 1039](#).

Table 126. RESTRUCTURE command keywords

Keyword	Required or optional	Description	Topic
ANALYZE	Optional	Specifies to analyze ADSs.	“ANALYZE keyword” on page 1040
DBD	Required	Specifies the database description (DBD) that contains the areas to restructure.	“DBD keyword” on page 1040
ERRORACT	Optional	Specifies the action for the utility to take at certain error conditions.	“ERRORACT keyword” on page 1041
IAREA	Optional	Specifies one or more input areas to restructure.	“IAREA keyword” on page 1041
ITASKCTL	Optional	Specifies the maximum number of input areas to process concurrently.	“ITASKCTL keyword” on page 1042
OAREA	Optional	Specifies one or more output areas for the process.	“OAREA keyword” on page 1042
ORIGADS_PROC	Optional	Specifies whether to delete or retain the original ADSs.	“ORIGADS_PROC keyword” on page 1043
OTASKCTL	Optional	Specifies the maximum number of output areas to process concurrently.	“OTASKCTL keyword” on page 1044
PAD	Optional	Specifies the padding character to be used when the segment minimum length is increased due to DBD description change.	“PAD keyword” on page 1044
REPLRAND_AREALVL	Optional	Issues the REPLRAND command by using IMS Tools Online System Interface (TOSI) to replace the randomizer. Only certain areas are stopped while the randomizer is replaced.	“REPLRAND_AREALVL keyword” on page 1045
RUNTYPE	Optional	Specifies the operating mode.	“RUNTYPE keyword” on page 1045
SKIP_OLC	Optional	Specifies to skip online change.	“SKIP_OLC keyword” on page 1046
TAKEOVER_TIME	Optional	Specifies when the takeover phase should start.	“TAKEOVER_TIME keyword” on page 1046
UPDATE_NUMBER	Optional	Specifies the threshold for the number of online updates.	“UPDATE_NUMBER keyword” on page 1046
USR_MAX	Optional	Specifies the maximum number of USR (unloaded segment records) data sets that the Online Structure Change utility generates during the unload phase.	“USR_MAX keyword” on page 1047

ALLOCATE subcommand keywords

The ALLOCATE subcommand is a required subcommand. Use the ALLOCATE subcommand to provide the Online Structure Change utility the information that is required to allocate restructured ADSs.

For the ALLOCATE subcommand, you can specify the keywords that are listed in the following table. For the format of the ALLOCATE subcommand, see [“ALLOCATE subcommand keywords” on page 1047](#). Also see [“Example: Specifying the ALLOCATE subcommand” on page 1087](#).

Table 127. ALLOCATE subcommand keywords

Keyword	Required or optional	Description	Topic
ATTRIBUTE	Required	Specifies the data set to pass on to Access Method Services for allocating new VSAM clusters.	“ATTRIBUTE keyword” on page 1048
OAREA	Required	Specifies one or more output areas.	“OAREA keyword” on page 1048

JCLPARM subcommand keywords

The JCLPARM subcommand is a required subcommand. Use the JCLPARM subcommand to specify the options to be applied to OSC internal jobs.

For the JCLPARM subcommand, you can specify the keywords that are listed in the following table. For the format of the JCLPARM command, see “JCLPARM subcommand keywords” on page 1049.

Table 128. JCLPARM subcommand keywords

Keyword	Required or optional	Description	Topic
COMPRESS	Optional	Specifies that the output records in the unloaded files contain compressed segments.	“COMPRESS keyword” on page 1052
CONFIGLIB_DSN	Optional	Specifies the name of the library that contains configuration module HFP# <i>imsid</i> or HFPSETOP.	“CONFIGLIB_DSN keyword” on page 1052
DATAComp_SUFFIX	Optional	Specifies the suffix portion of the VSAM data component name.	“DATAComp_SUFFIX keyword” on page 1052
DBDLIB_DSN	Required	Specifies the name of the active DBDLIB library.	“DBDLIB_DSN keyword” on page 1053
DBRCGRP	Required	Specifies the identifier that is assigned to a group of DBRC instances (DBRC sharing group) that access the same RECON data set in the IMSplex. This keyword is for the production IMS systems.	“DBRCGRP keyword” on page 1053
DBRCGRP_APPLY	Optional	Specifies the identifier that is assigned to a group of DBRC instances (DBRC sharing group) that access the same RECON data set in the IMSplex. This keywords is for the IMS DBCTL system.	“DBRCGRP_APPLY keyword” on page 1054
DBR_NOFEOV	Optional	Adds the NOFEOV keyword to the /DBRECOVERY command that is issued to stop the DEDB or DEDB areas in the production IMS systems.	“DBR_NOFEOV keyword” on page 1054
DBR_NOFEOV_APPLY	Optional	Adds the NOFEOV keyword to the TOSI DBRSYNC command that is issued to stop the DEDB or DEDB areas in the IMS DBCTL system.	“DBR_NOFEOV_APPLY keyword” on page 1054
DCLASS_TEMP	Optional	Specifies the SMS data class for the temporary data sets that the Online Structure Change utility dynamically allocates.	“DCLASS_TEMP keyword” on page 1055
DDNPREFIX_NEW	Optional	Specifies the DD name prefix for the new area data sets that the Online Structure Change utility adds to the production environment. This keyword is required when adding new areas.	“DDNPREFIX_NEW keyword” on page 1055
DDNPREFIX_SHADOW	Required	Specifies the DD name prefix for the shadow area data sets and restructured area data sets that the Online Structure Change utility allocates.	“DDNPREFIX_SHADOW keyword” on page 1056
DDNPREFIX_UIC	Required	Specifies the DD name prefix for the user image copy data sets that the Online Structure Change utility allocates.	“DDNPREFIX_UIC keyword” on page 1056
DSNMASK_NEW	Optional	Specifies the data set name or the mask for the new area data sets that the Online Structure Change utility adds to the production environment. This keyword is required when adding new areas.	“DSNMASK_NEW keyword” on page 1056
DSNMASK_SHADOW	Required	Specifies the data set name or the mask for the shadow area data sets and restructured area data sets that the Online Structure Change utility dynamically allocates.	“DSNMASK_SHADOW keyword” on page 1057
DSNMASK_UIC	Required	Specifies the data set name or the mask for the user image copy data sets that the Online Structure Change utility dynamically allocates.	“DSNMASK_UIC keyword” on page 1057
ENQDSN	Optional	Specifies the name of the ENQ data set in which the utility generates internal SYSIN files.	“ENQDSN keyword” on page 1058

Table 128. JCLPARM subcommand keywords (continued)

Keyword	Required or optional	Description	Topic
FPLIB_DSN	Required	Specifies the name of the IMS Fast Path Solution Pack SHFPLMDO library.	“FPLIB_DSN keyword” on page 1058
IMSID_APPLY	Required	Specifies the IMS DBCTL system ID that the Online Structure Change utility uses for the apply job.	“IMSID_APPLY keyword” on page 1059
IMSLIB_DSN	Required	Specifies the name of the production IMS SDFSRESL library.	“IMSLIB_DSN keyword” on page 1059
IMSLIB_DSN_APPLY	Required	Specifies the name of the IMS DBCTL SDFSRESL library that is used for the apply job.	“IMSLIB_DSN_APPLY keyword” on page 1059
IMSPLEX	Required	Specifies the name of the IMSplex group that share databases and message queues for the production IMS systems.	“IMSPLEX keyword” on page 1060
IMSPLEX_APPLY	Optional	Specifies the name of the IMSplex group that share databases and message queues for the IMS DBCTL system.	“IMSPLEX_APPLY keyword” on page 1060
IMSRESLB_EDC_DSN	Optional	Specifies the name of the library that contains the edit/compression routines that the active ACB member specifies.	“IMSRESLB_EDC_DSN keyword” on page 1060
IMSRESLB_RND_DSN	Required	Specifies the name of the library that contains the randomizing routine that the active ACB member specifies.	“IMSRESLB_RND_DSN keyword” on page 1061
INTRDR	Required	Specifies the member name of the internal reader program in the PROCLIB library.	“INTRDR keyword” on page 1061
JCLDSN	Required	Specifies the name of the OSC internal job data set in which the utility generates OSC internal jobs.	“JCLDSN keyword” on page 1061
JOBCARDx	Optional	Specifies JOB statements for the OSC internal jobs.	“JOBCARDx keyword” on page 1062
JOBN_PREFIX	Optional	Specifies the prefix for OSC internal job names.	“JOBN_PREFIX keyword” on page 1062
LOGDSN	Required	Specifies the name prefix for the OSC internal job log data set.	“LOGDSN keyword” on page 1063
MCLASS_TEMP	Optional	Specifies the SMS management class for the temporary data sets that the Online Structure Change utility dynamically allocates.	“MCLASS_TEMP keyword” on page 1063
NBA_APPLY	Optional	Specifies the IMS DBCTL Fast Path database buffers to be made available for the apply job.	“NBA_APPLY keyword” on page 1064
NEWDBDLIB_DSN	Required	Specifies the name of the new DBDLIB library that contains the database description for the restructured DEDB.	“NEWDBDLIB_DSN keyword” on page 1064
NEWRESLB_EDC_DSN	Optional	Specifies the name of the library that contains the edit/compression routines that the new ACB member specifies.	“NEWRESLB_EDC_DSN keyword” on page 1064
NEWRESLB_RND_DSN	Required	Specifies the name of the library that contains the randomizer that the new ACB member specifies.	“NEWRESLB_RND_DSN keyword” on page 1065
OBA_APPLY	Optional	Specifies, for the apply job, additional IMS DBCTL page-fixed buffers to be made available when the normal allotment is used up.	“OBA_APPLY keyword” on page 1065
PROCLIB_DSN	Required	Specifies the name of the PROCLIB library that the production IMS systems use.	“PROCLIB_DSN keyword” on page 1066
PROCLIB_DSN_APPLY	Required	Specifies the name of the IMS DBCTL PROCLIB library. This library is used during the apply job.	“PROCLIB_DSN_APPLY keyword” on page 1066
PSBNAME_APPLY	Required	Specifies the name of the PSB that is used for the apply job.	“PSBNAME_APPLY keyword” on page 1066
RECONx_DSN	Required	Specifies the names of RECON data sets that the production IMS systems use.	“RECONx_DSN keyword” on page 1067
RECONx_DSN_APPLY	Required	Specifies the names of IMS DBCTL RECONx data sets for the apply job.	“RECONx_DSN_APPLY keyword” on page 1067

Table 128. JCLPARM subcommand keywords (continued)

Keyword	Required or optional	Description	Topic
SCLASS_TEMP	Optional	Specifies the SMS storage class for the temporary data sets that the Online Structure Change utility dynamically allocates.	“SCLASS_TEMP keyword” on page 1067
SGLXLOAD	Optional	Specifies the name of the IMS Tools Base load module library.	“SGLXLOAD keyword” on page 1068
SPACE	Optional	Specifies the space requirements for the output data sets that the Online Structure Change utility dynamically allocates to hold unloaded segment data records.	“SPACE keyword” on page 1068
SPOC_WAITTIME	Optional	Specifies the wait time that applies to each of the commands in the batch SPOC utility SYSIN file.	“SPOC_WAITTIME keyword” on page 1069
SYSAFF	Required	Specifies the JES2 member that is eligible to process the job (representing system affinity).	“SYSAFF keyword” on page 1069
SYSAFF_APPLY	Optional	Species the JES2 member that is eligible to process the apply job (representing system affinity).	“SYSAFF_APPLY keyword” on page 1070
TOIXCFGRP_APPLY	Required	Specifies the XCF group that IMS Tools Online System Interface uses to interface with the apply job.	“TOIXCFGRP_APPLY keyword” on page 1070
TOSILOAD	Required	Specifies the name of the load module library for IMS Tools Online System Interface (TOSI).	“TOSILOAD keyword” on page 1070
UNIT_TEMP	Optional	Specifies the physical device type for the temporary data sets that the Online Structure Change utility dynamically allocates.	“UNIT_TEMP keyword” on page 1071
USRHLQ	Required	Specifies the high-level qualifier for the output data sets in which the Online Structure Change utility generates unloaded segment data records.	“USRHLQ keyword” on page 1071

LOADCTL subcommand keywords

The LOADCTL subcommand is an optional subcommand. The LOAD subcommand specifies the options for loading segments into the restructured ADSs.

For the LOADCTL subcommand, you can specify the keywords that are listed in the following table. For the format of the LOADCTL command, see [“LOADCTL subcommand keywords” on page 1071](#).

Table 129. LOADCTL subcommand keywords

Keyword	Required or optional	Description	Topic
INSERT_LIMIT_COUNT	Optional	Specifies the maximum number of segments that can be inserted close to their root segment.	“INSERT_LIMIT_COUNT keyword” on page 1072
LOCATION	Optional	Specifies the location to insert dependent segments, either DOVF or IOVF.	“LOCATION keyword” on page 1072
SEGMENT	Required	Specifies the name of the segment that should be loaded to a specific location.	“SEGMENT keyword” on page 1073

Topics:

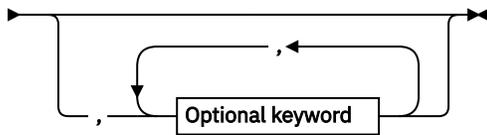
- [“GLOBAL command keywords” on page 1036](#)
- [“RESTRUCTURE command keywords” on page 1039](#)
- [“ALLOCATE subcommand keywords” on page 1047](#)
- [“JCLPARM subcommand keywords” on page 1049](#)
- [“LOADCTL subcommand keywords” on page 1071](#)

GLOBAL command keywords

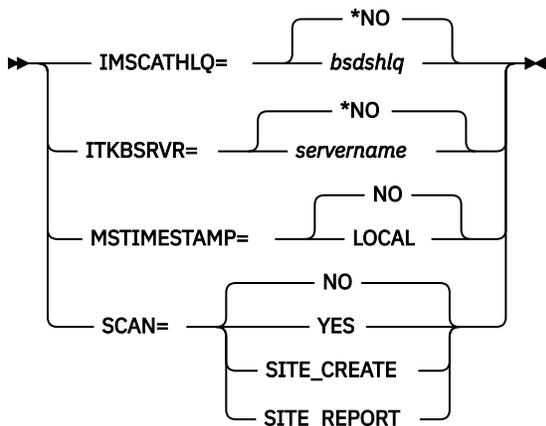
Use the GLOBAL command to establish implicit job step keyword values. GLOBAL is a required command for the HFPSYSIN DD statement. It must be the first command in the input stream.

You can specify the following keywords for the GLOBAL command.

►► GLOBAL — IMSID= *ims_id* — , — TOIXCFGRP= *TOI_XCF_group_name* —►



Optional keywords:

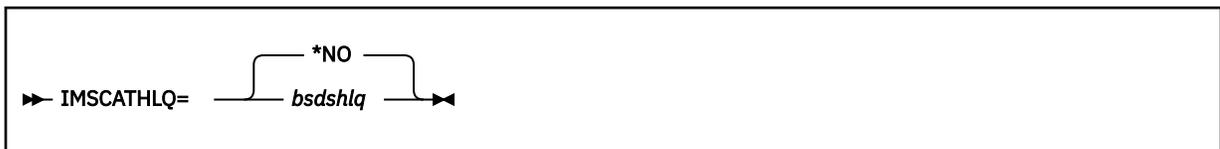


IMSCATHLQ keyword

Use the IMSCATHLQ keyword to specify the high-level qualifier of the bootstrap data set of the IMS directory.

IMSCATHLQ is an optional keyword for the GLOBAL command.

Format:



bsdshlq

Reads the ACB from the IMS directory instead of the ACB library by using IMS Tools Catalog Interface. *bsdshlq* specifies the high-level qualifier of the bootstrap data set of the IMS directory.

***NO**

Reads the ACB from the ACB library.

Default:

IMSCATHLQ=*NO

Site default support:

Not supported.

IMSID keyword

Use the IMSID keyword to specify the production IMS system ID. The OSC process runs on the IMS system that is identified by this ID.

IMSID is a required keyword for the GLOBAL command.

Format:

```
►► IMSID= ims_id ◄◄
```

ims_id

Specify a 1- to 4-byte IMS system ID.

Default:

None.

Site default support:

Not supported.

ITKBSRVR keyword

Use the ITKBSRVR keyword to specify the name of the IMS Tools KB server XCF group.

ITKBSRVR is an optional keyword for the GLOBAL command.

Format:

```
►► ITKBSRVR= servername ◄◄  
                  *NO
```

servername

Store reports in the repository of the specified IMS Tools KB server.

*NO

Do not store reports in the repository of IMS Tools KB.

Default:

ITKBSRVR=*NO

Site default support:

Supported.

MSTIMESTAMP keyword

Use the MSTIMESTAMP keyword to specify whether to include a timestamp in the messages.

MSTIMESTAMP is an optional keyword for the GLOBAL command.

Format:

```
►► MSTIMESTAMP= LOCAL ◄◄  
                  NO
```

LOCAL

Include a local timestamp in the messages.

NO

Do not include a timestamp in the messages.

Default:

MSTIMESTAMP=NO

Site default support:

Supported.

SCAN keyword

Use the SCAN keyword to verify the syntax of command, keywords, and parameters without running OSC functions.

The Scan function also allocates some (but not all) internal resources for running the process function.

SCAN is an optional keyword for the GLOBAL command.

Format:**NO**

Verify the syntax and parameters of the command and execute OSC processes.

YES

Verify the syntax and parameters of the command but do not execute OSC processes.

SITE_CREATE

Create a site default table. More than one command, their related subcommands, and their keywords can be specified to define site default values for the keywords. This parameter specifies that the command analysis is to be run, but does not get access to DBRC.

SITE_REPORT

Report on the site default table. The commands, subcommands, and keywords are ignored. This parameter does not get access to DBRC.

Default:

SCAN=NO

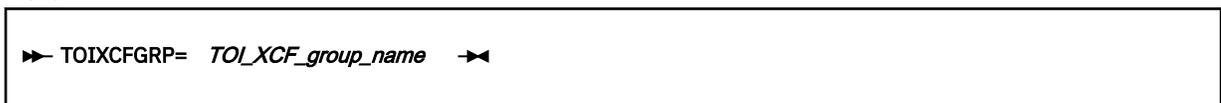
Site default support:

Not supported.

TOIXCFGRP keyword

Use the TOIXCFGRP keyword to specify the XCF group that IMS Tools Online System Interface uses to interface with the Online Structure Change utility.

TOIXCFGRP is a required keyword for the GLOBAL command.

Format:***TOI_XCF_group_name***

Specify a 1- to 5-alphanumeric character to generate the XCF group name. IMS Tools Online System Interface prefixes this string with characters "TOI" to generate the XCF group name.

Default:

None.

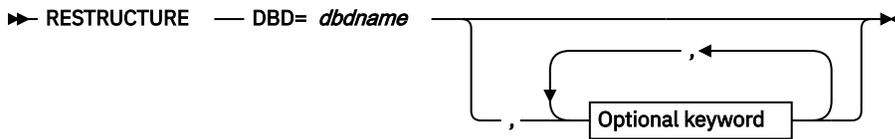
Site default support:

Supported.

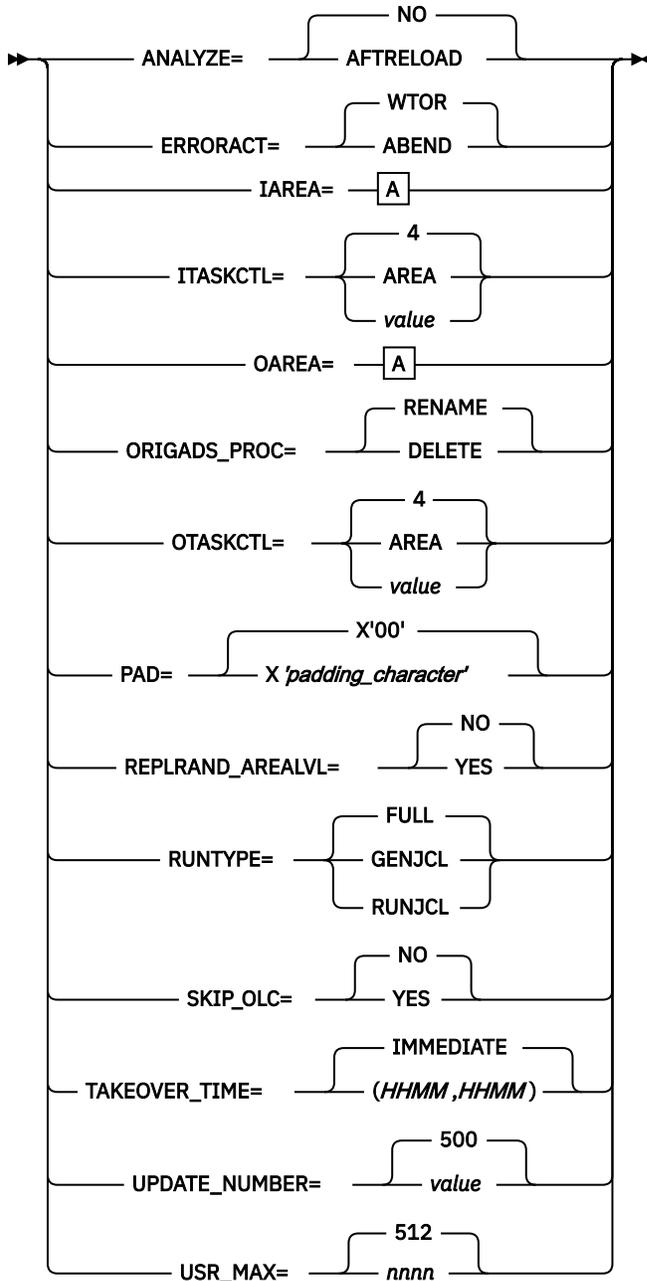
RESTRUCTURE command keywords

Use the RESTRUCTURE command to specify how you want to change the structure of DEDB areas. RESTRUCTURE is a required command for the HFPSYSIN DD statement.

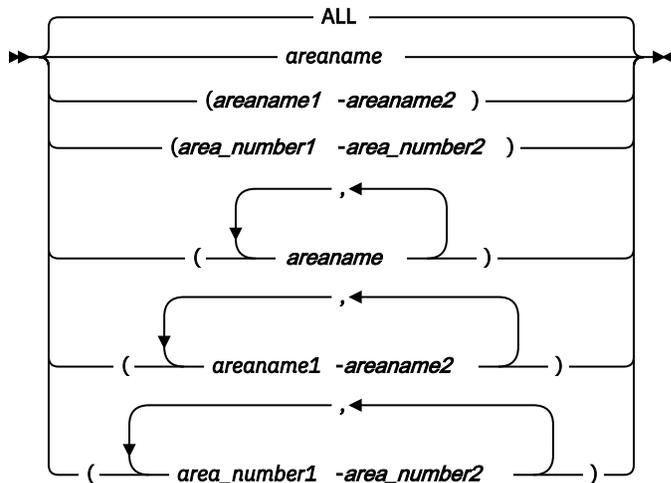
You can specify the following keywords for the RESTRUCTURE command.



Optional keywords:



A:



ANALYZE keyword

Use the ANALYZE keyword to analyze ADSs.

ANALYZE is an optional keyword for the RESTRUCTURE command.

Format:



AFTRELOAD

Analyze the restructured ADSs after the restructured ADSs are reloaded.

NO

Do not analyze ADSs.

Default:

ANALYZE=NO

Site default support:

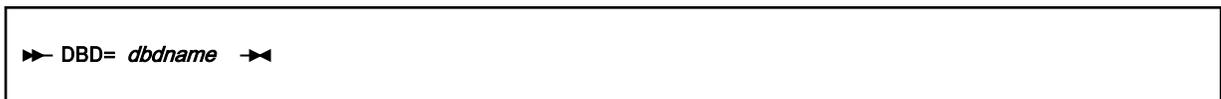
Supported.

DBD keyword

Use the DBD keyword to specify the database description (DBD) that contains the areas to restructure.

DBD is a required keyword for the RESTRUCTURE command.

Format:



dbdname

Specify a 1- to 8-character database name.

Default:

None.

Site default support:

Not supported.

ERRORACT keyword

Use the ERRORACT keyword to specify the action for the utility to take at certain error conditions. ERRORACT is an optional keyword for the RESTRUCTURE command.

Format:



WTOR

Issue message HFPT0512A on WTOR to prompt the operator for the next action to take.

ABEND

End the processing with a user abend code.

Default:

ERRORACT=WTOR

Site default support:

Supported.

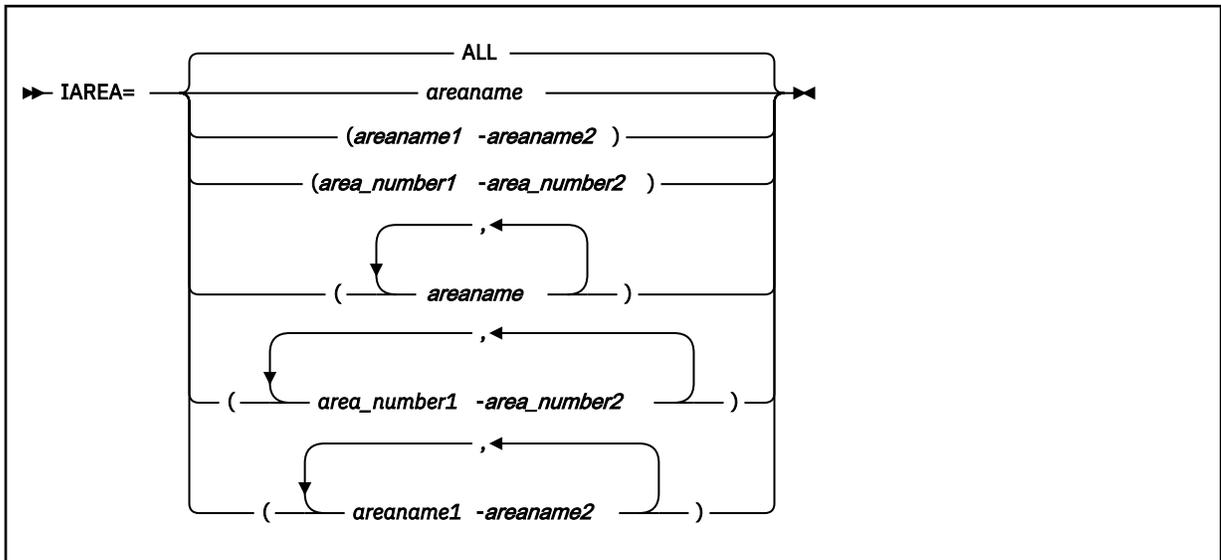
IAREA keyword

Use the IAREA keyword to specify one or more input areas to restructure.

IAREA is an optional keyword for the RESTRUCTURE command.

For some examples of how to specify the IAREA keyword, see [“Example: Specifying input and output areas”](#) on page 1086.

Format:



ALL

Process all areas.

areaname

Specify a 1- to 8-character area name.

You can use certain wild card characters to specify areas. Supported wild card characters are the asterisk (*), which represents from 0 to 8 characters, and the percent sign (%), which represents any single character.

area_number1-area_number2

Specify the range of areas by area numbers.

area_number1 and *area_number2* are area numbers, which are decimal numbers assigned sequentially to the areas within a database. The value for *area_number1* and *area_number2* can be in the range of 1 - 2048. *area_number2* must be equal to or greater than *area_number1*. At least one area must exist in the specified range.

areaname1-areaname2

Specify the range of areas by area names.

areaname1 and *areaname2* must be specified in alphabetical order. At least one area must exist in the specified range.

Default:

IAREA=ALL

Site default support:

Not supported.

ITASKCTL keyword

Use the ITASKCTL keyword to specify the maximum number of input areas to process concurrently.

ITASKCTL is an optional keyword for the RESTRUCTURE command.

Format:



AREA

The number of input areas.

value

Specify a value in the range of 1 - 2048. Specify a value less than or equal to the number of input areas. If you specify a value that is greater than the number of input areas, the value is replaced by the number of input areas.

Default:

ITASKCTL=4

Site default support:

Supported.

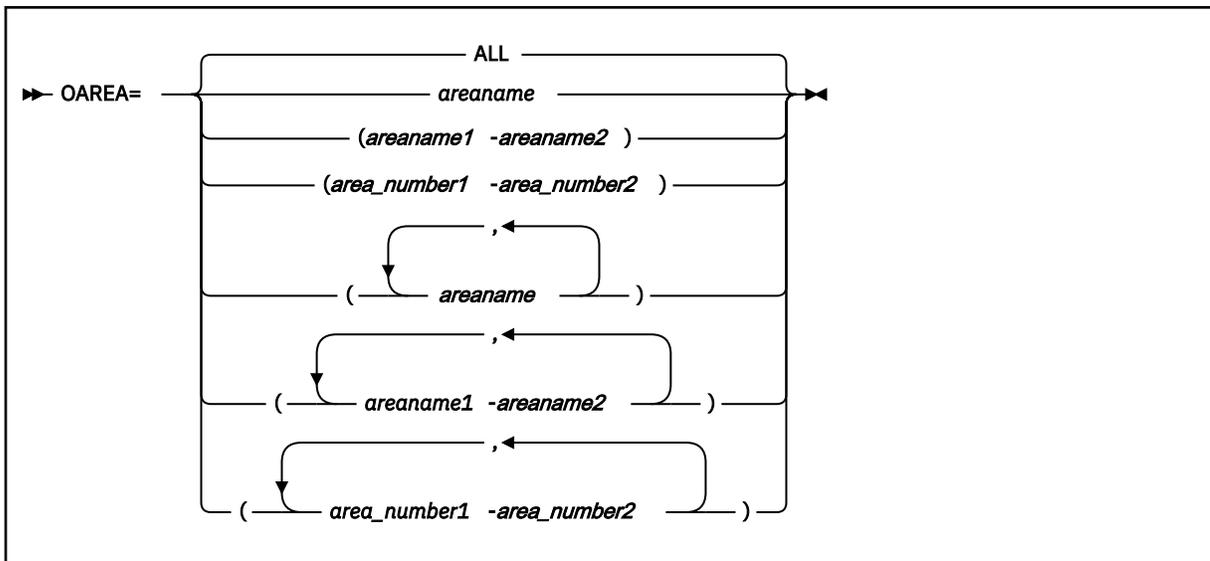
OAREA keyword

Use the OAREA keyword to specify one or more output areas.

OAREA is an optional keyword for the RESTRUCTURE command.

For some examples of how to specify the OAREA keyword, see [“Example: Specifying input and output areas”](#) on page 1086.

Format:



ALL

Use all areas of the DEDB as output areas.

areaname

Specify a 1- to 8-character area name.

You can use certain wild card characters to specify areas. Supported wild card characters are the asterisk (*), which represents from 0 to 8 characters, and the percent sign (%), which represents any single character.

area_number1-area_number2

Specify the range of areas by area numbers.

area_number1 and area_number2 are area numbers, which are decimal numbers assigned sequentially to the areas within a database. The value for area_number1 and area_number2 can be in the range of 1 - 2048. area_number2 must be equal to or greater than area_number1. At least one area must exist in the specified range.

areaname1-areaname2

Specify the range of areas by area names.

areaname1 and areaname2 must be specified in alphabetical order. At least one area must exist in the specified range.

Default:

OAREA=ALL

Site default support:

Not supported.

ORIGADS_PROC keyword

Use the ORIGADS_PROC keyword to specify whether to delete or retain the original area data sets (ADSs). ORIGADS_PROC is an optional keyword for the RESTRUCTURE command.

Format:



RENAME

Retain the original ADSs. During the takeover phase, each original ADS is renamed by adding suffix ".T". If MADS is used, all the copies of the MADS are renamed.

DELETE

Delete the original ADSs. Deletion is done during the takeover phase.

Usage note:

If you specify ORIGADS_PROC=RENAME, make sure that you delete the renamed original ADSs (ADSs with ".T" suffix) before you run the Online Structure Change utility again. If the renamed ADSs from an earlier OSC job remain and the utility tries to rename ADSs to the names of the existing ADSs, the job fails.

Default:

ORIGADS_PROC=RENAME

Site default support:

Supported.

OTASKCTL keyword

Use the OTASKCTL keyword to specify the maximum number of output areas to process concurrently.

OTASKCTL is an optional keyword for the RESTRUCTURE command.

Format:



AREA

The number of output areas.

value

Specify a value in the range of 1 - 2048. The value must be less than or equal to the number of output areas. If you specify a value that is greater than the number of output areas, it will be replaced by the number of output areas. This value is set to the ITASKCTL keyword of the OSC internal reload job.

Default:

OTASKCTL=4

Site default support:

Supported.

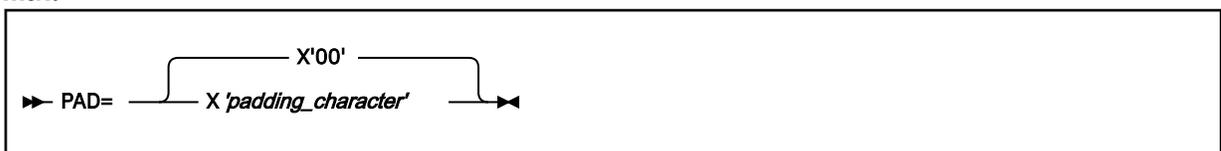
PAD keyword

Use the PAD keyword to specify the padding character to be used when the segment minimum length is increased due to DBD description change.

The PAD keyword is effective for fixed-length segments and variable-length segments.

PAD is an optional keyword for the RESTRUCTURE command.

Format:



padding_character

Two hexadecimal digits, prefixed by X and enclosed in single quotation marks ('). For example, X'01'.

Default:

PAD=X'00'

Site default support:

Supported.

REPLRAND_AREALVL keyword

Use the REPLRAND_AREALVL keyword to issue the TOSI REPLRAND command to replace the randomizer. Only certain areas are stopped while the randomizer is replaced.

REPLRAND_AREALVL is an optional keyword for the RESTRUCTURE command.

Format:



YES

Issue the TOSI REPLRAND command.

NO

Do not issue the TOSI REPLRAND command.

Default:

REPLRAND_AREALVL=NO

Site default support:

Not supported.

RUNTYPE keyword

Use the RUNTYPE keyword to specify the operating mode for the Online Structure Change utility job. Three run mode types are supported; generate and execute OSC internal jobs, only generate OSC internal jobs, and execute pre-generated OSC internal jobs.

For more information about operating modes, see [“OSC jobs and operating modes” on page 1009](#).

RUNTYPE is an optional keyword for the RESTRUCTURE command.

Format:



FULL

Generate JCL for OSC internal jobs and execute them.

GENJCL

Generate JCL for OSC internal jobs but do not execute them.

RUNJCL

Do not generate JCL for OSC internal jobs but execute OSC internal jobs by submitting OSC internal jobs that were generated in advance with the GENJCL mode.

Default:

RUNTYPE=FULL

Site default support:

Not supported.

SKIP_OLC keyword

Use the SKIP_OLC keyword to skip online change.



Attention: Use this keyword with caution. If there is no change to the DBD definition and edit/compression routines, you can use this keyword to skip ACB online change. However, if you specify this keyword when DBD definition or edit/compression routines are changed, ACB online change is skipped and may break the database.

SKIP_OLC is an optional keyword for the RESTRUCTURE command.

Format:**YES**

Skip ACB online change.

NO

Do not skip ACB online change.

Default:

SKIP_OLC=NO

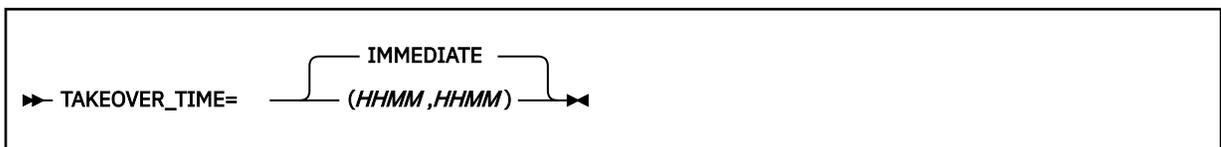
Site default support:

Not supported.

TAKEOVER_TIME keyword

Use the TAKEOVER_TIME keyword to specify when the takeover phase should start.

TAKEOVER_TIME is an optional keyword for the RESTRUCTURE command.

Format:**IMMEDIATE**

Takeover phase starts without delay.

HHMM

Specify the time to perform the takeover phase. For the first *HHMM*, specify the start time. For the second *HHMM*, specify the end time. The values must be in 24-hour format, in the range of 0000 to 2359.

Default:

TAKEOVER_TIME=IMMEDIATE

Site default support:

Not supported.

UPDATE_NUMBER keyword

Use the UPDATE_NUMBER keyword to specify the threshold for the number of online updates.

/DBR command is issued against the online DEDB or online areas if the number of captured online updates is below the threshold value.

UPDATE_NUMBER is an optional keyword for the RESTRUCTURE command.

Format:



value

Specify the threshold value for the number of online updates. The value must be in the range of 100 - 1000.

Default:

UPDATE_NUMBER=500

Site default support:

Supported.

USR_MAX keyword

Use the USR_MAX keyword to limit the maximum number of USR (unloaded segment records) data sets that the Online Structure Change utility generates during the unload phase.

USR_MAX is an optional keyword for the RESTRUCTURE command.

Format:



nnnn

Specifies the maximum number of USR data sets. The value must be in the range of 1 - 9999.

Default:

USR_MAX=512

Site default support:

Supported.

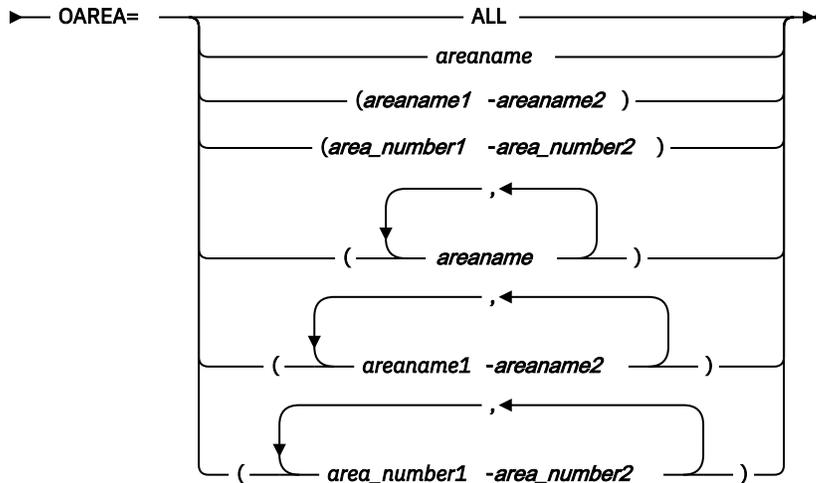
ALLOCATE subcommand keywords

Use the ALLOCATE subcommand to provide the Online Structure Change utility the information that is required to allocate restructured ADSs. ALLOCATE is a required subcommand for the HFPSYSIN DD statement.

The ATTRIBUTE keyword of the ALLOCATE subcommand specifies the data set that contains VSAM cluster information that the Online Structure Change utility uses to dynamically define VSAM clusters.

You can specify the following keywords for the ALLOCATE subcommand.

►► ALLOCATE — ATTRIBUTE=(IDCAMS,DSN= 'dsname') — , —►



ATTRIBUTE keyword

Use the ATTRIBUTE keyword to specify the data set to pass on to Access Method Services for allocating new VSAM clusters.

Requirement: A sequential or partitioned data set must be preallocated before the OSC job is submitted. For more information about creating this data set, see [“Example: Specifying the ALLOCATE subcommand” on page 1087.](#)

ATTRIBUTE is a required keyword for the ALLOCATE subcommand.

Format:

```
>> -----ATTRIBUTE=(IDCAMS,DSN='dsname')-----><
```

IDCAMS,DSN='dsname'

A sequential or partitioned data set is passed to Access Method Services as input for allocating the new VSAM cluster. Whether the new VSAM cluster is allocated or not is verified.

Enclose the data set name in single quotation marks (').

If the OSC job processes multiple areas, the parameter must be specified as follows:
IDCAMS,DSN=' dsname (&AREA) '. Where &AREA is the mask variable for area name.

Default:

None.

Site default support:

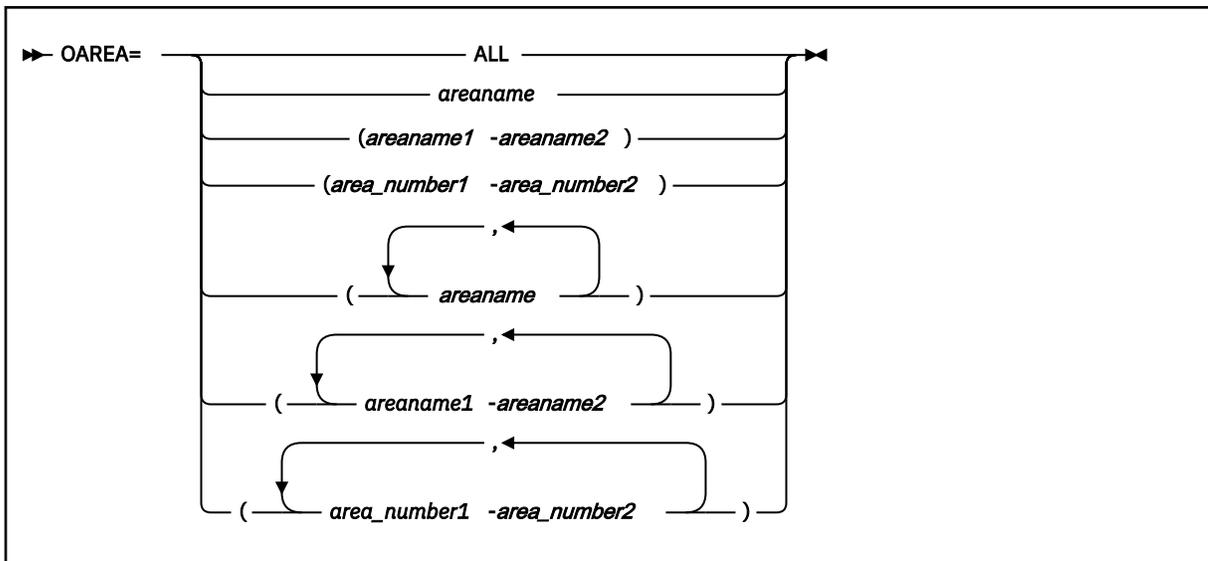
Not supported.

OAREA keyword

Use the OAREA keyword to specify one or more output areas for the process. The Online Structure Change utility allocates shadows ADSs for the areas that the OAREA keyword specifies.

OAREA is a required keyword for the ALLOCATE subcommand.

Format:



ALL

Use all areas of the DEDB as output areas.

areaname

Specify a 1- to 8-character area name.

You can also use certain wild card characters to specify areas. Supported wild card characters are the asterisk (*), which represents from 0 to 8 characters, and the percent sign (%), which represents any single character.

area_number1-area_number2

Specify the range of areas by area numbers.

area_number1 and *area_number2* are area numbers, which are decimal numbers assigned sequentially to the areas within a database. The value for *area_number1* and *area_number2* can be in the range of 1 - 2048. *area_number2* must be equal to or greater than *area_number1*. At least one area must exist in the specified range.

areaname1-areaname2

Specify the range of areas by area names.

areaname1 and *areaname2* must be specified in alphabetical order. At least one area must exist in the specified range.

Default:

None.

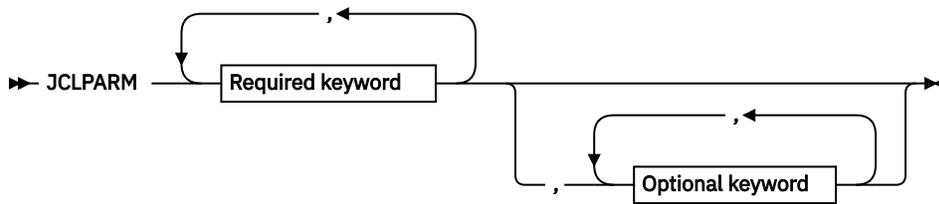
Site default support:

Not supported.

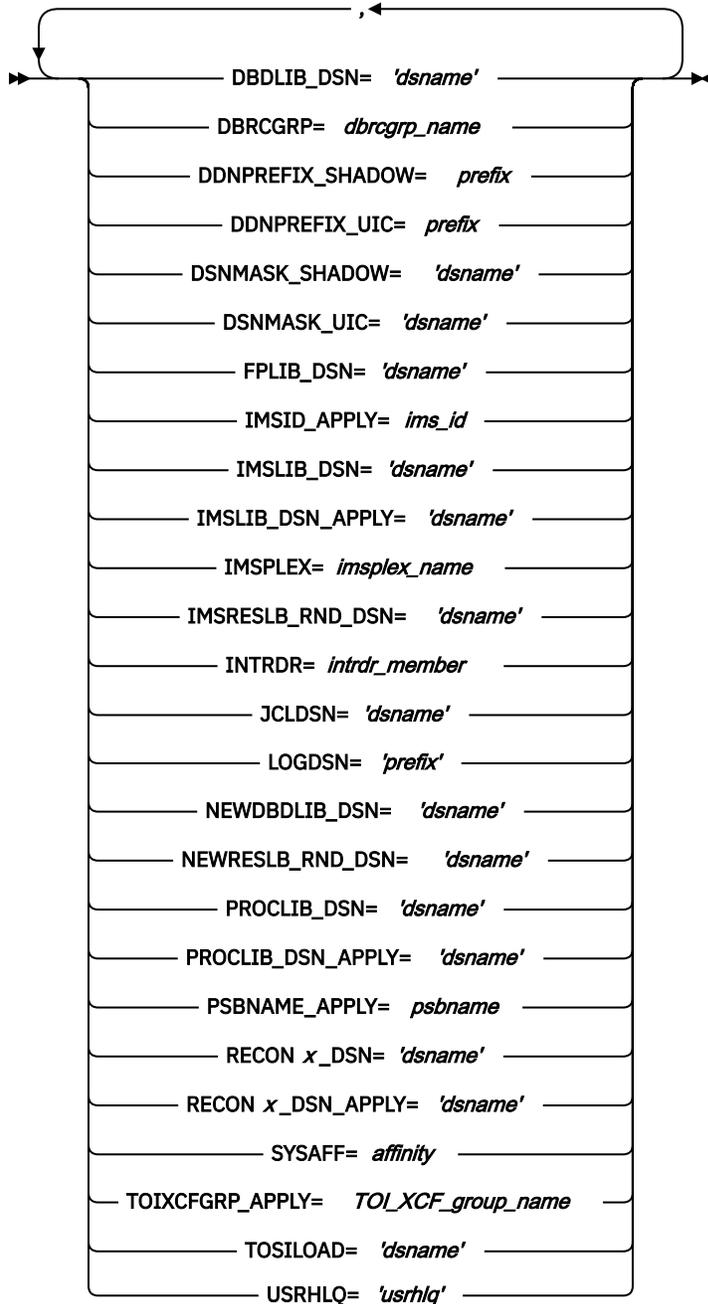
JCLPARM subcommand keywords

Use the JCLPARM subcommand to specify the options to be applied to OSC internal jobs. JCLPARM is a required subcommand for the HFPSYSIN DD statement.

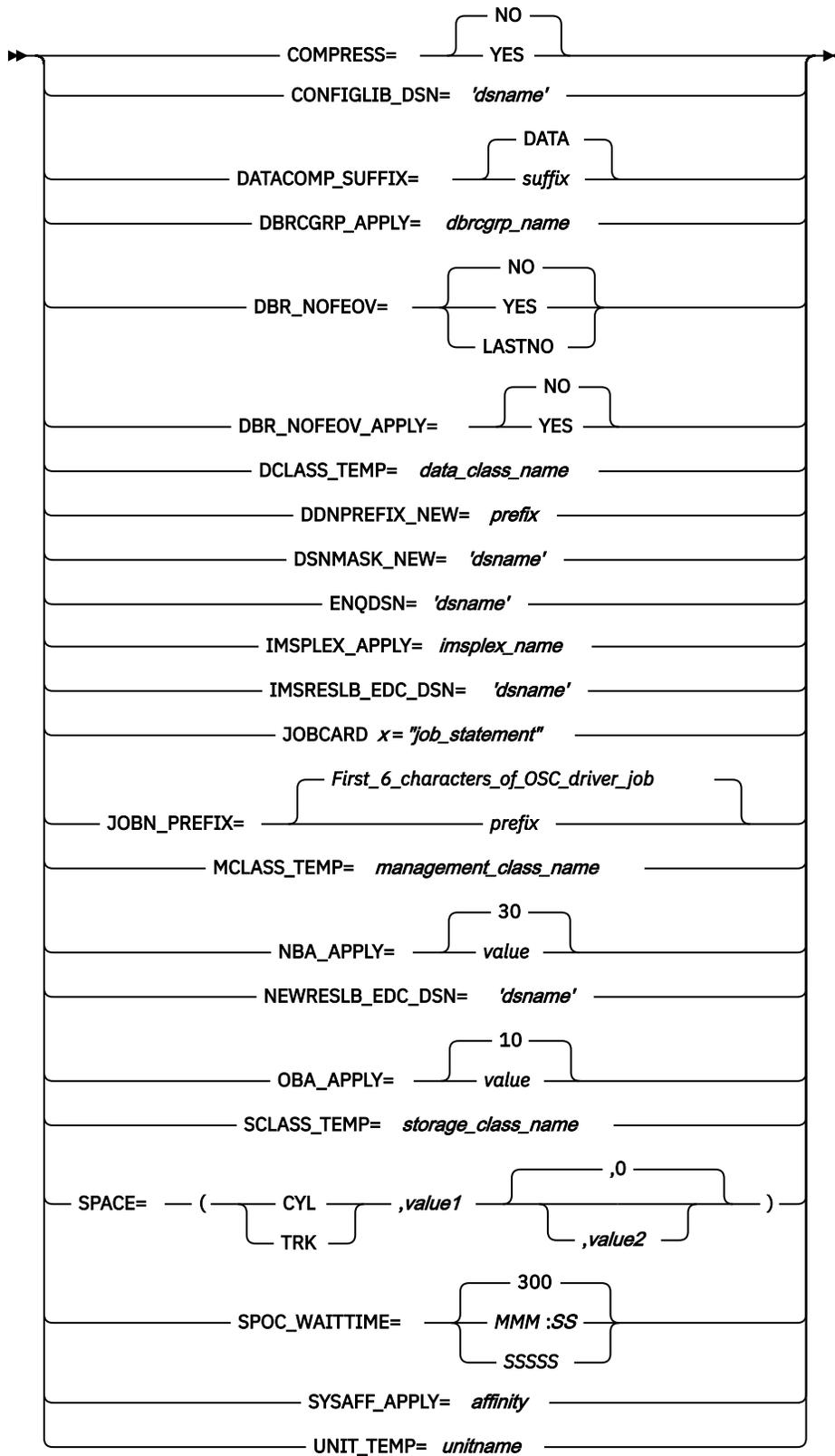
You can specify the following keywords for the JCLPARM subcommand.



Required keywords:



Optional keywords:



COMPRESS keyword

Use the COMPRESS keyword to specify that the output records in the unloaded files contain compressed segments.

Compression is done only when a segment edit/compression routine is defined for the segments during DBDGEN. If the COMPRTN operand is not specified for the SEGM statement of the DBD statement, this keyword is ignored.

COMPRESS is an optional keyword for the JCLPARM subcommand.

Format:



YES

Output records contain compressed segments.

NO

Output records contain uncompressed segments.

Default:

COMPRESS=NO

Site default support:

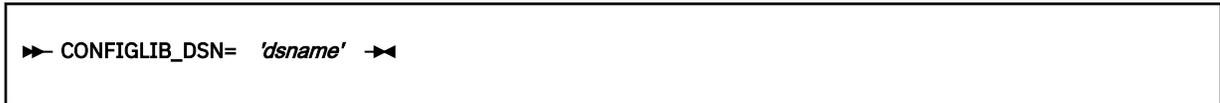
Supported.

CONFIGLIB_DSN keyword

Use the CONFIGLIB_DSN keyword to specify the library that contains configuration module HFP#*imsid* or HFPSETOP.

CONFIGLIB_DSN is an optional keyword for the JCLPARM subcommand.

Format:



dsname

Specify a 1- to 44-character data set name. Enclose the data set name in single quotation marks (').

Default:

None.

Site default support:

Not supported.

DATACOMP_SUFFIX keyword

Use the DATACOMP_SUFFIX keyword to specify the suffix portion of the VSAM data component name.

DATACOMP_SUFFIX is an optional keyword for the JCLPARM subcommand.

The suffix you specify for the DATACOMP_SUFFIX keyword is used to determine the data component name for shadow ADSs and UICs. The data component name is *cluster_name* + *suffix*. For shadow ADSs, the cluster name is determined by the DSNMASK_SHADOW keyword. For UICs, the cluster name is determined by the DSNMASK_UIC keyword.

For restructured ADSs:

- If the DATA parameter is specified in the IDCAMS control statement, the data component name must be *cluster_name* + *suffix* and the suffix portion must match the DATACOMP_SUFFIX keyword value.

- If the DATA parameter is not specified in the IDCAMS control statement, if DATACOMP_SUFFIX=DATA, the cluster name of the restructured ADSs must be less than or equal to 38 characters. If DATACOMP_SUFFIX=D, the cluster name for the restructured ADS must be 39 to 42 characters.

For more information, see [“VSAM cluster names for area data sets \(ADSs\) and user image copies \(UICs\)”](#) on page 1016.

Format:



suffix

Specify a 1- to 8-character suffix for the VSAM data component name.

Default:

DATACOMP_SUFFIX=DATA

Site default support:

Supported.

DBDLIB_DSN keyword

Use the DBDLIB_DSN keyword to specify the name of the active DBDLIB library.

DBDLIB_DSN is a required keyword for the JCLPARM subcommand.

Format:



dsname

Specify a 1- to 44-character data set name. Enclose the data set name in single quotation marks (').

Default:

None.

Site default support:

Not supported.

DBRCGRP keyword

Use the DBRCGRP keyword to specify the identifier that is assigned to a group of DBRC instances (DBRC sharing group) that access the same RECON data set in the IMSplex. This keyword is for the production IMS systems.

DBRCGRP is a required keyword for the JCLPARM subcommand.

Format:



dbrcgrp_name

Specify a 1- to 3-character DBRC group name.

Default:

None.

Site default support:

Not supported.

DBRCGRP_APPLY keyword

Use the DBRCGRP_APPLY keyword to specify the identifier that is assigned to a group of DBRC instances (DBRC sharing group) that access the same RECON data set in the IMSplex. This keyword is for the IMS DBCTL system.

DBRCGRP_APPLY is an optional keyword for the JCLPARM subcommand.

Format:

```
➤➤ DBRCGRP_APPLY= dbrcgrp_name ➤➤
```

dbrcgrp_name

Specify a 1- to 3-character DBRC group name.

Default:

None.

Site default support:

Not supported.

DBR_NOFEOV keyword

Use the DBR_NOFEOV keyword to add the NOFEOV keyword to the /DBRECOVERY command that is issued to stop the DEDB or DEDB areas in the production IMS systems.

DBR_NOFEOV is an optional keyword for the JCLPARM subcommand.

Format:

```
➤➤ DBR_NOFEOV= { NO | YES | LASTNO } ➤➤
```

YES

Specify that there is no forced end of volume, so that the log does not switch to the next OLDS and simple checkpoint is not taken.

NO

Specify that the system assumes an end-of-volume condition, so that the log switches automatically. This switch to the next OLDS is marked as a recovery point for log archiving purposes. IMS issues a simple checkpoint after the active log data set has been switched to the next OLDS.

LASTNO

Specify that the log switches automatically only after the last area is processed. This option is effective for area-level structure changes.

Default:

DBR_NOFEOV=NO

Site default support:

Supported.

DBR_NOFEOV_APPLY keyword

Use the DBR_NOFEOV_APPLY keyword to add the NOFEOV keyword to the TOSI DBRSYNC command, which is issued to stop the DEDB or DEDB areas in the IMS DBCTL system.

DBR_NOFEOV_APPLY is an optional keyword for the JCLPARM subcommand.

Format:**YES**

Specify that there is no forced end of volume, so that the log does not switch to the next OLDS and simple checkpoint is not taken.

NO

Specify that the system assumes an end-of-volume condition, so that the log switches automatically. This switch to the next OLDS is marked as a recovery point for log archiving purposes. IMS issues a simple checkpoint after the active log data set has been switched to the next OLDS.

Default:

DBR_NOFEOV_APPLY=NO

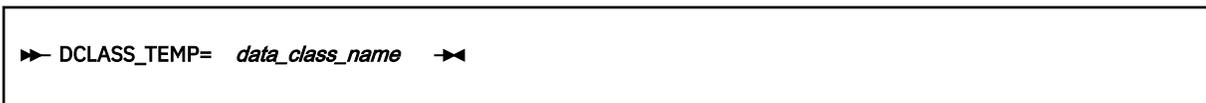
Site default support:

Supported.

DCLASS_TEMP keyword

Use the DCLASS_TEMP keyword to specify the SMS data class for the temporary data sets that the Online Structure Change utility dynamically allocates.

DCLASS_TEMP is an optional keyword for the JCLPARM subcommand.

Format:***data_class_name***

Specify a 1- to 8-character data class name.

Default:

None.

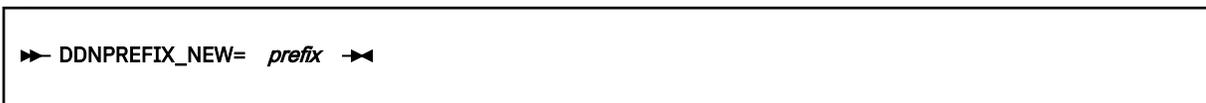
Site default support:

Not supported.

DDNPREFIX_NEW keyword

Use the DDNPREFIX_NEW keyword to specify the DD name prefix for the new area data sets that the Online Structure Change utility adds to the production environment. Area numbers (4 digits) are appended to form the DD names.

DDNPREFIX_NEW is an optional keyword for the JCLPARM subcommand. However, you must specify this keyword when adding new areas.

Format:***data_class_name***

Specify a 1- to 4-character prefix for the DD names.

Default:

None.

Site default support:

Not supported.

DDNPREFIX_SHADOW keyword

Use the DDNPREFIX_SHADOW keyword to specify the DD name prefix for the shadow area data sets and restructured area data sets that the Online Structure Change utility allocates. Area numbers (4 digits) are appended to form the DD names.

DDNPREFIX_SHADOW is a required keyword for the JCLPARM subcommand.

Format:

```
▶▶ DDNPREFIX_SHADOW= prefix ▶▶
```

prefix

Specify a 1- to 4-character prefix for the DD names.

Default:

None.

Site default support:

Not supported.

DDNPREFIX_UIC keyword

Use the DDNPREFIX_UIC keyword to specify the DD name prefix for the user image copy data sets that the Online Structure Change utility allocates. Area numbers (4 digits) are appended to form the DD names.

DDNPREFIX_UIC is a required keyword for the JCLPARM subcommand.

Format:

```
▶▶ DDNPREFIX_UIC= prefix ▶▶
```

prefix

Specify a 1- to 4-character prefix for the DD names.

Default:

None.

Site default support:

Not supported.

DSNMASK_NEW keyword

Use the DSNMASK_NEW keyword to specify the data set name or the mask for the new area data sets that the Online Structure Change utility adds to the production environment.

DSNMASK_NEW is an optional keyword for the JCLPARM subcommand. However, you must specify this keyword when adding new areas.

Format:

```
▶▶ DSNMASK_NEW= 'dsname' ▶▶
```

dsname

Specify a data set name or a data set mask. Enclose the data set name in single quotation marks ('').

Use the following variables, as necessary, to create the mask:

&DBD

DBD name

&AREA

Area name

&AREA#

3-digit relative area number.

&AREAn#*n*-digit relative area number. *n* is 1 - 5.**Default:**

None.

Site default support:

Not supported.

DSNMASK_SHADOW keyword

Use the DSNMASK_SHADOW keyword to specify the data set name or the mask for the shadow area data sets and restructured area data sets that the Online Structure Change utility dynamically allocates.

DSNMASK_SHADOW is a required keyword for the JCLPARM subcommand.

Format:

```
►► DSNMASK_SHADOW= 'dsname' ◄◄
```

dsname

Specify a data set name or a data set mask. Enclose the data set name in single quotation marks (').

Use the following variables, as necessary, to create the mask:

&DBD

DBD name

&AREA

Area name

&AREA#

3-digit relative area number.

&AREAn#*n*-digit relative area number. *n* is 1 - 5.**Default:**

None.

Site default support:

Not supported.

DSNMASK_UIC keyword

Use the DSNMASK_UIC keyword to specify the data set name or the mask for the user image copy data sets that the Online Structure Change utility dynamically allocates.

DSNMASK_UIC is a required keyword for the JCLPARM subcommand.

Format:

```
►► DSNMASK_UIC= 'dsname' ◄◄
```

dsname

Specify a data set name or a data set mask. Enclose the data set name in single quotation marks (').

Use the following variables, as necessary, to create the mask:

&DBD

DBD name

&AREA

Area name

&AREA#

3-digit relative area number.

&AREAn#*n*-digit relative area number. *n* is 1 - 5.**Default:**

None.

Site default support:

Not supported.

ENQDSN keyword

Use the ENQDSN keyword to specify the name of the ENQ data set in which the utility generates internal SYSIN files.

Requirements:

- An ENQ data set must be preallocated before you submit the OSC driver job. If multiple OSC driver jobs use the same ENQ data set, the data set must be a PDSE.
- If you plan to execute multiple OSC driver jobs simultaneously in an IMSplex, all the SYSIN files must be generated in a single ENQ data set. Ensure that the ENQDSN keywords in all OSC driver jobs specify the same ENQ data set.
- The ENQDSN keyword and the JCLDSN keyword must not specify the same data set.

ENQDSN is an optional keyword for the JCLPARM subcommand. However, you must specify the ENQDSN keyword when SKIP_OLC=NO. If SKIP_OLC=YES keyword is specified, the ENQDSN keyword is ignored.

For more information about SYSIN files generated in ENQ data sets, see [“Internal SYSIN files” on page 1013](#).

Format:

```
►► ENQDSN= 'dsname' ◄◄
```

dsname

Specify a 1- to 44-character data set name. Enclose the data set name in single quotation marks (').

Default:

None.

Site default support:

Not supported.

FPLIB_DSN keyword

Use the FPLIB_DSN keyword to specify the name of the IMS Fast Path Solution Pack SHFPLMD0 library.

FPLIB_DSN is a required keyword for the JCLPARM subcommand.

Format:

```
►► FPLIB_DSN= 'dsname' ◄◄
```

dsname

Specify a 1- to 44-character data set name. Enclose the data set name in single quotation marks (').

Default:

None.

Site default support:

Not supported.

IMSID_APPLY keyword

Use the IMSID_APPLY keyword to specify the IMS DBCTL system ID that the Online Structure Change utility uses for the apply job.

IMSID_APPLY is a required keyword for the JCLPARM subcommand.

Format:

```
▶▶ IMSID_APPLY= ims_id ◀◀
```

ims_id

Specify a 1- to 4-character IMS system ID.

Default:

None.

Site default support:

Not supported.

IMSLIB_DSN keyword

Use the IMSLIB_DSN keyword to specify the name of the production IMS SDFSRESL library.

IMSLIB_DSN is a required keyword for the JCLPARM subcommand.

Format:

```
▶▶ IMSLIB_DSN= 'dsname' ◀◀
```

dsname

Specify a 1- to 44-character data set name. Enclose the data set name in single quotation marks (').

Default:

None.

Site default support:

Not supported.

IMSLIB_DSN_APPLY keyword

Use the IMSLIB_DSN_APPLY keyword to specify the name of the IMS DBCTL SDFSRESL library that is used for the apply job.

IMSLIB_DSN_APPLY is a required keyword for the JCLPARM subcommand.

Format:

```
▶▶ IMSLIB_DSN_APPLY= 'dsname' ◀◀
```

dsname

Specify a 1- to 44-character data set name. Enclose the data set name in single quotation marks (').

Default:

None.

Site default support:

Not supported.

IMSPLEX keyword

Use the IMSPLEX keyword to specify the name of the IMSplex group that share databases and message queues for the production IMS systems.

IMSPLEX is a required keyword for the JCLPARM subcommand.

Format:

```
►► IMSPLEX= imsplex_name ◄◄
```

imsplex_name

Specify a 1- to 5-character IMSplex group name.

Default:

None.

Site default support:

Not supported.

IMSPLEX_APPLY keyword

Use the IMSPLEX keyword to specify the name of the IMSplex group that share databases and message queues for the IMS DBCTL system.

IMSPLEX_APPLY is an optional keyword for the JCLPARM subcommand.

Format:

```
►► IMSPLEX_APPLY= imsplex_name ◄◄
```

imsplex_name

Specify a 1- to 5-character IMSplex group name.

Default:

None.

Site default support:

Not supported.

IMSRESLB_EDC_DSN keyword

Use the IMSRESLB_EDC_DSN keyword to specify the name of the library that contains the edit/compression routines that the active ACB member specifies.

IMSRESLB_EDC_DSN is an optional keyword for the JCLPARM subcommand. However, if an edit/compression routine is defined, you must specify this keyword.

Format:

```
►► IMSRESLB_EDC_DSN= 'dsname' ◄◄
```

dsname

Specify a 1- to 44-character data set name. Enclose the data set name in single quotation marks (').

Default:

None.

Site default support:

Not supported.

IMSRESLB_RND_DSN keyword

Use the IMSRESLB_RND_DSN keyword to specify the name of the library that contains the randomizing routine that the active ACB member specifies.

IMSRESLB_RND_DSN is a required keyword for the JCLPARM subcommand.

Format:

```
►► IMSRESLB_RND_DSN= 'dsname' ◄◄
```

dsname

Specify a 1- to 44-character data set name. Enclose the data set name in single quotation marks (').

Default:

None.

Site default support:

Not supported.

INTRDR keyword

Use the INTRDR keyword to specify the member name of the internal reader program in the PROCLIB library.

INTRDR is a required keyword for the JCLPARM subcommand.

Format:

```
►► INTRDR= intrdr_member ◄◄
```

intrdr_member

Specify a 1- to 8-character member name.

Default:

None.

Site default support:

Not supported.

JCLDSN keyword

Use the JCLDSN keyword to specify the name of the OSC internal job data set in which the utility generates OSC internal jobs.

Requirement: An OSC internal job data set must be preallocated before you submit the OSC driver job. If multiple OSC driver jobs use the same OSC internal job data set, the data set must be a PDSE.

JCLDSN is a required keyword for the JCLPARM subcommand.

Format:

```
▶▶ JCLDSN= 'dsname' ▶▶
```

dsname

Specify a 1- to 44-character data set name. Enclose the data set name in single quotation marks (').

Default:

None.

Site default support:

Not supported.

JOB CARDx keyword

Use the JOB CARDx keyword to specify JOB statements for the OSC internal jobs.

JOB CARDx is an optional keyword for the JCLPARM subcommand.

The JOB statement in the OSC internal jobs can be up to five lines. You can specify up to five JOB CARDx keywords.

The Online Structure Change utility uses the string that you specify for the JOB CARD1 keyword to build JOB statements in columns 16 through 70.

If JOB CARD2 to JOB CARD5 are specified, the Online Structure Change utility uses the strings to build JOB statements in columns 3 through 57. If you specify a comment JOB statement, subsequent JOB statements must also be comment JOB statements.

Note: JOB CARDx keywords must not contain the JOB statement parameter DSENQSHR.

Format:

```
▶▶ JOB CARD x=" job_statement " ▶▶
```

x

Job card number, from 1- to 5.

job_statement

Specify a 1- to 55-character job statement. Enclose the job statement in double quotation marks (").

Default:

None.

Site default support:

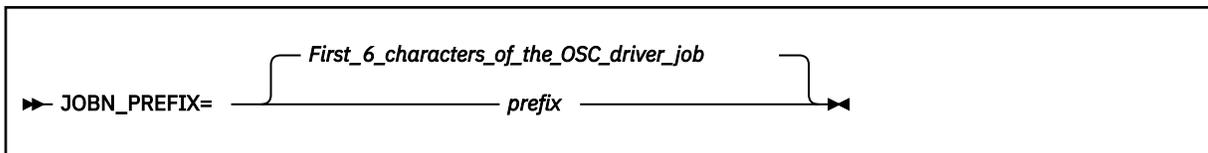
Not supported.

JOBN_PREFIX keyword

Use the JOBN_PREFIX keyword to specify the prefix for OSC internal job names.

JOBN_PREFIX is an optional keyword for the JCLPARM subcommand.

You can specify the first six characters for OSC internal job names. The last two characters of OSC internal job names are fixed. For OSC internal job names, see [“Online Structure Change utility internal jobs” on page 1010](#).

Format:**prefix**

Specify a 6-character prefix for OSC internal job names.

Default:

The first 6 characters of the OSC driver job name.

Site default support:

Not supported.

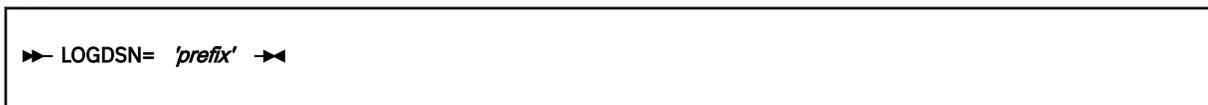
LOGDSN keyword

Use the LOGDSN keyword to specify the name prefix for the OSC internal job log data set.

Requirement: An OSC internal job log data set must be preallocated before you submit the OSC driver job.

LOGDSN is a required keyword for the JCLPARM subcommand.

The Online Structure Change utility stores job logs from OSC internal jobs in a data set that the LOGDSN keyword specifies. The log data set name is *prefix.OSC_driver_jobname*. For example, if you specify LOGDSN='HPFP.OSC' and the OSC driver job name is OSC10000, the log data set name is HPFP.OSC.OSC10000.

Format:**prefix**

Specify a 1- to 13-character high-level qualifier. Enclose the high-level qualifier in single quotation marks (').

Default:

None.

Site default support:

Not supported.

MCLASS_TEMP keyword

Use the MCLASS_TEMP keyword to specify the SMS management class for the temporary data sets that the Online Structure Change utility dynamically allocates.

MCLASS_TEMP is an optional keyword for the JCLPARM subcommand.

Format:**management_class_name**

Specify a 1- to 8-character management class name.

Default:

None.

Site default support:

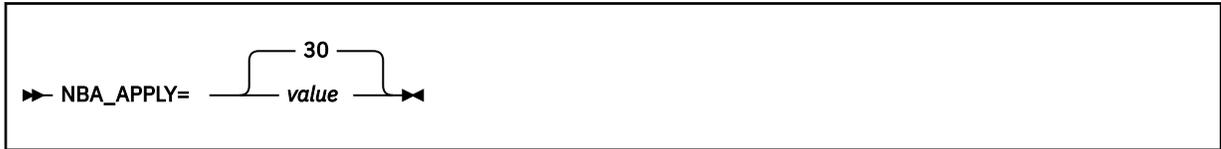
Not supported.

NBA_APPLY keyword

Use the NBA_APPLY keyword to specify the IMS DBCTL Fast Path database buffers to be made available for the apply job.

NBA_APPLY is an optional keyword for the JCLPARM subcommand.

Format:



value

Specify a value in the range of 1 - 9999.

Default:

NBA_APPLY=30

Site default support:

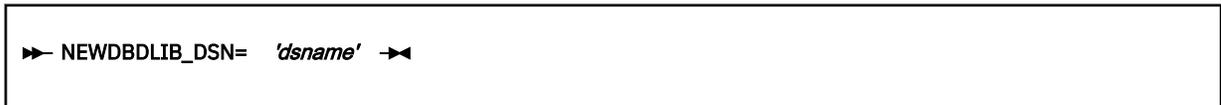
Not supported.

NEWDBDLIB_DSN keyword

Use the NEWDBDLIB_DSN keyword to specify the name of the new DBDLIB library that contains the database description for the restructured DEDB.

NEWDBDLIB_DSN is a required keyword for the JCLPARM subcommand.

Format:



dsname

Specify a 1- to 44-character data set name. Enclose the data set name in single quotation marks (').

Default:

None.

Site default support:

Not supported.

NEWRESLB_EDC_DSN keyword

Use the NEWRESLB_EDC_DSN keyword to specify the name of the library that contains the edit/compression routines that the new ACB member specifies.

NEWRESLB_EDC_DSN is an optional keyword for the JCLPARM subcommand. However, if an edit/compression routine is defined in the new ACB member, you must specify this keyword.

The library that you specify for the NEWRESLB_EDC_DSN keyword varies depending on how you want to change the segment edit/compression exit routines in production IMS systems:

- If you want to replace segment edit/compression exit routines, specify the new library to the NEWRESLB_EDC_DSN keyword. The Online Structure Change utility replaces the segment edit/compression exit routines in the production IMS library (specified by the IMSRESLB_EDC_DSN keyword) with those found in the new library (specified by the NEWRESLB_EDC_DSN keyword).
- If you want to change the definitions of segment edit/compression routines without replacing any segment edit/compression exit routines, specify the library that contains the new routines to both IMSRESLB_EDC_DSN and NEWRESLB_EDC_DSN keywords.
- If you do not want to replace segment edit/compression exit routines nor change the definitions of the routines, specify the current library to both NEWRESLB_EDC_DSN and IMSRESLB_EDC_DSN keywords.

For more information, see [“Replacing randomizer and segment edit/compression exit routines”](#) on page 1016.

Format:

```
➤ NEWRESLB_EDC_DSN= 'dsname' ➤
```

dsname

Specify a 1- to 44-character data set name. Enclose the data set name in single quotation marks (').

Default:

None.

Site default support:

Not supported.

NEWRESLB_RND_DSN keyword

Use the NEWRESLB_RND_DSN keyword to specify the name of the library that contains the randomizer that the new ACB member specifies.

NEWRESLB_RND_DSN is a required keyword for the JCLPARM subcommand.

The library that you specify for the NEWRESLB_RND_DSN keyword varies depending on how you want to change the randomizer in production IMS systems:

- If you want to replace the randomizer, specify the new library to the NEWRESLB_RND_DSN keyword. The Online Structure Change utility replaces the randomizer in the production IMS library (specified by the IMSRESLB_RND_DSN keyword) with the randomizer found in the new library (specified by the NEWRESLB_RND_DSN keyword).
- If you want to change the definitions of the randomizer without replacing the randomizer, specify the library that contains the new randomizer to both IMSRESLB_RND_DSN and NEWRESLB_RND_DSN keywords.
- If you do not want to replace the randomizer nor change the definitions of the randomizer, specify the current library to both NEWRESLB_RND_DSN and IMSRESLB_RND_DSN keywords.

Format:

```
➤ NEWRESLB_RND_DSN= 'dsname' ➤
```

dsname

Specify a 1- to 44-character data set name. Enclose the data set name in single quotation marks (').

Default:

None.

Site default support:

Not supported.

OBA_APPLY keyword

Use the OBA_APPLY keyword to specify, for the apply job, additional IMS DBCTL page-fixed buffers to be made available when the normal allotment is used up.

OBA_APPLY is an optional keyword for the JCLPARM subcommand.

Format:**value**

Specify a value in the range of 0 -9999.

Default:

OBA_APPLY=10

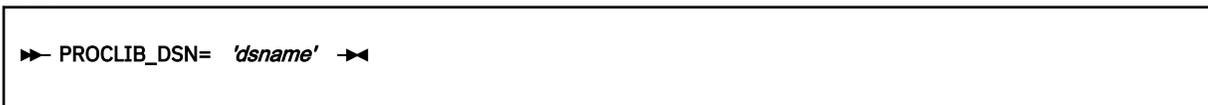
Site default support:

Not supported.

PROCLIB_DSN keyword

Use the PROCLIB_DSN keyword to specify the name of the PROCLIB library that the production IMS systems use.

PROCLIB_DSN is a required keyword for the JCLPARM subcommand.

Format:**dsname**

Specify a 1- to 44-character data set name. Enclose the data set name in single quotation marks (').

Default:

None.

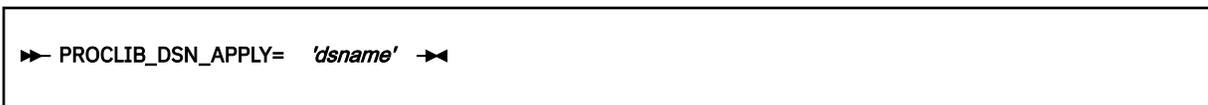
Site default support:

Not supported.

PROCLIB_DSN_APPLY keyword

Use the PROCLIB_DSN_APPLY keyword to specify the name of the IMS DBCTL PROCLIB library. This library is used during the apply job.

PROCLIB_DSN_APPLY is a required keyword for the JCLPARM subcommand.

Format:**dsname**

Specify a 1- to 44-character data set name. Enclose the data set name in single quotation marks (').

Default:

None.

Site default support:

Not supported.

PSBNAME_APPLY keyword

Use the PSBNAME_APPLY keyword to specify the name of the PSB that is used for the apply job.

PROCLIB_DSN_APPLY is a required keyword for the JCLPARM subcommand.

Format:

```
➤ PROCLIB_DSN_APPLY= psbname ➤
```

psbname

Specify a 1- to 8-character PSB name.

Default:

None.

Site default support:

Not supported.

RECON_x_DSN keyword

Use the RECON_x_DSN keyword to specify the names of RECON data sets that the production IMS systems use.

RECON_x_DSN is a required keyword for the JCLPARM subcommand.

Format:

```
➤ RECON x_DSN= 'dsname' ➤
```

dsname

Specify a 1- to 44-character data set name. Enclose the data set name in single quotation marks (').

Default:

None.

Site default support:

Not supported.

RECON_x_DSN_APPLY keyword

Use the RECON_x_DSN_APPLY keyword to specify the names of IMS DBCTL RECON_x data sets for the apply job.

RECON_x_DSN_APPLY is a required keyword for the JCLPARM subcommand.

Format:

```
➤ RECON x_DSN_APPLY= 'dsname' ➤
```

dsname

Specify a 1- to 44-character data set name. Enclose the data set name in single quotation marks (').

Default:

None.

Site default support:

Not supported.

SCLASS_TEMP keyword

Use the SCLASS_TEMP keyword to specify the SMS storage class for the temporary data sets that the Online Structure Change utility dynamically allocates.

SCLASS_TEMP is an optional keyword for the JCLPARM subcommand.

Format:

```
▶▶ SCLASS_TEMP= storage_class_name ▶▶
```

storage_class_name

Specify a 1- to 8-character storage class name.

Default:

None.

Site default support:

Not supported.

SGLXLOAD keyword

Use the SGLXLOAD keyword to specify the name of the IMS Tools Base load module library.

SGLXLOAD is an optional keyword for the JCLPARM subcommand. However, if IMSCATHLQ=*bsdshlq* is specified, you must specify this keyword.

Format:

```
▶▶ SGLXLOAD= 'dsname' ▶▶
```

dsname

Specify a 1- to 44-character data set name. Enclose the data set name in single quotation marks (').

Default:

None.

Site default support:

Not supported.

SPACE keyword

Use the SPACE keyword to specify the space requirements for the output data sets that the Online Structure Change utility dynamically allocates to hold unloaded segment data records.

SPACE is an optional keyword for the JCLPARM subcommand.

Format:

```
▶▶ SPACE= ( CYL TRK ,value1 ,value2 ,0 ) ▶▶
```

The first parameter specifies the unit type for allocation. The following values are supported:

CYL

Allocation unit is in cylinders.

TRK

Allocation unit is in tracks.

The second parameter specifies the value for the primary space for the allocated data set:

value1

Acceptable values are 1 - 16777215. The value represents the number of allocation units to be allocated.

The third parameter is optional.

SYSAFF_APPLY keyword

Use the SYSAFF_APPLY keyword to specify the JES2 member that is eligible to process the apply job (representing system affinity).

SYSAFF_APPLY is an optional keyword for the JCLPARM subcommand.

Format:

```
►► SYSAFF_APPLY= affinity ◄◄
```

affinity

Specify a 1- to 4-character JES2 member name.

Default:

None.

Site default support:

Not supported.

TOIXCFGRP_APPLY keyword

Use the TOIXCFGRP_APPLY keyword to specify the XCF group that IMS Tools Online System Interface uses to interface with the apply job.

TOIXCFGRP_APPLY is a required keyword for the JCLPARM subcommand.

Format:

```
►► TOIXCFGRP_APPLY= TOI_XCF_group_name ◄◄
```

TOI_XCF_group_name

Specify a 1- to 5-alphanumeric character to generate the XCF group name. IMS Tools Online System Interface prefixes this string with characters "TOI" to generate the XCF group name.

Default:

None.

Site default support:

Supported.

TOSILOAD keyword

Use the TOSILOAD keyword to specify the name of the load module library for IMS Tools Online System Interface (TOSI).

TOSILOAD is a required keyword for the JCLPARM subcommand.

Format:

```
►► TOSILOAD= 'dsname' ◄◄
```

dsname

Specify a 1- to 44-character data set name. Enclose the data set name in single quotation marks (').

Default:

None.

Site default support:

Not supported.

UNIT_TEMP keyword

Use the UNIT_TEMP keyword to specify the physical device type for the temporary data sets that the Online Structure Change utility dynamically allocates.

UNIT_TEMP is an optional keyword for the JCLPARM subcommand.

Format:

```
▶▶ UNIT_TEMP= unitname ◀◀
```

unitname

Specify a 1- to 8-character alphanumeric name of the device type.

Default:

None.

Site default support:

Not supported.

USRHLQ keyword

Use the USRHLQ keyword to specify the high-level qualifier for the output data sets in which the Online Structure Change utility generates unloaded segment data records.

USRHLQ is a required keyword for the JCLPARM subcommand.

Format:

```
▶▶ USRHLQ= 'usrhlq' ◀◀
```

usrhlq

Specify a 1- to 26-character high-level qualifier. Enclose the high-level qualifier in single quotation marks (').

Default:

None.

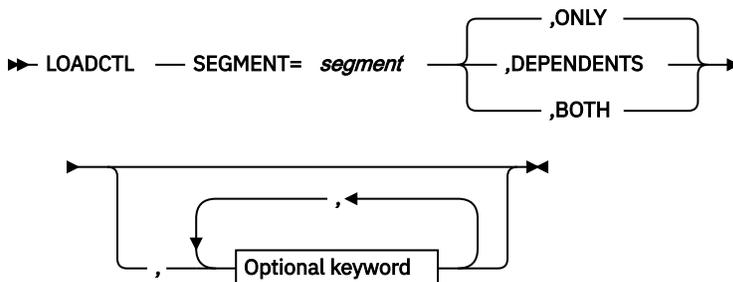
Site default support:

Not supported.

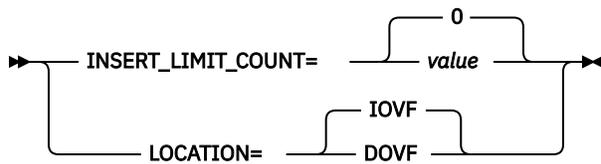
LOADCTL subcommand keywords

Use the LOADCTL subcommand to specify the options for loading segments into the restructured ADSs. LOADCTL is an optional subcommand for the HFPSYSIN DD statement.

You can specify the following keywords for the LOADCTL subcommand.



Optional keywords:



INSERT_LIMIT_COUNT keyword

Use the INSERT_LIMIT_COUNT keyword to specify the maximum number of segments that can be inserted close to their root segment.

This keyword is effective for the segments that are identified by the SEGMENT keyword.

INSERT_LIMIT_COUNT is an optional keyword for the LOADCTL subcommand.

Format:



value

Specify the number of segments to insert close to their root segment. The value must be in the range of 0 - 32767. 0 specifies that all segments identified by the SEGMENT keyword are stored far from their root segment.

Default:

INSERT_LIMIT_COUNT=0

Site default support:

Supported.

LOCATION keyword

Use the LOCATION keyword to specify the location to insert dependent segments, either DOVF or IOVF.

When the number of segments to load, which are selected by the SEGMENT keyword, exceeds the value of the INSERT_LIMIT_COUNT keyword, all the remaining segments are loaded to the location where the LOCATION keyword specifies.

LOCATION is an optional keyword for the LOADCTL subcommand.

Format:



IOVF

Segments are loaded to the IOVF CI space.

DOVF

Segments are loaded to the DOVF CI space. If no space is available in DOVF CIs, remaining segments are loaded to the IOVF CI space.

Default:

LOCATION=IOVF

Site default support:

Supported.

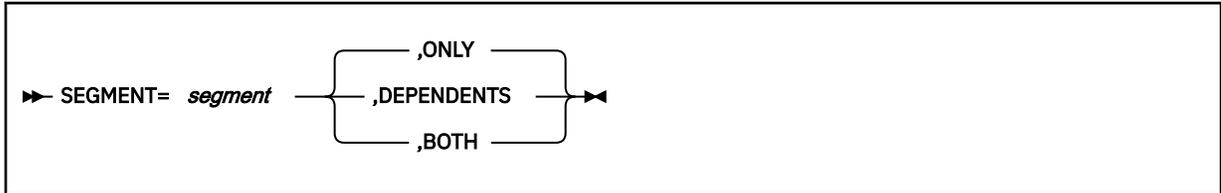
SEGMENT keyword

Use the SEGMENT keyword to specify the name of the segment that should be loaded to a specific location.

Requirement: The name of the segment you specify for the SEGMENT keyword must match the segment name coded for the SEGM statement in the DBD.

SEGMENT is a required keyword for the LOADCTL subcommand.

Format:



segment

Specify a 1- to 8-character alphanumeric name of the segment that is coded on the SEGM statement of the DBD. You cannot specify the root segment or the SDEP segment.

ONLY

The LOADCTL subcommand options apply only to the specified segment. You can specify O instead of ONLY.

DEPENDENTS

The LOADCTL subcommand options apply to all the dependent segments of the specified segment, but not to the segment itself. You can specify D instead of DEPENDENTS.

BOTH

The LOADCTL subcommand options apply to both the specified segment and its dependent segments. You can specify B instead of BOTH.

Default:

None. If only the first parameter (*segment*) is supplied without the second parameter, ONLY is applied as the second parameter.

Site default support:

Not supported.

Chapter 40. Online Structure Change utility reports

Online Structure Change utility reports are generated in the HFPRPTS data set.

The following table summarizes the reports that the Online Structure Change utility generates. The table also shows whether each report is stored in the IMS Tools KB Output repository.

If you want to export OSC reports, use the export capability provided by IMS Tools Base. For more information, see the topic "Exporting reports" in the *Tools Base IMS Tools Knowledge Base User's Guide*. For the RECONID parameter of the export utility, you must specify NORECON (RECONID=NORECON) to export OSC reports.

Table 130. Online Structure Change utility reports

Reports	Whether the report can be stored in the IMS Tools KB Output repository
Audit report	Yes
Processing report	Yes
DBD Definition report	Yes
Base Configuration report	Yes
OSC Internal Jobs Scenario report	Yes
Apply report	No

Topics:

- [“Audit report” on page 1075](#)
- [“Processing report” on page 1077](#)
- [“DBD Definition report” on page 1079](#)
- [“Base Configuration report” on page 1082](#)
- [“OSC Internal Jobs Scenario report” on page 1082](#)
- [“Apply report” on page 1083](#)

Audit report

The Audit report provides information about the HFPSYSIN control statements applied to the job, EXEC statement parameters, and GLOBAL command keyword values.

The following figure is an example of the Audit report.

```

0.....1.....2.....3.....4.....5.....6.....7.....8
1234567890123456789012345678901234567890123456789012345678901234567890
- 1: GLOBAL 01440016
- 2: MSTIMESTAMP=LOCAL, 01450016
- 3: IMSID=IFF5, 01460016
- 4: SCAN=NO, 01470016
- 5: TOIXCFGRP=F5TOI 01480016
- 6: RESTRUCTURE 01490016
- 7: ANALYZE=AFTRELOAD, 01500016
- 8: ORIGADS_PROC=RENAME, 01510016
- 9: DBD=DEDB0009, 01520016
- 10: IAREA=ALL, 01530016
- 11: OAREA=ALL, 01540016
- 12: RUNTYPE=RUNJCL, /* CHANGE EXECUTION TYPE 01550016
- 13: SKIP_OLC=NO, 01560020
- 14: TAKEOVER_TIME=IMMEDIATE, 01570016
- 15: UPDATE_NUMBER=700 01580016
- 16: JCLPARM 01590016
- 17: COMPRESS=NO, 01600016
- 18: CONFIGLIB_DSN='IMSDEV.FPT.OSC.USERLIB', 01610016
- 19: DATACOMP_SUFFIX=D, 01620016
- 20: DBDLIB_DSN='IMSDEV.IFF5.DBDLIB', 01630016
- 21: DBRCGRP=005, 01640016
- 22: DCLASS_TEMP=DEFAULT, 01650016
- 23: DDNPREFIX_NEW=DD9N, 01660016
- 24: DDNPREFIX_SHADOW=DD9S, 01670016
- 25: DDNPREFIX_UIC=DD9U, 01680016
- 26: DSNMASK_NEW='IMSDEV.IFF5.DEDB0009.&AREA.ADS1', 01690016
- 27: DSNMASK_SHADOW='IMSDEV.IFF5.DEDB0009.&AREA.ADS2', 01700016
- 28: DSNMASK_UIC='IMSDEV.IFF5.UIC.&AREA.ADS1', 01710016
- 29: FPLIB_DSN='IMSTJP.FPS210.SHFPLMD0', 01720016
- 30: IMSID_APPLY=IFFK, 01730016
- 31: IMSLIB_DSN='IMS.IFJ1.SDFSRESL', 01740016
- 32: IMSLIB_DSN_APPLY='IMS.IFJ3.SDFSRESL', 01750016
- 33: IMSPLEX=FPFJ1, 01760016
- 34: IMSRESLB_RND_DSN='IMSDEV.FPT.DATACAP.RMOD', 01770016
- 35: IMSRESLB_EDC_DSN='IMSDEV.FPT.DATACAP.EDC', 01780016
- 36: INTRDR=IFF50SDT, 01790016
- 37: JCLDSN='IMSDEV.FPT.OS500000.J', 01800016
- 38: ENQDSN='IMSDEV.FPT.OS500000.ENQ', 01810019
- 39: JOBCARD1='MSGLEVEL=(1,1),CLASS=A,MSGCLASS=A,', 01830016
- 40: JOBCARD2=' USER=,PASSWORD=,', 01840016
- 41: JOBCARD3=' REGION=300M', 01850016
- 42: JOBN_PREFIX=055D09, 01860016
- 43: LOGDSN='IMSDEV.FPT', 01870016
- 44: MCLASS_TEMP=STANDARD, 01880016
- 45: NBA_APPLY=0020, 01890016
- 46: NEWDBDLIB_DSN='IMSDEV.IFF5.DBDLIB.STAGE', 01900016
- 47: NEWRESLB_RND_DSN='IMSDEV.FPT.DATACAP.RMOD.NEW', 01910016

```

```

0.....1.....2.....3.....4.....5.....6.....7.....8
1234567890123456789012345678901234567890123456789012345678901234567890
- 48: NEWRESLB_EDC_DSN='IMSDEV.FPT.DATACAP.EDC', 01920016
- 49: OBA_APPLY=0010, 01930016
- 50: PROCLIB_DSN='IMSDEV.IFF5.PROCLIB', 01940016
- 51: PROCLIB_DSN_APPLY='IMSDEV.IFFK.PROCLIB', 01950016
- 52: PSBNAME_APPLY=DDLTPE1, 01960016
- 53: RECON1_DSN='IMSDEV.IFF5.RECON1', 01970016
- 54: RECON2_DSN='IMSDEV.IFF5.RECON2', 01980016
- 55: RECON3_DSN='IMSDEV.IFF5.RECON3', 01990016
- 56: RECON1_DSN_APPLY='IMSDEV.IFFK.RECON1', 02000016
- 57: RECON2_DSN_APPLY='IMSDEV.IFFK.RECON2', 02010016
- 58: RECON3_DSN_APPLY='IMSDEV.IFFK.RECON3', 02020016
- 59: SCLASS_TEMP=SHARED, 02030016
- 60: SYSAFF=RSIE, 02040016
- 61: SPACE=(CYL,10,5), 02050016
- 62: SYSAFF_APPLY=RSIF, 02060016
- 63: TOIXCFGRP_APPLY=FKTOI, 02070016
- 64: TOSILOAD='IMSTJP.ITB160.SFOILOAD', 02080018
- 65: UNIT_TEMP=SYSDA, 02090016
- 66: USRHLQ='IMSDEV.FPT.TSXXXX', 02100016
- 67: ALLOCATE 02110016
- 68: OAREA=ALL, 02120016
- 69: ATTRIBUTE=(IDCAMS, 02130016
- 70: DSN='IMSDEV.FPT.OSC2I15.D2.@IDCAMS(&AREA)') 02140016
- 71: LOADCTL 02150016
- 72: INSERT_LIMIT_COUNT=50, 02160016
- 73: LOCATION=DOVF, 02170016
- 74: SEGMENT=(DD11,ONLY) 02180016
PROCESSING INFORMATION
EXEC
- DBRCGRP : 005
- IMSID :
- IMSPLEX : FPFJ1
GLOBAL
- IMSCATHLQ : *NO
- IMSID : IFF5
- ITKBDATA : *NO
- ITKBLOAD : *NO
- ITKBSVR : *NO
- MSTIMESTAMP : LOCAL
- SCAN : NO
- TOIXCFGRP : F5TOI

```

Figure 385. Audit report

HFPSYSIN

Shows an echo of the control statements that you specified in the HFPSYSIN data set.

PROCESSING INFORMATION

Shows the parameters of the EXEC statement and the GLOBAL command in HFPSYSIN.

Processing report

The Processing report contains the values that the command process used. The values shown in the report are determined by the values of HFPSYSIN, the site default options, and the system default.

The following figure is an example of the Processing report.

```

IMS HPFP UTILITIES - OSC RESTRUCTURE                                "Processing report"                                PAGE: 1
5698-FPP V2R1                                                    2020-08-06 20:02:13
RESTRUCTURE STARTED DATE: 2020-08-06 TIME: 20:02:13 ENDED DATE: 2020-08-06 TIME: 20:02:13
RESTRUCTURE PROCESSING OPTION
- ANALYZE : AFTRELOAD
- DBD : DEDB0009
- ERRORACT : ABEND
- IAREA : ALL
- ITASKCTL : 4
- OAREA : ALL
- ORIGADS_PROC : RENAME
- OTASKCTL : 4
- PAD : X'00'
- REPLRAND_AREALVL : NO
- RUNTYPE : RUNJCL
- SKIP_OLC : NO
- TAKEOVER_TIME : IMMEDIATE
- UPDATE_NUMBER : 700
- USR_MAX : 512
- IMSACB : IMSDEV.IFF5.ACBLIB
- NEWACB : IMSDEV.IFF5.ACBLIB.STAGE
RESTRUCTURE SUBCOMMAND PROCESSING OPTION
ALLOCATE
- ATTRIBUTE : (IDCAMS,DSN='IMSDEV.FPT.OSC2I15.TSXXXX.D2.@IDCAMS(&AREA)',EROPT=ABORT)
- OAREA : ALL
RESTRUCTURE SUBCOMMAND PROCESSING OPTION
JCLPARM
- COMPRESS : NO
- CONFIGLIB_DSN : IMSDEV.FPT.TSXXXX.OSC.USERLIB
- DATACOMP_SUFFIX : D
- DBDLIB_DSN : IMSDEV.IFF5.DBDLIB
- DBR_NOFE0V : NO
- DBR_NOFE0V_APPLY : NO
- DBRCGRP : 005
- DBRCGRP_APPLY :
- DCLASS_TEMP : DEFAULT
- DDNPREFIX_NEW : DD9N
- DDNPREFIX_SHADOW : DD9S
- DDNPREFIX_UIC : DD9U
- DSMASK_NEW : 'IMSDEV.IFF5.DEDB0009.&AREA.ADS1'
- DSMASK_SHADOW : 'IMSDEV.IFF5.DEDB0009.&AREA.ADS2'
- DSMASK_UIC : 'IMSDEV.IFF5.UIC.&AREA.ADS1'
- ENODSN : IMSDEV.FPT.OS500000.ENQ
- FPLIB_DSN : IMSTJP.FPS210.TSXXXX.SHFPLM00
- IMSID_APPLY : IFFK
- IMSLIB_DSN : IMS.IFJ1.SDFSRESL
- IMSLIB_DSN_APPLY : IMS.IFJ3.SDFSRESL
- IMSPLEX : PPFJ1
- IMSPLEX_APPLY :
- IMSRESLB_EDC_DSN : IMSDEV.FPT.TSXXXX.DATACAP.EDC
IMS HPFP UTILITIES - OSC RESTRUCTURE                                "Processing report"                                PAGE: 2
5698-FPP V2R1                                                    2020-08-06 20:02:13
- IMSRESLB_RND_DSN : IMSDEV.FPT.TSXXXX.DATACAP.RMOD
- INTRDR : IFF50SDT
- TOSILOAD : IMSTJP.FPT.OSC.ITB160.SFOILOAD
- JCLDSN : IMSDEV.FPT.OS500000.J
- JOBCARD1 : "MSGLEVEL=(1,1),CLASS=A,MSGCLASS=A,"
- JOBCARD2 : " USER=,PASSWORD=,"
- JOBCARD3 : " REGION=300M"
- JOBCARD4 :
- JOBCARD5 :
- JOBN_PREFIX : OS5D09
- LOGDSN : IMSDEV.FPT
- MCLASS_TEMP : STANDARD
- NBA_APPLY : 0020
- NEWDBDLIB_DSN : IMSDEV.IFF5.DBDLIB.STAGE
- NEWRESLB_EDC_DSN : IMSDEV.FPT.TSXXXX.DATACAP.EDC.NEW
- NEWRESLB_RND_DSN : IMSDEV.FPT.TSXXXX.DATACAP.RMOD
- OBA_APPLY : 0010
- PROCLIB_DSN : IMSDEV.IFF5.PROCLIB
- PROCLIB_DSN_APPLY : IMSDEV.IFFK.PROCLIB
- PSBNAME_APPLY : DDLTFPE1
- RECON1_DSN : IMSDEV.IFF5.RECON1
- RECON2_DSN : IMSDEV.IFF5.RECON2
- RECON3_DSN : IMSDEV.IFF5.RECON3
- RECON1_DSN_APPLY : IMSDEV.IFFK.RECON1
- RECON2_DSN_APPLY : IMSDEV.IFFK.RECON2
- RECON3_DSN_APPLY : IMSDEV.IFFK.RECON3
- SCLASS_TEMP : SHARED
- SGLXLOAD :
- SPACE : (CYL,10,5)
- SPOC_WAITTIME : 300
- SYSAFF : RSIE
- SYSAFF_APPLY : RSIF
- TOIXCFGRP_APPLY : FKTOI
- UNIT_TEMP : SYSDA
- USRHLQ : IMSDEV.FPT.TSXXXX
RESTRUCTURE SUBCOMMAND PROCESSING OPTION
LOADCTL
- SEGMENT : (DD11,ONLY)
- LOCATION : DOVF
- INSERT_LIMIT_COUNT : 50

```

Figure 386. Processing report

RESTRUCTURE PROCESSING OPTION

Shows the values that the command process used.

RESTRUCTURE SUBCOMMAND PROCESSING OPTION

Show the values that the subcommand process used.

DBD Definition report

The DBD Definition report contains information about the DBD in the ACB libraries that the IMSACB DD and the NEWACB DD statements specify.

The Online Structure Change utility generates two DBD Definition reports, one for IMSACB and another for NEWACB. If there is a difference between the two, an asterisk (*) is printed next to the item that is different. If the difference is significant, an ampersand sign (&) is printed next to the item.

The following figures show examples of the DBD Definition report.

```

IMS HPFP UTILITIES - OSC RESTRUCTURE                "DBD definition report"
5698-FPP V2R1                                         PAGE: 1
                                                    2019-10-21 13:50:58
- ACBLIB      : IMSACB
- ACB DSNAME  : IMSDEV.IFF#.ACBLIB
- DBD NAME    : DEDBGS1D
- RANDOMIZER  : RMODA1D
- IMS LEVEL   : VERSION 15 RELEASE 1
DEDB DEFINITION DIFFERENCE DETECTED.
DB LARGEST INFO: CI-SIZE  UOW-1  NO(RAP'S/UOW)  *: DIFFERENCE  &: SEVERE DIFFERENCE
                  4,096    50                30      20      400
AREA:
AREA:
AREA NO.  AREA NAME  CI-SIZE  UOW=  ROOT=  TOTAL CI'S  SDEP START
          (BASE  DOVF  IOVF  TOTAL)  (BLOCK#  RBA(HEX))
==> 1  S1DA010  4,096* (50,20)*  (240,90)*  4,500  3,000  4,462  11,962  -  -
==> 2  S1DA020  4,096* (50,20)*  (240,90)*  4,500  3,000  4,462  11,962  -  -
==> 3  S1DA021  4,096* (50,20)*  (240,90)*  4,500  3,000  4,462  11,962  -  -
==> 4  S1DA030  4,096* (50,20)*  (240,90)*  4,500  3,000  4,462  11,962  -  -
==> 5  S1DA031  4,096* (50,20)*  (240,90)*  4,500  3,000  4,462  11,962  -  -
==> 6  S1DA040  4,096* (50,20)*  (240,90)*  4,500  3,000  4,462  11,962  -  -
==> 7  S1DA050  4,096* (50,20)*  (240,90)*  4,500  3,000  4,462  11,962  -  -
==> 8  S1DA060  4,096* (50,20)*  (240,90)*  4,500  3,000  4,462  11,962  -  -
SEGMENT:
+-----+
| LEGEND FOR SEGMENT INFORMATION |
+-----+
| R: ROOT SEGMENT  D: DDEP SEGMENT  S: SDEP SEGMENT |
| F: FIXED LENGTH  V: VARIABLE LENGTH  I: COMP INIT  Y: PCL DEFINED |
+-----+
SEG. CODE  SEG. NAME  HIER LVL  PARENT S. CODE  TYPE  FIX|VAR  PARENT PCL SSP  LENGTH  KEY  COMP-RTN  SEC.
          (MAX  MIN  OFF  LEN)  (NAME  INIT  IDX)
          (400  20  2  16)
==> 1  ROOTSEG1  1  -  R  V  -  -  400  20  2  16  -  -  -
==> 2  DD1  2  1  D  V  -  -  400  30  2  5  -  -  -
==> 3  DD11  3  2  D  F  Y  -  100  100  2  5  -  -  -
==> 4  DD12  3  2  D  V  -  -  400  40  2  5  -  -  -
==> 5  DD121  4  4  D  V  Y  -  400  30  -  -  -  -  -
==> 6* DD2  2  1  D  V  Y  -  400  40  2  8  -  -  -
*** END OF DATABASE DEFINITION REPORT ***

```

Figure 387. DBD Definition report (IMSACB)

- ACBLIB : NEWACB
- ACB DSNAME : IMSDEV.IFF#.ACBLIB.STAGE
- DBD NAME : DEDBGS1D
- RANDOMIZER : RMODA1D
- IMS LEVEL : VERSION 15 RELEASE 1

DEDB DEFINITION DIFFERENCE DETECTED.
DB LARGEST INFO: CI-SIZE UOW-1 NO(RAP'S/UOW) UOW-2 SEG-LEN

AREA:		CI-SIZE	UOW=	ROOT=	TOTAL CI'S				SDEP START	
AREA NO.	AREA NAME				BASE	DOVF	IOVF	TOTAL	BLOCK#	RBA (HEX)
==> 1	S1DA010	8,192*	(30,5)*	(200,20)*	4,500	900	595	5,995	-	-
==> 2	S1DA020	8,192*	(30,5)*	(200,20)*	4,500	900	595	5,995	-	-
==> 3	S1DA021	8,192*	(30,5)*	(200,20)*	4,500	900	595	5,995	-	-
==> 4	S1DA030	8,192*	(30,5)*	(200,20)*	4,500	900	595	5,995	-	-
==> 5	S1DA031	8,192*	(30,5)*	(200,20)*	4,500	900	595	5,995	-	-
==> 6	S1DA040	8,192*	(30,5)*	(200,20)*	4,500	900	595	5,995	-	-
==> 7	S1DA050	8,192*	(30,5)*	(200,20)*	4,500	900	595	5,995	-	-
==> 8	S1DA060	8,192*	(30,5)*	(200,20)*	4,500	900	595	5,995	-	-
==> 9	S1DA061*	8,192	(30,5)	(200,20)	4,500	900	595	5,995	-	-

SEGMENT:

```

+-----+
| LEGEND FOR SEGMENT INFORMATION |
+-----+
| R: ROOT SEGMENT  D: DDEP SEGMENT  S: SDEP SEGMENT |
| F: FIXED LENGTH  V: VARIABLE LENGTH I: COMP INIT   Y: PCL DEFINED |
+-----+

```

SEG. CODE	SEG. NAME	HIER LVL	PARENT S.CODE	TYPE	FIX VAR	PARENT		LENGTH		KEY		COMP-RTN		SEC. IDX
						PCL	SSP	MAX	MIN	OFF	LEN	NAME	INIT	
1	ROOTSEG1	1	-	R	V	-	-	400	20	2	16	-	-	-
2	DD1	2	1	D	V	-	-	400	30	2	5	-	-	-
3	DD11	3	2	D	F	Y	-	100	100	2	5	-	-	-
4	DD12	3	2	D	V	-	-	400	40	2	5	-	-	-
5	DD121	4	4	D	V	Y	-	400	30	-	-	-	-	-
==> 6	DD1A*	2	1	D	V	-	-	400	30	2	5	-	-	-
==> 7*	DD2	2	1	D	V	Y	-	400	40	2	8	-	-	-

*** END OF DATABASE DEFINITION REPORT ***

Figure 388. DBD Definition report (NEWACB)

ACBLIB

The name of the ACBLIB DD statement.

ACB DSNAME

The name of the ACBLIB data set specified by the *acblib* DD statement. This data set contains the DBD definition.

DBD NAME

The name of the DBD.

RANDOMIZER

The name of the randomizing module.

IMS LEVEL

The IMS level of the source data set from which the DBD definition information is reported.

DB LARGEST INFO

The largest value of the areas in the DBD as follows:

CI-SIZE

The largest CI size of areas in the DBD.

UOW-1

The largest number of CIs per UOW of areas in the DBD.

NO(RAP'S/UOW)

The largest number of RAP CIs per UOW of areas in the DBD.

UOW-2

The largest number of DOVF CIs per UOW of areas in the DBD.

SEG-LEN

The largest length of segments of areas in the DBD.

AREA NO.

The area number.

AREA NAME

The area name.

CI-SIZE

The CI size of the area.

UOW=

The UOW= parameter value for the area that is defined in the DBD.

ROOT=

The ROOT= parameter value for the area that is defined in the DBD.

TOTAL CI'S BASE

The total number of BASE CIs for the area that is defined in the DBD.

TOTAL CI'S DOVF

The total number of DOVF CIs for the area that is defined in the DBD.

TOTAL CI'S IOVF

The total number of IOVF CIs for the area that is defined in the DBD. IOVF bitmap CIs are not included.

TOTAL CI'S TOTAL

The total number of CIs for the area that is defined in the DBD. The first, second, IOVF bitmap, work UOW, and SDEP part CIs are not included.

SDEP START BLOCK#

The first block number of CI in the SDEP part of the area.

SDEP START RBA (HEX)

The hexadecimal value of the first CI in the SDEP part of the area.

SEG.CODE

The segment code.

SEG.NAME

The segment name.

HIER LVL

The hierarchical level of the segment.

PARENT S.CODE

The parent segment code of the segment.

TYPE

The type of the segment as follows:

R

Root segment.

S

Sequential dependent segment.

D

Direct dependent segment.

FIX|VAR

The attribute of the segment as follows:

F

Fixed-length segment.

V

Variable-length segment.

PARENT PCL

Indicates whether the parent of the segment has a PCL pointer. Y means that the parent of the segment has a PCL pointer.

PARENT SSP

Indicates the number of subset pointers that the parent of the segment has, if any are defined.

LENGTH MAX

The maximum length of the segment that is defined in the DBD.

LENGTH MIN

The minimum length of the segment that is defined in the DBD.

KEY OFF

The offset of the key field of the segment, if one is defined.

KEY LEN

The key length of the segment, if one is defined.

COMP-RTN NAME

The name of the segment edit/compression exit routine of the segment, if one is defined.

COMP-RTN INIT

I indicates that the COMPRTN= INIT subparameter is defined in the DBD.

SEC. IDX

The number of secondary index databases in which the segment is defined as a target segment.

Base Configuration report

The Base Configuration report, which is generated by the OSC driver job, displays the base configuration parameters that are defined in the HFP#*imsid* or the HFPSETOP module.

The following figure shows an example of the Base Configuration report.

```

IMS HFPF UTILITIES - OSC RESTRUCTURE                "Base configuration report"                PAGE: 1
5698-FPP V2R1                                       2019-10-21 13:50:42
BASE CONFIGURATION MODULE                          : HFPSETOP (10/01/2019 CREATED)
BASE CONFIGURATION PARAMETERS
- ULOGID                                           : F1
- XCFGROUP                                         : XCFIFF#

```

Figure 389. Base Configuration report

BASE CONFIGURATION MODULE

Shows which base configuration module is used, and when it was created.

BASE CONFIGURATION PARAMETERS

Shows the base configuration parameters that are specified in the base configuration module.

OSC Internal Jobs Scenario report

The OSC Internal Jobs Scenario report, which is generated by the OSC driver job, summarizes the scenario of OSC internal jobs that the Online Structure Change utility submits during the OSC driver job.

The following figure shows an example of the OSC Internal Jobs Scenario report.

```

IMS HFPF UTILITIES - OSC RESTRUCTURE                "OSC internal jobs scenario report"          PAGE: 1
5698-FPP V2R1                                       2020-08-06 20:02:13
DSN FOR INTERNAL JOBS JCL: IMSDEV.FPT.OS500000.J
INTERNAL JOBS PROCESSED WHILE OSC IS BEING EXECUTED:
PHASE  JOB ID  JOBNAME  LPAR  DESCRIPTION
-----
INITIAL  CHKRTN  OS5D09E0  RSIE  CHECKS IF THE RANDOMIZER OR EDIT/COMPRESSION ROUTINES ARE IN USE BY ANY OTHER DEDB.
        INVAPPLY  OS5D09A0  RSIF  INVOKES THE CHANGE APPLY JOB.
        DEFSADS  OS5D09F0  RSIE  ALLOCATES SHADOW ADS AND REGISTERS THEM TO RECON.
        INITSADS  OS5D09I0  RSIE  INITIALIZES SHADOW ADS.
COPY    COPYSADS  OS5D09C0  RSIE  COPIES DATA FROM ORIGINAL ADS TO SHADOW ADS.
        STOPSADS  OS5D09P0  RSIE  STOPS SHADOW ADS AND UNREGISTERS THEM FROM RECON.
UNLOAD  UNLOAD   OS5D09U0  RSIF  UNLOADS DATA FROM SHADOW ADS TO UNLOADED FILES.
RELOAD  RELOAD   OS5D09R0  RSIF  RELOADS DATA INTO RESTRUCTURED ADS.
        ANALYZE  OS5D09K0  RSIF  ANALYZES RESTRUCTURED ADS.
APPLY   STAAPPLY  OS5D09J*  RSIF  STARTS THE DEDB AND AREAS IN DBCTL SUBSYSTEM. NUMBER OF JOBS : 0001
TAKEOVER DBRPROD  OS5D09D0  RSIE  STOPS THE DEDB AND AREAS IN IMS SYSTEMS.
        CKPTSTAT  OS5D09H0  RSIE  FLUSHES XCF BUFFERS.
        DBRDBCTL  OS5D09S0  RSIF  STOPS THE DEDB AND AREAS IN DBCTL SUBSYSTEM.
        UNDBBCTL  OS5D09B0  RSIF  UNREGISTERS RESTRUCTURED ADS AND ICDS FROM DBCTL RECON.
        COPYRTN  OS5D09Y0  RSIE  COPIES NEW RANDOMIZING OR EDIT/COMPRESSION ROUTINES FROM NEWRESLB TO IMSRESLB.
        OLC      OS5D09M0  RSIE  ISSUES 'INITIATE OLC' COMMAND TO CHANGE THE ACBLIB.
        RENAME  OS5D09N0  RSIE  DELETES ORIGINAL ADS, RENAMES RESTRUCTURED ADS TO ORIGINAL NAMES, ISSUES NOTIFY.UIC.
        STAPROD  OS5D09T0  RSIE  STARTS THE DEDB AND AREAS IN IMS SYSTEMS.
TERM    TERMPROC  OS5D09L0  RSIE  TERMINATION PROCESSING.

```

Figure 390. OSC Internal Jobs Scenario report

DSN FOR INTERNAL JOBS JCL

The name of the data set in which the Online Structure Change utility generated OSC internal jobs.

PHASE

The OSC process phase. Indicates in which phase the OSC internal job is submitted.

JOB ID

job ID.

JOBNAME

Job name.

LPAR

The LPAR where the OSC internal job is run.

DESCRIPTION

Description for the OSC internal job.

Apply report

The Apply report provides, for each area and the entire DEDB, the number of DL/I calls that the Online Structure Change utility issued during the apply phase. The calls are categorized by call types. Apply reports are generated during the apply job.

The following figure shows an example of the Apply report.

```
IMS HPFP UTILITIES - OSC RESTRUCTURE                "Apply report"
5698-FPP V2R1
APPLY DBDNAME: DEDBGS1D
- AREA NO: 1 AREANAME: S1DA010
                                     PAGE: 1
                                     2019-10-21 13:50:58
                                     CALL TYPE
SEGMENT SEGMENT SEGMENT
CODE    NAME    HIERARCHY    ISRT    DLET    REPL
-----
1 ROOTSEG1 1          40      0      0
2 DD1      2          40      0      0
3 DD11     3          40      0      0
4 DD12     3          40      0      0
5 DD121    4          0       0      0
6 DD1A     2          0       0      0
7 DD2      2          40      0      0
-----
** TOTAL **                200      0      0
- AREA NO: 2 AREANAME: S1DA020
                                     CALL TYPE
SEGMENT SEGMENT SEGMENT
CODE    NAME    HIERARCHY    ISRT    DLET    REPL
-----
1 ROOTSEG1 1          40      0      0
2 DD1      2          40      0      0
3 DD11     3          40      1      0
4 DD12     3          40      0      0
5 DD121    4          0       0      0
6 DD1A     2          0       0      0
7 DD2      2          40      0      0
-----
** TOTAL **                200      1      0
IMS HPFP UTILITIES - OSC RESTRUCTURE                "Apply report"
5698-FPP V2R1
APPLY DBDNAME: DEDBGS1D
                                     PAGE: 4
                                     2019-10-21 13:50:58
                                     CALL TYPE
SEGMENT SEGMENT SEGMENT
CODE    NAME    HIERARCHY    ISRT    DLET    REPL    SYNC
-----
1 ROOTSEG1 1          80      0      0
2 DD1      2          80      0      0
3 DD11     3          80      1      0
4 DD12     3          80      0      0
5 DD121    4          0       0      0
6 DD1A     2          0       0      0
7 DD2      2          80      0      0
-----
-                                               81
** TOTAL **                400      1      0      81
```

Figure 391. Apply report

AREA NO

The area number.

AREA NAME

The area name.

SEGMENT CODE

The segment code.

SEGMENT NAME

The segment name.

SEGMENT HIERARCHY

The hierarchical level of the segment.

CALL TYPE

The type of the DL/I call, as follows:

ISRT

DL/I ISRT call

DLET

DL/I DLET call

REPL

DL/I REPL call

SYNC

DL/I SYNC call

Chapter 41. Examples for the Online Structure Change utility

The following topics provide examples for coding JCL statements and control statements for the Online Structure Change utility.

Topics:

- [“Example: Restructuring an online DEDB” on page 1085](#)
- [“Example: Specifying input and output areas” on page 1086](#)
- [“Example: Specifying the ALLOCATE subcommand” on page 1087](#)

Example: Restructuring an online DEDB

The following figure shows a JCL example for restructuring an online DEDB.

Before running this example, you must generate a new ACBLIB. To run this example in an IMS managed ACBs environment, you must also populate the new ACB definition in the staging data set of the IMS directory.

This example restructures online database DEDB000F. The following changes are also made during the restructuring process:

- Replaces the randomizer.
- Adds two areas.
- Changes the UOW, ROOT, and CISIZE parameters.

```
//HFPEXE EXEC PGM=HFPOSC00,REGION=100M,
//      PARM=' IMSPLEX=FPFJ1,DBRCGRP=001'
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//      DD DISP=SHR,DSN=ITB.SFOILOAD
//      DD DISP=SHR,DSN=ITB.SGLXLOAD
//      DD DISP=SHR,DSN=IMSVS.SDFSRESL
//      DD DISP=SHR,DSN=IMSVS.PGMLIB
//      DD DISP=SHR,DSN=HPFP.OSC.USERLIB    <== HFPSETOP
//HFPPRINT DD SYSOUT=*
//HFPRPTS  DD SYSOUT=*
//*
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//NEWACB  DD DISP=SHR,DSN=IMSVS.ACBLIB.STAGE
//*
//SYSOUT  DD SYSOUT=A
//HFP3XTRA DD DUMMY
//*
//HFPSYSIN DD *
GLOBAL
  MSTIMESTAMP=LOCAL,
  IMSID=IFF#,
  SCAN=NO,
  TOIXCFGRP=F#TOI
RESTRUCTURE
  DBD=DEDB000F,
  IAREA=ALL,
  OAREA=ALL,
  ITASKCTL=1,
  OTASKCTL=1,
  RUNTYPE=FULL,
  ERRORACT=ABEND,
  TAKEOVER_TIME=IMMEDIATE
```

Figure 392. Restructuring an online DEDB (Part 1 of 2)

```

JCLPARM
  COMPRESS=NO,
  TOSILOAD='ITB.SFOILOAD',
  PSBNAME_APPLY=DDLTPE1,
  IMSID_APPLY=IFF@,
  NBA_APPLY=0020,
  OBA_APPLY=0010,
  INTRDR=IFF#OSCG,
  CONFIGLIB_DSN='HPFP.OSC.USERLIB',
  DBDLIB_DSN='IMSVS.DBDLIB',
  DBR_NOFEOV=NO,
  DBR_NOFEOV_APPLY=YES,
  DBRCGRP=00$,
  DCLASS_TEMP=DEFAULT,
  DDNPREFIX_NEW=DDNN,
  DDNPREFIX_SHADOW=DDNS,
  DDNPREFIX_UIC=DDNU,
  DSNMASK_NEW='HPFP.DEDB000F.&AREA.ADS1',
  DSNMASK_SHADOW='HPFP.DEDB000F.&AREA.ADS2',
  DSNMASK_UIC='HPFP.UIC.DEDB000F.&AREA.ADS1',
  ENQDSN='HPFP.OSC.@ENQDSN',
  FPLIB_DSN='HPFP.SHFPLMD0',
  IMSLIB_DSN='IMSVS.SDFSRESL',
  IMSLIB_DSN_APPLY='IMSVS.IFJ3.SDFSRESL',
  IMSPLEX=FPFJ1,
  IMSRESLB_RND_DSN='IMSVS.DATACAP.RMOD',
  IMSRESLB_EDC_DSN='IMSVS.DATACAP.RMOD',
  JCLDSN='HPFP.OSC.@GENJCLS',
  JOBCARD1="MSGLEVEL=(1,1),CLASS=A,MSGCLASS=A,",
  JOBCARD2="      USER=,PASSWORD=,",
  JOBCARD3="      REGION=300M",
  JOBN_PREFIX=OST00F,
  MCLASS_TEMP=STANDARD,
  NEWRESLB_RND_DSN='IMSVS.DATACAP.RMOD.NEW',
  NEWRESLB_EDC_DSN='IMSVS.DATACAP.RMOD',
  NEWDBDLIB_DSN='IMSVS.DBDLIB.STAGE',
  PROCLIB_DSN='IMSVS.IFF#.PROCLIB',
  PROCLIB_DSN_APPLY='IMSVS.IFF@.PROCLIB',
  RECON1_DSN='IMSVS.IFF#.RECON1',
  RECON2_DSN='IMSVS.IFF#.RECON2',
  RECON3_DSN='IMSVS.IFF#.RECON3',
  RECON1_DSN_APPLY='IMSVS.IFF@.RECON1',
  RECON2_DSN_APPLY='IMSVS.IFF@.RECON2',
  RECON3_DSN_APPLY='IMSVS.IFF@.RECON3',
  SCLASS_TEMP=SHARED,
  SPACE=(CYL,5,3),
  SYSAFF=RSIE,
  SYSAFF_APPLY=RSIF,
  LOGDSN='HPFP.OSC',
  TOIXCFGRP_APPLY=F@TOI,
  USRHLQ='HPFP.OSC.TEST'
ALLOCATE
  OAREA=ALL,
  ATTRIBUTE=(IDCAMS,
  DSN='HPFP.OSC.@IDCAMS(&AREA)')
/*

```

Figure 393. Restructuring an online DEDB (Part 2 of 2)

Example: Specifying input and output areas

Use the IAREA keyword of the RESTRUCTURE command to specify one or more input areas, and the OAREA keyword of the RESTRUCTURE command to specify one or more output areas.

The IAREA and the OAREA keywords support the following values:

ALL

All areas of the DEDB. This is the default value.

areaname

A single area. Specify an area with its name. The name must be 1- to 8- characters.

areaname1-areaname2

Multiple areas. Specify the range of areas by area names.

area_number1-area_number2

Multiple areas. Specify the range of areas by area numbers.

For example, to restructure areas from area#5 to area#10, and area#15 to area#20, specify as follows:

```
RESTRUCTURE
  IAREA=(5-10,15-20),
  OAREA=(5-10,15-20)
```

To add a new area (AREA010) by splitting AREA001 into two areas (AREA001 and AREA010), specify as follows:

```
RESTRUCTURE
  IAREA=(AREA001),
  OAREA=(AREA001,AREA010)
```

To delete an area (AREA002) by merging AREA001 and AREA002 into one area (AREA001), specify as follows:

```
RESTRUCTURE
  IAREA=(AREA001,AREA002),
  OAREA=(AREA001)
```

A structure change that affects the entire DEDB, for example, changing the randomizing module or the segment edit/compression routine, both IAREA and OAREA keywords must specify ALL:

```
RESTRUCTURE
  IAREA=ALL,
  OAREA=ALL
```

The default values for the IAREA and OAREA keywords are ALL. These two keywords cannot be specified in the site default table.

If a segment record is randomized to an area that is not specified by the OAREA keyword, the utility discards the segment and issues an error message in the unload phase or the apply phase to alert this condition.

When adding new areas, you must provide allocation information for the new areas with the following keywords:

- DDNPREFIX_NEW keyword
- DSNMASK_NEW keyword
- The data set specified for the ATTRIBUTE keyword (ALLOCATE subcommand) must contain a member with the name of the new area.

Example: Specifying the ALLOCATE subcommand

The ALLOCATE subcommand specifies the information that the Online Structure Change utility uses to dynamically allocate restructured ADSs.

The ATTRIBUTE keyword of the ALLOCATE subcommand specifies the partition data set that the Online Structure Change utility passes to Access Method Services for allocating new VSAM clusters. The partitioned data set must have one or more members that contain the IDCAMS control statement for each area. The data set must be preallocated before you submit the OSC driver job.

For example, you want to restructure area#1, area#2, and area#3.

You can specify IDCAMS control statements in a data set or in multiple data sets. If you specify IDCAMS control statements in multiple data sets, there must be one ALLOCATE subcommand for each partitioned data set.

If you specify the IDCAMS control statement for each area in one data set, you create three members in the data set; AREA001, AREA002, and AREA003. Then you specify the data set with one ALLOCATE subcommand as follows:

```
ALLOCATE
  OAREA=(AREA001-AEA003),
  ATTRIBUTE=(IDCAMS,DSN='HPFP.IDCAMS(&AREA)')
```

If you specify the IDCAMS control statement in different data sets, for example, AREA001 and AREA002 in data set HPFP.IDCAMS1, and AREA003 in data set HPFP.IDCAMS2, then you specify two sets of ALLOCATE subcommands as follows:

```
ALLOCATE
  OAREA=(AREA001,AREA002),
  ATTRIBUTE=(IDCAMS,DSN='HPFP.IDCAMS1(&AREA)')
ALLOCATE
  OAREA=(AREA003),
  ATTRIBUTE=(IDCAMS,DSN='HPFP.IDCAMS2(&AREA)')
```

The following figure shows an example of IDCAMS control statements for area D090A010:

```
DEFINE CLUSTER -
  (NAME(IMSDEV.IEF5.DEDB0009.D09A010.ADS2) -
  NONINDEXED CYL(120) UNIQUE -
  CONTROLINTERVALSIZE(8192) RECORDSIZE(8185 8185) -
  FOR(999) SHAREOPTIONS(3 3) SPEED)
```

Part 6. References

These topics provide technical references for IMS HP Fast Path Utilities.

Topics:

- [Chapter 42, “Performance tips and best practices for using IMS HP Fast Path Utilities,” on page 1091](#)
- [Chapter 43, “DEDDB monitoring and tuning guide,” on page 1111](#)
- [Chapter 44, “Viewing the reports stored in IMS Tools KB,” on page 1171](#)
- [Chapter 45, “Record layouts,” on page 1173](#)
- [Chapter 46, “Exit routine option and its interface,” on page 1197](#)
- [Chapter 47, “Segment edit/compression facility support,” on page 1201](#)
- [Chapter 48, “Determining how to process SDEP segments,” on page 1205](#)
- [Chapter 49, “How to read syntax diagrams,” on page 1207](#)

Chapter 42. Performance tips and best practices for using IMS HP Fast Path Utilities

Learn the techniques for improving the performance of IMS HP Fast Path Utilities jobs and the options that simplify the use of IMS HP Fast Path Utilities.

Topics:

- [“Tips and best practices for IMS Fast Path Advanced Tool” on page 1091](#)
- [“Tips and best practices for IMS Fast Path Online Tools” on page 1106](#)
- [“Tips and best practices for the Online Space Management utility” on page 1109](#)
- [“Tips and best practices for the Online Structure Change utility” on page 1110](#)

Tips and best practices for IMS Fast Path Advanced Tool

FPA supports many keywords that simplify the management of DEDBs. The following topics describe how to use those keywords to improve the performance of FPA jobs and further simplify management of DEDBs.

Tips and best practices for the FPA Unload function

The FPA Unload function supports several keywords that improve the performance of FPA jobs and that further simplify management of DEDBs.

Improving performance

Specify the number of areas to unload concurrently

When you unload multiple areas, use the ITASKCTL keyword to control the number of areas to unload concurrently. The default value of the ITASKCTL keyword is 4. Change the value to optimize the run time and CPU time of the FPA Unload job.

Increasing the value of the ITASKCTL keyword might affect the resources in the following ways:

- Reduced job run time.
- Slightly increased CPU time.
- Increased region size.

For more information, see [“ITASKCTL keyword” on page 486](#).

Pre-load CIs into data space storage

If you specify ICACHE=YES, FPA pre-loads the control intervals (CIs) from the DEDB areas into data space storage and accesses the CIs in data space storage instead of accessing the CIs in VSAM or image copy data sets. Therefore, the run time of the job decreases. The pre-loaded CIs include those in BASE (only when the input data sets are image copy data sets), IOVF, and SDEP portions of the areas.

ICACHE=NO is recommended only for the following situations:

- You specify a large value for the ITASKCTL keyword. In this case, data from all concurrently processed DEDB areas is pre-loaded into data space storage concurrently, and the amount of data space that the thread tasks of FPA use might be considerable.
- The data sets are VSAM ESDS data sets, and no segments or only a few segments exist in the IOVF and SDEP portions of the areas. In this case, there is no advantage to pre-loading IOVF and SDEP portions into data spaces.

For more information, see [“ICACHE keyword” on page 472](#).

Skip the segment edit/compression exit routine process

If the job is for non-reorg mode and the segment edit/compression exit routines in IMSACB and NEWACB are the same, specify EDITCOMP_SKIP=YES to skip the segment edit/compression exit routine call. Skipping this process shortens the job run time and the CPU time that the segment edit/compression exit routine needs to compress or expand segment records.

Warning: If the segment edit/compression exit routines in IMSACB and NEWACB have the same name but different contents, running the job with the EDITCOMP_SKIP=YES option might destroy the database.

For more information, see [“EDITCOMP_SKIP keyword” on page 459](#).

Using the function more effectively and efficiently

Perform syntax check

If you specify SCAN=YES on the GLOBAL command, FPA does not run the unload process but it checks the syntax of the commands, keywords, and parameters that you specify in the JCL. Use this option if you want to verify the syntax of HFPSYSIN control statements when you create new JCL.

For more information, see [“SCAN keyword” on page 523](#).

Generate image copies during the job

If you specify ICOUT=YES, after FPA unloads the areas, FPA starts IMS HP Image Copy to generate image copies of the areas. If you use this option, you do not need to code another job step to generate image copies.

When you specify ICOUT=YES, you must also specify the IMS HP Image Copy library on the STEPLIB DD statement. You can also use the ICEIN DD statement to specify the control statements for IMS HP Image Copy.

For more information, see [“ICOUT keyword” on page 476](#).

Sort unloaded segment records during the job

If you specify SORT=YES, FPA sorts the unloaded segment records during the unload process. If you use this option, you do not need to code another job step to sort the unloaded segment records.

For more information, see [“SORT keyword” on page 541](#).

Restart the job from where it failed

When you unload multiple areas, specify the RESTART=YES option. Then if the job fails, you resubmit the job to start it from where it failed. FPA skips the areas that were unloaded in the previous run and unloads only the remaining areas.

For more information, see [“RESTART keyword” on page 514](#) and [“Example 12: Unloading all areas of a DEDB that are registered with DBRC and creating their image copies with the checkpoint restart option” on page 84](#).

Resolving problems

WER039A INSUFFICIENT VIRTUAL STORAGE

If this message was issued during the SYNC SORT process, use the xTAKCTL keyword to reduce the number of areas to process concurrently, and specify the following DD statement in your JCL.

```
//$ORTPARM DD*  
VSCORE=256K,VSCORET=3M,MINCORE=256K  
/*
```

ICE046A 0 SORT CAPACITY EXCEEDED - RECORD COUNT nnnnnnnn

If this message was issued during the DFSORT process, specify SORTOPT_FILSZ=AUTO so that FPA automatically estimates the number of intermediate records, calculates the value of the FILSZ parameter, and adds the FILSZ parameter to the SORT control statement. The SORTOPT_FILSZ=AUTO option helps to prevent jobs from failing due to lack of sort work space.

Tips and best practices for the FPA Reload function

The FPA Reload function supports several keywords that improve the performance of FPA Reload jobs and that further simplify management of DEDBs.

Improving performance

Specify the number of areas to reload concurrently

When you reload multiple areas, use the ITASKCTL keyword to control the number of areas to reload concurrently. The default value of the ITASKCTL keyword is 4. Change the value to optimize the run time and CPU time of the FPA Reload job.

Increasing the value of the ITASKCTL keyword might affect the resources in the following ways:

- Reduced job run time.
- Slightly increased CPU time.
- Increased region size.

For more information, see [“ITASKCTL keyword” on page 486](#).

Specify the number of areas to analyze concurrently

When you reload multiple areas, use the ATASKCTL keyword to control the number of areas to analyze concurrently. The default value of the ATASKCTL keyword is the value that you specify for the ITASKCTL keyword. Change the value to optimize the run time and CPU time of the FPA Reload job. The ATASKCTL keyword is effective except when PTRCHKLVL=NONE is specified.

Increasing the value of the ATASKCTL keyword might affect the resources in the following ways:

- Reduced job run time.
- Slightly increased CPU time.
- Increased region size.

For more information, see [“ATASKCTL keyword” on page 443](#).

Use data space storage to reload data into IOVF and SDEP portions

If you specify OCACHE=YES, FPA uses data space storage to reload CIs into the IOVF and SDEP portions of the areas. This option improves access to CIs and reduces the run time of the job.

OCACHE=YES is effective when the area data set is large or when many segments are placed in the IOVF portion, the SDEP portion, or both portions of the area.

For more information, see [“OCACHE keyword” on page 499](#).

Using the function more effectively and efficiently

Perform syntax check

If you specify SCAN=YES on the GLOBAL command, FPA does not run the reload process but it checks the syntax of the commands, keywords, and parameters that you specify in the JCL. Use this option if you want to verify the syntax of HFPSYSIN control statements when you create new JCL.

For more information, see [“SCAN keyword” on page 523](#).

Generate image copies during the job

If you specify ICOUT=YES, after FPA reloads the areas, FPA starts IMS HP Image Copy to generate image copies of the areas. If you use this option, you do not need to code another job step to generate image copies.

When you specify ICOUT=YES, you must specify the IMS HP Image Copy library on the STEPLIB DD statement. You can also use the ICEIN DD statement to specify the control statements for IMS HP Image Copy.

For more information, see [“ICOUT keyword” on page 476](#).

Analyze DEDB areas during the job

If you specify PTRCHKLVL=FULL or QUICK, FPA analyzes the reloaded areas. If you use this option, you do not need to code another job step to analyze DEDB areas. If you also specify the REPORT subcommand, FPA generates reports that you can use to analyze the reloaded areas.

For more information, see [“PTRCHKLVL keyword” on page 506](#).

Sort unloaded segment records during the job

If you specify SORT=YES, FPA sorts the unloaded segment records before the Reload process. If you use this option, you do not need to code another job step to sort the unloaded segment records.

For more information, see [“SORT keyword” on page 541](#).

Resolving problems

WER039A INSUFFICIENT VIRTUAL STORAGE

If this message was issued during the SYNC SORT process, use the xTAKCTL keyword to reduce the number of areas to process concurrently, and specify the following DD statement in your JCL.

```
//$ORTPARM DD*  
VSCORE=256K,VSCORET=3M,MINCORE=256K  
/*
```

ICE046A 0 SORT CAPACITY EXCEEDED - RECORD COUNT nnnnnnnn

If this message was issued during the DFSORT process, specify SORTOPT_FILSZ=AUTO so that FPA automatically estimates the number of intermediate records, calculates the value of the FILSZ parameter, and adds the FILSZ parameter to the SORT control statement. The SORTOPT_FILSZ=AUTO option helps to prevent jobs from failing due to lack of sort work space.

Tips and best practices for the FPA Change function

The FPA Change function supports several keywords that improve the performance of FPA Change jobs and that further simplify management of DEDBs.

Improving performance

Specify the number of areas to unload concurrently

When you change multiple areas, use the ITASKCTL keyword to control the number of areas to unload concurrently. The default value of the ITASKCTL keyword is 4. Change the value to optimize the run time and CPU time of the FPA Change job.

Increasing the value of the ITASKCTL keyword might affect the resources in the following ways:

- Reduced job run time.
- Slightly increased CPU time.
- Increased region size.

For more information, see [“ITASKCTL keyword” on page 486](#).

Specify the number of areas to reload concurrently

When you change multiple areas, use the OTASKCTL keyword to control the number of areas to reload concurrently. The default value of the OTASKCTL keyword is 4. Change the value to optimize the run time and CPU time of the FPA Change job.

Increasing the value of the OTASKCTL keyword might affect the resources in the following ways:

- Reduced job run time.
- Slightly increased CPU time.
- Increased region size.

For more information, see [“OTASKCTL keyword” on page 503](#).

Specify the number of areas to analyze concurrently

When you change multiple areas, use the ATASKCTL keyword to control the number of areas to analyze concurrently. The default value of the ATASKCTL keyword is the value that you specify for the OTASKCTL keyword. Change the ATASKCTL value to optimize the run time and CPU time of the FPA Change job. The ATASKCTL keyword is effective except when PTRCHKLVL=NONE is specified.

Increasing the value of the ATASKCTL keyword might affect the resources in the following ways:

- Reduced job run time.
- Slightly increased CPU time.
- Increased region size.

For more information, see [“ATASKCTL keyword” on page 443](#).

Pre-load CIs into data space storage

If you specify ICACHE=YES, FPA pre-loads the control intervals (CIs) from the DEDB areas into data space storage and accesses the CIs in data space storage instead of accessing the CIs in VSAM or image copy data sets. Therefore, the run time of the job decreases. The pre-loaded CIs include those in BASE (only when the input data sets are image copy data sets), IOVF, and SDEP portions of the areas.

ICACHE=NO is recommended only for the following situations:

- You specify a large value for the ITASKCTL keyword. In this case, data from all concurrently processed DEDB areas is pre-loaded into data space storage concurrently, and the amount of data space that the thread tasks of FPA use might be considerable.
- The data sets are VSAM ESDS data sets, and no segments or only a few segments exist in the IOVF and SDEP portions. In this case, there is no advantage to pre-loading IOVF and SDEP portions into data spaces.

For more information, see [“ICACHE keyword” on page 472](#).

Use data space storage to reload data into IOVF and SDEP portions

If you specify OCACHE=YES, FPA uses data space storage to reload CIs into IOVF and SDEP portions of the areas. This option improves access to CIs and reduces the run time of the job.

OCACHE=YES is effective when the area data set is large or when many segments are placed in the IOVF portion, the SDEP portion, or both portions of the areas.

For more information, see [“OCACHE keyword” on page 499](#).

Skip the segment edit/compression exit routine process

If the job is for non-reorg mode and the segment edit/compression exit routines in IMSACB and OLDACB are the same, specify EDITCOMP_SKIP=YES to skip the segment edit/compression exit routine call. Skipping this process shortens the job run time and the CPU time that the segment edit/compression exit routine needs to compress or expand segment records.

Warning: If the segment edit/compression exit routines in IMSACB and OLDACB have the same name but different contents, running the job with the EDITCOMP_SKIP=YES option might destroy the database.

For more information, see [“EDITCOMP_SKIP keyword” on page 459](#).

Using the function more effectively and efficiently

Perform syntax check

If you specify SCAN=YES on the GLOBAL command, FPA does not run the change process but it checks the syntax of the commands, keywords, and parameters that you specify in the JCL. Use this option if you want to verify the syntax of HFPSYSIN control statements when you create new JCL.

For more information, see [“SCAN keyword” on page 523](#).

Generate image copies during the job

If you specify ICOUT=YES, after FPA changes the areas, FPA starts IMS HP Image Copy to generate image copies of the areas. If you use this option, you do not need to code another job step to generate image copies.

When you specify ICOUT=YES, you must specify the IMS HP Image Copy library on the STEPLIB DD statement. You can also use the ICEIN DD statement to specify the control statements for IMS HP Image Copy.

For more information, see [“ICOUT keyword” on page 476](#).

Analyze DEDB areas during the job

If you specify PTRCHKLVL=FULL or QUICK, FPA analyzes the reorganized or restructured areas. If you use this option, you do not need to code another job step to analyze DEDB areas. If you also specify the REPORT subcommand, FPA generates reports that you can use to analyze the reorganized or restructured areas.

For more information, see [“PTRCHKLVL keyword” on page 506](#).

Resolving problems

WER039A INSUFFICIENT VIRTUAL STORAGE

If this message was issued during the SYNC SORT process, use the xTAKCTL keyword to reduce the number of areas to process concurrently, and specify the following DD statement in your JCL.

```
//$ORTPARM DD*  
VSCORE=256K,VSCORET=3M,MINCORE=256K  
/*
```

ICE046A 0 SORT CAPACITY EXCEEDED - RECORD COUNT nnnnnnnn

If this message was issued during the DFSORT process, specify SORTOPT_FILSZ=AUTO so that FPA automatically estimates the number of intermediate records, calculates the value of the FILSZ parameter, and adds the FILSZ parameter to the SORT control statement. The SORTOPT_FILSZ=AUTO option helps to prevent jobs from failing due to lack of sort work space.

Tips and best practices for the FPA Reorg function

The FPA Reorg function supports several keywords that improve the performance of FPA Reorg jobs and that further simplify management of DEDBs.

Improving performance

Specify the number of areas to reorganize concurrently

When you reorganize multiple areas, use the ITASKCTL keyword to control the number of areas to reorganize concurrently. The default value of the ITASKCTL keyword is 4. Change the value to optimize the run time and CPU time of the FPA Reorg job.

Increasing the value of the ITASKCTL keyword might affect the resources in the following ways:

- Reduced job run time.
- Slightly increased CPU time.
- Increased region size.

For more information, see [“ITASKCTL keyword” on page 486](#).

Specify the number of areas to analyze concurrently

When you reorganize multiple areas, use the ATASKCTL keyword to control the number of areas to analyze concurrently. The default value of the ATASKCTL keyword is the value that you specify for the ITASKCTL keyword. Change the value to optimize the run time and CPU time of the FPA Reorg job. The ATASKCTL keyword is effective except when PTRCHKLVL=NONE is specified.

Increasing the value of the ATASKCTL keyword might affect the resources in the following ways:

- Reduced job run time.
- Slightly increased CPU time.
- Increased region size.

For more information, see [“ATASKCTL keyword” on page 443](#).

Pre-load CIs into data space storage

If you specify ICACHE=YES, FPA pre-loads the control intervals (CIs) from the DEDB areas into data space storage and accesses the CIs in data space storage instead of accessing the CIs in VSAM data sets. Therefore, the run time of the job decreases. The pre-loaded CIs include those IOVF and SDEP portions of the areas.

ICACHE=NO is recommended only for the following situations:

- You specify a large value for the ITASKCTL keyword. In this case, data from all concurrently processed DEDB areas is pre-loaded into data space storage concurrently, and the amount of data space that the thread tasks of FPA use might be considerable.
- No segments or only a few segments exist in the IOVF and SDEP portions. In this case, there is no advantage to pre-loading IOVF and SDEP portions into data spaces.

For more information, see [“ICACHE keyword” on page 472](#).

Using the function more effectively and efficiently

Perform syntax check

If you specify SCAN=YES on the GLOBAL command, FPA does not run the reorganization process but it checks the syntax of the commands, keywords, and parameters that you specify in the JCL. Use this option if you want to verify the syntax of HFPSYSIN control statements when you create new JCL.

For more information, see [“SCAN keyword” on page 523](#).

Generate image copies during the job

If you specify ICOUT=YES, after FPA reorganizes the areas, FPA starts IMS HP Image Copy to generate image copies of the areas. If you use this option, you do not need to code another job step to generate image copies.

When you specify ICOUT=YES, you must specify the IMS HP Image Copy library on the STEPLIB DD statement. You can also use the ICEIN DD statement to specify the control statements for IMS HP Image Copy.

For more information, see [“ICOUT keyword” on page 476](#).

Analyze DEDB areas during the job

If you specify PTRCHKLVL=FULL or QUICK, FPA analyzes the reorganized areas. If you use this option, you do not need to code another job step to analyze DEDB areas. If you also specify the REPORT subcommand, FPA generates reports that you can use to analyze the reorganized areas.

For more information, see [“PTRCHKLVL keyword” on page 506](#).

Resolving problems

WER039A INSUFFICIENT VIRTUAL STORAGE

If this message was issued during the SYNC SORT process, use the xTAKCTL keyword to reduce the number of areas to process concurrently, and specify the following DD statement in your JCL.

```
//SORTPARM DD*
VSCORE=256K,VSCORET=3M,MINCORE=256K
/*
```

ICE046A 0 SORT CAPACITY EXCEEDED - RECORD COUNT nnnnnnnn

If this message was issued during the DFSORT process, specify SORTOPT_FILSZ=AUTO so that FPA automatically estimates the number of intermediate records, calculates the value of the FILSZ

parameter, and adds the FILSZ parameter to the SORT control statement. The SORTOPT_FILSZ=AUTO option helps to prevent jobs from failing due to lack of sort work space.

Tips and best practices for the FPA Analyze function

The FPA Analyze function supports several keywords that improve the performance of FPA jobs and that further simplify management of DEDBs.

Improving performance

Specify the number of areas to analyze concurrently

When you analyze multiple areas, use the ATASKCTL keyword to control the number of areas to analyze concurrently. The default value of the ATASKCTL keyword is 4. Change the value to optimize the run time and CPU time of the FPA Analyze job.

Increasing the value of the ATASKCTL keyword might affect the resources in the following ways:

- Reduced job run time.
- Slightly increased CPU time.
- Increased region size.

For more information, see [“ATASKCTL keyword” on page 443](#).

Specify the number of secondary indexes to analyze concurrently

When you analyze multiple secondary indexes, use the IDXTASKCTL keyword to control the number of secondary indexes to analyze concurrently. The default value of the IDXTASKCTL keyword is 4. Change the value to optimize the run time and CPU time of the FPA Analyze job. The IDXTASKCTL keyword is effective when you also specify the INDEXDBD keyword.

Increasing the value of the IDXTASKCTL keyword might affect the resources in the following ways:

- Reduced job run time.
- Slightly increased CPU time.
- Increased region size.

For more information, see [“IDXTASKCTL keyword” on page 479](#).

Pre-load CIs into data space storage

If the input area data sets are image copy data sets and you specify ICACHE=YES, FPA pre-loads the control intervals (CIs) from the DEDB areas into data space storage and accesses the CIs in data space storage instead of accessing the CIs in image copy data sets. Therefore, the run time of the job decreases. The pre-loaded CIs include those in BASE, IOVF, and SDEP portions of the areas.

Consider specifying ICACHE=NO only when you specify a large value for the ATASKCTL keyword. In this case, data from all concurrently processed DEDB areas is pre-loaded into data space storage concurrently, and the amount of data space that the thread tasks of FPA use might be considerable.

For more information, see [“ICACHE keyword” on page 472](#).

Using the function more effectively and efficiently

Perform syntax check

If you specify SCAN=YES on the GLOBAL command, FPA does not run the analyze process but it checks the syntax of the commands, keywords, and parameters that you specify in the JCL. Use this option if you want to verify the syntax of HFPSYSIN control statements when you create new JCL.

For more information, see [“SCAN keyword” on page 523](#).

Generate image copies during the job

If you specify ICOUT=YES, after FPA analyzes the areas, FPA starts IMS HP Image Copy to generate image copies of the areas. If you use this option, you do not need to code another job step to generate image copies.

When you specify ICOUT=YES, you must specify the IMS HP Image Copy library on the STEPLIB DD statement. You can also use the ICEIN DD statement to specify the control statements for IMS HP Image Copy.

For more information, see [“ICOUT keyword” on page 476](#).

Resolving problems

WER039A INSUFFICIENT VIRTUAL STORAGE

If this message was issued during the SYNC SORT process, use the xTAKCTL keyword to reduce the number of areas to process concurrently, and specify the following DD statement in your JCL.

```
//$ORTPARM DD*  
VSCORE=256K,VSCORET=3M,MINCORE=256K  
/*
```

ICE046A 0 SORT CAPACITY EXCEEDED - RECORD COUNT nnnnnnnn

If this message was issued during the DFSORT process, specify SORTOPT_FILSZ=AUTO so that FPA automatically estimates the number of intermediate records, calculates the value of the FILSZ parameter, and adds the FILSZ parameter to the SORT control statement. The SORTOPT_FILSZ=AUTO option helps to prevent jobs from failing due to lack of sort work space.

Tips and best practices for the FPA Tune function

The FPA Tune function supports several keywords that improve the performance of FPA Tune jobs and that further simplify management of DEDBs.

Improving performance

Specify the number of areas to scan concurrently

When you tune multiple areas, use the ITASKCTL keyword to control the number of areas to scan concurrently. The default value of the ITASKCTL keyword is 4. Change the value to optimize the run time and CPU time of the FPA Tune job.

Increasing the value of the ITASKCTL keyword might affect the resources in the following ways:

- Reduced job run time.
- Slightly increased CPU time.
- Increased region size.

For more information, see [“ITASKCTL keyword” on page 486](#).

Specify the number of areas to simulate concurrently

When you tune multiple areas, use the OTASKCTL keyword to control the number of areas to simulate concurrently. The default value of the OTASKCTL keyword is the value that you specify for the ITASKCTL keyword. Change the value to optimize the run time and CPU time of the FPA Tune job.

Increasing the value of the OTASKCTL keyword might affect the resources in the following ways:

- Reduced job run time.
- Slightly increased CPU time.
- Increased region size.

For more information, see [“OTASKCTL keyword” on page 503](#).

Pre-load CIs into data space storage

If the input area data sets are image copy data sets and you specify ICACHE=YES, FPA pre-loads the control intervals (CIs) from the DEDB areas into data space storage and accesses the CIs in data space storage instead of accessing the CIs in image copy data sets. Therefore, the run time of the job decreases. The pre-loaded CIs include those in BASE, IOVF, and SDEP portions of the areas.

Consider specifying ICACHE=NO only when you specify a large value for the ITASKCTL keyword. In this case, data from all concurrently processed DEDB areas is pre-loaded into data space storage concurrently, and the amount of data space that the thread tasks of FPA use might be considerable.

When the input area data sets are VSAM ESDS data sets, FPA sets the ICACHE parameter to NO.

For more information, see [“ICACHE keyword” on page 472](#).

Using the function more effectively and efficiently

Perform syntax check

If you specify SCAN=YES on the GLOBAL command, FPA does not run the tune process but it checks the syntax of the commands, keywords, and parameters that you specify in the JCL. Use this option if you want to verify the syntax of HFPSYSIN control statements when you create new JCL.

For more information, see [“SCAN keyword” on page 523](#).

Resolving problems

WER039A INSUFFICIENT VIRTUAL STORAGE

If this message was issued during the SYNCSORT process, use the xTAKCTL keyword to reduce the number of areas to process concurrently, and specify the following DD statement in your JCL.

```
//$ORTPARM DD*  
VSCORE=256K,VSCORET=3M,MINCORE=256K  
/*
```

ICE046A 0 SORT CAPACITY EXCEEDED - RECORD COUNT nnnnnnnn

If this message was issued during the DFSORT process, specify SORTOPT_FILSZ=AUTO so that FPA automatically estimates the number of intermediate records, calculates the value of the FILSZ parameter, and adds the FILSZ parameter to the SORT control statement. The SORTOPT_FILSZ=AUTO option helps to prevent jobs from failing due to lack of sort work space.

Tips and best practices for the FPA Extract function

The FPA Extract function supports several keywords that improve the performance of FPA Extract jobs and that further simplify management of DEDBs.

Improving performance

Specify the number of areas to extract concurrently

When you extract multiple areas, use the ITASKCTL keyword to control the number of areas to extract concurrently. The default value of the ITASKCTL keyword is 4. Change the value to optimize the run time and CPU time of the FPA Extract job.

Increasing the value of the ITASKCTL keyword might affect the resources in the following ways:

- Reduced job run time.
- Slightly increased CPU time.
- Increased region size.

For more information, see [“ITASKCTL keyword” on page 486](#).

Pre-load CIs into data space storage

If you specify ICACHE=YES, FPA pre-loads the control intervals (CIs) from the DEDB areas into data space storage and accesses the CIs in data space storage instead of accessing the CIs in VSAM or image copy data sets. Therefore, the run time of the job decreases. The pre-loaded CIs include those in BASE (only when the input data sets are image copy data sets), IOVF, and SDEP portions of the areas.

ICACHE=NO is recommended only for the following situations:

- You specify a large value for the ITASKCTL keyword. In this case, data from all concurrently processed DEDB areas is pre-loaded into data space storage concurrently, and the amount of data space that the thread tasks of FPA use might be considerable.
- The data sets are VSAM ESDS data sets, and no segments or only a few segments exist in the IOVF and SDEP portions. In this case, there is no advantage to pre-loading IOVF and SDEP portions into data spaces.

For more information, see [“ICACHE keyword” on page 472](#).

Using the function more effectively and efficiently

Perform syntax check

If you specify SCAN=YES on the GLOBAL command, FPA does not run the extract process but it checks the syntax of the commands, keywords, and parameters that you specify in the JCL. Use this option if you want to verify the syntax of HFPSYSIN control statements when you create new JCL.

For more information, see [“SCAN keyword” on page 523](#).

Sort extracted segment records during the job

If you specify SORT=YES, FPA sorts the extracted segment records during the extract process. If you use this option, you do not need to code another job step to sort the unloaded segment records.

For more information, see [“SORT keyword” on page 541](#).

Resolving problems

WER039A INSUFFICIENT VIRTUAL STORAGE

If this message was issued during the SYNCSORT process, use the xTAKCTL keyword to reduce the number of areas to process concurrently, and specify the following DD statement in your JCL.

```
//$ORTPARM DD*
VSCORE=256K,VSCORET=3M,MINCORE=256K
/*
```

ICE046A 0 SORT CAPACITY EXCEEDED - RECORD COUNT nnnnnnnn

If this message was issued during the DFSORT process, specify SORTOPT_FILSZ=AUTO so that FPA automatically estimates the number of intermediate records, calculates the value of the FILSZ parameter, and adds the FILSZ parameter to the SORT control statement. The SORTOPT_FILSZ=AUTO option helps to prevent jobs from failing due to lack of sort work space.

Tips and best practices for the FPA DMAC Print function

The FPA DMAC Print function supports a keyword that helps you use the function more effectively and efficiently.

Improving performance

The FPA DMAC Print function does not support keywords that improve job performance because the job run time and the CPU time are generally short.

Using the function more effectively and efficiently

Perform syntax check

If you specify SCAN=YES on the GLOBAL command, FPA does not run the DMAC print process but it checks the syntax of the commands, keywords, and parameters that you specify in the JCL. Use this option if you want to verify the syntax of HFPSYSIN control statements when you create new JCL.

For more information, see [“SCAN keyword” on page 523](#).

Tips and best practices for the FPA Build Index function

The FPA Build Index function supports several keywords that improve the performance of FPA Build Index jobs and help you use the function more effectively and efficiently.

Improving performance

Specify the number of areas to scan concurrently

When you build indexes for multiple areas, use the ITASKCTL keyword to control the number of areas to scan concurrently. The default value of the ITASKCTL keyword is 4. Change the value to optimize the run time and CPU time of the FPA Build Index job.

Increasing the value of the ITASKCTL keyword might affect the resources in the following ways:

- Reduced job run time.
- Slightly increased CPU time.
- Increased region size.

For more information, see [“ITASKCTL keyword” on page 486](#).

Specify the number of secondary indexes to build concurrently

When you build multiple secondary indexes, use the OTASKCTL keyword to control the number of secondary indexes to build concurrently. The default value of the OTASKCTL keyword is 4. Change the value to optimize the run time and CPU time of the FPA Build Index job.

Increasing the value of the OTASKCTL keyword might affect the resources in the following ways:

- Reduced job run time.
- Slightly increased CPU time.
- Increased region size.

For more information, see [“OTASKCTL keyword” on page 503](#).

Pre-load CIs into data space storage

If you specify ICACHE=YES, FPA pre-loads the control intervals (CIs) from the DEDB areas into data space storage and accesses the CIs in data space storage instead of accessing the CIs in VSAM or image copy data sets. Therefore, the run time of the job decreases. The pre-loaded CIs include those in BASE (only when the input data sets are image copy data sets), IOVF, and SDEP portions of the areas.

ICACHE=NO is recommended only for the following situations:

- You specify a large value for the ITASKCTL keyword. In this case, data from all concurrently processed DEDB areas is pre-loaded into data space storage concurrently, and the amount of data space that the thread tasks of FPA use might be considerable.
- The data sets are VSAM ESDS data sets, and no segments or only a few segments exist in the IOVF and SDEP portions. In this case, there is no advantage to pre-loading IOVF and SDEP portions into data spaces.

For more information, see [“ICACHE keyword” on page 472](#).

Using the function more effectively and efficiently

Perform syntax check

If you specify SCAN=YES on the GLOBAL command, FPA does not run the build index process but it checks the syntax of the commands, keywords, and parameters that you specify in the JCL. Use this option if you want to verify the syntax of HFPSYSIN control statements when you create new JCL.

For more information, see [“SCAN keyword” on page 523](#).

Sort build pointer segment records during the job

If you specify SORT=YES, FPA sorts the build pointer segment records during the build index process. If you use this option, you do not need to code another job step to sort the build pointer segment records.

For more information, see [“SORT keyword” on page 541](#).

Automatically estimate the space required for build pointer segment record data sets

FPA can automatically estimate the data set size and dynamically allocate the data sets that are required for build pointer segment records. Specify the following keywords:

- Specify the BUILDMODE=SCAN option on the INDEXBLD command.
- Specify the FILECTL subcommand.
- Specify the DISP=NEW option on the FILECTL subcommand.

Do not specify the SPACE keyword on the FILECTL subcommand. If you specify the SPACE keyword, FPA uses the values that you specify for the keyword to allocate the data sets.

Resolving problems

WER039A INSUFFICIENT VIRTUAL STORAGE

If this message was issued during the SYNC SORT process, use the xTAKCTL keyword to reduce the number of areas to process concurrently, and specify the following DD statement in your JCL.

```
//$ORTPARM DD*  
VSCORE=256K,VSCORET=3M,MINCORE=256K  
/*
```

Tips and best practices for the FPA Resync function

The FPA Resync function supports several keywords that improve the performance of FPA Resync jobs and that further simplify management of DEDBs.

Improving performance

Specify the number of areas to scan concurrently

When you synchronize multiple areas with their secondary index databases, use the ITASKCTL keyword to control the number of areas to scan concurrently. The default value of the ITASKCTL keyword is 4. Change the value to optimize the run time and CPU time of the FPA Resync job.

Increasing the value of the ITASKCTL keyword might affect the resources in the following ways:

- Reduced job run time.
- Slightly increased CPU time.
- Increased region size.

For more information, see [“ITASKCTL keyword” on page 486](#).

Specify the number of secondary indexes to verify concurrently

When you synchronize multiple secondary indexes with their DEDB areas, use the OTASKCTL keyword to control the number of secondary indexes to verify concurrently. The default value of the OTASKCTL keyword is 4. Change the value to optimize the run time and CPU time of the FPA Resync job.

Increasing the value of the OTASKCTL keyword might affect the resources in the following ways:

- Reduced job run time.
- Slightly increased CPU time.
- Increased region size.

For more information, see [“OTASKCTL keyword” on page 503](#).

Specify the interval between synchronization points

Use the SYNC keyword to reduce the number of SYNC calls that FPA issues during a job. The default value of the SYNC keyword is 1000. Therefore, for every 1000 segments, FPA issues a SYNC call to update secondary index databases.

For more information, see [“SYNC keyword” on page 547](#).

Pre-load CIs into data space storage

If you specify ICACHE=YES, FPA pre-loads the control intervals (CIs) from the DEDB areas into data space storage and accesses the CIs in data space storage instead of accessing the CIs in VSAM data sets. Therefore, the run time of the job decreases. The pre-loaded CIs include those in IOVF and SDEP portions of the areas.

ICACHE=NO is recommended only for the following situations:

- You specify a large value for the ITASKCTL keyword. In this case, data from all concurrently processed DEDB areas is pre-loaded into data space storage concurrently, and the amount of data space that the thread tasks of FPA use might be considerable.
- No segments or only a few segments exist in the IOVF and SDEP portions. In this case, there is no advantage to pre-loading IOVF and SDEP portions into data spaces.

For more information, see [“ICACHE keyword” on page 472](#).

Synchronize secondary indexes concurrently by submitting multiple jobs

The FPA Resync function cannot concurrently update secondary indexes in one job. To update multiple secondary indexes concurrently, create multiple Resync jobs and submit them at the same time.

For more information, see the following topics:

- [“Two-step method” on page 323](#)
- [“Verify update method and direct update method when area scan and sort are done in different jobs” on page 322](#)
- [“Example 2: Synchronization using the two-step method” on page 347](#)
- [“Example 3: Synchronization using the verify update method \(running area scan and sort processes separately\)” on page 349](#)

Using the function more effectively and efficiently

Perform syntax check

If you specify SCAN=YES on the GLOBAL command, FPA does not run the build index process but it checks the syntax of the commands, keywords, and parameters that you specify in the JCL. Use this option if you want to verify the syntax of HFPSYSIN control statements when you create new JCL.

For more information, see [“SCAN keyword” on page 523](#).

Automatically estimate the space required for unverified resync pointer segment record data sets

FPA can automatically estimate the data set size and dynamically allocate the data sets that are required for unverified resync pointer segment records. Specify the following keywords:

- Specify the RESYNCMODE=AREASCAN option on the INDEXBLD command.
- Specify the FILECTL subcommand.
- Specify the DISP=NEW option on the FILECTL subcommand.

Do not specify the SPACE keyword on the FILECTL subcommand. If you specify the SPACE keyword, FPA uses the values that you specify for the keyword to allocate the data sets.

Resolving problems

WER039A INSUFFICIENT VIRTUAL STORAGE

If this message was issued during the SYNC SORT process, use the xTAKCTL keyword to reduce the number of areas to process concurrently, and specify the following DD statement in your JCL.

```
//$ORTPARM DD*  
VSCORE=256K,VSCORET=3M,MINCORE=256K  
/*
```

Tips and best practices for the FPA Freespace Reclaim function

The FPA Freespace Reclaim function supports several keywords that improve the performance of FPA Freespace Reclaim jobs and help you use the function more effectively and efficiently.

Improving performance

Suppress the printing of Secondary Index Freespace Reclaim report

If you specify FS_RECLAIM_REPORT=YES, FPA generates the Secondary Index Freespace Reclaim report. FPA issues IDCAMS EXAMINE commands to collect information from the secondary index database data sets, and this process increases the elapsed time and CPU time required for the Freespace Reclaim job. The job runs faster if you specify FS_RECLAIM_REPORT=NO to suppress the printing of the report.

For more information, see [“FS_RECLAIM_REPORT keyword”](#) on page 468.

Using the function more effectively and efficiently

Perform syntax check

If you specify SCAN=YES on the GLOBAL command, FPA does not run the build index process but it checks the syntax of the commands, keywords, and parameters that you specify in the JCL. Use this option if you want to verify the syntax of HFPSYSIN control statements when you create new JCL.

For more information, see [“SCAN keyword”](#) on page 523.

Tips and best practices for the FPA Initialize function

The FPA Initialize function supports several keywords that improve the performance of FPA Initialize jobs and that further simplify management of DEDBs.

Improving performance

Specify the number of areas to initialize concurrently

When you initialize multiple areas, use the OTASKCTL keyword to control the number of areas to initialize concurrently. The default value of the OTASKCTL keyword is 4. Change the value to optimize the run time and CPU time of the FPA Initialize job.

Increasing the value of the OTASKCTL keyword might affect the resources in the following ways:

- Reduced job run time.
- Slightly increased CPU time.
- Increased region size.

For more information, see [“OTASKCTL keyword”](#) on page 503.

Using the function more effectively and efficiently

Perform syntax check

If you specify SCAN=YES on the GLOBAL command, FPA does not run the initialize process but it checks the syntax of the commands, keywords, and parameters that you specify in the JCL. Use this option if you want to verify the syntax of HFPSYSIN control statements when you create new JCL.

For more information, see [“SCAN keyword” on page 523](#).

Tips and best practices for IMS Fast Path Online Tools

FPO supports many keywords that simplify the management of DEDBs. The following topics describe how to use those keywords to improve the performance of FPO jobs and further simplify management of DEDBs.

Tips and best practices for the FPO Online Pointer Checker utility

The FPO Online Pointer Checker utility (OPC) supports several keywords that improve the performance of OPC jobs and that further simplify management of DEDBs.

Improving performance

Perform a quick scan of online areas

If you specify the FULLSTEP=NO option, OPC performs a quick scan to check the database integrity and collects pointer verification data. You can use the data as input for the DEDB Pointer Checker utility of FPB to run an in-depth pointer check and produce a complete set of analysis reports. A job that uses FULLSTEP=NO runs faster than a job that uses FULLSTEP=YES. Consequently, online data availability is improved. The disadvantage to using FULLSTEP=NO is that you must run FPB jobs to perform in-depth pointer check and obtain a complete set of analysis reports.

For more information, see [“FULLSTEP keyword” on page 882](#).

Reduce the time required for the image copy process

If you specify the IC=(YES,ICEIN) option, OPC starts IMS HP Image Copy to generate image copies of the areas. If you also specify the DBBUF keyword on the ICEIN DD, which is the DD statement for controlling the IMS HP Image Copy process, you can control the number of buffers that IMS HP Image Copy uses to access the areas. Specifying a large value for the DBBUF keyword generally improves performance.

For more information, see the DBBUF keyword in the *IMS High Performance Image Copy User's Guide*.

Using the function more effectively and efficiently

Perform syntax check

If you specify PCHKONLY=YES, OPC does not run the pointer check process but it checks the syntax of the control statements, keywords, and parameters that you specify in the JCL. Use this option if you want to verify the syntax of control statements when you create new JCL.

For more information, see [“PCHKONLY keyword” on page 901](#).

Generate a complete set of analysis reports

If you specify TYPRUN=RPT and FULLSTEP=YES options, OPC performs in-depth pointer check and generates a complete set of analysis reports in one job step.

For more information, see [“TYPRUN keyword” on page 935](#) and [“FULLSTEP keyword” on page 882](#).

Generate image copies during the job

If you specify IC=YES or IC=(YES,*parameter*), OPC starts the IMS Database Image Copy utility (DFSUDMPO) or IMS HP Image Copy to generate image copies of the areas. If you use this option, you do not need to code another job step to generate image copies.

When you specify IC=(YES,*parameter*), you must specify the IMS HP Image Copy library on the STEPLIB DD statement. If you specify IC=(YES,ICEIN), you can use the ICEIN DD statement to specify the control statements for IMS HP Image Copy.

Tips and best practices for the FPO Online Data Extract utility

The FPO Online Data Extract utility (ODE) supports keywords that further simplify management of DEDBs.

Improving performance

ODE does not support keywords for improving the job performance.

Using the function more effectively and efficiently

Perform syntax check

If you specify PCHKONLY=YES, ODE does not run the extract process but it checks the syntax of the control statements, keywords, and parameters that you specify in the JCL. Use this option if you want to verify the syntax of control statements when you create new JCL.

For more information, see [“PCHKONLY keyword” on page 901](#).

Sort extracted records during the job

If you specify SORT=YES but no ODESORT DD statement, ODE sorts the extracted records during the job. If you use this option, you do not need to code another job step to sort the extracted records.

For more information, see [“SORT keyword” on page 927](#).

Tips and best practices for the FPO Online Area Extender utility

The FPO Online Area Extender utility (OAE) supports several keywords that improve the performance of OAE jobs and that further simplify management of DEDBs.

Improving performance

Select the fastest run mode

OAE supports three run modes: shadow expansion, original expansion with the EXTEND_TYPE=COPY option, and original expansion with the EXTEND_TYPE=RENAME option. To extend the area quickly, use shadow expansion. If you want to use original expansion, original expansion with the EXTEND_TYPE=RENAME option is faster than original expansion with EXTEND_TYPE=COPY. Before you select a mode, see [Select the simplest run mode](#), and review the additional required tasks that accompany each mode.

Using the function more effectively and efficiently

Perform syntax check

If you specify PCHKONLY=YES, OAE does not run the extend process but it checks the syntax of the control statements, keywords, and parameters that you specify in the JCL. Use this option if you want to verify the syntax of control statements when you create new JCL.

For more information, see [“PCHKONLY keyword” on page 901](#).

Select the simplest run mode

OAE supports three run modes: shadow expansion, original expansion with the EXTEND_TYPE=COPY option, and original expansion with the EXTEND_TYPE=RENAME option. The simplest approach is to use original expansion with the EXTEND_TYPE=COPY option. Original expansion with the EXTEND_TYPE=RENAME option is simpler than shadow expansion.

The following additional tasks accompany each mode:

- If you use shadow expansion, before you run the OAE job, you must manually allocate the target area data sets and register them to the RECON data sets.
- If you use original expansion with the EXTEND_TYPE=RENAME option, after the OAE job runs, you must manually run the OAE Rename program. During the rename process, the area is temporarily stopped in all sharing IMS systems.
- If you use original expansion with the EXTEND_TYPE=COPY option, no additional tasks are required.

For more information, see [“EXTEND_TYPE keyword” on page 875](#).

Perform area expansion multiple times without modifying the JCL job

If you need to expand an area occasionally, specify the expansion size as a percentage. For the IOVF portion, OAE reads the size to expand from the IOVF_UOW keyword. For the SDEP portion, OAE reads the size to expand from the SDEP_CI keyword. OAE accepts integers and percentages. If you specify the values as percentages, you can reuse the job without modification.

For more information, see [“IOVF_UOW keyword” on page 889](#) and [“SDEP_CI keyword” on page 913](#).

Tips and best practices for the FPO Online Expert Reorganization utility

The FPO Online Expert Reorganization utility (OER) supports several keywords that improve the performance of OER jobs and that further simplify management of DEDBs.

Improving performance

Limit the number of UOWs to reorganize

If you specify the number of UOWs to reorganize on the REORGLIM keyword of the REORG statement, you can minimize the impact on other critical applications that access the area and maximize the effect of reorganization.

For more information, see [“REORGLIM keyword” on page 906](#).

Perform a quick scan of online areas

If you specify the PTRCHK=YES option, OER calls OPC during the job. If you also specify the FULLSTEP=NO option, OPC performs a quick scan to check the database integrity and collects pointer verification data. You can use the data as input for the DEDB Pointer Checker utility of FPB to run an in-depth pointer check and produce a complete set of analysis reports. A job that uses FULLSTEP=NO runs faster than a job that uses FULLSTEP=YES. Consequently, online data availability is improved. The disadvantage to using FULLSTEP=NO is that you must run FPB jobs to perform in-depth pointer check and obtain a complete set of analysis reports.

For more information, see [“PTRCHK keyword” on page 902](#) and [“FULLSTEP keyword” on page 882](#).

Reduce the time required for the image copy process

If you specify the PTRCHK=YES option, OER calls OPC during the job. If you also specify the IC=(YES,ICEIN) option, OPC starts IMS HP Image Copy to generate image copies of the areas. If you also specify the DBBUF keyword on the ICEIN DD, which is the DD statement for controlling the IMS HP Image Copy process, you can control the number of buffers that IMS HP Image Copy uses to access the areas. Specifying a large value for the DBBUF keyword generally improves performance.

For more information, see the DBBUF keyword in the *IMS High Performance Image Copy User's Guide*.

Using the function more effectively and efficiently

Perform syntax check

If you specify PCHKONLY=YES, OER does not run the reorganization process but it checks the syntax of control statements, keywords, and parameters that you specify in the JCL. Use this option if you want to verify the syntax of control statements when you create new JCL.

For more information, see [“PCHKONLY keyword” on page 901](#).

Generate a complete set of analysis reports

If you specify PTRCHK=YES, OER starts OPC after it reorganizes the area. If you also specify TYPRUN=RPT and FULLSTEP=YES options, OPC performs in-depth pointer check and generates a complete set of analysis reports in one job step.

For more information, see [“PTRCHK keyword” on page 902](#), [“TYPRUN keyword” on page 935](#), and [“FULLSTEP keyword” on page 882](#).

Generate image copies during the job

If you specify PTRCHK=YES, OER starts OPC after it reorganizes the area. If you also specify IC=YES or IC=(YES,*parameter*), OPC starts the IMS Database Image Copy utility (DFSUDMPO) or IMS HP Image Copy to generate image copies of the areas. If you use this option, you do not need to code another job step to generate image copies.

When you specify IC=(YES,*parameter*), you must specify the IMS HP Image Copy library on the STEPLIB DD statement. If you specify IC=(YES,ICEIN), you can use the ICEIN DD statement to specify the control statements for IMS HP Image Copy.

Tips and best practices for the FPO Online DMAC Print utility

The FPO Online DMAC Print utility (ODM) supports a keyword that further simplifies management of DEDBs.

Improving performance

ODM does not support keywords for improving the job performance.

Using the function more effectively and efficiently

Perform syntax check

If you specify PCHKONLY=YES, ODM does not run the DMAC printing process but it checks the syntax of the control statements, keywords, and parameters that you specify in the JCL. Use this option if you want to verify the syntax of control statements when you create new JCL.

For more information, see [“PCHKONLY keyword” on page 901](#).

Tips and best practices for the Online Space Management utility

The Online Space Management utility (OSM) supports several keywords that simplify the management of DEDBs. The following sections describe how to use those keywords to improve the performance of OSM jobs and further simplify management of DEDBs.

Improving performance

Limit the number of UOWs to reorganize

If you specify the number of UOWs to reorganize on the REORGLIM keyword of the SETREORGPARM subcommand, you can minimize the impact on other critical applications that access the area and maximize the effect of reorganization.

For more information, see [“REORGLIM keyword” on page 997](#).

Select the faster run mode for area expansion

If you specify the EXTEND_TYPE=RENAME option for the SETEXTENDPARM subcommand, you can expand the area faster than when you specify EXTEND_TYPE=COPY.

Consideration: If you use the EXTEND_TYPE=RENAME option, after the OSM job runs, you must manually run the OAE Rename program. During the rename process, the area is temporarily stopped in all sharing IMS systems.

For more information, see [“EXTEND_TYPE keyword” on page 989](#).

Using the function more effectively and efficiently

Perform syntax check

If you specify SCAN=YES on the GLOBAL command, OSM does not run the reorganization or area expansion process but it checks the syntax of the control statements, keywords, and parameters that you specify in the JCL. Use this option if you want to verify the syntax of control statements when you create new JCL.

Note: The IMS online system must be started before you submit the job.

For more information, see [“SCAN keyword” on page 983](#).

Select simpler run mode for area expansion

If you specify the EXTEND_TYPE=RENAME option on the SETEXTENDPARM subcommand, you must manually run the OAE Rename program after the job. However, if you specify the EXTEND_TYPE=COPY option, you do not need manual intervention after the job.

For more information, see [“EXTEND_TYPE keyword” on page 989](#).

Perform area expansion multiple times without modifying the JCL job

If you need to expand an area occasionally, specify the expansion size as a percentage. For the IOVF portion, OSM reads the size to expand from the IOVF_UOW keyword. For the SDEP portion, OSM reads the size to expand from the SDEP_CI keyword. OSM accepts integers and percentages. If you specify the values as percentages, you can reuse the job without modification.

For more information, see [“IOVF_UOW keyword” on page 990](#) and [“SDEP_CI keyword” on page 991](#).

Tips and best practices for the Online Structure Change utility

The Online Structure Change utility (OSC) supports several keywords that simplify DEDB structure change. The following sections describe how to use those keywords to improve the performance of OSC jobs and further simplify DEDB structure change.

Improving performance

Specify the number of areas to restructure concurrently

When you restructure multiple areas, use the ITASKCTL and OTASKCTL keywords to control the number of areas to restructure concurrently. The default values of the ITASKCTL and OTASKCTL keyword are 4. Change the value to optimize the run time and CPU time of the OSC job.

Increasing the value of the ITASKCTL and OTASKCTL keyword might affect the resources in the following ways:

- Reduced job run time
- Slightly increased CPU time
- Increased region size.

For more information, see [“ITASKCTL keyword” on page 1042](#) and [“OTASKCTL keyword” on page 1044](#).

Using the function more effectively and efficiently

Perform syntax check

If you specify SCAN=YES on the GLOBAL command, the Online Structure Change utility does not run the restructure process but it checks the syntax of the commands, keywords, and parameters that you specify in the JCL. Use this option if you want to verify the syntax of HFPSYSIN control statements when you create new JCL.

For more information, see [“SCAN keyword” on page 1038](#).

Analyze DEDB areas during the job

If you specify ANALYZE=AFTERLOAD, the Online Structure Change utility generates a xxxxxxK0 internal job to analyze the restructured areas during the reload phase.

For more information, see [“ANALYZE keyword” on page 1040](#).

Chapter 43. DEDB monitoring and tuning guide

You can use the utilities of IMS Fast Path Solution Pack to collect statistics data from DEDB areas, be notified of exceptional states of areas, and generate statistical reports to tune the areas.

The following topics explain how to monitor DEDB areas by providing examples of the control statement keywords and the DD statements that you can use to monitor DEDB areas.

Tips:

- The *IMS Solution Packs IMS Database Space Tuning Guide* also provides details about DEDB monitoring and tuning. The information in the *IMS Solution Packs IMS Database Space Tuning Guide* pertains to policy-based monitoring of DEDBs. However, even if you do not use policy-based monitoring, you can use the guide to learn how to interpret the analysis reports and what tuning activity is recommended for specific conditions of DEDB areas.

Before using the guide, see [“Comparison of the exception detection methods”](#) on page 1120 to identify the corresponding Policy Services exceptions for the exceptional states that you observed by using the threshold capabilities of IMS HP Fast Path Utilities.

- You can automate DEDB monitoring tasks by using the Online Space Management utility in the active autonomies environment. The utility automates free space reorganization, IOVF extension, and SDEP extension of areas. In this environment, Autonomics Director periodically collects and evaluates sensor data, makes recommendations, and starts Online Space Management utility jobs to perform the recommended actions. For more information see [“Online Space Management utility in the active autonomies environment”](#) on page 963.

Topics:

- [“Summary of DEDB monitoring and tuning steps”](#) on page 1111
- [“Collecting and storing statistical data for DEDB areas”](#) on page 1112
- [“Detecting exceptional states of DEDB areas”](#) on page 1120
- [“Determining the actions to resolve the detected exceptions”](#) on page 1125
- [“JCL examples for monitoring and tuning DEDBs”](#) on page 1129

Summary of DEDB monitoring and tuning steps

DEDB monitoring involves collecting various types of statistical data that represent states of the areas. DEDB tuning typically involves setting the database and area attributes to minimize the physical I/O requirements. The utilities of IMS Fast Path Solution Pack provide the functions that you need to monitor and tune your DEDBs.

DEDB tuning is a subjective process. Tuning decisions that you make depend on application activities that were performed and that will be performed against the database. Generally, however, you must consider the following factors:

Free space utilization

Predetermining the reorganization criteria to maximize the free space utilization

Definition and sizing of UOW and overflow

Selecting the optimal CI size and the UOW and ROOT parameter values to minimize the overflow to IOVF

Randomizing efficiency

Tuning the randomizer to optimize utilization of RAP CIs and to minimize overflow of database records

A tuned DEDB area has certain performance characteristics. A well-tuned area typically has the following characteristics:

No or modest use of IOVF

- The percentage of UOWs that use IOVF is low or zero.
- The average and maximum numbers of IOVF CIs that are used by a UOW are low.
- The percentage of database records that use IOVF is low.

Sufficient free space in RAA for each UOW

- The percentage of free space in each section of the area—that is, the RAA BASE section, the DOVF section, and the IOVF section—is sufficient after a free space reorganization.
- The percentage of available IOVF CIs is sufficient so that IOVF CIs are not used up during application activities.

Minimum I/O to access both root segments and database records

- The average and maximum physical I/O that is required to retrieve a root segment are few.
- The average and maximum physical I/O that is required to retrieve a database record are few.

DEDB monitoring, therefore, involves collecting this statistical data and detecting exceptional states of DEDB areas. DEDB tuning involves determining the best approach to resolve exceptions to achieve the ideal state of the area. The utilities of IMS Fast Path Solution Pack help you perform these activities through the following steps:

Step 1: Run an IMS Fast Path Solution Pack utility to collect statistics data of the area.

IMS Fast Path Solution Pack provides various utilities to collect statistical data of a DEDB area. You can select from these utilities to collect statistical data for detecting exceptions.

For the methods for collecting statistics data, see [“Collecting and storing statistical data for DEDB areas” on page 1112.](#)

Step 2: Detect exceptional states of the area.

You can determine if the area is in an exceptional state by using the methods that are provided by IMS Fast Path Solution Pack utilities, IMS Tools Base Policy Services, or user programs.

For the methods for detecting exceptional states, see [“Detecting exceptional states of DEDB areas” on page 1120.](#)

Step 3: Determine actions to resolve the detected exceptions.

If an exceptional state of an area is detected, you must review the area. The utilities of IMS Fast Path Solution Pack provide various reports that help you review the area and understand the space utilization problems.

For details about analysis and tuning of DEDB areas by using the utility reports, see [“Determining the actions to resolve the detected exceptions” on page 1125.](#)

Collecting and storing statistical data for DEDB areas

You can select from various utilities of IMS Fast Path Solution Pack to collect statistical data to detect exceptions.

Typically the following items are monitored to detect exceptional states of a DEDB area:

- Overflow usage
- Free space availability
- Free space relationships
- Root I/O
- Record I/O
- Synonym chain length

You can use one of the following IMS Fast Path Solution Pack utilities to collect data for all or subset of the preceding monitoring items:

- DEDB analysis utilities:
 - The Analyze function of IMS Fast Path Advanced Tool (FPA)
 - DEDB Pointer Checker of IMS Fast Path Basic Tools (FPB)
 - Online Pointer Checker of IMS Fast Path Online Tools (FPO)
- Reorganization utilities:
 - The Change, Reload, and Reorg functions of FPA (with the Analyze process)
 - Online Expert Reorganization utility (with the OPC process)
- Image copy utility:
 - The DEDB HASH Check function of IMS HP Image Copy

Also, you have the following two options to store the collected data:

- Statistics history data set (History file and History2 file)
- Sensor Data repository

The following topics explain the JCL specifications that are required to run these utilities and to specify the resource for storing the collected data:

- [“Storing data in the statistics history data set \(History file\)” on page 1113](#)
- [“Storing data in the statistics history data set \(History2 file\)” on page 1115](#)
- [“Storing data in the Sensor Data repository of IMS Tools KB” on page 1117](#)

Storing data in the statistics history data set (History file)

To collect and store statistics data in the statistics history data set (History file), you must specify certain parameters in the utility JCL.

About this task

The History records of DEDB areas can be created and stored as an option of various IMS Fast Path Solution Pack utilities that generate statistics data reports.

The History file is a sequential data set of fixed-length records. The physical characteristics of the History file are DSORG=PS, RECFM=FB, and LRECL=100. The data in the History file is expected to be used by user application programs that analyze the state of DEDB areas.

By using the FPA Analyze function, the FPA Change function, the FPA Reload function, and the FPA Reorg function, you can choose to collect all the statistics data items or only a subset of the data items. By default, only the data items that belong to the first three categories in the following list are collected:

- Overflow usage
- Free space availability
- Free space relationship
- Root I/O
- Record I/O
- Synonym chain length

To collect all the data items, you must specify PTRCHKLVL=FULL.

For a list of data items and the layout of the History file records, see [“History file records layout” on page 1173](#).

Collecting only a subset of the data items is faster than collecting all the data items and is usually sufficient to detect symptoms of the space utilization exceptions. Use the PTRCHKLVL=FULL option to collect all data items if you want to see how efficiently the root segments are randomized and how efficiently database records can be read hierarchically and sequentially in terms of the number of I/O operations.

The History records can also be populated by using Online Pointer Checker, Online Expert Reorganization, and DEDB Pointer Checker. When the History file function is activated, these utilities collect and store all the data items in all the six categories.

Procedure

Use one of the following methods:

Storing data by using a function of FPA

To store the data in the History file by using the FPA Analyze function, the FPA Change function, the FPA Reload function, or the FPA Reorg function, complete the following steps:

- a. Prepare the History file. For instructions for preparing the History file for the FPA functions, see [“Preprocess for the Analyze process: Initializing permanent data set” on page 197](#).
- b. Specify the HFP AHST DD statement or the DD statement that is specified by the HISTORYDD keyword parameter to specify the HISTORY data set.
- c. Specify the REPORT subcommand option.
- d. Optional: If you want to collect all the data items, specify the PTRCHKLVL=FULL option. If you want to collect only the subset of the data items, you can skip this step.
- e. Run the job.

You can find coding examples in the following topics:

- [“FPA Analyze JCL examples for monitoring and tuning DEDBs” on page 1129](#)
- [“FPA Reload JCL examples for monitoring and tuning DEDBs” on page 1135](#)
- [“FPA Change JCL examples for monitoring and tuning DEDBs” on page 1141](#)
- [“FPA Reorg JCL examples for monitoring and tuning DEDBs” on page 1148](#)

Storing data by using OPC

To store the data in the History file by using the Online Pointer Checker utility, complete the following steps:

- a. Prepare the History file. For instructions for preparing the History file for the FPO utilities, see [“Preprocess for OPC: Initializing permanent data set” on page 682](#).
- b. Specify the FPX AHST DD statement or the DD statement that is specified by the HISTORYDD keyword parameter to specify the HISTORY data set.
- c. Specify the TYPRUN=RPT option for the ANALYZE statement.
- d. Specify the FULLSTEP=YES option for the ANALYZE statement.
- e. Run the job.

You can find a coding example in [“Online Pointer Checker JCL examples for monitoring and tuning DEDBs” on page 1157](#).

Storing data by using OER

To store the data in the History file by using the Online Expert Reorganization utility, complete the following steps:

- a. Prepare the History file. For instructions for preparing the History file for the FPO utilities, see [“Preprocess for OPC: Initializing permanent data set” on page 682](#).
- b. Specify the FPX AHST DD statement or the DD statement that is specified by the HISTORYDD keyword parameter to specify the HISTORY data set.
- c. Specify the PTRCHK=YES option for the REORG statement to call the OPC process.
- d. Specify the TYPRUN=RPT option for the PTRCHK statement.
- e. Specify the FULLSTEP=YES option for the PTRCHK statement.
- f. Specify the CHKUOW=ALL option for the PTRCHK statement.

Note: The options specified for the PTRCHK statement are applied to the OPC process.

g. Run the job.

You can find a coding example in [“Online Expert Reorganization JCL examples for monitoring and tuning DEDBs”](#) on page 1162.

Storing data by using the DEDB Pointer Checker utility of FPB

To store the data in the History file by using the FABADA5 program of the DEDB Pointer Checker utility, complete the following steps:

- a. Prepare the History file. For instructions for preparing the History file for the DEDB Pointer Checker utility, see the topic "Preprocess for DEDB Pointer Checker" in the *IMS Fast Path Solution Pack: IMS Fast Path Basic Tools User's Guide*.
- b. Add the HISTORY DD statement to specify the HISTORY data set.
- c. Run the job.

You can find a coding example in [“DEDB Pointer Checker JCL examples for monitoring and tuning DEDBs”](#) on page 1154.

Storing data in the statistics history data set (History2 file)

To collect and store UOW-group level statistics data in the History2 data set (History2 file), you must specify certain parameters and DD statements in the utility JCL.

About this task

The History2 records for UOW groups that are defined for a DEDB area can be created and stored as an option of the following processes:

- Analyze process of FPA
- OPC process of FPO
- DEDB Pointer Checker utility of FPB

A set of UOW groups is defined by a sequential data set called the *UOW group definition data set*. A control statement record in the UOW group definition data set defines a group of UOWs of an area by a UOW range that is bounded by a start UOW number and a stop UOW number.

The History2 file is a sequential data set of fixed-length records. The physical characteristics of the History2 file are DSORG=PS, RECFM=FB, and LRECL=200. The data in the History2 file is expected to be used by user application programs that analyze the state of DEDB areas at the UOW-group level.

The data items in the following categories are collected for each UOW group and are stored in a History2 file record:

- Overflow usage
- Free space availability
- Free space relationship
- Root I/O
- Record I/O
- Synonym chain length

Information about the UOW and ROOT parameter values and the CI size that is defined in the DBD and the low and high UOW numbers that define a UOW group are also stored in the History2 file record.

For more information, see the following resources:

- For the layout of UOW group definition records, see [“UOW group definition record layout”](#) on page 1191.
- For a list of data items and the layout of History2 file records, see [“History2 file records layout”](#) on page 1175.

Procedure

Use one of the following methods:

Storing data by using a function of FPA

To store the data in the History2 file by using the FPA Analyze function, the FPA Change function, the FPA Reload function, or the FPA Reorg function, complete the following steps:

- a. Prepare the History2 file. For instructions for preparing the History2 file for the FPA functions, see [“Preprocess for the Analyze process: Initializing permanent data set” on page 197.](#)
- b. Prepare the content of the UOW group definition data set to define UOW groups by UOW ranges.
- c. Specify the REPORT subcommand and the PTRCHKLVL=FULL option for the Analyze process.
- d. Specify the HFXAUOWC DD statement for the Analyze process to identify the UOW group definition data set.
- e. Specify the HFXAHST2 DD statement for the Analyze process to identify the History2 file.
- f. Run the job.

You can find coding examples in the following topics:

- [“FPA Analyze JCL examples for monitoring and tuning DEDBs” on page 1129](#)
- [“FPA Reload JCL examples for monitoring and tuning DEDBs” on page 1135](#)
- [“FPA Change JCL examples for monitoring and tuning DEDBs” on page 1141](#)
- [“FPA Reorg JCL examples for monitoring and tuning DEDBs” on page 1148](#)

Storing data by using OPC

To store the data in the History2 file by using the Online Pointer Checker utility, complete the following steps:

- a. Prepare the History2 file. For instructions for preparing the History2 file for the FPO utilities, see [“Preprocess for OPC: Initializing permanent data set” on page 682.](#)
- b. Prepare the content of the UOW group definition data set to define UOW groups by UOW ranges.
- c. Specify the FFXAUOWC DD statement for the OPC process to identify the UOW group definition data set.
- d. Specify the FFXAHST2 DD statement for the OPC process to identify the History2 file.
- e. Specify the TYPRUN=RPT option for the ANALYZE statement.
- f. Specify the FULLSTEP=YES option for the ANALYZE statement.
- g. Run the job.

You can find a coding example in [“Online Pointer Checker JCL examples for monitoring and tuning DEDBs” on page 1157.](#)

Storing data by using OER

To store the data in the History2 file by using the Online Expert Reorganization utility, complete the following steps:

- a. Prepare the History2 file. For instructions for preparing the History2 file for the FPO utilities, see [“Preprocess for OPC: Initializing permanent data set” on page 682.](#)
- b. Prepare the content of the UOW group definition data set to define UOW groups by UOW ranges.
- c. Specify the FFXAUOWC DD statement for the OPC process to identify the UOW group definition data set.
- d. Specify the FFXAHST2 DD statement for the OPC process to identify the History2 file.
- e. Specify the PTRCHK=YES option for the REORG statement to call the OPC process.
- f. Specify the TYPRUN=RPT option for the PTRCHK statement.
- g. Specify the FULLSTEP=YES option for the PTRCHK statement.
- h. Specify the CHKUOW=ALL option for the PTRCHK statement.

Note: The options specified for the PTRCHK statement are applied to the OPC process.

i. Run the job.

You can find a coding example in [“Online Expert Reorganization JCL examples for monitoring and tuning DEDBs”](#) on page 1162.

Storing data by using the DEDB Pointer Checker utility of FPB

To store the data in the History2 file by using the DEDB Pointer Checker utility, complete the following steps:

- a. Prepare the History2 file. For instructions for preparing the History2 file for the DEDB Pointer Checker utility, see the topic "Preprocess for DEDB Pointer Checker" in the *IMS Fast Path Solution Pack: IMS Fast Path Basic Tools User's Guide*.
- b. Prepare the content of the UOW group definition data set to define UOW groups by UOW ranges.
- c. Specify the TYPRUN=RPT or MODEL option for the FABADA1 job step.
- d. Specify the UOWCTL DD statement for the FABADA5 job step to identify the UOW group definition data set.
- e. Specify the HISTORY2 DD statement for the FABADA5 job step to identify the History2 file.
- f. Run the job.

You can find a coding example in [“DEDB Pointer Checker JCL examples for monitoring and tuning DEDBs”](#) on page 1154.

Storing data in the Sensor Data repository of IMS Tools KB

To collect and store statistics data in the Sensor Data repository of IMS Tools Knowledge Base (IMS Tools KB), use the FPA functions, the OPC process of FPO, Stand-alone DB Sensor, or the DEDB HASH Check function of IMS HP Image Copy.

About this task

The Sensor Data repository is a centralized statistics data repository of IMS Tools Knowledge Base. The statistics data for DEDB areas can be collected and stored by the functions of FPA, the OPC process of FPO, and Stand-alone DB Sensor. You can also use the DEDB HASH Check function of IMS HP Image Copy, but this function collects and stores only a subset of the sensor data.

By using the FPA Analyze function, the FPA Change function, the FPA Reload function, and the FPA Reorg function, you can choose to collect all the statistics data items or only a subset of the data items. By default, only the data items that belong to the first three categories in the following list are collected:

- Overflow usage
- Free space availability
- Free space relationship
- Root I/O
- Record I/O
- Synonym chain length
- Segment and record occurrence

To collect all the data items, you must specify PTRCHKLVL=FULL.

Collecting only a subset of the data items is faster than collecting all the data items and is usually sufficient to detect symptoms of the space utilization exceptions. Use the PTRCHKLVL=FULL option and collect all data items if you want to see how efficiently the root segments are randomized and how efficiently database records can be read hierarchically and sequentially in terms of the number of I/O operations.

The statistics data that is stored in the Sensor Data repository is referred to as *sensor data* and is used by various IMS Tools products. The sensor data is stored in the Sensor Data repository as a group (or a set) of

records made up of *data element values*. A data element consists of a data element tag and a data element value pair.

With OPC and the internal OPC process in OER, you can choose to collect all the statistics data items or only a subset of the data items. To collect all the data items, you must specify TYPRUN=RPT and FULLSTEP=YES.

For details about Sensor Data repository and a list of data elements, see the *Tools Base Policy Services User's Guide*.

Procedure

Use one of the following methods:

Storing data by using a function of FPA

To store the sensor data in the Sensor Data repository by using the FPA Analyze function, the FPA Change function, the FPA Reload function, or the FPA Reorg function, specify the following options for the HFPSYSIN control statements:

- a. Specify the ITKBSRVR keyword for the GLOBAL command, and specify the IMS Tools KB XCF group name for its parameter.
- b. Specify the SENSOR=YES option for the function command.
- c. If you want to collect all the data items, specify the PTRCHKLVL=FULL option, the SENSOR_DBREC=YES option, and the SENSOR_UOW=YES option for the function command.

Tips:

- If you want to collect only the subset of the data items, you can skip this step.
- If you do not want to store the sensor data for the data elements that belong to the synonym chain length, root I/O, and record I/O categories in the Sensor Data repository, specify SENSOR_DBREC=NO. For a list of data elements that are collected by the SENSOR_DBREC keyword, see [“SENSOR_DBREC keyword” on page 536](#).
- If you do not want to store the sensor data for the data elements that are related to UOW in the Sensor Data repository, specify SENSOR_UOW=NO.

You can find coding examples in the following topics:

- [“FPA Analyze JCL examples for monitoring and tuning DEDBs” on page 1129](#)
- [“FPA Reload JCL examples for monitoring and tuning DEDBs” on page 1135](#)
- [“FPA Change JCL examples for monitoring and tuning DEDBs” on page 1141](#)
- [“FPA Reorg JCL examples for monitoring and tuning DEDBs” on page 1148](#)

Storing data by using OPC

To store the sensor data in the Sensor Data repository by using OPC, specify the following options for the FPXIN control statements:

- a. Specify the ITKBSRVR keyword for the GLOBAL statement, and specify the IMS Tools KB XCF group name for its parameter.
- b. Specify the SENSOR=YES option for the ANALYZE statement.
- c. If you want to collect all the data items, specify the following options for the ANALYZE statement:
 - TYPRUN=RPT
 - FULLSTEP=YES
 - SENSOR_DBREC=YES
 - SENSOR_UOW=YES

If you want to collect only the subset of the data items, specify one of the following options:

- TYPRUN=PTRSCAN
- TYPRUN=FS

- TYPRUN=PTRALL
- TYPRUN=RPT and FULLSTEP=NO

Tips:

- If you do not want to store the sensor data for the data elements that belong to the synonym chain length, root I/O, and record I/O categories in the Sensor Data repository, specify SENSOR_DBREC=NO. For a list of data elements that are collected by the SENSOR_DBREC keyword, see [“SENSOR_DBREC keyword” on page 923](#).
- If you do not want to store the sensor data for the data elements that are related to UOW in the Sensor Data repository, specify SENSOR_UOW=NO.

You can find a coding example in [“Online Pointer Checker JCL examples for monitoring and tuning DEDBs” on page 1157](#).

Storing data by using OER

To store the sensor data in the Sensor Data repository by using OER, specify the following options for the FPXIN control statements:

- Specify the ITKBSRVR keyword for the GLOBAL statement, and specify the IMS Tools KB XCF group name for its parameter.
- Specify the PTRCHK=YES option for the REORG statement to call the OPC process.
- Specify the SENSOR=YES option for the PTRCHK statement.
- If you want to collect all the data items, specify the following options for the PTRCHK statement:
 - TYPRUN=RPT
 - FULLSTEP=YES
 - SENSOR_DBREC=YES
 - SENSOR_UOW=YES

If you want to collect only the subset of the data items, specify one of the following options:

- TYPRUN=PTRSCAN
- TYPRUN=FS
- TYPRUN=PTRALL
- TYPRUN=RPT and FULLSTEP=NO

Tips:

- If you do not want to store the sensor data for the data elements that belong to the synonym chain length, root I/O, and record I/O categories in the Sensor Data repository, specify SENSOR_DBREC=NO. For a list of data elements that are collected by the SENSOR_DBREC keyword, see [“SENSOR_DBREC keyword” on page 923](#).
 - If you do not want to store the sensor data for the data elements that are related to UOW in the Sensor Data repository, specify SENSOR_UOW=NO.
- Specify the CHKUOW=ALL option for the PTRCHK statement.

Note: The options specified for the PTRCHK statement are applied to the OPC process.

You can find a coding example in [“Online Expert Reorganization JCL examples for monitoring and tuning DEDBs” on page 1162](#).

Storing data by using Stand-alone DB Sensor

To store the sensor data in the Sensor Data repository by using Stand-alone DB Sensor, run the DB Sensor job.

Tips:

- If you do not want to store the sensor data for the data elements that belong to the synonym chain length, root I/O, and record I/O categories in the Sensor Data repository, specify SENSOR_DBREC=NO.

- If you do not want to store the sensor data for the data elements that are related to UOW in the Sensor Data repository, specify SENSOR_UOW=NO.

For information about using Stand-alone DB Sensor and coding examples, see the *IMS Solution Packs Data Sensor User's Guide*.

Storing data by using the DEDB HASH Check function of IMS HP Image Copy

To store the subset of sensor data in the Sensor Data repository by using the DEDB HASH Check function of IMS HP Image Copy, specify the following options for the ICEIN control statements:

- Specify the DEDBPC=Y option for the GLOBAL command.
- Specify the ITKBSRVR keyword for the GLOBAL command, and specify the IMS Tools KB XCF group name for its parameter.
- Specify the SENSOR=YES option for the GLOBAL command.

You can find a coding example in [“IMS HP Image Copy JCL examples for monitoring and tuning DEDBs” on page 1167](#).

When the DEDB HASH Check function of IMS HP Image Copy is used, only the following data items are stored in the Sensor Data repository:

- Overflow usage
- Free space availability
- Free space relationship

The following data items are not stored:

- Root I/O
- Record I/O
- Synonym chain length

For information about using the DEDB HASH Check function, see the *IMS High Performance Image Copy User's Guide*.

Detecting exceptional states of DEDB areas

You can detect exceptional states of DEDB areas by using the threshold capabilities of IMS HP Fast Path Utilities, Policy Services, or user-written programs.

The following three methods are supported for detecting exceptional states of a DEDB area.

- [“Method 1: Using thresholds in FPA Analyze process” on page 1123](#)
- [“Method 2: Using thresholds in DEDB Pointer Checker” on page 1123](#)
- [“Method 3: Using Policy Services” on page 1124](#)

See [“Comparison of the exception detection methods” on page 1120](#) to understand the difference between the methods.

You can also use your own programs to detect exceptional states of DEDB areas by using the records that are stored in the History or History 2 file. These records can be used to calculate statistical values for your own analysis. However, this method is beyond the scope of this information.

Comparison of the exception detection methods

Exceptions that can be detected by each method are different. Use the method that best meets your preferences.

Method 1: Using thresholds in the FPA Analyze process

The THRESHOLD subcommand for the FPA Analyze process defines the thresholds for detecting exceptions. The FPA Analyze process can be called from the following functions and utilities:

- FPA Analyze function
- FPA Change function (with the PTRCHKLVL=QUICK or PTRCHKLVL=FULL option)
- FPA Reload function (with the PTRCHKLVL=QUICK or PTRCHKLVL=FULL option)
- FPA Reorg function (with the PTRCHKLVL=QUICK or PTRCHKLVL=FULL option)
- FPO Online Pointer Checker utility (with the FULLSTEP=YES and TYPRUN=RPT options)
- FPO Online Expert Reorganization utility (with the PTRCHK=YES, FULLSTEP=YES, TYPRUN=RPT, and CHKUOW=ALL options)

Note: The statistics data that is collected by OPC and OER is processed by the FPA Analyze process, which is invoked internally in the same job step.

In [Table 131 on page 1122](#), the keywords that can be specified for the THRESHOLD subcommand, which define the thresholds for detecting exceptions, are listed in the column "Method 1: THRESHOLD parameters (FPA Analyze and Tune process)"

Method 2: Using thresholds in the DEDB Pointer Checker process

The THRCTL JCL DD statement for the FABADA5 program of the DEDB Pointer Checker utility defines the thresholds for detecting exceptions. This method can be used by the following utilities:

- FPB DEDB Pointer Checker
- FPO Online Pointer Checker (with the FULLSTEP=NO and one of the TYPRUN=RPT, TYPRUN=FS, and TYPRUN=MODEL options)
- FPO Online Expert Reorganization (with the PTRCHK=YES, FULLSTEP=NO, and one of the TYPRUN=RPT, TYPRUN=FS, and TYPRUN=MODEL options)

In [Table 131 on page 1122](#), the exception threshold items that can be specified on the THRCTL JCL DD statement are listed in the column "Method 2: FABADA5 exception thresholds (DEDB Pointer Checker utility)."

Method 3: Using Policy Services

IMS Tools Base Autonomics Director uses IMS Tools Base Policy Services to detect exceptions based on the user-defined reorganization policy. Sensor data that is stored by IMS Fast Path Solution Pack utilities is used for policy-based exception detection.

In a policy, you can define up to three levels of thresholds for each exception class. Exception levels that are supported are warning, severe, and critical. In [Table 131 on page 1122](#), the exception classes that can be detected by Policy Services and rules that are used to detect those exceptions are listed in the column "Method 3: Policy Services exceptions."

Method 4: Using thresholds in the FPA Tune process

The THRESHOLD subcommand for the FPA Tune process defines the thresholds for detecting exceptions. The FPA Tune process can be called from the FPA Tune function.

In [Table 131 on page 1122](#), the keywords that can be specified for the THRESHOLD subcommand, which define the thresholds for detecting exceptions, are listed in the column "Method 1: THRESHOLD parameters (FPA Analyze and Tune process)."

Exceptions that can be detected in each method

The following table summarizes the exceptions that can be detected by using each method.

Table 131. Exceptions that can be detected in each method

Exception category	Method 1: THRESHOLD parameters (FPA Analyze and Tune process)	Method 2: FABADA5 exception thresholds (DEDB Pointer Checker process)	Method 3: Policy Services exceptions and originating rules	Description
Overflow usage	IOVFUSED_PCT	Not available	DEDB_EXCESSIVE_IOVF_CI_USED Rule: IBM.DEDB_OVERFLOW.70	The percentage of the number of IOVF CIs used exceeded a threshold.
	Not available	Available	Not available	The percentage of IOVF CIs that are available in the area is lower than a threshold.
	Use IOVFUSED_PCT as an alternative.		Use DEDB_EXCESSIVE_IOVF_CI_USED as an alternative.	
	Not available	Not available	DEDB_EXCESS_RAP_CI_USING_OVFLOW Rule: IBM.DEDB_OVERFLOW.80	The percentage of RAP CIs that are using the overflow area among all used RAP CIs exceeded a threshold.
	RAPOVFL_PCT	Available	Not available	The percentage of RAP CIs that are using the overflow area among all allocated RAP CIs exceeded a threshold.
	UOWDOVF_PCT	Available	DEDB_EXCESS_PCT_UOWS_USING_DOVF Rule: IBM.DEDB_OVERFLOW.10	The percentage of UOWs that are using DOVF exceeded a threshold.
	UOWIOVF_PCT	Available	DEDB_EXCESS_PCT_UOWS_USING_IOVF Rule: IBM.DEDB_OVERFLOW.20	The percentage of UOWs that are using IOVF exceeded a threshold.
	UOWIOVF_AVG	Available	DEDB_EXCESS_AVG_IOVF_CI_PER_UOW Rule: IBM.DEDB_OVERFLOW.40	The average use of IOVF CIs per UOW exceeded a threshold.
	UOWIOVF_MAX	Not available	DEDB_UOW_USING_EXCESSIVE_IOVF_CI Rule: IBM.DEDB_OVERFLOW.50	The maximum use of IOVF CIs by a UOW exceeded a threshold.
	Not available	Available	DEDB_EXCESS_MIN_IOVF_CI_PER_UOW Rule: IBM.DEDB_OVERFLOW.60	The minimum number of IOVF CIs that are used by a UOW exceeded a threshold.
	Not available	Not available	DEDB_EXCESS_NUM_UOWS_USING_IOVF Rule: IBM.DEDB_OVERFLOW.30	The maximum use of IOVF CIs by a UOW exceeded a threshold.
RECIOVF_PCT	Available	DEDB_EXCESSIVE_DBREC_USING_IOVF Rule: IBM.DEDB_OVERFLOW.90	The percentage of DB records that are using IOVF exceeded a threshold.	
Free space availability	RAAFS_PCT	Available	DEDB_FREE_SPACE_AVAIL_IN_RAA Rule: IBM.DEDB_FS.10	The percentage of free space in RAA BASE fell below a threshold.
	DOVFFS_PCT	Available	DEDB_FREE_SPACE_AVAIL_IN_DOVF Rule: IBM.DEDB_FS.20	The percentage of free space in DOVF fell below a threshold.
	IOVFFS_PCT	Available	DEDB_FREE_SPACE_AVAIL_IN_IOVF Rule: IBM.DEDB_FS.30	The percentage of free space in DOVF fell below a threshold.
	Not available	Available	DEDB_FREE_SPACE_AVAIL_IN_SDEP Rule: IBM.DEDB_FS.40	The percentage of free space in SDEP fell below a threshold.
Free space relationship	FS_RAADOVFIOVF	Available	DEDB_FREE_SPACE_IN_RAA_VS_OVFLOW Rule: IBM.DEDB_FS.70	Free spaces in RAA, DOVF, and IOVF are used inefficiently.
	FS_DOVFIOVF	Available	DEDB_FREE_SPACE_AVAIL_IN_OVFLOW Rule: IBM.DEDB_FS.60	Free spaces in DOVF and IOVF are used inefficiently.
	FS_RAADOVF	Not available	DEDB_FREE_SPACE_IN_RAA_VS_DOVF Rule: IBM.DEDB_FS.40	Free spaces in RAA BASE and DOVF are used inefficiently.
	FS_RAATIOVF	Not available	DEDB_FREE_SPACE_IN_RAA_VS_IOVF Rule: IBM.DEDB_FS.50	Free spaces in RAA BASE and IOVF are used inefficiently.
Root I/O	ROOTIO_AVG	Available	DEDB_EXCESSIVE_AVG_NUM_ROOT_IO Rule: IBM.DEDB_ROOT_IO.10	The average number of I/Os that are required to read a root segment exceeded a threshold.
	ROOTIO_MAX	Available	DEDB_ROOT_SEGMENT_WITH_EXCESS_IO Rule: IBM.DEDB_ROOT_IO.20	The maximum number of I/Os that are required to read a root segment exceeded a threshold.
Record I/O	RECIO_AVG	Available	DEDB_EXCESSIVE_AVG_NUM_RECORD_IO Rule: IBM.DEDB_DBREC_IO.10	The average number of I/Os that are required to read a database record exceeded a threshold.
	RECIO_MAX	Available	DEDB_DBRECORD_WITH_EXCESSIVE_IO Rule: IBM.DEDB_DBREC_IO.20	The maximum number of I/Os that are required to read a database record exceeded a threshold.

Table 131. Exceptions that can be detected in each method (continued)

Exception category	Method 1: THRESHOLD parameters (FPA Analyze and Tune process)	Method 2: FABADA5 exception thresholds (DEDB Pointer Checker process)	Method 3: Policy Services exceptions and originating rules	Description
Synonym chain	SYN_PCT_LEN	Available	Not available	The percentage of RAPs that have long synonym chains exceeded a threshold.
	SYN_LEN_AVG	Available	DEDB_EXCESSIVE_AVG_LEN_SYNONYMS Rule: IBM.DEDB_SYN_LEN.10	The average length of RAP synonym chains exceeded a threshold.
	SYN_MAX	Available	DEDB_LONG_SYNONYM_CHAIN Rule: IBM.DEDB_SYN_LEN.20	The maximum length of RAP synonym chains exceeded a threshold.
	RAP_LONGSYN_PCT (only for the TUNE process)	Not available	Not available	The percentage of RAPs that have long synonym chains among all used RAPs exceeded a threshold.
Segment and record occurrence	Not available	Not available	NUMBER_OF_DB_RECORDS Rule: IBM.NUM_DBRECORDS.10	The number of database records exceeded a threshold.
	Not available	Not available	EXCESSIVE_SEGMENT_OCCURRENCES Rule: IBM.DEDB_SEGM_CNT.10	The number of segment occurrences exceeded a threshold.

Method 1: Using thresholds in FPA Analyze process

Use the THRESHOLD subcommand keywords to define the threshold values for the FPA Analyze process.

Procedure

1. Ensure that the PTRCHKLVL=FULL option is specified for the ANALYZE command.
2. Code the THRESHOLD subcommand and its keywords and parameters.

See “THRESHOLD subcommand” on page 435 for the format, the keywords, and the parameters of the THRESHOLD subcommand.

You can find coding examples in the following topics:

- [“FPA Analyze JCL examples for monitoring and tuning DEDBs” on page 1129](#)
- [“FPA Reload JCL examples for monitoring and tuning DEDBs” on page 1135](#)
- [“FPA Change JCL examples for monitoring and tuning DEDBs” on page 1141](#)
- [“FPA Reorg JCL examples for monitoring and tuning DEDBs” on page 1148](#)

Results

When a threshold exception is detected, a return code of 4 is issued, and one or more 0510W messages are printed in the Process Summary of Analysis report.

For an example of the exception notification by the 0510W messages, see the following topics:

- [“Process Summary of Analysis report” on page 590](#)
- [Message “0510W” on page 1423](#)

Method 2: Using thresholds in DEDB Pointer Checker

Define the threshold values for the FPB DEDB Pointer Checker FABADA5 program in the THRCTL DD data set.

About this task

You can implement threshold-based database monitoring with DEDB Pointer Checker by selecting one of the following TYPRUN modes for the FABADA1 job step:

TYPRUN=FS

Integrity verification scan and free space reports

TYPRUN=RPT

Full integrity checking and report generation

TYPRUN=MODEL

Full integrity checking and generation of report and tuning data

When TYPRUN=FS, the following exceptions can be monitored:

- Overflow usage exceptions
- Free space availability exceptions
- Free space relationship exceptions

For TYPRUN=RPT or MODEL, in addition to the free space values, the following exceptions can be monitored:

- Root I/O exceptions
- Record I/O exceptions
- Synonym chain exceptions

TYPRUN=RPT is the system default value for the FABADA1 job step.

Procedure

Code the threshold values in the THRCTL DD data set for the FABADA5 program. Specify appropriate values for each of the thresholds.

For the format of threshold values in the THRCTL DD data set, see the topic "FABADA5 THRCTL DD data set" in the *IMS Fast Path Solution Pack: IMS Fast Path Basic Tools User's Guide*.

You can find a coding example in "[DEDB Pointer Checker JCL examples for monitoring and tuning DEDBs](#)" on page 1154

Results

If the utility detects threshold exceptions, a return code of 99 is set for the utility job step, and one or more FABAO599W messages are printed in the SYSPRINT data set. When you receive this signal, review your database.

For an example of the exception notification by the FABAO599W messages, see the following topics in the *IMS Fast Path Solution Pack: IMS Fast Path Basic Tools User's Guide*:

- Topic "FABADA5 SYSPRINT DD data set"
- Message FABAO599W

Method 3: Using Policy Services

You can use IMS Tools Base Policy Services to define policies for detecting exceptional states of DEDB areas and notifying you about exceptions.

Procedure

Use the Policy Services ISPF user interface to tune the IBM-provided default policy for DEDB areas or to define a policy for a specific area or for a specific group of areas. Then, use the Autonomics Director ISPF user interface to schedule policy evaluation for your database resources.

In the policy evaluation process, Policy Services evaluates the sensor data that is stored in the Sensor Data repository and detects exceptional states of the DEDB areas. If Policy Services detects exceptions,

you are notified about the exceptions by an HFP2xxxx message and an accompanying list of detected exceptions.

For more information, see the following resources:

- For a summary of policy-based database monitoring tasks and the recommended actions to remove the detected exceptions, see the *IMS Solution Packs IMS Database Space Tuning Guide*.
- For details about exceptions that can be detected by Policy Services and for instructions for using Policy Services, see the *Tools Base Policy Services User's Guide*.
- For instructions for running the policy evaluation process, see the *Tools Base Autonomics Director User's Guide*.

Method 4: Using thresholds in FPA Tune process

Use the THRESHOLD subcommand keywords to define the threshold values for the FPA Tune process.

Procedure

Code the THRESHOLD subcommand and its keywords and parameters.

See [“THRESHOLD subcommand” on page 435](#) for the format, the keywords, and the parameters of the THRESHOLD subcommand.

You can find coding examples in [“Examples for the Tune function” on page 253](#).

Results

The Area Tuning Summary report contains the exceptions and area statistics based on the thresholds specified by the THRESHOLD subcommand keywords. The report contains the results obtained from the current area condition and the simulated condition. For an example of the Area Tuning Summary report, see [“Area Tuning Summary report” on page 649](#).

When a threshold exception is detected, one or more 0510W messages are printed in the Process Summary of Analysis report, the TUNED Process Summary of Analysis report, or both.

For an example of the exception notification by the 0510W messages, see the following topics:

- [“Process Summary of Analysis report” on page 590](#)
- [“TUNED reports” on page 653](#)
- Message [“0510W” on page 1423](#)

Determining the actions to resolve the detected exceptions

When an exceptional state of an area is detected, review the area to determine the appropriate resolution.

In general, if you find that the database record profile and the number of database records in the area are relatively unchanged, reorganizing the free space in the area might resolve the exception. Conversely, if you find that the number of database records or the record profile has changed significantly, performance tuning for the area, expansion of the area, or both might be required.

You can use the tutorial that is provided in the *IMS Solution Packs IMS Database Space Tuning Guide* to determine the actions to resolve exception conditions. The tutorial uses sample reports for a DEDB area to help you understand how each of the IMS HP Fast Path utility reports is related with detected exceptions and how you can use these reports to analyze and tune DEDB areas.

Generating the analysis reports

IMS Fast Path Solution Pack utilities can generate various reports that help you to analyze and tune DEDB areas.

About this task

The analysis reports can be generated by the Analyze, Change, Reload, and Reorg functions of FPA, the OPC and the OER utilities of FPO, the DEDB Pointer Checker utility of FPB, and IMS HP Image Copy. However, each function or utility cannot generate all of the reports.

The following table summarizes the reports that each function or utility can generate.

Table 132. Analysis reports printed by each utility

Report number	Report	FPA	FPO	FPB	IMS HP Image Copy
1	Freespace Analysis report	Printable	Printable	Printable	Printable
2	DB Record Profile Analysis report	Printable	Printable	Printable	-
3	DB Record Placement Analysis report	Printable	Printable	Printable	-
4	Segment Placement Analysis report	Printable	Printable	Printable	-
5	Segment I/O Analysis report	Printable	Printable	Printable	-
6	Synonym Chain Analysis report	Printable	Printable	Printable	-
7	UOW report (free space and overflow statistics)	Printable	Printable	Printable	-
8	UOW report (full statistics including the number of database records and the average database record length)	Printable	Printable	Printable	-
9	UOW Exception report	Printable	Printable	Printable	-
10	Largest Database Records report	Printable	Printable	Printable	-
11	Segment Length Distribution report	Printable	-	-	Printable
12	Consolidated Largest Database Records report	Printable	Printable	-	-

Procedure

Use the following methods to generate the analysis reports:

Generating the reports by using a function of FPA

To generate the reports by using the FPA Analyze function, the FPA Change function, the FPA Reload function, or the FPA Reorg function, complete these steps:

- a. Specify the REPORT subcommand.
This specification generates reports 1 and 7.
- b. Specify the PTRCHKLVL=FULL option.
This specification generates reports 2, 3, 4, 5, 6, and 8.
- c. Specify the LARGEREC_REPORT=YES option.
This specification generates report 10, and is also required to generate report 12.
- d. Specify a value that is equal to or greater than 1 for the second parameter of the LARGEREC_COUNT keyword.
This specification generates report 12.
- e. Specify the SEGLDIST=YES option.
This specification generates report 11.
- f. Run the job.

Notes:

- The FPA Analyze function can use an image copy of the area, instead of the area data set, to generate these reports.
- The UOW Exception report (report 9) is generated only for those UOWs in which one of the conditions that are defined by the RBASEFS, RDOVFFS, and UDOVFFS parameters (in the HFPAPARM DD statement) is met.

You can find coding examples in the following topics:

- [“FPA Analyze JCL examples for monitoring and tuning DEDBs” on page 1129](#)
- [“FPA Reload JCL examples for monitoring and tuning DEDBs” on page 1135](#)
- [“FPA Change JCL examples for monitoring and tuning DEDBs” on page 1141](#)
- [“FPA Reorg JCL examples for monitoring and tuning DEDBs” on page 1148](#)

Generating the reports by using the Tune function of FPA

By using the Tune function, you can generate analysis reports for the current areas as well as for the simulated areas. The FPA Tune function also generates an Area Tuning Summary report that contains information about threshold exceptions, statistics, and DBD definitions of the current areas and the simulated areas.

With the Tune function, reports 1 through 8 are always generated. To generate other reports, complete these steps:

- Specify the LARGEREC_REPORT=YES option.
This specification generates report 10, and is also required to generate report 12.
- Specify a value that is equal to or greater than 1 for the second parameter of the LARGEREC_COUNT keyword.
This specification generates report 12.
- Specify the SEGLDIST=YES option.
This specification generates report 11.
- Run the job.

Note: The UOW Exception report (report 9) is generated only for those UOWs in which one of the conditions that are defined by the RBASEFS, RDOVFFS, and UDOVFFS parameters (in the HFPAPARM DD statement) is met.

You can find coding examples in [“Examples for the Tune function” on page 253](#).

Generating the reports by using a utility of FPO

To generate the reports by using the Online Pointer Checker utility or the Online Expert Reorganization utility, complete these steps:

- Specify the TYPRUN=RPT option.
- If you use the Online Expert Reorganization utility, also specify the PTRCHK=YES option.
- Specify the FULLSTEP=YES option.
This specification generates reports 1, 2, 3, 4, 5, 6, 7, and 8.
- Specify the LARGEREC_REPORT=YES option.
This specification generates report 10, and is also required to generate report 12.
- Specify a value that is equal to or greater than 1 for the second parameter of the LARGEREC_COUNT keyword.
This specification generates report 12.
- Run the job.

Note: The UOW Exception report (report 9) is generated only for those UOWs in which one of the conditions that are defined by the EXC_RBASEFS, EXC_RDOVFFS, and EXC_UDOVFFS parameters (in the FPXIN DD statement) is met.

You can find coding examples in the following topics:

- [“Online Pointer Checker JCL examples for monitoring and tuning DEDBs” on page 1157](#)

- [“Online Expert Reorganization JCL examples for monitoring and tuning DEDBs” on page 1162](#)

Generating the reports by using the DEDB Pointer Checker utility of FPB

To generate the reports by using the FABADA5 program of the DEDB Pointer Checker utility, complete these steps:

- Specify the LARGEREC DD statement and the TYPRUN=MODEL option.
This specification generates report 10.
The FABADA5 program generates reports 1, 2, 3, 4, 5, 6, 7, and 8 by default.
- Run the job.

Notes:

- The DEDB Pointer Checker utility can use an image copy of the area, instead of the area data set, to generate these reports.
- The UOW Exception report (report 9) is generated only for those UOWs in which one of the conditions that are defined by the RBASEFS, RDOVFFS, and UDOVFFS parameters (in the PARMCTL DD statement) is met.

You can find a coding example in [“DEDB Pointer Checker JCL examples for monitoring and tuning DEDBs” on page 1154](#)

Generating the reports by using the DEDB HASH Check function of IMS HP Image Copy

To generate the reports by using the DEDB HASH Check function of IMS HP Image Copy, complete these steps:

- Use the FABJMAIN program (the EXEC PGM parameter must specify FABJMAIN).
- Specify the DEDB HASH Check option (DEDBPC=Y).
This specification generates report 1.
- Specify the SEGLDIST parameter as the second parameter of the DEDBPC keyword (DEDBPC=(Y,SEGLDIST)).
This specification generates report 11.
- Run the job.

You can find a coding example in [“IMS HP Image Copy JCL examples for monitoring and tuning DEDBs” on page 1167.](#)

Related reference

Freespace Analysis report

The Freespace Analysis report provides the information about the free space in an area.

DB Record Profile Analysis report

The DB Record Profile Analysis report contains the analysis information of the database records in an area.

DB Record Placement Analysis report

The DB Record Placement Analysis report includes the number of database records that are contained in each of the seven placement categories.

Segment Placement Analysis report

The Segment Placement Analysis report provides an overview of where the segments reside in the database and can be used as introductory information for I/O analysis.

Segment I/O Analysis report

The Segment I/O Analysis report provides information for the average database record.

Synonym Chain Analysis report

The Synonym Chain Analysis report provides detailed information about the frequency and length of synonym chains and statistics about root placement and physical I/O requirements.

UOW report

The optional report for UOW analysis provides detailed information about each UOW.

UOW Exception report

The UOW Exception report contains the analysis information of the UOW that satisfies the criteria specified by the HFPAPARM keywords.

Largest Database Records report

The optional report for the largest database records analysis provides a description for each of the large database records, starting with the largest database record.

JCL examples for monitoring and tuning DEDBs

Use the following JCL examples to monitor and tune DEDB areas.

The following topics provide JCL examples that you can use to monitor and tune DEDBs. For each utility that you can use to monitor and tune DEDBs, these topics provide JCL examples for collecting statistics data, storing the data in the Sensor Data repository or history data sets, and generating analysis reports.

- [“FPA Analyze JCL examples for monitoring and tuning DEDBs” on page 1129](#)
- [“FPA Reload JCL examples for monitoring and tuning DEDBs” on page 1135](#)
- [“FPA Change JCL examples for monitoring and tuning DEDBs” on page 1141](#)
- [“FPA Reorg JCL examples for monitoring and tuning DEDBs” on page 1148](#)
- [“DEDB Pointer Checker JCL examples for monitoring and tuning DEDBs” on page 1154](#)
- [“Online Pointer Checker JCL examples for monitoring and tuning DEDBs” on page 1157](#)
- [“Online Expert Reorganization JCL examples for monitoring and tuning DEDBs” on page 1162](#)
- [“IMS HP Image Copy JCL examples for monitoring and tuning DEDBs” on page 1167](#)

FPA Analyze JCL examples for monitoring and tuning DEDBs

Use these JCL examples to monitor and tune DEDBs by using the FPA Analyze function.

Subsections:

- [“JCL template for running the FPA Analyze function” on page 1129](#)
- [“Example 1: Storing statistics in the HISTORY data set” on page 1130](#)
- [“Example 2: Storing statistics in the HISTORY2 data set” on page 1130](#)
- [“Example 3: Storing statistics in the Sensor Data repository” on page 1131](#)
- [“Example 4: Specifying FPA threshold values” on page 1131](#)
- [“Example 5: Generating free space analysis reports” on page 1132](#)
- [“Example 6: Generating reports to obtain database records and randomizing information” on page 1132](#)
- [“Example 7: Generating reports to obtain large database records information” on page 1133](#)
- [“Example 8: Generating reports for all UOWs” on page 1134](#)
- [“Example 9: Generating reports for UOWs in which exceptions were detected” on page 1134](#)

JCL template for running the FPA Analyze function

The following figure shows a JCL template for running the FPA Analyze function.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input area data sets (ADSs) to be validated are dynamically allocated. The allocation information (DD name and DS name of the ADSs) is obtained from DBRC.
- The PTRCHKLVL keyword is not specified; therefore, the default value (PTRCHKLVL=QUICK) is applied, and fast scan of database integrity verification is performed.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
ANALYZE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1)
/*

```

Figure 394. JCL template for FPA Analyze

Example 1: Storing statistics in the HISTORY data set

The following figure shows example JCL for storing the statistical information of an area that is registered in DBRC into the HISTORY data set.

Prerequisite: If you are using the HISTORY data set for the first time, initialize the permanent data sets by following the procedures in “Preprocess for the Analyze process: Initializing permanent data set” on page 197. Also see “History file records layout” on page 1173 for information about the layout of the History file records.

In this example:

- The HFP AHST DD statement specifies the HISTORY data set.

Tip: Instead of using the HFP AHST DD statement, you can specify a different DD statement that is specified by the HISTORYDD= keyword parameter.

- PTRCHKLVL=FULL for the ANALYZE command specifies that the integrity verification process runs in FULL mode.
- The REPORT subcommand specifies that the statistics are stored in the HISTORY data set.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFP AHST DD DISP=(MOD,KEEP,KEEP),DSN=HPFP.HISTORY
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
ANALYZE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL
REPORT
/*

```

Figure 395. FPA Analyze example 1: Storing statistics in the HISTORY data set

Example 2: Storing statistics in the HISTORY2 data set

The following figure shows example JCL for storing the statistical information of a UOW group into the HISTORY2 data set.

Prerequisite: If you are using the HISTORY2 data set for the first time, initialize the permanent data sets by following the procedures in “Preprocess for the Analyze process: Initializing permanent data set” on page 197. Also see “History2 file records layout” on page 1175 for information about the layout of the History2 file records.

In this example:

- The HFP AHST2 DD statement specifies the HISTORY2 data set.

- The HFP AUOWC DD statement specifies the group of UOWs for the area.
- PTRCHKLVL=FULL for the ANALYZE command specifies that the integrity verification process runs in FULL mode.
- The REPORT subcommand specifies that the statistics are stored in the HISTORY2 data set.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPAST2 DD DISP=(MOD,KEEP,KEEP),DSN=HPFP.HISTORY2
//HFP AUOWC DD DISP=SHR,DSN=HPFP.UOWCTL
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
ANALYZE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL
REPORT
/*
```

Figure 396. FPA Analyze example 2: Storing statistics in the HISTORY2 data set

Example 3: Storing statistics in the Sensor Data repository

The following figure shows example JCL for storing the statistical information of an entire area that is registered in DBRC into the Sensor Data repository.

Prerequisite: To run this JCL, an IMS Tools Knowledge Base server and its repositories must be configured.

In this example:

- The IMS Tools Base library (ITBVS.SHKTLOAD) is concatenated to the STEPLIB DD.
- ITKBSRVR= for the GLOBAL command specifies the IMS Tools KB server XCF group.
- PTRCHKLVL=FULL for the ANALYZE command specifies that the integrity verification process runs in FULL mode.
- SENSOR=YES for the ANALYZE command specifies that the statistics are stored in the Sensor Data repository.
- SENSOR_DBREC=YES for the ANALYZE command specifies that the statistics that are related to database records and root segment distribution are collected.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=ITBVS.SHKTLOAD
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES,
  ITKBSRVR=FPQSRVFP
ANALYZE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL,
  SENSOR=YES,
  SENSOR_DBREC=YES
/*
```

Figure 397. FPA Analyze example 3: Storing statistics in the Sensor Data repository

Example 4: Specifying FPA threshold values

The following figure shows example JCL for specifying the threshold values of FPA.

In this example:

- PTRCHKLVL=FULL for the ANALYZE command specifies that the integrity verification process runs in FULL mode.
- The REPORT subcommand specifies that the Process Summary of Analysis report is generated.
- The THRESHOLD subcommand enables the threshold settings.
- The keywords and their parameters for the THRESHOLD subcommand specify the threshold values.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
ANALYZE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL
REPORT
THRESHOLD
  RAAFS_PCT=30,
  RECIO_MAX=6
/*
```

Figure 398. FPA Analyze example 4: Specifying FPA threshold values

See [“THRESHOLD subcommand” on page 435](#) for the keywords that you can specify on the THRESHOLD subcommand.

Example 5: Generating free space analysis reports

The following figure shows example JCL for generating the reports that are required for free space analysis.

In this example, the REPORT subcommand specifies that the Freespace Analysis report is generated.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
ANALYZE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1)
REPORT
/*
```

Figure 399. FPA Analyze example 5: Generating reports for free space analysis

Example 6: Generating reports to obtain database records and randomizing information

The following figure shows example JCL for generating the reports that contain information about database records and randomizing.

In this example:

- PTRCHKLVL=FULL for the ANALYZE command specifies that the integrity verification process runs in FULL mode.
- The REPORT subcommand specifies that the DEDB Area Analysis report is generated.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB   DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
ANALYZE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL
REPORT
/*

```

Figure 400. FPA Analyze example 6-1: Generating reports to obtain database records and randomizing information

The following figure shows example JCL for generating the reports that contain SDEP record information, prefix length, and data flags to facilitate diagnosis of the database.

In this example:

- SDEPCHKLVL=FULL for the ANALYZE command specifies that a Segment Placement Analysis report that includes SDEP information is generated.
- SEGLSTAT=DETAILED keyword specifies that a DB Record Profile Analysis report that includes prefix length and data flags is generated.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB   DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
GLOBAL
  DBRC=YES
ANALYZE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  SDEPCHKLVL=FULL,
  PTRCHKLVL=FULL
REPORT
  SEGLSTAT=DETAILED

```

Figure 401. FPA Analyze example 6-2: Generating reports to obtain SDEP record information, prefix length, and data flags

Example 7: Generating reports to obtain large database records information

The following figure shows example JCL for generating the reports that contain information about the large database records.

In this example:

- PTRCHKLVL=FULL for the ANALYZE command specifies that the integrity verification process runs in FULL mode.
- The REPORT subcommand, LARGEREC_REPORT=YES, and LARGEREC_COUNT=(10,20) specify that the Largest Database Records report and the Consolidated Largest Database Records report are generated.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
ANALYZE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL
REPORT
  LARGEREC_REPORT=YES,
  LARGEREC_COUNT=(10,20)
/*

```

Figure 402. FPA Analyze example 7: Generating reports to obtain large database records information

Example 8: Generating reports for all UOWs

The following figure shows example JCL for generating the reports for all UOWs.

In this example:

- PTRCHKLVL=FULL for the ANALYZE command specifies that the integrity verification process runs in FULL mode.
- **Tip:** Use this option when you want to obtain information about database record length.
- The REPORT subcommand specifies that the UOW reports are generated.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
ANALYZE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1)
REPORT
/*

```

Figure 403. FPA Analyze example 8: Generating reports for all UOWs

Example 9: Generating reports for UOWs in which exceptions were detected

The following figure shows example JCL for generating the reports only for the UOWs in which exceptions were detected.

In this example:

- PTRCHKLVL=FULL for the ANALYZE command specifies that the integrity verification process runs in FULL mode.
- **Tip:** Use this option when you want to obtain information about database record length.
- The REPORT subcommand specifies that the UOW Exception reports are generated.
- The HFPA Parm DD statement specifies the threshold values for generating the UOW Exception reports.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPAPARM DD *
           RDOVFFS=40 UDOVFFS=35 RBASEFS=40
/*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
ANALYZE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL
REPORT
/*

```

Figure 404. FPA Analyze example 9: Generating reports for UOWs in which exceptions were detected

FPA Reload JCL examples for monitoring and tuning DEDBs

Use these JCL examples to monitor and tune DEDBs by using the FPA Reload function.

Subsections:

- [“JCL template for running the FPA Reload function” on page 1135](#)
- [“Example 1: Storing statistics in the HISTORY data set” on page 1136](#)
- [“Example 2: Storing statistics in the HISTORY2 data set” on page 1136](#)
- [“Example 3: Storing statistics in the Sensor Data repository” on page 1137](#)
- [“Example 4: Specifying FPA threshold values” on page 1138](#)
- [“Example 5: Generating free space analysis reports” on page 1138](#)
- [“Example 6: Generating reports to obtain database records and randomizing information” on page 1139](#)
- [“Example 7: Generating reports to obtain large database records information” on page 1139](#)
- [“Example 8: Generating reports for all UOWs” on page 1140](#)
- [“Example 9: Generating reports for UOWs in which exceptions were detected” on page 1140](#)

JCL template for running the FPA Reload function

The following figure shows a JCL template for running the FPA Reload function.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input unloaded segment records data set for the area is specified by the IDSNMASK='HPFP.USRFILE.&AREA' parameter. This data set is allocated dynamically.
- The output ADSs to be reloaded are dynamically allocated. The allocation information (DD statement name and DS name of the ADSs) is obtained from DBRC.

Requirement: You must define the output ADSs before running this example.

- After the Reload process, fast scan of database integrity verification is performed to analyze the reloaded areas. The PTRCHKLVL keyword is not specified; therefore, the default value (PTRCHKLVL=QUICK) is applied.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//        DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  IDSNMASK='HPFP.USRFILE.&AREA',
  OAREA=(DB22AR0,DB22AR1)
/*

```

Figure 405. JCL template for FPA Reload

Example 1: Storing statistics in the HISTORY data set

The following figure shows example JCL for storing the statistical information of a reloaded area that is registered in DBRC into the HISTORY data set.

Prerequisite: If you are using the HISTORY data set for the first time, initialize the permanent data sets by following the procedures in [“Preprocess for the Analyze process: Initializing permanent data set”](#) on page 197. Also see [“History file records layout”](#) on page 1173 for information about the layout of the History file records.

In this example:

- The HFP AHST DD statement specifies the HISTORY data set.
- **Tip:** Instead of using the HFP AHST DD statement, you can specify a different DD statement that is specified by the HISTORYDD= keyword parameter.
- PTRCHKLVL=FULL for the RELOAD command specifies that the integrity verification process runs in FULL mode.
- The REPORT subcommand specifies that the statistics are stored in the HISTORY data set.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//        DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFP AHST DD DISP=(MOD,KEEP,KEEP),DSN=HPFP.HISTORY
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  IDSNMASK='HPFP.USRFILE.&AREA',
  OAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL
REPORT
/*

```

Figure 406. FPA Reload example 1: Storing statistics in the HISTORY data set

Example 2: Storing statistics in the HISTORY2 data set

The following figure shows example JCL for storing the statistical information of a UOW group of a reloaded area into the HISTORY2 data set.

Prerequisite: If you are using the HISTORY2 data set for the first time, initialize the permanent data sets by following the procedures in [“Preprocess for the Analyze process: Initializing permanent data set”](#) on page 197. Also see [“History2 file records layout”](#) on page 1175 for information about the layout of the History2 file records.

In this example:

- The HFPAHST2 DD statement specifies the HISTORY2 data set.
- The HFPAUOWC DD statement specifies the group of UOWs for the area.
- PTRCHKLVL=FULL for the RELOAD command specifies that the integrity verification process runs in FULL mode.
- The REPORT subcommand specifies that the statistics are stored in the HISTORY2 data set.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//         DD DISP=SHR,DSN=IMSVS.SDFSRESL
//         DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPAHST2 DD DISP=(MOD,KEEP,KEEP),DSN=HPFP.HISTORY2
//HFPAUOWC DD DISP=SHR,DSN=HPFP.UOWCTL
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  IDSNMASK='HPFP.USRFILE.&AREA',
  OAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL
REPORT
/*
```

Figure 407. FPA Reload example 2: Storing statistics in the HISTORY2 data set

Example 3: Storing statistics in the Sensor Data repository

The following figure shows example JCL for storing the statistical information of an entire reloaded area that is registered in DBRC into the Sensor Data repository.

Prerequisite: To run this JCL, an IMS Tools Knowledge Base server and its repositories must be configured.

In this example:

- The IMS Tools Base library (ITBVS.SHKTLOAD) is concatenated to the STEPLIB DD.
- ITKBSRVR= for the GLOBAL command specifies the IMS Tools KB server XCF group.
- PTRCHKLVL=FULL for the RELOAD command specifies that the integrity verification process runs in FULL mode.
- SENSOR=YES for the RELOAD command specifies that the statistics are stored in the Sensor Data repository.
- SENSOR_DBREC=YES for the RELOAD command specifies that the statistics that are related to database records and root segment distribution are collected.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=ITBVS.SHKTLOAD
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES,
  ITKBSRVR=FPQSRVFP
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  IDSNMASK='HPFP.USRFIL.&AREA',
  OAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL,
  SENSOR=YES,
  SENSOR_DBREC=YES
/*

```

Figure 408. FPA Reload example 3: Storing statistics in the Sensor Data repository

Example 4: Specifying FPA threshold values

The following figure shows example JCL for specifying the threshold values of FPA.

In this example:

- PTRCHKLVL=FULL for the RELOAD command specifies that the integrity verification process runs in FULL mode.
- The REPORT subcommand specifies to generate the Process Summary of Analysis report.
- The THRESHOLD subcommand enables the threshold settings.
- The keywords and their parameters for the THRESHOLD subcommand specify the threshold values.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  IDSNMASK='HPFP.USRFIL.&AREA',
  OAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL
REPORT
THRESHOLD
  RAAFS_PCT=30,
  RECIO_MAX=6
/*

```

Figure 409. FPA Reload example 4: Specifying FPA threshold values

See [“THRESHOLD subcommand” on page 435](#) for the keywords that you can specify on the THRESHOLD subcommand.

Example 5: Generating free space analysis reports

The following figure shows example JCL for generating the reports that are required for free space analysis.

In this example, the REPORT subcommand specifies that the Freespace Analysis report is generated.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB   DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  IDSNMASK='HPFP.USRFIL.&AREA',
  OAREA=(DB22AR0,DB22AR1)
REPORT
/*

```

Figure 410. FPA Reload example 5: Generating reports for free space analysis

Example 6: Generating reports to obtain database records and randomizing information

The following figure shows example JCL for generating the reports that contain information about database records and randomizing.

In this example:

- PTRCHKLVL=FULL for the RELOAD command specifies that the integrity verification process runs in FULL mode.
- The REPORT subcommand specifies that the DEDB Area Analysis report is generated.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB   DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  IDSNMASK='HPFP.USRFIL.&AREA',
  OAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL
REPORT
/*

```

Figure 411. FPA Reload example 6: Generating reports to obtain database records and randomizing information

Example 7: Generating reports to obtain large database records information

The following figure shows example JCL for generating the reports that contain information about the large database records.

In this example:

- PTRCHKLVL=FULL for the RELOAD command specifies that the integrity verification process runs in FULL mode.
- The REPORT subcommand, LARGEREC_REPORT=YES, and LARGEREC_COUNT=(10,20) specify that the Largest Database Records report and the Consolidated Largest Database Records report are generated.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  IDSNMASK='HPFP.USRFIL.&AREA',
  OAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL
REPORT
  LARGEREC_REPORT=YES,
  LARGEREC_COUNT=(10,20)
/*

```

Figure 412. FPA Reload example 7: Generating reports to obtain large database records information

Example 8: Generating reports for all UOWs

The following figure shows example JCL for generating the reports for all UOWs.

In this example:

- PTRCHKLVL=FULL for the RELOAD command specifies that the integrity verification process runs in FULL mode.
- **Tip:** Use this option when you want to obtain information about database record length.
- The REPORT subcommand specifies that the UOW reports are generated.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  IDSNMASK='HPFP.USRFIL.&AREA',
  OAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL
REPORT
/*

```

Figure 413. FPA Reload example 8: Generating reports for all UOWs

Example 9: Generating reports for UOWs in which exceptions were detected

The following figure shows example JCL for generating the reports only for the UOWs in which exceptions were detected.

In this example:

- PTRCHKLVL=FULL for the RELOAD command specifies that the integrity verification process runs in FULL mode.
- **Tip:** Use this option when you want to obtain information about database record length.
- The REPORT subcommand specifies that the UOW Exception reports are generated.
- The HFPAPARM DD statement specifies the threshold values for generating the UOW Exception reports.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPAPARM DD *
           RDOVFFS=40 UDOVFFS=35 RBASEFS=40
/*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
RELOAD
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  IDSNMASK='HPFP.USRFILE.&AREA',
  OAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL
REPORT
/*

```

Figure 414. FPA Reload example 9: Generating reports for UOWs in which exceptions were detected

FPA Change JCL examples for monitoring and tuning DEDBs

Use these JCL examples to monitor and tune DEDBs by using the FPA Change function.

Subsections:

- [“JCL template for running the FPA Change function” on page 1141](#)
- [“Example 1: Storing statistics in the HISTORY data set” on page 1142](#)
- [“Example 2: Storing statistics in the HISTORY2 data set” on page 1143](#)
- [“Example 3: Storing statistics in the Sensor Data repository” on page 1143](#)
- [“Example 4: Specifying FPA threshold values” on page 1144](#)
- [“Example 5: Generating free space analysis reports” on page 1145](#)
- [“Example 6: Generating reports to obtain database records and randomizing information” on page 1145](#)
- [“Example 7: Generating reports to obtain large database records information” on page 1146](#)
- [“Example 8: Generating reports for all UOWs” on page 1146](#)
- [“Example 9: Generating reports for UOWs in which exceptions were detected” on page 1147](#)

JCL template for running the FPA Change function

The following figure shows a JCL template for running the FPA Change function to reorganize multiple areas that are registered with DBRC while renaming the input area data sets (ADSs).

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input ADSs are specified by the IDSNMASK='HPFP.&AREA.ADS1' parameter. These data sets are allocated dynamically.
- The input ADSs are renamed by using a mask before processing the CHANGE command.
- The output ADSs are defined by the ALLOCATE subcommand before processing the CHANGE command.
- The output ADSs to be reorganized are dynamically allocated. The allocation information (DD statement name and DS name of the ADSs) is obtained from DBRC.
- After the reorganization, fast scan of database integrity verification is performed to analyze the areas. The PTRCHKLVL keyword is not specified; therefore, the default value (PTRCHKLVL=QUICK) is applied.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
CHANGE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  IDSNMASK='HPFP.&AREA.ADS1',
  OAREA=(DB22AR0,DB22AR1)
IRENAME
  IAREA=(DB22AR0,DB22AR1),
  DSNAME='HPFP.OLD.&AREA.ADS1'
ALLOCATE
  OAREA=(DB22AR0,DB22AR1),
  SPACE=(CYL,500),
  VOLSER=HFPVOL
/*

```

Figure 415. JCL template for FPA Change

Example 1: Storing statistics in the HISTORY data set

The following figure shows example JCL for storing the statistical information of a reorganized area that is registered in DBRC into the HISTORY data set.

Prerequisite: If you are using the HISTORY data set for the first time, initialize the permanent data sets by following the procedures in [“Preprocess for the Analyze process: Initializing permanent data set”](#) on page 197. Also see [“History file records layout”](#) on page 1173 for information about the layout of the History file records.

In this example:

- The HFP AHST DD statement specifies the HISTORY data set.
- **Tip:** Instead of using the HFP AHST DD statement, you can specify a different DD statement that is specified by the HISTORYDD= keyword parameter.
- PTRCHKLVL=FULL for the CHANGE command specifies that the integrity verification process runs in FULL mode.
- The REPORT subcommand specifies that the statistics are stored in the HISTORY data set.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFP AHST DD DISP=(MOD,KEEP,KEEP),DSN=HPFP.HISTORY
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
CHANGE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  IDSNMASK='HPFP.&AREA.ADS1',
  OAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL
IRENAME
  IAREA=(DB22AR0,DB22AR1),
  DSNAME='HPFP.OLD.&AREA.ADS1'
ALLOCATE
  OAREA=(DB22AR0,DB22AR1),
  SPACE=(CYL,500),
  VOLSER=HFPVOL
REPORT
/*

```

Figure 416. FPA Change example 1: Storing statistics in the HISTORY data set

Example 2: Storing statistics in the HISTORY2 data set

The following figure shows example JCL for storing the statistical information of a UOW group of a reorganized area into the HISTORY2 data set.

Prerequisite: If you are using the HISTORY2 data set for the first time, initialize the permanent data sets by following the procedures in “Preprocess for the Analyze process: Initializing permanent data set” on page 197. Also see “History2 file records layout” on page 1175 for information about the layout of the History2 file records.

In this example:

- The HFP AHST2 DD statement specifies the HISTORY2 data set.
- The HFPAUOWC DD statement specifies the group of UOWs for the area.
- PTRCHKLVL=FULL for the CHANGE command specifies that the integrity verification process runs in FULL mode.
- The REPORT subcommand specifies that the statistics are stored in the HISTORY2 data set.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//        DD DISP=SHR,DSN=IMSVS.SDFSRESL
//        DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFP AHST2 DD DISP=(MOD,KEEP,KEEP),DSN=HPFP.HISTORY2
//HFPAUOWC DD DISP=SHR,DSN=HPFP.UOWCTL
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
CHANGE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  IDSNMASK=' HPFP.&AREA.ADS1 ',
  OAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL
IRENAME
  IAREA=(DB22AR0,DB22AR1),
  DSNNAME=' HPFP.OLD.&AREA.ADS1 '
ALLOCATE
  OAREA=(DB22AR0,DB22AR1),
  SPACE=(CYL,500),
  VOLSER=HFPVOLL
REPORT
/*
```

Figure 417. FPA Change example 2: Storing statistics in the HISTORY2 data set

Example 3: Storing statistics in the Sensor Data repository

The following figure shows example JCL for storing the statistical information of an entire reorganized area that is registered in DBRC into the Sensor Data repository.

Prerequisite: To run this JCL, an IMS Tools Knowledge Base server and its repositories must be configured.

In this example:

- The IMS Tools Base library (ITBVS.SHKTLOAD) is concatenated to the STEPLIB DD.
- ITKBSRVR= for the GLOBAL command specifies the IMS Tools KB server XCF group.
- PTRCHKLVL=FULL for the CHANGE command specifies that the integrity verification process runs in FULL mode.
- SENSOR=YES for the CHANGE command specifies that the statistics are stored in the Sensor Data repository.
- SENSOR_DBREC=YES for the CHANGE command specifies that the statistics that are related to database records and root segment distribution are collected.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=ITBVS.SHKTLOAD
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB   DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES,
  ITKBSRVR=FPQSRVFP
CHANGE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  IDSNMASK='HPFP.&AREA.ADS1',
  OAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL,
  SENSOR=YES,
  SENSOR_DBREC=YES
IRENAME
  IAREA=(DB22AR0,DB22AR1),
  DSNAME='HPFP.OLD.&AREA.ADS1'
ALLOCATE
  OAREA=(DB22AR0,DB22AR1),
  SPACE=(CYL,500),
  VOLSER=HFPVOL
/*

```

Figure 418. FPA Change example 3: Storing statistics in the Sensor Data repository

Example 4: Specifying FPA threshold values

The following figure shows example JCL for specifying the threshold values of FPA.

In this example:

- PTRCHKLVL=FULL for the CHANGE command specifies that the integrity verification process runs in FULL mode.
- The REPORT subcommand specifies that the Process Summary of Analysis report is generated.
- The THRESHOLD subcommand enables the threshold settings.
- The keywords and their parameters for the THRESHOLD subcommand specify the threshold values.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB   DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
CHANGE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  IDSNMASK='HPFP.&AREA.ADS1',
  OAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL
IRENAME
  IAREA=(DB22AR0,DB22AR1),
  DSNAME='HPFP.OLD.&AREA.ADS1'
ALLOCATE
  OAREA=(DB22AR0,DB22AR1),
  SPACE=(CYL,500),
  VOLSER=HFPVOL
REPORT
THRESHOLD
  RAAFS_PCT=30,
  RECIO_MAX=6
/*

```

Figure 419. FPA Change example 4: Specifying FPA threshold values

See “THRESHOLD subcommand” on page 435 for the keywords that you can specify on the THRESHOLD subcommand.

Example 5: Generating free space analysis reports

The following figure shows example JCL for generating the reports that are required for free space analysis.

In this example, the REPORT subcommand specifies that the Freespace Analysis report is generated.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
CHANGE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  IDSNMASK='HPFP.&AREA.ADS1',
  OAREA=(DB22AR0,DB22AR1)
IRENAME
  IAREA=(DB22AR0,DB22AR1),
  DSNAME='HPFP.OLD.&AREA.ADS1'
ALLOCATE
  OAREA=(DB22AR0,DB22AR1),
  SPACE=(CYL,500),
  VOLSER=HFPPVOL
REPORT
/*
```

Figure 420. FPA Change example 5: Generating reports for free space analysis

Example 6: Generating reports to obtain database records and randomizing information

The following figure shows example JCL for generating reports that contain information about database records and randomizing.

In this example:

- PTRCHKLVL=FULL for the CHANGE command specifies that the integrity verification process runs in FULL mode.
- The REPORT subcommand specifies that the DEDB Area Analysis report is generated.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
CHANGE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  IDSNMASK='HPFP.&AREA.ADS1',
  OAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL
IRENAME
  IAREA=(DB22AR0,DB22AR1),
  DSNAME='HPFP.OLD.&AREA.ADS1'
ALLOCATE
  OAREA=(DB22AR0,DB22AR1),
  SPACE=(CYL,500),
  VOLSER=HFPVOL
REPORT
/*

```

Figure 421. FPA Change example 6: Generating reports to obtain database records and randomizing information

Example 7: Generating reports to obtain large database records information

The following figure shows example JCL for generating the reports that contain information about the large database records.

In this example:

- PTRCHKLVL=FULL for the CHANGE command specifies that the integrity verification process runs in FULL mode.
- The REPORT subcommand, LARGEREC_REPORT=YES, and LARGEREC_COUNT=(10,20) specify that the Largest Database Records report and the Consolidated Largest Database Records report are generated.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
CHANGE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  IDSNMASK='HPFP.&AREA.ADS1',
  OAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL
IRENAME
  IAREA=(DB22AR0,DB22AR1),
  DSNAME='HPFP.OLD.&AREA.ADS1'
ALLOCATE
  OAREA=(DB22AR0,DB22AR1),
  SPACE=(CYL,500),
  VOLSER=HFPVOL
REPORT
  LARGEREC_REPORT=YES,
  LARGEREC_COUNT=(10,20)
/*

```

Figure 422. FPA Change example 7: Generating reports to obtain large database records information

Example 8: Generating reports for all UOWs

The following figure shows example JCL for generating the reports for all UOWs.

In this example:

- PTRCHKLVL=FULL for the CHANGE command specifies that the integrity verification process runs in FULL mode.

Tip: Use this option when you want to obtain information about database record length.

- The REPORT subcommand specifies that the UOW reports are generated.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//         DD DISP=SHR,DSN=IMSVS.SDFSRESL
//         DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
CHANGE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  IDSNMASK='HPFP.&AREA.ADS1',
  OAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL
IRENAME
  IAREA=(DB22AR0,DB22AR1),
  DSNAME='HPFP.OLD.&AREA.ADS1'
ALLOCATE
  OAREA=(DB22AR0,DB22AR1),
  SPACE=(CYL,500),
  VOLSER=HFPVOL
REPORT
/*
```

Figure 423. FPA Change example 8: Generating reports for all UOWs

Example 9: Generating reports for UOWs in which exceptions were detected

The following figure shows example JCL for generating the reports only for the UOWs in which exceptions were detected.

In this example:

- PTRCHKLVL=FULL for the CHANGE command specifies that the integrity verification process runs in FULL mode.

Tip: Use this option when you want to obtain information about database record length.

- The REPORT subcommand specifies that the UOW Exception reports are generated.
- The HFPAPARM DD statement specifies the threshold values for generating the UOW Exception reports.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPAPARM DD *
           RDOVFFS=40 UDOVFFS=35 RBASEFS=40
/*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
CHANGE
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  IDSNMASK=' HPFP.&AREA.ADS1 ',
  OAREA=(DB22AR0,DB22AR1),
  PTRCHKLVL=FULL
IRENAME
  IAREA=(DB22AR0,DB22AR1),
  DSNAME=' HPFP.OLD.&AREA.ADS1 '
ALLOCATE
  OAREA=(DB22AR0,DB22AR1),
  SPACE=(CYL,500),
  VOLSER=HFPPVOL
REPORT
/*

```

Figure 424. FPA Change example 9: Generating reports for UOWs in which exceptions were detected

FPA Reorg JCL examples for monitoring and tuning DEDBs

Use these JCL examples to monitor and tune DEDBs by using the FPA Reorg function.

Subsections:

- [“JCL template for running the FPA Reorg function” on page 1148](#)
- [“Example 1: Storing statistics in the HISTORY data set” on page 1149](#)
- [“Example 2: Storing statistics in the HISTORY2 data set” on page 1150](#)
- [“Example 3: Storing statistics in the Sensor Data repository” on page 1150](#)
- [“Example 4: Specifying FPA threshold values” on page 1151](#)
- [“Example 5: Generating free space analysis reports” on page 1151](#)
- [“Example 6: Generating reports to obtain database records and randomizing information” on page 1152](#)
- [“Example 7: Generating reports to obtain large database records information” on page 1152](#)
- [“Example 8: Generating reports for all UOWs” on page 1153](#)
- [“Example 9: Generating reports for UOWs in which exceptions were detected” on page 1153](#)

JCL template for running the FPA Reorg function

The following figure shows a JCL template for running the FPA Reorg function to reorganize specific UOWs of multiple areas that are registered with DBRC. By using this JCL, you can reorganize the UOWs that satisfy either the RBASEFS or RDOVFFS condition.

In this example:

- RECON data sets are allocated dynamically by using the DFSMDA members in the IMSDALIB DD data set.
- The input area data sets (ADSs) to be validated is dynamically allocated. The allocation information (DD name and DS name of the ADSs) is obtained from DBRC.

- The UOWs that are to be reorganized are limited to the subset of UOWs that satisfy either the RBASEFS or RDOVFFS condition. The UOWs in this subset are ranked on the basis of statistics that are gathered by the Area Sensor.
- After the Reorg process, fast scan of database integrity verification is performed to analyze the reorganized UOWs. The PTRCHKLVL keyword is not specified; therefore, the default value (PTRCHKLVL=REORGED) is applied.
- The RFSSCAN keyword is not specified; therefore, the default value (RFSSCAN=(YES,YES)) is applied, and the statistics before and after reorganization are collected.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
REORG
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  RBASEFS=20,
  RDOVFFS=20
/*
```

Figure 425. JCL template for FPA Reorg

Example 1: Storing statistics in the HISTORY data set

The following figure shows example JCL for storing the statistical information of a reorganized area that is registered in DBRC into the HISTORY data set.

Prerequisite: If you are using the HISTORY data set for the first time, initialize the permanent data sets by following the procedures in “Preprocess for the Analyze process: Initializing permanent data set” on page 197. Also see “History file records layout” on page 1173 for information about the layout of the History file records.

In this example:

- The HFP AHST DD statement specifies the HISTORY data set.
- Tip:** Instead of using the HFP AHST DD statement, you can specify a different DD statement that is specified by the HISTORYDD= keyword parameter.
- PTRCHKLVL=FULL for the REORG command specifies that the integrity verification process runs in FULL mode.
 - The REPORT subcommand specifies that the statistics are stored in the HISTORY data set.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFP AHST DD DISP=(MOD,KEEP,KEEP),DSN=HPFP.HISTORY
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
REORG
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  RBASEFS=20,
  RDOVFFS=20,
  PTRCHKLVL=FULL
REPORT
/*
```

Figure 426. FPA Reorg example 1: Storing statistics in the HISTORY data set

Example 2: Storing statistics in the HISTORY2 data set

The following figure shows example JCL for storing the statistical information of a UOW group of a reorganized area into the HISTORY2 data set.

Prerequisite: If you are using the HISTORY2 data set for the first time, initialize the permanent data sets by following the procedures in “Preprocess for the Analyze process: Initializing permanent data set” on page 197. Also see “History2 file records layout” on page 1175 for information about the layout of the History2 file records.

In this example:

- The HFP AHST2 DD statement specifies the HISTORY2 data set.
- The HFPAUOWC DD statement specifies the group of UOWs for the area.
- PTRCHKLVL=FULL for the REORG command specifies that the integrity verification process runs in FULL mode.
- The REPORT subcommand specifies that the statistics are stored in the HISTORY2 data set.

```
//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFP AHST2 DD DISP=(MOD,KEEP,KEEP),DSN=HPFP.HISTORY2
//HFPAUOWC DD DISP=SHR,DSN=HPFP.UOWCTL
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
REORG
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  RBASEFS=20,
  RDOVFFS=20,
  PTRCHKLVL=FULL
REPORT
//*
```

Figure 427. FPA Reorg example 2: Storing statistics in the HISTORY2 data set

Example 3: Storing statistics in the Sensor Data repository

The following figure shows example JCL for storing the statistical information of an entire reorganized area that is registered in DBRC into the Sensor Data repository.

Prerequisite: To run this JCL, an IMS Tools Knowledge Base server and its repositories must be configured.

In this example:

- The IMS Tools Base library (ITBVS.SHKTLOAD) is concatenated to the STEPLIB DD.
- ITKBSRVR= for the GLOBAL command specifies the IMS Tools KB server XCF group.
- PTRCHKLVL=FULL for the REORG command specifies that the integrity verification process runs in FULL mode.
- SENSOR=YES for the REORG command specifies that the statistics are stored in the Sensor Data repository.
- SENSOR_DBREC=YES for the REORG command specifies that the statistics that are related to database records and root segment distribution are collected.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=ITBVS.SHKTLOAD
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES,
  ITKBSRVR=FPQSRVFP
REORG
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  RBASEFS=20,
  RDOVFFS=20,
  PTRCHKLVL=FULL,
  SENSOR=YES,
  SENSOR_DBREC=YES
/*

```

Figure 428. FPA Reorg example 3: Storing statistics in the Sensor Data repository

Example 4: Specifying FPA threshold values

The following figure shows example JCL for specifying the threshold values of FPA.

In this example:

- PTRCHKLVL=FULL for the REORG command specifies that the integrity verification process runs in FULL mode.
- The REPORT subcommand specifies that the Process Summary of Analysis report is generated.
- The THRESHOLD subcommand enables the threshold settings.
- The keywords and their parameters for the THRESHOLD subcommand specify the threshold values.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
REORG
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  RBASEFS=20,
  RDOVFFS=20,
  PTRCHKLVL=FULL
REPORT
THRESHOLD
  RAAFS_PCT=30,
  RECIO_MAX=6
/*

```

Figure 429. FPA Reorg example 4: Specifying FPA threshold values

See [“THRESHOLD subcommand” on page 435](#) for the keywords that you can specify on the THRESHOLD subcommand.

Example 5: Generating free space analysis reports

The following figure shows example JCL for generating reports that are required for free space analysis.

In this example:

- PTRCHKLVL=QUICK for the REORG command specifies that the integrity verification process runs in QUICK mode.
- The REPORT subcommand specifies that the Freespace Analysis report is generated.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
REORG
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  RBASEFS=20,
  RDOVFFS=20,
  PTRCHKLVL=QUICK
REPORT
/*

```

Figure 430. FPA Reorg example 5: Generating reports for free space analysis

Example 6: Generating reports to obtain database records and randomizing information

The following figure shows example JCL for generating the reports that contain information about database records and randomizing.

In this example:

- PTRCHKLVL=FULL for the REORG command specifies that the integrity verification process runs in FULL mode.
- The REPORT subcommand specifies that the DEDB Area Analysis reports are generated.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
REORG
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  RBASEFS=20,
  RDOVFFS=20,
  PTRCHKLVL=FULL
REPORT
/*

```

Figure 431. FPA Reorg example 6: Generating reports to obtain database records and randomizing information

Example 7: Generating reports to obtain large database records information

The following figure shows example JCL for generating the reports that contain information about the large database records.

In this example:

- PTRCHKLVL=FULL for the REORG command specifies that the integrity verification process runs in FULL mode.
- The REPORT subcommand, LARGEREC_REPORT=YES, and LARGEREC_COUNT=(10,20) specify that the Largest Database Records report and the Consolidated Largest Database Records report are generated.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB   DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
REORG
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  RBASEFS=20,
  RDOVFFS=20,
  PTRCHKLVL=FULL
REPORT
  LARGEREC_REPORT=YES,
  LARGEREC_COUNT=(10,20)
/*

```

Figure 432. FPA Reorg example 7: Generating reports to obtain large database records information

Example 8: Generating reports for all UOWs

The following figure shows example JCL for generating the reports for all UOWs.

In this example:

- PTRCHKLVL=FULL for the REORG command specifies that the integrity verification process runs in FULL mode.
- **Tip:** Use this option when you want to obtain information about database record length.
- The REPORT subcommand specifies that the UOW reports are generated.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB   DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
REORG
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  RBASEFS=20,
  RDOVFFS=20,
  PTRCHKLVL=FULL
REPORT
/*

```

Figure 433. FPA Reorg example 8: Generating reports for all UOWs

Example 9: Generating reports for UOWs in which exceptions were detected

The following figure shows example JCL for generating the reports only for the UOWs in which exceptions were detected.

In this example:

- PTRCHKLVL=FULL for the REORG command specifies that the integrity verification process runs in FULL mode.
- **Tip:** Use this option when you want to obtain information about database record length.
- The REPORT subcommand specifies that the UOW Exception reports are generated.
- The HFPAPARM DD statement specifies the threshold values for generating the UOW Exception reports.

```

//HFP      EXEC PGM=HFPMAIN0
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//          DD DISP=SHR,DSN=IMSVS.SDFSRESL
//          DD DISP=SHR,DSN=IMSVS.PGMLIB
//IMSACB  DD DISP=SHR,DSN=IMSVS.ACBLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//HFPAPARM DD *
           RDOVFFS=40 UDOVFFS=35 RBASEFS=40
/*
//HFPSYSIN DD *
GLOBAL
  DBRC=YES
  Reorg
  DBD=DEDBJN22,
  IAREA=(DB22AR0,DB22AR1),
  RBASEFS=20,
  RDOVFFS=20,
  PTRCHKLVL=FULL
  REPORT
/*

```

Figure 434. FPA Reorg example 9: Generating reports for UOWs in which exceptions were detected

DEDB Pointer Checker JCL examples for monitoring and tuning DEDBs

Use these JCL examples to monitor and tune DEDBs by using the DEDB Pointer Checker utility.

Subsections:

- [“JCL template for running the DEDB Pointer Checker utility” on page 1154](#)
- [“Example 1: Storing statistics in the HISTORY data set” on page 1154](#)
- [“Example 2: Storing statistics in the HISTORY2 data set” on page 1155](#)
- [“Example 3: Specifying FPB threshold values” on page 1155](#)
- [“Example 4: Generating reports to obtain database records and randomizing information” on page 1156](#)
- [“Example 5: Generating reports to obtain large database records information” on page 1156](#)
- [“Example 6: Generating reports for all UOWs” on page 1156](#)
- [“Example 7: Generating reports for UOWs in which exceptions were detected” on page 1156](#)

JCL template for running the DEDB Pointer Checker utility

The following figure shows a JCL template for running the DEDB Pointer Checker utility.

In this example, the EXEC statement specifies the cataloged procedure FABAPC that is located in your JCL library (SHFPSAMP). This specification runs the integrity verification process in FULL mode.

```

//JOB LIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
/*
//FPPC EXEC FABAPC,
// ACBLIB='IMSVS.ACBLIB',
// AREADS='HPFP.DB22AR0.ADS1'
/*
//FABADA1.SYSIN DD *
  DBDNAME=DEDBJN22,AREA=DB22AR0,VSAM,STATS,TYPRUN=RPT
/*

```

Figure 435. JCL template for DEDB Pointer Checker

Example 1: Storing statistics in the HISTORY data set

The following figure shows example JCL for storing the statistical information of an area into the HISTORY data set.

Prerequisite: If you are using the HISTORY data set for the first time, initialize the permanent data sets by following the procedures in the topic "Preprocess for DEDB Pointer Checker" in the *IMS Fast Path Solution Pack: IMS Fast Path Basic Tools User's Guide*. Also see [“History file records layout” on page 1173](#) for information about the layout of the History file records.

In this example, the HISTORY DD statement for the FABADA5 step specifies the HISTORY data set.

```
//JOB LIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//*
//FPPC EXEC FABAPC,
// ACBLIB='IMSVS.ACBLIB',
// AREADS='HPFP.DB22AR0.ADS1'
//*
//FABADA1.SYSIN DD *
DBDNAME=DEDBJN22,AREA=DB22AR0,VSAM,STATS,TYPRUN=RPT
/*
//FABADA5.HISTORY DD DISP=(MOD,KEEP,KEEP),DSN=HPFP.HISTORY
```

Figure 436. DEDB Pointer Checker example 1: Storing statistics in the HISTORY data set

Example 2: Storing statistics in the HISTORY2 data set

The following figure shows example JCL for storing the statistical information of a group of UOWs into the HISTORY2 data set.

Prerequisite: If you are using the HISTORY2 data set for the first time, initialize the permanent data sets by following the procedures in the topic "Preprocess for DEDB Pointer Checker" in the *IMS Fast Path Solution Pack: IMS Fast Path Basic Tools User's Guide*. Also see "[History2 file records layout](#)" on page 1175 for information about the layout of the History2 file records.

In this example:

- The HISTORY2 DD statement for the FABADA5 step specifies the HISTORY2 data set.
- The UOWCTL DD statement for the FABADA5 step specifies the group of UOWs for the area.

```
//JOB LIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//*
//FPPC EXEC FABAPC,
// ACBLIB='IMSVS.ACBLIB',
// AREADS='HPFP.DB22AR0.ADS1'
//*
//FABADA1.SYSIN DD *
DBDNAME=DEDBJN22,AREA=DB22AR0,VSAM,STATS,TYPRUN=RPT
/*
//FABADA5.HISTORY2 DD DISP=(MOD,KEEP,KEEP),DSN=HPFP.HISTORY2
//FABADA5.UOWCTL DD DISP=SHR,DSN=HPFP.UOWCTL
```

Figure 437. DEDB Pointer Checker example 2: Storing statistics in the HISTORY2 data set

Example 3: Specifying FPB threshold values

The following figure shows example JCL for specifying the threshold values of FPB.

In this example, the THRCTL DD statement for the FABADA5 step specifies the threshold values.

```
//JOB LIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//*
//FPPC EXEC FABAPC,
// ACBLIB='IMSVS.ACBLIB',
// AREADS='HPFP.DB22AR0.ADS1'
//*
//FABADA1.SYSIN DD *
DBDNAME=DEDBJN22,AREA=DB22AR0,VSAM,STATS,TYPRUN=RPT
/*
//FABADA5.THRCTL DD *
T 060 030
/*
```

Figure 438. DEDB Pointer Checker example 3: Specifying FPB threshold values

See the topic "FABADA5 THRCTL DD data set" in the *IMS Fast Path Solution Pack: IMS Fast Path Basic Tools User's Guide* for the threshold values that can be specified on the THRCTL DD statement.

Example 4: Generating reports to obtain database records and randomizing information

You can use the JCL template to generate the reports that contain information about database records and randomizing for both the area and all the UOWs of the area. See [“JCL template for running the DEDB Pointer Checker utility” on page 1154.](#)

Example 5: Generating reports to obtain large database records information

The following figure shows example JCL for generating the reports that contain information about the large database records.

In this example, TYPRUN=MODEL specifies that the root key value is appended to the segment information records for tuning purposes.

```
//JOB LIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//*
//FPPC EXEC FABAPC,
// ACBLIB='IMSVS.ACBLIB',
// AREADS='HPFP.DB22AR0.ADS1'
//*
//FABADA1.SYSIN DD *
// DBDNAME=DEDBJN22,AREA=DB22AR0,VSAM,STATS,TYPRUN=MODEL
//*
```

Figure 439. DEDB Pointer Checker example 5: Generating reports for large database records

Example 6: Generating reports for all UOWs

The following figure shows example JCL for generating the reports for all UOWs.

In this example, the UOWRPT DD statement for the FABADA5 step specifies the output data set for the UOW reports.

```
//JOB LIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//*
//FPPC EXEC FABAPC,
// ACBLIB='IMSVS.ACBLIB',
// AREADS='HPFP.DB22AR0.ADS1'
//*
//FABADA1.SYSIN DD *
// DBDNAME=DEDBJN22,AREA=DB22AR0,VSAM,STATS,TYPRUN=RPT
//*
//FABADA5.UOWRPT DD SYSOUT=*
```

Figure 440. DEDB Pointer Checker example 6: Generating reports for all UOWs

Example 7: Generating reports for UOWs in which exceptions were detected

The following figure shows example JCL for generating the reports only for the UOWs in which exceptions were detected.

In this example:

- The UOWEXCP DD statement for the FABADA5 step specifies the output data set for the UOW Exception reports.
- The PARMCTL DD statement for the FABADA5 step specifies the threshold values for generating the UOW Exception reports.

```

//JOB LIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
//*
//FPPC EXEC FABAPC,
// ACBLIB='IMSVS.ACBLIB',
// AREADS='HPFP.DB22AR0.ADS1'
//*
//FABADA1.SYSIN DD *
DBDNAME=DEDBJN22,AREA=DB22AR0,VSAM,STATS,TYPRUN=RPT
/*
//FABADA5.UOWEXCP DD SYSOUT=*
//FABADA5.PARMCTL DD *
RDOVFFS=40 UD0VFFS=35 RBASEFS=40
/*

```

Figure 441. DEDB Pointer Checker example 7: Generating reports for UOWs in which exceptions were detected

Online Pointer Checker JCL examples for monitoring and tuning DEDBs

Use these JCL examples to monitor and tune DEDBs by using the Online Pointer Checker utility.

Subsections:

- [“JCL template for running the Online Pointer Checker utility” on page 1157](#)
- [“Example 1: Storing statistics in the HISTORY data set” on page 1157](#)
- [“Example 2: Storing statistics in the HISTORY2 data set” on page 1158](#)
- [“Example 3: Storing statistics in the Sensor Data repository” on page 1158](#)
- [“Example 4: Specifying FPO threshold values” on page 1159](#)
- [“Example 5: Specifying FABADA5 threshold values” on page 1159](#)
- [“Example 6: Generating reports for database analysis” on page 1161](#)
- [“Example 7: Generating reports to obtain large database records information” on page 1161](#)
- [“Example 8: Generating reports for UOWs in which exceptions were detected” on page 1161](#)
- [“Example 9: Generating free space analysis reports” on page 1161](#)

JCL template for running the Online Pointer Checker utility

The following figure shows a JCL template for running the Online Pointer Checker utility.

In this example, TYPRUN=RPT is specified, which runs the integrity verification process in FULL mode.

```

//OPC EXEC FPUTIL,DBD=DEDBJN22,IMSID=IMS1
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
//FPXIN DD *
ANALYZE AREA=(DB22AR0,DB22AR1),
FULLSTEP=YES,
TYPRUN=RPT
/*

```

Figure 442. JCL template for Online Pointer Checker

Example 1: Storing statistics in the HISTORY data set

The following figure shows example JCL for storing the statistical information of an online area into the HISTORY data set.

Prerequisite: If you are using the HISTORY data set for the first time, initialize the permanent data sets by following the procedures in [“Preprocess for OPC: Initializing permanent data set” on page 682](#). Also see [“History file records layout” on page 1173](#) for information about the layout of the History file records.

In this example, the FPXAHST DD statement specifies the HISTORY data set.

Tip: Instead of using the FPXAHST DD statement, you can specify a different DD statement that is specified by the HISTORYDD= keyword parameter.

```
//OPC      EXEC FPUTIL, DBD=DEDBJN22, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//        DD DISP=SHR, DSN=IMSVS.SDFSRESL
//FPXAHST DD DISP=(MOD,KEEP,KEEP), DSN=HPFP.HISTORY
//FPXIN   DD *
        ANALYZE AREA=(DB22AR0,DB22AR1),
        FULLSTEP=YES,
        TYPRUN=RPT
/*
```

Figure 443. Online Pointer Checker example 1: Storing statistics in the HISTORY data set

Example 2: Storing statistics in the HISTORY2 data set

The following figure shows example JCL for storing the statistical information of a group of UOWs into the HISTORY2 data set.

Prerequisite: If you are using the HISTORY2 data set for the first time, initialize the permanent data sets by following the procedures in “Preprocess for OPC: Initializing permanent data set” on page 682. Also see “History2 file records layout” on page 1175 for information about the layout of the History2 file records.

In this example:

- The FPXAHST2 DD statement specifies the HISTORY2 data set.
- The FPXAUOWC DD statement specifies the group of UOWs for the area.

```
//OPC      EXEC FPUTIL, DBD=DEDBJN22, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//        DD DISP=SHR, DSN=IMSVS.SDFSRESL
//FPXAHST2 DD DISP=(MOD,KEEP,KEEP), DSN=HPFP.HISTORY2
//FPXAUOWC DD DISP=SHR, DSN=HPFP.UOWCTL
//FPXIN   DD *
        ANALYZE AREA=(DB22AR0,DB22AR1),
        FULLSTEP=YES,
        TYPRUN=RPT
/*
```

Figure 444. Online Pointer Checker example 2: Storing statistics in the HISTORY2 data set

Example 3: Storing statistics in the Sensor Data repository

The following figure shows example JCL for storing the statistical information of an entire online area into the Sensor Data repository.

Prerequisite: To run this JCL, an IMS Tools Knowledge Base server and its repositories must be configured.

In this example:

- The IMS Tools Base library (ITBVS.SHKTLOAD) is concatenated to the STEPLIB DD.
- ITKBSRVR= for the GLOBAL statement specifies the IMS Tools KB server XCF group.
- SENSOR=YES for the ANALYZE statement specifies that the statistics are stored in the Sensor Data repository.
- SENSOR_DBREC=YES for the ANALYZE statement specifies that the statistics that are related to database records and root segment distribution are collected.

```

//OPC      EXEC FPUTIL, DBD=DEDBJN22, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//         DD DISP=SHR, DSN=IMSVS.SDFSRESL
//         DD DISP=SHR, DSN=ITBVS.SHKTLOAD
//FPXIN    DD *
GLOBAL
ITKBSRVR=FPQSRVFP
ANALYZE AREA=(DB22AR0, DB22AR1),
FULLSTEP=YES,
TYPRUN=RPT,
SENSOR=YES,
SENSOR_DBREC=YES
/*

```

Figure 445. Online Pointer Checker example 3: Storing statistics in the Sensor Data repository

Example 4: Specifying FPO threshold values

The following figure shows example JCL for specifying the threshold values of FPO.

In this example:

- The THRESHOLD statement enables the threshold settings.
- The keywords and their parameter for the THRESHOLD statement specify the threshold values.

```

//OPC      EXEC FPUTIL, DBD=DEDBJN22, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//         DD DISP=SHR, DSN=IMSVS.SDFSRESL
//FPXIN    DD *
ANALYZE AREA=(DB22AR0, DB22AR1),
FULLSTEP=YES,
TYPRUN=RPT
THRESHOLD
RAAFS_PCT=30,
RECIO_MAX=6
/*

```

Figure 446. Online Pointer Checker example 4: Specifying FPO threshold values

See [“THRESHOLD control statement”](#) on page 859 for the keywords that you can specify on the THRESHOLD statement.

Example 5: Specifying FABADA5 threshold values

The following figure shows example JCL for running OPC and several programs of DEDB Pointer Checker. When you run this JCL, OPC collects the pointer and segment information, and the DEDB Pointer Checker programs evaluate the data by using the threshold values that are specified in this JCL.

You can use the cataloged procedures FABAPC and FPXAPC1 that are located in your JCL library (SHFPSAMP).

In this example:

- The AREA=(DB22AR0) keyword specifies that a single area is processed.
- The FULLSTEP=YES keyword is not specified.
- DAR120, DAR13SO, DAR3CSO, SORT12, and SORT3CS DD statements specify that the pointer and segment information is passed to the DEDB Pointer Checker job steps.
- After the OPC job step, all the job steps of DEDB Pointer Checker that are typically required for the TYPRUN=RPT option, except for the FABADA1 step, are specified in this JCL.
- The THRCTL DD statement of the FABADA5 step specifies the threshold values.

```

//OPC      EXEC FPUTIL, DBD=DEDBJN22, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//        DD DISP=SHR, DSN=IMSVS.SDFSRESL
//FPXIN   DD *
        ANALYZE AREA=(DB22AR0),
        TYPRUN=RPT
/*
//DAR120  DD DISP=(NEW, PASS), DSN=&&DA12,
//        UNIT=SYSDA, SPACE=(CYL, (1, 1), RLSE)
//DAR13S0 DD DISP=(NEW, PASS), DSN=&&DA13S,
//        UNIT=SYSDA, SPACE=(CYL, (1, 1), RLSE)
//DAR3CS0 DD DISP=(NEW, PASS), DSN=&&DA3CS,
//        UNIT=SYSDA, SPACE=(CYL, (1, 1), RLSE)
//SORT12  DD DISP=(NEW, PASS), DSN=&&SORT12,
//        UNIT=SYSDA, SPACE=(CYL, (1, 1), RLSE)
//SORT3CS DD DISP=(NEW, PASS), DSN=&&SORT3CS,
//        UNIT=SYSDA, SPACE=(CYL, (1, 1), RLSE)
//*
//SORT3CS EXEC PGM=SORT
//SYSOUT  DD SYSOUT=*
//SYSIN   DD DISP=(OLD, DELETE), DSN=&&SORT3CS
//SORTIN  DD DISP=(OLD, DELETE), DSN=&&DA3CS
//SORTOUT DD DISP=(NEW, PASS), DSN=&&SDA3CS,
//        UNIT=SYSDA, SPACE=(CYL, (1, 1), RLSE)
//SORTWK01 DD UNIT=SYSDA, SPACE=(CYL, (2, 2))
//SORTWK02 DD UNIT=SYSDA, SPACE=(CYL, (2, 2))
//SORTWK03 DD UNIT=SYSDA, SPACE=(CYL, (2, 2))
//*
//FABADA3 EXEC PGM=FABADA3
//SYSPRINT DD SYSOUT=*
//DAR13SI DD DISP=(OLD, DELETE), DSN=&&DA13S
//DAR3CSI DD DISP=(OLD, DELETE), DSN=&&SDA3CS
//DAR4ERRS DD DISP=(NEW, PASS), DSN=&&DA4ERRS,
//        UNIT=SYSDA, SPACE=(CYL, (2, 2), RLSE)
//DAR130  DD DISP=(NEW, PASS), DSN=&&RDA13,
//        UNIT=SYSDA, SPACE=(CYL, (1, 1), RLSE)
//SORT4CTL DD DISP=(NEW, PASS), DSN=&&SORT4,
//        UNIT=SYSDA, SPACE=(TRK, (2, 1), RLSE)
//SORTR13 DD DISP=(NEW, PASS), DSN=&&SORTR13,
//        UNIT=SYSDA, SPACE=(TRK, (2, 1), RLSE)
//*
//SORT12  EXEC PGM=SORT
//SORTLIB DD DSN=SYS1.SORTLIB, DISP=SHR
//SYSOUT  DD SYSOUT=*
//SYSIN   DD DISP=(OLD, DELETE), DSN=&&SORT12
//SORTWK01 DD UNIT=SYSDA, SPACE=(CYL, (2, 2))
//SORTWK02 DD UNIT=SYSDA, SPACE=(CYL, (2, 2))
//SORTWK03 DD UNIT=SYSDA, SPACE=(CYL, (2, 2))
//SORTIN  DD DISP=(OLD, DELETE), DSN=&&DA12
//SORTOUT DD DISP=(NEW, PASS), DSN=&&SDA12,
//        UNIT=SYSDA, SPACE=(TRK, (2, 1), RLSE)
//*
//SORTR13 EXEC PGM=SORT
//SYSOUT  DD SYSOUT=*
//SYSIN   DD DISP=(OLD, DELETE), DSN=&&SORTR13
//SORTIN  DD DISP=(OLD, DELETE), DSN=&&RDA13
//SORTOUT DD DISP=(NEW, PASS), DSN=&&SRDA13,
//        UNIT=SYSDA, SPACE=(CYL, (1, 1), RLSE)
//SORTWK01 DD UNIT=SYSDA, SPACE=(CYL, (2, 2))
//SORTWK02 DD UNIT=SYSDA, SPACE=(CYL, (2, 2))
//SORTWK03 DD UNIT=SYSDA, SPACE=(CYL, (2, 2))
//*
//FABADA5 EXEC PGM=FABADA5
//SYSPRINT DD SYSOUT=*
//REPORTS DD SYSOUT=*
//DAR12I  DD DISP=(OLD, DELETE), DSN=&&SDA12
//DAR13I  DD DISP=(OLD, DELETE), DSN=&&SRDA13
//THRCTL  DD *
T                060                                030
/*

```

Figure 447. Online Pointer Checker example 5: Specifying FABADA5 threshold values

See the topic "FABADA5 THRCTL DD data set" in the *IMS Fast Path Solution Pack: IMS Fast Path Basic Tools User's Guide* for the threshold values that can be specified on the THRCTL DD statement.

Example 6: Generating reports for database analysis

You can use the JCL template to generate the reports that contain information about database records and randomizing for both the area and all the UOWs of the area. See [“JCL template for running the Online Pointer Checker utility” on page 1157.](#)

Example 7: Generating reports to obtain large database records information

The following figure shows example JCL for generating the reports that contain information about the large database records.

In this example, the REPORT control statement, LARGEREC_REPORT=YES, and LARGEREC_COUNT=(10,20) specify that the Largest Database Records report and the Consolidated Largest Database Records report are generated.

```
//OPC      EXEC FPUTIL, DBD=DEDBJN22, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//        DD DISP=SHR, DSN=IMSVS.SDFSRESL
//FPXIN   DD *
ANALYZE AREA=(DB22AR0, DB22AR1),
        FULLSTEP=YES,
        TYPRUN=RPT
REPORT
        LARGEREC_REPORT=YES,
        LARGEREC_COUNT=(10,20)
/*
```

Figure 448. Online Pointer Checker example 7: Generating reports to obtain large database records information

Example 8: Generating reports for UOWs in which exceptions were detected

The following figure shows example JCL for generating the reports only for the UOWs in which exceptions were detected.

In this example, EXC_RDOVFS=, EXC_UDOVFFS=, and EXC_RBASEFS= for the ANALYZE statement specify the threshold values for generating the UOW Exception reports.

```
//OPC      EXEC FPUTIL, DBD=DEDBJN22, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//        DD DISP=SHR, DSN=IMSVS.SDFSRESL
//FPXIN   DD *
ANALYZE AREA=(DB22AR0, DB22AR1),
        FULLSTEP=YES,
        TYPRUN=RPT,
        EXC_RDOVFFS=40,
        EXC_UDOVFFS=35,
        EXC_RBASEFS=40
/*
```

Figure 449. Online Pointer Checker example 8: Generating reports for UOWs in which exceptions were detected

Example 9: Generating free space analysis reports

The following figure shows example JCL for generating the reports that are required for free space analysis.

In this example, TYPRUN=FS for the ANALYZE statement specifies that the Freespace Analysis report is generated.

```

//OPC      EXEC FPUTIL, DBD=DEDBJN22, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//         DD DISP=SHR, DSN=IMSVS.SDFSRESL
//         DD DISP=SHR, DSN=ITBVS.SHKTLOAD
//FPXIN    DD *
ANALYZE   AREA=(DB22AR0, DB22AR1),
          FULLSTEP=YES,
          TYPRUN=FS
/*

```

Figure 450. Online Pointer Checker example 9: Generating reports for free space analysis

Online Expert Reorganization JCL examples for monitoring and tuning DEDBs

Use these JCL examples to monitor and tune DEDBs by using the Online Expert Reorganization utility.

Subsections:

- [“JCL template for running the Online Expert Reorganization utility” on page 1162](#)
- [“Example 1: Storing statistics in the HISTORY data set” on page 1162](#)
- [“Example 2: Storing statistics in the HISTORY2 data set” on page 1163](#)
- [“Example 3: Storing statistics in the Sensor Data repository” on page 1163](#)
- [“Example 4: Specifying FPO threshold values” on page 1164](#)
- [“Example 5: Specifying FABADA5 threshold values” on page 1164](#)
- [“Example 6: Generating reports for database analysis” on page 1166](#)
- [“Example 7: Generating reports to obtain large database records information” on page 1166](#)
- [“Example 8: Generating reports for UOWs in which exceptions were detected” on page 1166](#)
- [“Example 9: Generating free space analysis reports” on page 1166](#)

JCL template for running the Online Expert Reorganization utility

The following figure shows a JCL template for running the Online Expert Reorganization utility.

In this example, the integrity verification process runs in FULL mode for the entire area because the following options are specified:

- PTRCHK=YES for the REORG statement
- TYPRUN=RPT and CHKUOW=ALL for the PTRCHK statement

```

//OER      EXEC FPUTIL, DBD=DEDBJN22, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//         DD DISP=SHR, DSN=IMSVS.SDFSRESL
//FPXIN    DD *
REORG     AREA=(DB22AR0, DB22AR1),
          PTRCHK=YES
          PTRCHK
          TYPRUN=RPT,
          CHKUOW=ALL,
          FULLSTEP=YES
/*

```

Figure 451. JCL template for Online Expert Reorganization

Example 1: Storing statistics in the HISTORY data set

The following figure shows example JCL for storing the statistical information of an online-reorganized area into the HISTORY data set.

Prerequisite: If you are using the HISTORY data set for the first time, initialize the permanent data sets by following the procedures in [“Preprocess for OPC: Initializing permanent data set” on page 682](#). Also see [“History file records layout” on page 1173](#) for information about the layout of the History file records.

In this example, the FPXAHST DD statement specifies the HISTORY data set.

Tip: Instead of using the FPXAHST DD statement, you can specify a different DD statement that is specified by the HISTORYDD= keyword parameter.

```
//OER      EXEC FPUTIL, DBD=DEDBJN22, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//        DD DISP=SHR, DSN=IMSVS.SDFSRESL
//FPXAHST DD DISP=(MOD,KEEP,KEEP), DSN=HPFP.HISTORY
//FPXIN   DD *
          REORG AREA=(DB22AR0, DB22AR1),
          PTRCHK=YES
          PTRCHK
          TYPRUN=RPT,
          CHKUOW=ALL,
          FULLSTEP=YES
/*
```

Figure 452. Online Expert Reorganization example 1: Storing statistics in the HISTORY data set

Example 2: Storing statistics in the HISTORY2 data set

The following figure shows example JCL for storing the statistical information of a group of UOWs into the HISTORY2 data set.

Prerequisite: If you are using the HISTORY2 data set for the first time, initialize the permanent data sets by following the procedures in “Preprocess for OPC: Initializing permanent data set” on page 682. Also see “History2 file records layout” on page 1175 for information about the layout of the History2 file records.

In this example:

- The FPXAHST2 DD statement specifies the HISTORY2 data set.
- The FPXAUOWC DD statement specifies the group of UOWs for the area.

```
//OER      EXEC FPUTIL, DBD=DEDBJN22, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//        DD DISP=SHR, DSN=IMSVS.SDFSRESL
//FPXAHST2 DD DISP=(MOD,KEEP,KEEP), DSN=HPFP.HISTORY2
//FPXAUOWC DD DISP=SHR, DSN=HPFP.UOWCTL

//FPXIN   DD *
          REORG AREA=(DB22AR0, DB22AR1),
          PTRCHK=YES
          PTRCHK
          TYPRUN=RPT,
          CHKUOW=ALL,
          FULLSTEP=YES
/*
```

Figure 453. Online Expert Reorganization example 2: Storing statistics in the HISTORY2 data set

Example 3: Storing statistics in the Sensor Data repository

The following figure shows example JCL for storing the statistical information of an entire online-reorganized area into the Sensor Data repository.

Prerequisite: To run this JCL, an IMS Tools Knowledge Base server and its repositories must be configured.

In this example:

- The IMS Tools Base library (ITBVS.SHKTLOAD) is concatenated to the STEPLIB DD.
- ITKBSRVR= for the GLOBAL statement specifies the IMS Tools KB server XCF group.
- SENSOR=YES for the PTRCHK statement specifies that the statistics are stored in the Sensor Data repository.
- SENSOR_DBREC=YES for the PTRCHK statement specifies that the statistics that are related to database records and root segment distribution are collected.

```

//OER      EXEC FPUTIL, DBD=DEDBJN22, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//         DD DISP=SHR, DSN=IMSVS.SDFSRESL
//         DD DISP=SHR, DSN=ITBVS.SHKTLOAD
//FPXIN    DD *
GLOBAL
ITKBSRVR=FPQSRVFP
REORG AREA=(DB22AR0, DB22AR1),
PTRCHK=YES
PTRCHK
TYPRUN=RPT,
CHKUOW=ALL,
FULLSTEP=YES,
SENSOR=YES,
SENSOR_DBREC=YES
/*

```

Figure 454. Online Expert Reorganization example 3: Storing statistics in the Sensor Data repository

Example 4: Specifying FPO threshold values

The following figure shows example JCL for specifying the threshold values of FPO.

In this example:

- The THRESHOLD statement enables the threshold settings.
- The keywords and their parameter for the THRESHOLD statement specify the threshold values.

```

//OER      EXEC FPUTIL, DBD=DEDBJN22, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//         DD DISP=SHR, DSN=IMSVS.SDFSRESL
//FPXIN    DD *
REORG AREA=(DB22AR0, DB22AR1),
PTRCHK=YES
PTRCHK
TYPRUN=RPT,
CHKUOW=ALL,
FULLSTEP=YES
THRESHOLD
RAAFS_PCT=30,
RECIO_MAX=6
/*

```

Figure 455. Online Expert Reorganization example 4: Specifying FPO threshold values

See “THRESHOLD control statement” on page 859 for the keywords that you can specify on the THRESHOLD statement.

Example 5: Specifying FABADA5 threshold values

The following figure shows example JCL for running OER and several programs of DEDB Pointer Checker. When you run this JCL, OPC, which is internally called by OER, collects the pointer and segment information, and the DEDB Pointer Checker programs evaluate the data by using the threshold values that are specified in this JCL.

You can use the cataloged procedures FABAPC and FPXAPC1 that are located in your JCL library (SHFPSAMP).

In this example:

- The AREA=(DB22AR0) keyword specifies that a single area is processed.
- The FULLSTEP=YES keyword is not specified.
- DAR120, DAR13SO, DAR3CSO, SORT12, and SORT3CS DD statements specify that the pointer and segment information is passed to the DEDB Pointer Checker job steps.
- After the OER job step, all the job steps of DEDB Pointer Checker that are typically required for the TYPRUN=RPT option, except for the FABADA1 step, are specified in this JCL.
- The THRCTL DD statement of the FABADA5 step specifies the threshold values.

```

//OER      EXEC FPUTIL, DBD=DEDBJN22, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//         DD DISP=SHR, DSN=IMSVS.SDFSRESL
//FPXIN    DD *
REORG AREA=(DB22AR0),
PTRCHK=YES
PTRCHK
TYPRUN=RPT,
CHKUOW=ALL
/*
//DAR120   DD DISP=(NEW, PASS), DSN=&&DA12,
//         UNIT=SYSDA, SPACE=(CYL, (1, 1), RLSE)
//DAR13SO  DD DISP=(NEW, PASS), DSN=&&DA13S,
//         UNIT=SYSDA, SPACE=(CYL, (1, 1), RLSE)
//DAR3CSO  DD DISP=(NEW, PASS), DSN=&&DA3CS,
//         UNIT=SYSDA, SPACE=(CYL, (1, 1), RLSE)
//SORT12   DD DISP=(NEW, PASS), DSN=&&SORT12,
//         UNIT=SYSDA, SPACE=(CYL, (1, 1), RLSE)
//SORT3CS  DD DISP=(NEW, PASS), DSN=&&SORT3CS,
//         UNIT=SYSDA, SPACE=(CYL, (1, 1), RLSE)
//*
//SORT3CS  EXEC PGM=SORT
//SYSOUT   DD SYSOUT=*
//SYSIN    DD DISP=(OLD, DELETE), DSN=&&SORT3CS
//SORTIN   DD DISP=(OLD, DELETE), DSN=&&DA3CS
//SORTOUT  DD DISP=(NEW, PASS), DSN=&&SDA3CS,
//         UNIT=SYSDA, SPACE=(CYL, (1, 1), RLSE)
//SORTWK01 DD UNIT=SYSDA, SPACE=(CYL, (2, 2))
//SORTWK02 DD UNIT=SYSDA, SPACE=(CYL, (2, 2))
//SORTWK03 DD UNIT=SYSDA, SPACE=(CYL, (2, 2))
//*
//FABADA3  EXEC PGM=FABADA3
//SYSPRINT DD SYSOUT=*
//DAR13SI  DD DISP=(OLD, DELETE), DSN=&&DA13S
//DAR3CSI  DD DISP=(OLD, DELETE), DSN=&&SDA3CS
//DAR4ERRS DD DISP=(NEW, PASS), DSN=&&DA4ERRS,
//         UNIT=SYSDA, SPACE=(CYL, (2, 2), RLSE)
//DAR130   DD DISP=(NEW, PASS), DSN=&&RDA13,
//         UNIT=SYSDA, SPACE=(CYL, (1, 1), RLSE)
//SORT4CTL DD DISP=(NEW, PASS), DSN=&&SORT4,
//         UNIT=SYSDA, SPACE=(TRK, (2, 1), RLSE)
//SORTR13  DD DISP=(NEW, PASS), DSN=&&SORTR13,
//         UNIT=SYSDA, SPACE=(TRK, (2, 1), RLSE)
//*
//SORT12   EXEC PGM=SORT
//SORTLIB  DD DSN=SYS1.SORTLIB, DISP=SHR
//SYSOUT   DD SYSOUT=*
//SYSIN    DD DISP=(OLD, DELETE), DSN=&&SORT12
//SORTWK01 DD UNIT=SYSDA, SPACE=(CYL, (2, 2))
//SORTWK02 DD UNIT=SYSDA, SPACE=(CYL, (2, 2))
//SORTWK03 DD UNIT=SYSDA, SPACE=(CYL, (2, 2))
//SORTIN   DD DISP=(OLD, DELETE), DSN=&&DA12
//SORTOUT  DD DISP=(NEW, PASS), DSN=&&SDA12,
//         UNIT=SYSDA, SPACE=(TRK, (2, 1), RLSE)
//*
//SORTR13  EXEC PGM=SORT
//SYSOUT   DD SYSOUT=*
//SYSIN    DD DISP=(OLD, DELETE), DSN=&&SORTR13
//SORTIN   DD DISP=(OLD, DELETE), DSN=&&RDA13
//SORTOUT  DD DISP=(NEW, PASS), DSN=&&SRDA13,
//         UNIT=SYSDA, SPACE=(CYL, (1, 1), RLSE)
//SORTWK01 DD UNIT=SYSDA, SPACE=(CYL, (2, 2))
//SORTWK02 DD UNIT=SYSDA, SPACE=(CYL, (2, 2))
//SORTWK03 DD UNIT=SYSDA, SPACE=(CYL, (2, 2))
//*
//FABADA5  EXEC PGM=FABADA5
//SYSPRINT DD SYSOUT=*
//REPORTS  DD SYSOUT=*
//DAR12I   DD DISP=(OLD, DELETE), DSN=&&SDA12
//DAR13I   DD DISP=(OLD, DELETE), DSN=&&SRDA13
//THRCTL   DD *
T          060                      030
/*

```

Figure 456. Online Expert Reorganization example 5: Specifying FABADA5 threshold values

See the topic "FABADA5 THRCTL DD data set" in the *IMS Fast Path Solution Pack: IMS Fast Path Basic Tools User's Guide* for the threshold values that can be specified on the THRCTL DD statement.

Example 6: Generating reports for database analysis

You can use the JCL template to generate the reports that contain information about database records and randomizing for both the area and all the UOWs of the area. See [“JCL template for running the Online Expert Reorganization utility” on page 1162.](#)

Example 7: Generating reports to obtain large database records information

The following figure shows example JCL for generating the reports that contain information about the large database records.

In this example, the REPORT control statement, LARGEREC_REPORT=YES, and LARGEREC_COUNT=(10,20) specify that the Largest Database Records report and the Consolidated Largest Database Records report are generated.

```
//FPXIN DD *
REORG AREA=(DB22AR0,DB22AR1),
PTRCHK=YES
PTRCHK
TYPRUN=RPT,
FULLSTEP=YES,
CHKUOW=ALL
REPORT
LARGEREC_REPORT=YES,
LARGEREC_COUNT=(10,20)
/*
```

Figure 457. Online Expert Reorganization example 7: Generating reports to obtain large database records information

Example 8: Generating reports for UOWs in which exceptions were detected

The following figure shows example JCL for generating the reports only for the UOWs in which exceptions were detected.

In this example, EXC_RDOVFS=, EXC_UDOVFFS=, and EXC_RBSEFS= for the PTRCHK statement specify the threshold values for generating the UOW Exception reports.

```
//OER EXEC FPUTIL,DBD=DEDBJN22,IMSID=IMS1
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
//FPXIN DD *
REORG AREA=(DB22AR0,DB22AR1),
PTRCHK=YES
PTRCHK
TYPRUN=RPT,
CHKUOW=ALL,
FULLSTEP=YES,
EXC_RDOVFFS=40,
EXC_UDOVFFS=35,
EXC_RBSEFS=40
/*
```

Figure 458. Online Expert Reorganization example 8: Generating reports for UOWs in which exceptions were detected

Example 9: Generating free space analysis reports

The following figure shows example JCL for generating the reports that are required for free space analysis.

In this example, TYPRUN=FS for the PTRCHK statement specifies that the Freespace Analysis report is generated.

```

//OPC      EXEC FPUTIL, DBD=DEDBJN22, IMSID=IMS1
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//         DD DISP=SHR, DSN=IMSVS.SDFSRESL
//         DD DISP=SHR, DSN=ITBVS.SHKTLOAD
//FPXIN    DD *
REORG AREA=(DB22AR0, DB22AR1),
        PTRCHK=YES
        PTRCHK
        TYPRUN=FS,
        CHKUOW=ALL,
        FULLSTEP=YES
/*

```

Figure 459. Online Expert Reorganization example 9: Generating reports for free space analysis

IMS HP Image Copy JCL examples for monitoring and tuning DEDBs

Use these JCL examples to monitor and tune DEDBs by using the IMS HP Image Copy with the DEDB HASH Check option.

Subsections:

- [“JCL template for running IMS HP Image Copy with the DEDB HASH Check option” on page 1167](#)
- [“Example 1: Storing statistics in the Sensor Data repository” on page 1167](#)
- [“Example 2: Generating free space analysis reports” on page 1168](#)
- [“Example 3: Generating the Segment Length Distribution report” on page 1168](#)

JCL template for running IMS HP Image Copy with the DEDB HASH Check option

The following figure shows a JCL template for running IMS HP Image Copy with the DEDB HASH Check option.

In this example:

- Fast scan of database integrity verification is performed for the DEDB areas.
- The input area data sets (ADSs) to be validated are dynamically allocated. The allocation information (DD name and DS name of the ADSs) is obtained from DBRC.

```

//HPICHASH EXEC PGM=FABJMAIN
//STEPLIB DD DISP=SHR, DSN=HPFP.SHFPLMD0
//         DD DISP=SHR, DSN=HPS.SHPSLMD0
//         DD DISP=SHR, DSN=IMSVS.SDFSRESL
//DFSRESLB DD DISP=SHR, DSN=IMSVS.SDFSRESL
//DFSPRINT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//ICEPRINT DD SYSOUT=*
//MSGOUT   DD SYSOUT=*
//REPORTS  DD SYSOUT=*
//IMS      DD DISP=SHR, DSN=IMSVS.DBDLIB
//IMSDALIB DD DISP=SHR, DSN=IMSVS.MDALIB
//ICEIN DD *
GLOBAL DBRC=Y,
        DEBPC=Y,
        ICHLQ=ICOUT.HFP,
        UNIT=SYSDA,
        SPACE=(CYL,100,100)
AIC     DBD=DEDBJN22, AREA=DB22AR0
AIC     DBD=DEDBJN22, AREA=DB22AR1
/*

```

Figure 460. JCL template for IMS HP Image Copy with the DEDB HASH Check option

Example 1: Storing statistics in the Sensor Data repository

The following figure shows example JCL for storing the statistical information of an entire area that is registered in DBRC into the Sensor Data repository.

Prerequisite: To run this JCL, an IMS Tools Knowledge Base server and its repositories must be configured.

In this example:

- The IMS Tools Base library (ITBVS.SHKTLOAD) is concatenated to the STEPLIB DD.
- ITKBSRVR= for the GLOBAL command specifies the IMS Tools KB server XCF group.
- SENSOR=YES for the GLOBAL command specifies that the statistics are stored in the Sensor Data repository.
- The HFPSRT DD statement specifies the output data set for the Sensor Data Statistics report.

```
//HPICHASH EXEC PGM=FABJMAIN
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=HPS.SHPSLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
// DD DISP=SHR,DSN=ITBVS.SHKTLOAD
//DFSRESLB DD DISP=SHR,DSN=IMSVS.SDFSRESL
//DFSPRINT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//ICEPRINT DD SYSOUT=*
//MSGOUT DD SYSOUT=*
//REPORTS DD SYSOUT=*
//HFPSRT DD SYSOUT=*
//IMS DD DISP=SHR,DSN=IMSVS.DBDLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//ICEIN DD *
GLOBAL DBRC=Y,
        DEDBPC=Y,
        ICHLQ=ICOUT.HFP,
        UNIT=SYSDA,
        SPACE=(CYL,100,100),
        ITKBSRVR=FPQSRVFP,
        SENSOR=YES
AIC DBD=DEDBJN23,AREA=DB23AR1
AIC DBD=DEDBJN23,AREA=DB23AR2
/*
```

Figure 461. IMS HP Image Copy example 1: Storing statistics in the Sensor Data repository

Example 2: Generating free space analysis reports

You can use the JCL template to generate the Freespace Analysis report that can be used for free space analysis. See [“JCL template for running IMS HP Image Copy with the DEDB HASH Check option”](#) on page 1167.

Example 3: Generating the Segment Length Distribution report

The following figure shows example JCL for generating the Segment Length Distribution report.

In this example, DEDBPC=(Y,SEGLDIST) for the GLOBAL command specifies that the Segment Length Distribution report is generated.

```

//HPICHASH EXEC PGM=FABJMAIN
//STEPLIB DD DISP=SHR,DSN=HPFP.SHFPLMD0
// DD DISP=SHR,DSN=HPS.SHPSLMD0
// DD DISP=SHR,DSN=IMSVS.SDFSRESL
//DFSRESLB DD DISP=SHR,DSN=IMSVS.SDFSRESL
//DFSPRINT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//ICEPRINT DD SYSOUT=*
//MSGOUT DD SYSOUT=*
//REPORTS DD SYSOUT=*
//IMS DD DISP=SHR,DSN=IMSVS.DBDLIB
//IMSDALIB DD DISP=SHR,DSN=IMSVS.MDALIB
//ICEIN DD *
GLOBAL DBRC=Y,
        DEDBPC=(Y,SEGLDIST),
        ICHLQ=ICOUT.HFP,
        UNIT=SYSDA,
        SPACE=(CYL,100,100)
AIC DBD=DEDBJN23,AREA=DB23AR1
AIC DBD=DEDBJN23,AREA=DB23AR2
/*

```

Figure 462. IMS HP Image Copy example 3: Generating the Segment Length Distribution report

Chapter 44. Viewing the reports stored in IMS Tools KB

There are many ways to use the ISPF user interface to view the IMS HP Fast Path Utilities reports that are stored in the IMS Tools KB repository.

The examples provided in the following topics show some of the typical ways. For details about viewing reports in IMS Tools KB, see the topic "Finding and viewing reports" in the *Tools Base IMS Tools Knowledge Base User's Guide*.

Topics:

- [“Viewing DEDB area space utilization trend” on page 1171](#)
- [“Viewing specific DEDB management information” on page 1171](#)

Viewing DEDB area space utilization trend

You can analyze the DEDB area space utilization trend from the history reports.

Procedure

1. For FPA, specify the ITKBSRVR keyword, the REPORT subcommand, and the PTRCHKLVL=FULL keyword in the ANALYZE JCL and submit the JCL. For FPO, specify the ITKBSRVR keyword and the FULLSTEP=YES keyword in the OPC JCL and submit the JCL.
2. Select option 6 **List of Reports** from the IMS Tools Knowledge Base main menu panel.
3. Select **HPFPU-FPA-DEDB Area Analysis report** from the **Report List** panel by using the **List** row action (S). Press Enter.
4. Use the Row Actions commands to view the history reports including the most recent report, and analyze the DEDB area space utilization trend from these reports.

Note: The most recent report and the history reports can be identified with the Step Start timestamp.

5. View all the reports that were generated in the same step as the DEDB Area Analysis report by using the **STEP** row action (T). Press Enter.

Viewing specific DEDB management information

You can view how a specific DEDB area has been managed through time from the history reports.

Procedure

1. Select option 1 **List of Databases with reports** from the IMS Tools Knowledge Base main menu panel.
2. Select the specific database from the **Database List** panel by using the **List** row action (S). Press Enter.
3. Use the Row Actions commands to select the reports that are related to the specific database and view how the specific DEDB area has been managed through time. For example:
 - The Reorganization Processing Summary report generated by the FPA Reorg function or the OER utility shows the information of the area definition, the reorganization processing options, the result of UOW evaluation, and the result of reorganization.
 - The UOW Space Statistics report generated by the FPA Reorg function or the OER utility shows the statistics before and after the reorganization for each reorganized UOW.
 - The Unloaded report and Reloaded report generated by the FPA Change function shows the segment counts and area totals.
 - The OAE messages generated by the OAE utility show the structure of the extended area in comparison with the original area.

Chapter 45. Record layouts

These reference topics provide information about record layouts and how to map records with macros.

Topics:

- [“Historical records layout” on page 1173](#)
- [“Largest database records layout” on page 1176](#)
- [“Unloaded segment records layout” on page 1177](#)
- [“Total segments records layout” on page 1183](#)
- [“Sequence error segment records layout” on page 1183](#)
- [“DURDBDFN records layout” on page 1185](#)
- [“ERRORSEG records layout” on page 1185](#)
- [“Extracted records layout” on page 1186](#)
- [“OUTFILE records layout” on page 1190](#)
- [“UOW group definition record layout” on page 1191](#)
- [“Build pointer segment records layout” on page 1193](#)
- [“Resync pointer segment records and duplicate pointer segment records layout” on page 1193](#)

Historical records layout

These topics present the layouts for the historical records that are created by the FPA Analyze function, the FPB DEDB Pointer Checker utility, and the OPC utility.

History file records layout

History file records are created by the FPA Analyze function, the FPB DEDB Pointer Checker utility, the FPO OPC utility, and the HPFPU Hash Check support for IMS HP Image Copy.

This topic contains product sensitive programming interface information.

PSPI

You can use the HFPAMHST macro (FPA, FPO, and HPFPU Hash Check support for IMS HP Image Copy) or the FABAMHST macro (FPB) to map these records.

The following table shows the record layout, which is a product-sensitive programming interface. This data set contains 100 bytes fixed length records. The block size, if coded in your JCL statement, must be a multiple of 100.

Table 133. Record layout of the History file

Field description	COBOL usage	Format
DBD name	X(8)	
Area name	X(8)	
Image copy date	X(8)	YYYYMMDD
Number of DB records	S9(9)	Packed
Average record length	S9(7)	Packed
Maximum record length	S9(7)	Packed
Minimum record length	S9(5)	Packed

Table 133. Record layout of the History file (continued)

Field description	COBOL usage	Format
Percentage of free space RAA BASE	S9(3)	Packed
Percentage of free space DOVF	S9(3)	Packed
Percentage of free space IOVF	S9(3)	Packed
Average synonym chain length	S9(5)V99	Packed
Maximum synonym chain length	S9(5)	Packed
Average root I/O	S9(5)V99	Packed
Maximum root I/O	S9(5)	Packed
Average record I/O	S9(5)V99	Packed
Maximum record I/O	S9(5)	Packed
Percentage of available IOVF CIs	S9(3)	Packed
Percentage of UOWs using IOVF	S9(3)	Packed
Average IOVF CIs used by a UOW	S9(5)V99	Packed
Maximum IOVF CIs used by a UOW	S9(5)	Packed
Percentage of records using IOVF	S9(3)	Packed
Percentage of RAA BASE CIs using either DOVF or IOVF	S9(3)	Packed
Percentage of UOWs using DOVF	S9(3)	Packed
Average DOVF CIs used by a UOW	S9(5)V99	Packed
Maximum DOVF CIs used by a UOW	S9(5)	Packed
Filler	X(9)	

If the average and/or maximum record length field exceeds 9999999, the value of this field is set to the maximum, 9999999. If the minimum record length field exceeds 99999, the field is set to maximum, 99999.

For the HPFPU Hash Check support for IMS HP Image Copy, the following fields are always set to 0.

- Average record length
- Maximum record length
- Minimum record length
- Average synonym chain length
- Maximum synonym chain length
- Average root I/O
- Maximum root I/O
- Average record I/O
- Maximum record I/O
- Percentage of records using IOVF



History2 file records layout

History2 file records are created by the FPA Analyze function, the FPB DEDB Pointer Checker utility, and the OPC utility. It contains the history data, which is totaled by UOW range.

This topic contains product sensitive programming interface information.



You can use the HFPAMHS2 macro (FPA and FPO) or the FABAMHS2 macro (FPB) to map the History2 record in the assembler language.

The following table presents the History2 file record format, which is a product-sensitive programming interface. This data set contains 200 bytes fixed length records. The block size, if coded in your JCL statement, must be a multiple of 200.

Table 134. Record layout of the History2 file

Field description	COBOL usage	Format
DBD name	X(8)	
Area name	X(8)	
Date of Analysis	S9(7)	YYYYDDD packed
filler	X(2)	
Low UOW Range Number	S9(8)	Binary
High UOW Range Number	S9(8)	Binary
Number of DB Records	S9(9)	Packed
Average Record Length	S9(7)	Packed
Maximum Record Length	S9(7)	Packed
Minimum Record length	S9(5)	Packed
P/C Freespace RAABASE	S9(3)	Packed
P/C Freespace DOVF	S9(3)	Packed
P/C Freespace IOVF	S9(3)	Packed
Average Syn Chain Length	S9(5)V99	Packed
Maximum Syn Chain Length	S9(5)	Packed
Average Root I/O	S9(5)V99	Packed
Maximum Root I/O	S9(5)	Packed
Average Record I/O	S9(5)V99	Packed
Maximum Record I/O	S9(5)	Packed
P/C Available IOVF CIs ¹	S9(3)	Packed
P/C UOWs Using IOVF	S9(3)	Packed
Average IOVF CIs Used By a UOW	S9(5)V99	Packed
Maximum IOVF CIs Used By a UOW	S9(5)	Packed
P/C Records Using IOVF	S9(3)	Packed
P/C RAABASE CIs Using either	S9(3)	Packed

Table 134. Record layout of the History2 file (continued)

Field description	COBOL usage	Format
DOVF or IOVF		
P/C UOWs Using DOVF	S9(3)	packed
Average DOVF CIs Used By a UOW	S9(5)V99	packed
Maximum DOVF CIs Used By a UOW	S9(5)	packed
Report Heading	X(16)	
P/C Usable Freespace RAABASE	S9(3)	packed
P/C Usable Freespace DOVF	S9(3)	packed
P/C Usable Freespace IOVF	S9(3)	packed
filler	X(1)	
UOW-1 Value ¹	S9(4)	Binary
UOW-2 Value ¹	S9(4)	Binary
Root-1 Value ¹	S9(8)	Binary
Root-2 Value ¹	S9(8)	Binary
Number of SDEP CIs ¹	S9(8)	Binary
CI size ¹	S9(4)	Binary
filler	X(42)	
P/C RAPs with space for Root	S9(3)	packed
P/C free space in SDEP portion ¹	S9(3)	packed
Repository Group	X(4)	
filler	X(12)	

Note: These items are computed for the entire area, and are not dependent on any UOW range specified.



Largest database records layout

Largest database records are created by the FPB DEDB Pointer Checker utility.

This topic contains product sensitive programming interface information.



You can use the FABALREC macro to map these records.

The following table shows the format of the largest database records, which is a product-sensitive programming interface.

Table 135. Record layout of the largest database records file

Field name	Assembler definition	Description
LRECARNM	CL8	Area name

Table 135. Record layout of the largest database records file (continued)

Field name	Assembler definition	Description
LRECAR#	PL3	Area number (packed decimal)
LRECSIZE	PL6	Record size (packed decimal)
LRECSEG#	PL5	Number of segments (packed decimal)
LRECRAP#	PL3	RAP number (packed decimal)
	XL3	(reserved)
LRECRBA	XL4	Root RAP RBA
LRECSRBA	XL4	Root segment RBA
	XL2	(reserved)
LRECKEYL	PL2	Root key length (packed decimal)
LRECKEY	XL256	Root key value

This record layout is a product-sensitive programming interface for the customer.



Unloaded segment records layout

Unloaded area records are created by the FPA Unload function and the FPB Unload utility.

For the format of the unloaded segment records that were created during an Unload process with FORMAT=USER specification, see [“Extracted records layout”](#) on page 1186.

This topic contains product sensitive programming interface information.



You can use the HFPZUSR macro (FPA), the FABCMUSR macro (FPB), or the UR#MUSRO (IMS DBT 2.1 compatibility) macro to map these records.

Subsections:

- [“Record layout of the unloaded segment records file”](#) on page 1177
- [“Record layout of the area information records file”](#) on page 1179
- [“Record layout of the DBD information records”](#) on page 1181

Record layout of the unloaded segment records file

The following table shows the record layout, which is a product-sensitive programming interface.

Table 136. Record layout of the unloaded segment records file

Field name	Assembler definition	Description
		"RDW" fields
USRRDW	H H	RDW "LL" field RDW "ZZ" field
		Sort key part of record
USRA#	H	Area number

Table 136. Record layout of the unloaded segment records file (continued)

Field name	Assembler definition	Description
USRRAP	XL4	RAP RBA
USRLCFLG	X	"Insert Limit Count" Flag X'00': Twin not greater than limit X'FF': Twin greater than limit
USRRKL	XL2	ROOT segment key length
USRRKV	XL nn	ROOT segment key value nn is the length of the root segment key field
USRLCG#	XL2	"Insert Limit Count" group
USRSCSQ2	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 2 Direct Dependent segments (binary zeros for Root Segment). <ul style="list-style-type: none"> • First byte is seg-code of the segment in this Segment data record • Second-forth bytes are a sequential counter of the occurrence of this segment type within this database record
USRSCSQ3	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 3 Direct Dependent segments (binary zeros for Root Segment).
USRSCSQ4	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 4 Direct Dependent segments (binary zeros for Root Segment).
USRSCSQ5	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 5 Direct Dependent segments (binary zeros for Root Segment).
USRSCSQ6	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 6 Direct Dependent segments (binary zeros for Root Segment).
USRSCSQ7	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 7 Direct Dependent segments (binary zeros for Root Segment).
USRSCSQ8	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 8 Direct Dependent segments (binary zeros for Root Segment).
USRSCSQ9	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 9 Direct Dependent segments (binary zeros for Root Segment).
USRSCSQA	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 10 Direct Dependent segments (binary zeros for Root Segment).
USRSCSQB	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 11 Direct Dependent segments (binary zeros for Root Segment).

Table 136. Record layout of the unloaded segment records file (continued)

Field name	Assembler definition	Description
USRSCSQC	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 12 Direct Dependent segments (binary zeros for Root Segment).
USRSCSQD	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 13 Direct Dependent segments (binary zeros for Root Segment).
USRSCSQE	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 14 Direct Dependent segments (binary zeros for Root Segment).
USRSCSQF	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 15 Direct Dependent segments (binary zeros for Root Segment).
DATA PORTION OF RECORD		
USRPFLG1	X	Processing Flag 1 X'01': More Segs on Twin Chain (But 'ISRT LIMCT'Reached) X'02': Unloaded Seg is compressed. X'04': Fixed-length segment (see note).
USRSEGCD	H	Segment Code
USRSHLVL	H	Hierarchical level of segment
USRPCD	H	"Hier. Parent's Seg Code" <ul style="list-style-type: none"> • First child segment: Hierarchical parent's segment code. • Second or subsequent child segment: Segment code of the segment itself.
USRPFLG2	X	Processing Flag 2 Bit(s) set if segment is target of a Subset pointer. Bits are set left to right for pointers 1 - 8
Blank	X	1 byte filler
USRSDATA	XLnnn	Segment Data First two bytes of USRSDATA are the length of the segment

Note: The first two bytes of USRSDATA are the length of the segment that was generated by the IMS system. From the viewpoint of application program, the first two bytes must be removed to get actual data.

Record layout of the area information records file

For each unloaded area, there is a special unloaded record, which is an area information record. The following table shows this record layout, which is a product-sensitive programming interface.

Table 137. Record layout of the area information records file

Field name	Assembler definition	Description
		"RDW" fields
USRRDW	H H	RDW "LL" field RDW "ZZ" field
		Sort key part of record
USRA#	H	Area number
USRRAP	XL4	X'00000000'
USRLCFLG	X	X'00'
USRRKL	XL2	ROOT segment key length
USRRKV	XL nn	ROOT segment key value nn is the length of the root segment key field
USRLCG#	XL2	X'0000'
USRSCSQ2	XL1,XL3	X'00',X'000000'
USRSCSQ3	XL1,XL3	X'00',X'000000'
USRSCSQ4	XL1,XL3	X'00',X'000000'
USRSCSQ5	XL1,XL3	X'00',X'000000'
USRSCSQ6	XL1,XL3	X'00',X'000000'
USRSCSQ7	XL1,XL3	X'00',X'000000'
USRSCSQ8	XL1,XL3	X'00',X'000000'
USRSCSQ9	XL1,XL3	X'00',X'000000'
USRSCSQA	XL1,XL3	X'00',X'000000'
USRSCSQB	XL1,XL3	X'00',X'000000'
USRSCSQC	XL1,XL3	X'00',X'000000'
USRSCSQD	XL1,XL3	X'00',X'000000'
USRSCSQE	XL1,XL3	X'00',X'000000'
USRSCSQF	XL1,XL3	X'00',X'000000'
		FORMAT AND SDEP INFORMATION
USRFMTID	USRLCFLG+5,XL2	Format type C'TR': Trimmed format null: DBT format
USRSDEP	USRLCFLG+7,XL2	SDEP option C'LO': SDEP=LOGICAL C'PH': SDEP=PHYSICAL change C'PN': SDEP=PHYSICAL accompanied by a DBD change (FPB only) null: SDEP=NO

Table 137. Record layout of the area information records file (continued)

Field name	Assembler definition	Description
USRSDLB	USRLCFLG+9,XL8	SDEP Logical Begin
USRSDLE	USRLCFLG+17,XL8	SDEP Logical End
USRSDTM	USRLCFLG+25,XL8	SDEP begin timestamp
USRSDFS	USRLCFLG+33,XL4	SDEP block # of first SDEP
USRSDFB	USRLCFLG+37,XL4	SDEP block # of beyond SDEP
		Data portion of record
USRPFLG1	X	X'00'
USRSEGCD	H	H'0'
USRSHLVL	H	H'0'
USRPCD	H	H'0'
USRPFLG2	X	X'00'
Blank	X	X'00'

If the unloaded area is empty, there is an area information record but no unloaded area record for the area.

Record layout of the DBD information records

The following tables show the records layout for the DBD information records in the USR files. These tables contain product-sensitive programming interface information.

The DBD information records are written into the USR files by the Unload process when ACBCHK=YES is specified. You can use the HFPZUSR macro (FPA) to map these records.

Table 138. Record layout of the first DBD information record

Field name	Assembler definition	Description
USRRDW	H H	RDW "LL" field RDW "ZZ" field
USRA#	H	X'0000'
USRDBDFG	XL2	DBD record type flag X'0001': The first record
USRDBDLL	H	Length of DBD data part
USRBDVVR	XL2	DBD information version
USRRCCNT	F	DBD records counter

Table 138. Record layout of the first DBD information record (continued)

Field name	Assembler definition	Description
USRACBDS	XL44	ACB DSNAME If the IMS management of ACBs is enabled, this field contains either of the following values: <ul style="list-style-type: none"> • *IMS DIRECTORY (CURRENT), if IMSCATACB_INPUT=CURRENT is specified. • *IMS DIRECTORY (PENDING), if IMSCATACB_INPUT=PENDING is specified.
USRDBDSZ	F	Size of DBD information
USRDDTAD	A	Address of UNLOAD DDT
USRINFO	3F	Unload information

Table 139. Record layout of the second and subsequent DBD information records

Field name	Assembler definition	Description
USRRDW	H H	RDW "LL" field RDW "ZZ" field
USRA#	H	X'0000'
USRDBDFG	XL2	DBD record type flag X'0002': The second record
USRBDLL	H	Length of DBD data part
USRBD	XLnn	DBD data part

Table 140. Record layout of the end DBD information record

Field name	Assembler definition	Description
USRRDW	H H	RDW "LL" field RDW "ZZ" field
USRA#	H	X'0000'
USRDBDFG	XL2	DBD record type flag X'FFFF': The end record
USRBDLL	H XL2	Length of DBD data part (reserved)
USRRCCNT	F	DBD records counter



Total segments records layout

Total segments records are created by the FPA Unload, Reload, Change functions, and the FPB DEDB Unload utility.

This topic contains product sensitive programming interface information.

PSPI

You can use the HFPZTOT macro (FPA) or the FABC1TAL macro (FPB) to map these records.

The following table shows the format of total segments record, which is a product-sensitive programming interface.

Table 141. Record layout of the total segments records file

Field name	Assembler definition	Description
TOTAREA#	PL2	Area number (packed decimal)
TOTSEGMN	CL8	Segment name
TOTSEGCT	PL5	Segment count
TOTUPDATE	PL4	Unloaded or reloaded date (00YYDDD)
TOTUTIME	PL4	Unloaded or reloaded time (OHHMMSS)
TOTRECPR	CL1	FPA Unload: 'U' FPA Reload: 'R' FPA Change: 'C' FPB Unload: 'H'
TOTRECVE	CL1	HFP Version C'2'
TOTRECRL	CL1	HFP Release C'1'
TOTAREA2	PL3	Area number for GT240
TOTDBDNM	CL8	DBD name (FPA only)
TOTARNM	CL8	AREA name (FPA only)
	XL15	Reserved

PSPI

If you want to use this macro for an application, TOTAREA2 field should be used instead of TOTAREA#. TOTAREA# field is supported for the purpose of compatibility, to serve users who use OEM's HFPTOTAL data set.

Sequence error segment records layout

Sequence error segment records are created by the FPA Unload and Change functions.

This topic contains product sensitive programming interface information.

PSPI

You can use the HFPZSER macro to map these records.

The following table shows the layout, which is a product-sensitive programming interface.

Table 142. Record layout of the sequence error segment file

Field name	Assembler definition	Description
		"RDW" fields
SERRDW	H H	RDW "LL" field RDW "ZZ" field
SERDBD_NAME	CL8	DBD name
SERAREA_NAME	CL8	Area name
SERAREA_NUMBER2	XL2	Area number
SERSEGNAME	CL8	Segment name
SERSEGLEVEL	XL1	Segment level
SERSEGCODE	XL1	Segment code
SERSCSQ2	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 2 Direct Dependent segments (binary zeros for Root Segment). <ul style="list-style-type: none"> • First byte is seg-code of the segment in this Segment data record • Second-forth bytes are a sequential counter of the occurrence of this segment type within this database record
SERSCSQ3	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 3 Direct Dependent segments (binary zeros for Root Segment).
SERSCSQ4	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 4 Direct Dependent segments (binary zeros for Root Segment).
SERSCSQ5	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 5 Direct Dependent segments (binary zeros for Root Segment).
SERSCSQ6	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 6 Direct Dependent segments (binary zeros for Root Segment).
SERSCSQ7	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 7 Direct Dependent segments (binary zeros for Root Segment).
SERSCSQ8	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 8 Direct Dependent segments (binary zeros for Root Segment).
SERSCSQ9	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 9 Direct Dependent segments (binary zeros for Root Segment).
SERSCSQA	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 10 Direct Dependent segments (binary zeros for Root Segment).

Table 142. Record layout of the sequence error segment file (continued)

Field name	Assembler definition	Description
SERSCSQB	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 11 Direct Dependent segments (binary zeros for Root Segment).
SERSCSQC	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 12 Direct Dependent segments (binary zeros for Root Segment).
SERSCSQD	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 13 Direct Dependent segments (binary zeros for Root Segment).
SERSCSQE	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 14 Direct Dependent segments (binary zeros for Root Segment).
SERSCSQF	XL1,XL3	Segment code and segment sequence number. Field for Hierarchical Level 15 Direct Dependent segments (binary zeros for Root Segment).
SERSEGPKEYL	XL2	The length of SERSEGPKEYL
SERSEGPKEY	XLnnn	Parent's concatenated key
SERSEGDATA	XLnnn	Segment data 1st two bytes of SERSEGDATA is the length of the segment

When a segment is detected as a sequence error, the second to forth bytes of SERSCSQx, which corresponds to the segment level with the key sequence error, show the number of occurrences of previous segments that have no sequence errors. Similarly, in the sequence error segment data record for its child segment, the second to forth bytes of SERSCSQx, which corresponds to the segment level with the key sequence error, show the number of occurrences of previous segments that have no sequence errors.



DURDBDFN records layout

DURDBDFN records are created by the FPB DEDB Unload utility and the FPB Database Definition Record Create utility.

For a database that contains 240 or fewer areas, the record format is provided by a macro, FABCMDT, which is a product-sensitive user interface. When the number is greater than 240, the record format and its use are not disclosed to the users of FPB.

If you want to retrieve unloaded segment data for GT240 DEDB, use FABCUR7 so that you do not need to consider the format of the DURDBDFN.

If you want to create the reload segment records data set (DURDzzzO or XDzzzzzO), the SORT control statement data set (DURSzzzO or XSzzzzzO), and the DURDBDFN data set for GT240 DEDB, use FABCUR6 so that there will be no need for user's applications to generate the DURDBDFN data set.

ERRORSEG records layout

ERRORSEG records are created by the FPB DEDB Reload utility.

This topic contains product sensitive programming interface information.

The ERRORSEG record has a specific record layout.

The following table shows the layout for the ERRORSEG file records, which is a product-sensitive programming interface.

Table 143. Record layout of the ERRORSEG records file

Field name	Assembler definition	Description
ERLLZZ	H H	RDW "LL" Field RDW "ZZ" Field
ERAREA#	H	Area number
ERRAPRBA	XL4	RAP RBA
ERILCFLG	X	"Insert Limit Count" Flag => set to X'00'
ERROOTKL	XL2	ROOT Seg key length
ERROOTKY	XLnn	ROOT Seg Key Value => nn is the length of the root segment key field
ERILCG#	XL2	"Insert Limit Count" Group => set to X'0000'
ERSEGNAM	CL8	Segment name
ERRSGDAT	XLnn	Segment Data => This part is the same as USRSDATA in the unload record.

Extracted records layout

Extracted records are created by the FPA Extract function and the FPO ODE utility.

This topic contains product sensitive programming interface information.

Subsections:

- [“Standard format” on page 1186](#)
- [“DBT Unload/Reload format” on page 1189](#)
- [“HD Unload format” on page 1189](#)
- [“User format” on page 1189](#)

Standard format

You can use the HFPZSTD macro to map these records. For more information, see:

- [Table 144 on page 1187](#) for the record layout of the DBD control information record for SORT=NO.
- [Table 145 on page 1187](#) for the record layout of the segment data record for SORT=NO.
- [Table 146 on page 1188](#) for the record layout of the DBD control information record for SORT=YES.
- [Table 147 on page 1188](#) for the record layout of the segment data record for SORT=YES.

The following table shows the record layout of the DBD control information record for SORT=NO, which is a product-sensitive programming interface.

Table 144. Record layout of the DBD control information records file for SORT=NO

Field name	Assembler definition	Description
PFXLL	H	RDW LL
PFXZZ	H	RDW ZZ
PFXFLG	C	C' ' (blank)
PFXFLG1	X	Reserved
PFXSEGC	H	X'0000'
STDNOSE	H	Number of segment table entries
STDSETE		Segment table entry
STDSESC	H	Segment code
STDSEHI	H	Segment hierarchical level
STDSEPA	H	Parent segment code
STDSEGN	CL8	Segment name

The following table shows the record layout of the segment data record for SORT=NO, which is a product-sensitive programming interface.

Table 145. Record layout of the segment data records file for SORT=NO

Field name	Assembler definition	Description
PFXLL	H	RDW LL
PFXZZ	H	RDW ZZ
PFXFLG	C	C' ' (blank)
PFXFLG1	X	Bit-value Flag <ul style="list-style-type: none"> • X'04' Fixed segment • X'02' Compressed segment. This flag is set on when the COMPRTN is specified for the segment in the DBD definition and either of the following conditions is met: <ul style="list-style-type: none"> – For the ODE utility of FPO, EXPAND=NO is specified on the FPXCTL statement. – For the Extract function of FPA, COMPRESS=YES is specified on the EXTRACT command.
PFXSEGC	H	Segment code
STDCCKL	H	Concatenated key length
STDCCK	X (Variable)	Concatenated key
STDDATAL	H	Extracted segment data length
STDDATA	X (Variable)	Extracted segment data

The following table shows the record layout of the DBD control information record for SORT=YES, which is a product-sensitive programming interface.

Table 146. Record layout of the DBD control information records file for SORT=YES

Field name	Assembler definition	Description
PFXLL	H	RDW LL
PFXZZ	H	RDW ZZ
PFXFLG	C	C'S' (blank)
PFXFLG1	X	Reserved
PFXOFFST	H	Offset of segment table entries
PFXRKEYL	H	X'0000'
PFXRKEY	X (Variable)	Root key (Null value)
PFXSCSQ		SCSQ table. The number of entries is one less than the number of hierarchical levels in DMB. There is no entry for the Root segment.
PFXSQSC	X	X'00'
PFXSQSCT	XL3	X'000000'
SRTDNOSE	H	Number of segment table entries
SRTSETE		Segment table entry
SRTSESC	H	Segment code
SRTSEHI	H	Segment hierarchical level
SRTSEPA	H	Parent segment code
SRTSEGN	CL8	Segment name

The following table shows the record layout of the segment data record for SORT=YES, which is a product-sensitive programming interface.

Table 147. Record layout of the segment data records file for SORT=YES

Field name	Assembler definition	Description
PFXLL	H	RDW LL
PFXZZ	H	RDW ZZ
PFXFLG	C	C'S' (blank)
PFXFLG1	X	Bit-value Flag <ul style="list-style-type: none"> • X'04' Fixed segment • X'02' Compressed segment. This flag is set on when the COMPRTN is specified for the segment in the DBD definition and either of the following conditions is met: <ul style="list-style-type: none"> – For the ODE utility of FPO, EXPAND=NO is specified on the FPXCTL statement. – For the Extract function of FPA, COMPRESS=YES is specified on the EXTRACT command.
PFXOFFST	H	Offset of segment code entry

Table 147. Record layout of the segment data records file for SORT=YES (continued)

Field name	Assembler definition	Description
PFXRKEYL	H	Root key length
PFXRKEY	X (Variable)	Root key
PFXSCSQ		SCSQ table. The number of entries is one less than the number of hierarchical levels in DMB. There is no entry for the root segment.
PFXSQSC	X	Segment code in SCSQ table
PFXSQSCT	XL3	Sequential counter of the occurrence of this segment.
PFXSEGC	H	Segment code
STDCCKL	H	Concatenated key length (does not include the root key length)
STDCCK	X (Variable)	Concatenated key (does not include the root key)
STDDATAL	H	Extracted segment data length
STDDATA	X (Variable)	Extracted segment data



DBT Unload/Reload format

For the DBT Unload/Reload format, see [“Unloaded segment records layout”](#) on page 1177.

HD Unload format

For the HD Unload format, see the DFSURGUF macro that is provided by IMS.

User format

The following figure shows the layout of the user format extracted file or the user format unloaded file.

The user-defined header and the trailer records are defined by the USER_RECORD subcommand.

User-defined prefix and suffix are defined as follows:

- For the FPA Extract function and the FPA Unload function, they are defined by the SEGREC_PREFIX keyword and the SEGREC_SUFFIX keyword on the EXTRACT command or the UNLOAD command.
- For the ODE utility of FPO, they are defined by the SEGREC_PREFIX keyword and the SEGREC_SUFFIX keyword on the FPXCTL statement.

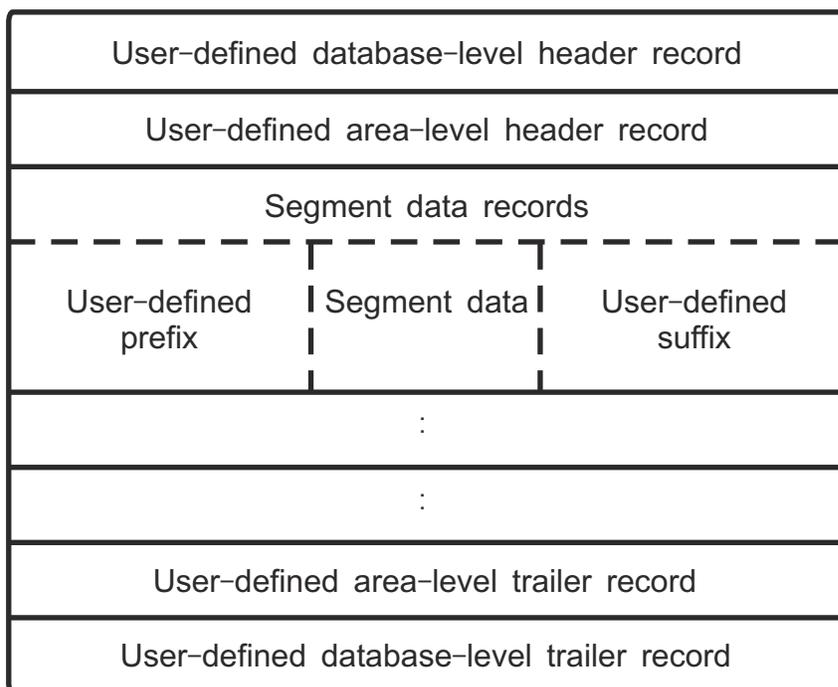


Figure 463. Layout of the user format extracted file

Related reference

BREAK keyword(See this topic for the BREAK keyword for FPA) Use the BREAK keyword to specify the level and the timing for the trigger event for user-defined header and trailer of an extract or an unloaded file.

BREAK keyword(See this topic for the BREAK keyword for FPO) The BREAK keyword specifies the level and the timing for the trigger event for user-defined header and trailer of an extract file.

FIELDS keyword(See this topic for the FIELDS keyword for FPA) Use the FIELDS keyword to specify the content of the segment data records, user-defined header records, or user-defined trailer records that are written to the output file.

FIELDS keyword (for USER_RECORD)(See this topic for the FIELDS keyword for FPO) The FIELDS keyword specifies the user-defined header records, or the user-defined trailer records that are written to the output file.

SEGREC_PREFIX keyword(See this topic for the SEGREC_PREFIX keyword for FPA) Use the SEGREC_PREFIX keyword to specify the contents of the fields within the prefix portion of the segment records, which are to be written to an extract or an unload file.

SEGREC_PREFIX keyword(See this topic for the SEGREC_PREFIX keyword for FPO) The SEGREC_PREFIX keyword specifies the contents of the fields within the prefix portion of the segment records, which are to be written to an extract file.

SEGREC_SUFFIX keyword(See this topic for the SEGREC_SUFFIX keyword for FPA) Use the SEGREC_SUFFIX keyword to specify the contents of the fields within the suffix portion of the segment records, which are written to an extract or an unload file.

SEGREC_SUFFIX keyword(See this topic for the SEGREC_SUFFIX keyword for FPO) The SEGREC_SUFFIX keyword specifies the contents of the fields within the suffix portion of the segment records, which are written to an extract file.

OUTFILE records layout

OUTFILE DD data set records are created by the DEDB/HD Unload Conversion utility.

This topic contains product sensitive programming interface information.



The OUTFILE record has a specific record layout.

The following table shows the layout for the OUTFILE file records, which is a product-sensitive programming interface.

Table 148. Record layout of the OUTFILE DD data set records file

Field description	Assembler definition	Format
LL of variable length record	XL2	Unsigned binary
ZZ of variable length record	XL2	Unsigned binary
Segment name	CL8	Characters First record contains blanks in this field
Filler	XL1	
Segment code	CL3	Numeric characters First record contains C'00' in this field
Filler	XL1	
Segment data	OXL32743	The first two bytes of the data contain the segment length (LL)



UOW group definition record layout

UOW group definition records are used by the FPA Analyze function and the FPB DEDB Pointer Checker utility.

The data set that contains the UOW group definition records must have the following attributes: DSORG=PS, RECFM=FB, and LRECL=80. If the BLKSIZE is specified, it must be a multiple of 80. The data set can contain records for more than one area. The UOW ranges can overlap and can be sorted in reverse order, but the records must be sorted by the area number.

The following two types of records are supported:

ARDIGIT=3 (default)

The area number is specified by a 3-digit decimal number.

ARDIGIT=5

The area number is specified by a 5-digit decimal number.

You must use one of these record types for all records in the data set. The type of the record must match the specification of the ARDIGIT parameter for the utility that uses the data set. The number of digits of the area number field ("AREA# IN DBD") in the record must have the same number of digits as the ARDIGIT value.

You can specify the ARDIGIT parameter in the following data sets or control statements:

- The PARMCTL data set for the FPB DEDB Pointer Checker utility
- The HFPAPARM data set for the FPA Analyze function and the Analyze subfunction for the Reload, Change, and Reorg functions
- The ANALYZE statement for FPO Online Pointer Checker utility
- The PTRCHK statement for FPO Online Expert Reorganization utility

The following table summarizes the UOW group definition record layout when the ARDIGIT parameter is 3.

Table 149. UOW group definition record layout when ARDIGIT is 3

Field description	COBOL usage
AREA number defined in DBD	9(3)
Start UOW number	9(7)
Stop UOW number	9(7)
Filler	X(3)
Description of the UOW group	X(16)
Filler	X(44)

The following table summarizes the UOW group definition record layout when the ARDIGIT parameter is 5.

Table 150. UOW group definition record layout when ARDIGIT is 5

Field description	COBOL usage
AREA number defined in DBD	9(5)
Start UOW number	9(7)
Stop UOW number	9(7)
Filler	X(3)
Description of the UOW group	X(16)
Filler	X(42)

Area number defined in DBD

Is the area number of the DEDB database that contains the target UOW range. This field contains a right-aligned 3- or 5-digit number, with leading zeros if necessary.

Start UOW number

Specifies the UOW number that is used to start processing. This field contains a right-aligned 7-digit number, with leading zeros if necessary.

Stop UOW number

Specifies the UOW number that is used to stop processing. This field contains a right-aligned 7-digit number, with leading zeros if necessary.

Description of the UOW group

Describes the UOW group that is defined by the start UOW number and the stop UOW number in the record. This 16-byte character string is recorded in the History2 file record or the sensor data record that corresponds to the UOW group.

The following examples show the UOW group definition records:

UOW group definition records when ARDIGIT is 3:

```
001000000000000001 UOW 0-1
001000000200000003 UOW 2-3
```

UOW group definition records when ARDIGIT is 5:

```
00001000000000000001 UOW 0-1
00001000000200000003 UOW 2-3
```

Build pointer segment records layout

Build pointer segment records are created by the FPA Build Index function.

This topic contains product sensitive programming interface information.

Build pointer segment records consists of the record type, a header record, and a pointer segment record. Each record consists of two parts; prefix part and data part.

You can use the HFPZBPR macro to map the build pointer segment records.

The following table shows the record layout, which is a product-sensitive programming interface.

PSPI

Table 151. Record layout of build pointer segment records (prefix part)

Field name	Assembler definition	Description
BPRTYPE	XL1	Record type X'00': Header record X'10': Pointer segment record
	XL3	(reserved)
BPRDATA	XLnn	Data part

Table 152. Record layout of build pointer segment records (data part in header record)

Field name	Assembler definition	Description
BPRDDN	CL8	DD name of the pointer segment record data set
BPRIDBD	CL8	Secondary index DBD name

Table 153. Record layout of build pointer segment records (data part in pointer segment record)

Field name	Assembler definition	Description
BPRPSEG	XLnn	Index pointer segment data

PSPI

Resync pointer segment records and duplicate pointer segment records layout

Resync pointer segment records are created by the FPA Resync function and duplicate pointer segment records are created by the FPA Build Index function.

This topic contains product sensitive programming interface information.

Resync pointer segment records consists of the record type, a header record, a pointer segment record, and a trailer record. Duplicate pointer segment records consists of the record type, a header record, and a pointer segment record. Each record consists of two parts; prefix part and data part.

You can use the HFPZRPR macro to map the resync pointer segment records and duplicate pointer segment records.

The following table shows the record layout, which is a product-sensitive programming interface.

Table 154. Record layout of pointer segment records (prefix part)

Field name	Assembler definition	Description
RPRRDW	H H	RDW "LL" field RDW "ZZ" field
RPRID	XL1 XL3	Record ID X'00': Header record X'01': Processed area record X'10': Pointer segment record X'F0': Trailer record (reserved)
RPRDATAP	XLnn	Data part

Table 155. Record layout of pointer segment records (data part in the header record)

Field name	Assembler definition	Description
RPRHDDDN	CL8	DD name of the pointer segment record data set
RPRHDDBDN	CL8	Secondary index DBD name
RPRHDRTYP	CL1 XL1	Record Type C'R': Resync record C'D': Duplicate record (reserved)
RPRHDPSLN	H	Length of the index pointer segment
RPRHDDATE	PL4	Date that the pointer segment records were generated Format is P'YYYYDDDF'
RPRHDTIME	PL4	Time that the pointer segment records were generated Format is P'OHHMMSSF'

Table 156. Record layout of processed area records (data part in the processed area record)

Field name	Assembler definition	Description
RPRARNO	H	Processed area number

Table 157. Record layout of pointer segment records (data part in the pointer segment record)

Field name	Assembler definition	Description
RPRPSA#	H	Area number that the index source segment resides

Table 157. Record layout of pointer segment records (data part in the pointer segment record) (continued)

Field name	Assembler definition	Description
RPRPSTYPE	CL1	How the index pointer segment is reflected to the secondary index C'D': The pointer segment will be deleted C'I': The pointer segment will be inserted C'M': The pointer segment is duplicated C'U': The pointer segment is unverified
RPRPSREC	XLnn	Index pointer segment data

Table 158. Record layout of pointer segment records (data part in the trailer record)

Field name	Assembler definition	Description
RPRTLDEL#	F	Number of pointer segment records to be deleted
RPRTLIS#	F	Number of pointer segment records to be inserted
RPRTLUNV#	F	Number of unverified pointer segment records

PSPI

Chapter 46. Exit routine option and its interface

The programs for DEDB Unload and DEDB Reload of FPA and FPB, DEDB Change of FPA, the DEDB Reload Segment Data Set Create utility, and the DEDB Unloaded Segment Data Set Retrieve utility support a user exit routine.

Topics:

- [“Exit routine option overview” on page 1197](#)
- [“Exit routine interface” on page 1197](#)
- [“Exit routine FABCRPCX” on page 1199](#)

Exit routine option overview

For each segment processed, the exit routine specified by the EXITRTN=, IEXITRTN=, and OEXITRTN= keyword parameters will be invoked.

The exit routine is invoked as follows:

- The programs for DEDB Unload of FPA and FPB, and DEDB Change of FPA will invoke the exit routine with an expand function.
- The programs for DEDB Reload of FPA and FPB, and DEDB Change of FPA will invoke the exit routine with the specified function.
- DEDB Reload Segment Data Set Create utility will invoke the exit routine with a compress function.
- DEDB Unloaded Segment Data Set Retrieve utility will invoke the exit routine with an expand function.

If a segment is compressed by an IMS segment edit/compression routine, it will be expanded by the edit/compression routine before calling a user exit routine. After returning from the user exit routine, the segment will be compressed by the edit/compression routine again.

Note: All exit routines should be reentrant so that they can be invoked by multiple subtasks. If the routine is non-re-entrant, then processing of the user exit routine is serialized between multiple subtasks, and will lower the performance in some cases. Exit routines must be capable of AMODE=31.

Exit routine interface

When the exit routine is invoked, register 1 contains the address of a 10-address parameter list.

Each parameter in the list is as follows:

Parameter 1

Address of the 4-byte function code for this call. Valid codes are:

EXP

segment expansion request

CMP

segment compression request

END

clean up request

The function codes are left-adjusted in the field.

Parameter 2

Address of an 18-byte area containing the following area information:

DBD name

8 bytes

area name

8 bytes

area number

2-byte binary expression

Note: When the exit routine is called from the programs for DEDB Unload of FPA and FPB, and DEDB Change of FPA, and when a DEDB record is randomized by using the new definition in the new ACB library member, the area name and the area number in Parameter 2 are the result of the randomization.

Parameter 3

Address of a 10-byte area containing the following segment information:

segment name

8 bytes

segment code

2-byte binary expression

Parameter 4

Address of a 2-byte area where the exit routine will return the status code. The status code is as follows:

blank

The exit routine successfully compressed or expanded the segment. The modified segment has been moved to the area addressed by Parameter 7.

NO

The segment does not require expansion or compression. The area addressed by Parameter 6 and the area addressed by Parameter 7 has not been changed.

E1

A warning condition was detected by the exit routine. The area addressed by Parameter 6 and the area addressed by Parameter 7 has not been changed. The message area addressed by Parameter 8 will be printed by the calling program to the message output data set. The calling program will continue execution.

E2

An error condition was detected by the exit routine. The area addressed by Parameter 6 and the area addressed by Parameter 7 has not been changed. The message area addressed by Parameter 8 will be printed by the calling program to the message output data set. The calling program will issue an error message and abend.

T1

This status code is only for the programs for DEDB Unload of FPA and FPB, and DEDB Change of FPA. The exit routine does not write the segment to the output data set and does not proceed down the hierarchical path (all dependent segments of the segment being deleted are ignored, that is, the exit routine is not invoked for these segments and does not write to the output data set).

T2

This status code is only for the programs for DEDB Unload of FPA and FPB, and DEDB Change of FPA. The exit routine does not write the segment to the output data set. It is the exit routine's responsibility to also return a T2 status code for any dependent segments of the segment being deleted. Failure to T2 a dependent segment will cause the unload program to terminate with a message.

Parameter 5

Address of an 8-byte area (two full-words) that are as follows:

1st full-word

Set to binary zeros and left unmodified by the calling program. The exit routine will use it as an anchor to anchor all work areas required by the exit routine. The exit routine can recognize that initialization must be performed by the binary zero field on the first call. At the END call, the exit routine should clean up the anchor.

2nd full-word

Contains an address that points to 28 full-words of storage set to binary zeros and left unmodified by the calling Program. The exit routine will use them as an anchor point for common storage in a multitasking environment.

Parameter 6

Address of the area that contains the DEDB segment (LL + segment data) that is to be compressed or expanded.

Parameter 7

Address of the area into which the exit routine is to move the compressed or expanded segment. The exit routine can modify this field only when a blank status code is returned. The field is ten bytes larger than the maximum segment size, and the exit routine should not move segment data larger than this size.

Parameter 8

Address of a 128-byte message area. The exit routine should set a message in this area whenever an E1 or E2 status code is being returned. A message can also be returned on an END call. It is the exit routine's responsibility to manage the contents of the message area, although it will be initialized to spaces by the caller. The caller will issue the message contained in this area to the message output data set whenever the status code is E1 or E2. The message will also be issued by the caller on an END call. The caller always formats the message into two print lines. The first printed line is 80 bytes and the second line is 48 bytes. There is no need to provide carriage control as part of the message.

Parameter 9

Address of a 1-byte area indicating whether a segment is being put into the independent overflow area (IF). The indicator is as follows:

I

Put the segments into IOVF.

blank

Put the segments where they would normally go.

The programs for DEDB Unload and DEDB Reload of FPA and FPB, DEDB Change of FPA, the DEDB Reload Segment Data Set Create utility, and the DEDB Unloaded Segment Data Set Retrieve utility do not support this function but keeps this parameter as is for the existing user exit routine interface compatibility.

Parameter 10

Address of the DEDB exit/compression routine if compression is used.

Exit routine FABCRPCX

Exit routine FABCRPCX can be used within the FPB Unload function.

When the exit routine FABCRPCX is specified in FPB Unload Reorg mode, FPB Unload passes the information to invoke the randomizer to the exit routine. FABCRPCX can validate the area number and RAP RBA of the segment that is stored in the area by comparing them with those returned by the randomizer. FABCRPCX does not support the XCI randomizer interface run in batch.

When the exit routine FABCRPCX is invoked in FPB Unload, register 0 contains the parameter list that has 9 full-word parameters in addition to register 1, which contains the parameter list that has 10 full-word parameters. Each parameter and value in the list is as follows:

Parameter 1

The address of 8-byte randomizer module.

Parameter 2

The entry point of randomizing module that FPB loaded from RMODLIB or JOBLIB/STEPLIB.

Parameter 3

The number of entries in the MRMB (total number of AREAs in the DEDB).

Parameter 4

Address of the first MRMB the routine uses.

Parameter 5

The size of an entry in the MRMB.

Parameter 6

Total number of RAPs in the DEDB.

Parameter 7

The address of the 8-byte IMSID.

Parameter 8

The value of RAP RBA.

Parameter 9

The address of the area that contains the root key (LL + root key value).

Restriction: The exit routine FABCRPCX must be specified in FPB Unload Reorg mode.

Chapter 47. Segment edit/compression facility support

This topic helps you produce the unloaded data set records with expanded segment data from compressed segments. It contains the functional capabilities of this facility.

This topic documents Product-Sensitive programming interfaces and associated guidance information provided by the programs for DEDB Unload and DEDB Reload of FPA and FPB, and DEDB Change of FPA.

The Product-Sensitive programming interfaces are provided to allow the customer installation to perform tasks such as tailoring, monitoring, modification, or diagnosis of this IBM product. The use of such interfaces creates dependencies on the detailed design or implementation of the IBM product. The Product-Sensitive interfaces should be used only for these specialized purposes. Because of their dependencies on detailed design and implementation, it is to be expected that programs written to such interfaces may need to be changed in order to run with new product releases or versions, or as a result of service.

Action of the programs for DEDB Unload, DEDB Reload, and DEDB Change

The following table shows the action of the programs for DEDB Unload and DEDB Reload of FPA and FPB, and DEDB Change of FPA.

Table 159. Action of DEDB Unload, Reload, and Change programs

Definition/Specification			System action		
COMPRESS=	EXIT of original ACB library	EXIT of new ACB library	EXIT call by Unload program		EXIT call by Reload program
			Original ACB library	New ACB library	
YES	No	No	No	No	No
YES	No	Exit A	No	Compress	No
YES	Exit A	No	Expand	No	No
YES	Exit A	Exit A	No	No	No
YES	Exit A	Exit B	Expand	Compress	No
NO	No	No	No	No	No
NO	No	Exit A	No	No	Compress
NO	Exit A	No	Expand	No	No
NO	Exit A	Exit A	Expand	No	Compress
NO	Exit A	Exit B	Expand	No	Compress

If a new ACB library not specified, the definitions of the original ACB library are applied to the new ACB library.

If the EXITRTN keyword is specified for the DEDB Unload, the DEDB Reload, or the DEDB Change program, some of these actions are not applicable. These programs always pass uncompressed segments to the user exit routine. Therefore, to process a compressed segment, the edit/compression routine is called with the expand function first, and then the user exit is called. When the user exit returns, the edit/compression routine is called with the compression function.

Interfaces

This topic describes the segment edit/compression routine interface environment created by the programs for DEDB Unload and DEDB Reload of FPA and FPB, and DEDB Change of FPA. The segment edit/compression routine interface environment created by these programs conforms to the published interface as defined in *IMS Exit Routines* with one exception; when the segment edit/compression routine is invoked by IMS, register 1 contains the address of EPST. However, when these programs call a segment edit/compression routine, register 1 is set to 0 indicating that it was not called by IMS.

The segment edit/compression routine is called by the DEDB Unload, the DEDB Reload, and the DEDB Change programs in four ways:

- To compress a segment (compress interface is used).
- To expand a segment (expand interface is used).
- When a segment edit/compression routine is loaded by these FPA and FPB programs (open interface is used).
- At the end of these FPA and FPB programs (close interface is used).

The routine is called with open/close interface only if INIT is specified for the COMPRTN= parameter on the SEGM statement for DBDGEN.

Database definition record

The database definition record in DURDBDFN data set is created by DEDB Unload program of FPB (FABCUR1) and is used by DEDB Reload program of FPB (FABCUR3). This record is the complete set of the database description table (mapped by FABCMDDT macro) created by FABDGDD routine.

- Two flags are introduced in flag byte 'DDTFLG1' for global information in DDT section of the table.
 - DDTFCMY (X'02')
 - ON specifies that COMPRESS=YES is specified.
 - OFF specifies that COMPRESS=NO is specified.

This flag is set on by DEDB Unload program when COMPRESS=YES parameter is specified on DATABASE control statement.

- DDTFCMP (X'01')
- ON specifies that segment edit/compression routine is defined in some segment.
- OFF specifies that no segment edit/compression routine is defined in any segment.

This flag is set on by FABDGDD routine when the routine creates the database description table based on DMB in ACB library. It specifies that at least one segment, with a segment edit/compression routine definition, exists in DMB.

- Two flags are introduced in attribute flag byte 'SDTFLG1' in SDT section of the table.
 - SDTFCMP (X'01')

- ON specifies that segment edit/compression routine is defined for the segment.
- OFF specifies that no segment edit/compression routine is defined for the segment.

This flag is set on by FABDGDD routine when the routine creates the database description table based on DMB in ACB library. It specifies that a CMPC control block, with the segment edit/compression routine definition for the segment, exists.

- SDTFCAL (X'02')

This is a work flag used by FABCUR1.

- CMPC list is created following the last SDT (Segment Description Table). Total number of the entries of the list (12 bytes) is the same as the number of the segments defined for the database. Each entry

consists of, the CMPC offset from the top of DDT (full-word), the minimum CMPC length defined by DBDGEN (half-word), and a reserved field (six bytes). Zero for the offset specifies there is no segment edit/compression routine, and therefore no CMPC is defined for the corresponding segment.

- The existence of CMPC list can be determined by the flag bit DDTFCMP of flag byte DDTFLG1.
- The CMPC control blocks themselves, in which segment edit/compression routines are defined by DBDGEN, are created following the CMPC list. CMPC control block in the database description table is the same as that in DMB except for the CMPCPEP field which contains the actual segment edit/compression routine entry point.

Unloaded segment record

A flag is introduced in the flag byte 'USRPFLG1' (processing flag #1) in the unloaded segment record which is mapped by FABCUR macro of FPB or HFPZUSR macro of FPA.

```
USRPCOMP (X' 02')  
  ON specifies that the record contains compressed segment  
  OFF specifies that the record contains expanded segment
```

This flag is set 'ON' by the programs for FPA DEDB Unload and FPB DEDB Unload. The programs for FPA DEDB Reload and FPB DEDB Reload check this flag and the definition of the segment edit/compression routine in the database definition record. If the two values show inconsistency, the programs for FPA DEDB Reload and FPB DEDB Reload terminate, issuing a message.

Chapter 48. Determining how to process SDEP segments

IMS HP Fast Path Utilities provides several utilities to process SDEP segments. You can use these utilities to analyze and maintain SDEP segments.

The sequential dependent part holds sequential dependent segments from roots in all UOWs in the area. Sequential dependent segments are stored in chronological order without regard to the root or UOW that contains the root. When the sequential dependent part is full, it is reused from the beginning. However, before the sequential dependent part can be reused, you must delete a contiguous portion, or all the sequential dependent segments in that part.

Subsections:

- [“Analyzing an SDEP part” on page 1205](#)
- [“Maintaining an SDEP part” on page 1205](#)
- [“Retrieving SDEP segments” on page 1206](#)

Analyzing an SDEP part

IMS HP Fast Path Utilities provides the functions for DEDB integrity verification and DEDB analysis and report.

- DEDB integrity verification

You can select the method for running the integrity verification for the SDEP part.

The methods are; Cross-reference validation of pointers, Checksum validation of pointers, and Not running the SDEP processing. IMS HP Fast Path Utilities provides options to specify the severity level of the messages that are issued when detecting orphan SDEP segments or SDEP pointers that point to the IOVF portion. IMS HP Fast Path Utilities provides integrity verification reports.

- DEDB analysis and report

The complete analysis reports are provided to effectively manage the space utilization for SDEP part. For the details, see [Chapter 43, “DEDB monitoring and tuning guide,” on page 1111](#).

The SDEP Space Utilization utility generates reports that show trends in SDEP space utilization. You can use the information provided to schedule database expansion and to forecast future DASD requirements, thus utilize the SDEP.

Maintaining an SDEP part

IMS HP Fast Path Utilities provides the following functions to maintain an SDEP part:

- FPA Unload and Reload, Change, FPB Unload and Reload

The Unload function provides the following three options for maintaining SDEP segments.

- LOGICAL option

SDEP segments are unloaded in logical order. The valid SDEP segments are extracted during the unload process. They will be inserted into the new database during the reload process and will retain their logical order (entry sequence) within each database record. The SDEP marker will be lost at unload. When reducing the size of SDEP part, you must make sure that the SDEP size after being reduced has enough space to restore all SDEP CIs that are between SDEP logical begin and logical end.

In case of FPA, SORT=YES is required in the Unload or the Reload function.

In case of FPB, unloaded segment records must be sorted prior to running a reload process.

– PHYSICAL option

SDEP segments are unloaded in physical order. All SDEP data between the SDEP logical begin and the logical end are extracted during unload. The SDEP data will be reloaded into the SDEP part of the area created during reload. The reloaded SDEP data will be in the same physical order between the SDEP logical begin and the logical end, as prior to the unload. When SDEP=PHYSICAL is specified, the performance of the unload processing will be faster than that of SDEP=LOGICAL.

To reduce the size of SDEP part, in FPA, use the RELOCATE_SDEP option for the Unload and the Change functions. In FPB, use the SDEPRELOCATE option for the DEDB Reload utility. When reducing the size of SDEP part with the FPA RELOCATE_SDEP option or the FPB SDEPRELOCATE option, ensure that the SDEP size after being reduced has enough space to restore all SDEP CIs that are between SDEP logical begin and logical end.

The following restrictions apply when the SDEP=PHYSICAL option is specified:

- The SDEP marker at unload can be kept to reload.
 - Unload segment records of the SDEP CIs are unloaded as 512-byte blocks of CI data, not as discrete segments.
 - The number of 512-byte blocks is shown in the reports generated by Unload and Reload. In these reports, text (PHYSICAL) is printed after the number to imply SDEP=PHYSICAL.
 - Compressed segments will not be expanded.
 - If the FPA Unload function, the Change function, or the FPB Unload utility is accompanied by a DBD change, the following restrictions apply:
 - The characteristic of the randomizer defined in the new DBD must be area specific because the root segments must be stored in the original area. In FPB, specification of RMODTYPE=S is also required.
 - The CI size of the ADS to be reloaded cannot be changed.
 - If the FPA Unload function, the Change function, or the FPB Unload utility is accompanied by a DBD change or the SDEPRELOCATE=YES option is selected for the FPB Reload utility, absolute RBA value at unload to identify the SDEP marker might no longer be effective but root segment and the field value of the SDEP marker at unload will remain effective to identify the marker SDEP for SDEP Scan/Delete Utilities.
- NO option
- SDEP segments are not unloaded.
- FPO OAE

OAE lets you extend the size of the DEDB area dynamically by enlarging the sequential dependent (SDEP) portion, the independent overflow (IOVF) portion of the DEDB area, or both. Because this is accomplished without taking the DEDB area offline, the process is not disruptive to applications accessing the area. For the details, see [Chapter 22, “Extending a DEDB online with FPO,” on page 747.](#)

Retrieving SDEP segments

Most of the time, SDEP segments are retrieved all at once, using the DEDB Sequential Dependent Scan utility. If you later must relate SDEP segments to their roots, you must plan for root identification as part of the SDEP segment data.

The FPA Extract function or the FPO ODE function can be used as an alternative to writing your own application for extracting DEDB data and adding supplementary information to the data. You can add various supplementary information to DEDB data by using the control statement keywords provided by the functions.

FPO ODE also shows detailed information of the SDEP segments, such as whether they are valid segments, logically deleted segments, in-doubt segments, aborted segments, or SDEP control segments within Logical begin and Logical end.

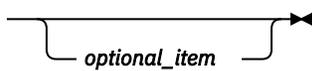
Chapter 49. How to read syntax diagrams

The following rules apply to the syntax diagrams that are used in this information:

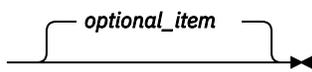
- Read the syntax diagrams from left to right, from top to bottom, following the path of the line. The following conventions are used:
 - The >>--- symbol indicates the beginning of a syntax diagram.
 - The ---> symbol indicates that the syntax diagram is continued on the next line.
 - The >--- symbol indicates that a syntax diagram is continued from the previous line.
 - The --->< symbol indicates the end of a syntax diagram.
- Required items appear on the horizontal line (the main path).

▶▶ *required_item* ▶▶

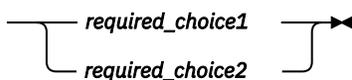
- Optional items appear below the main path.

▶▶ *required_item* ▶▶

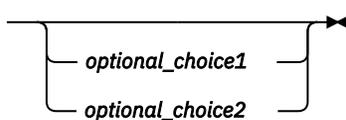
If an optional item appears above the main path, that item has no effect on the execution of the syntax element and is used only for readability.

▶▶ *required_item* ▶▶

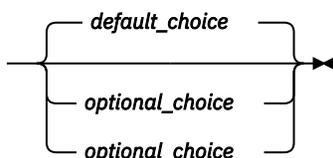
- If you can choose from two or more items, they appear vertically, in a stack.
If you *must* choose one of the items, one item of the stack appears on the main path.

▶▶ *required_item* ▶▶

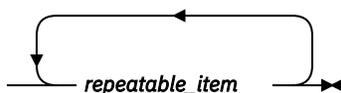
If choosing one of the items is optional, the entire stack appears below the main path.

▶▶ *required_item* ▶▶

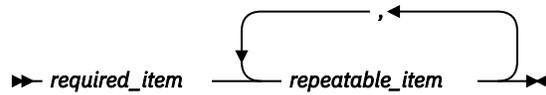
If one of the items is the default, it appears above the main path, and the remaining choices are shown below.

▶▶ *required_item* ▶▶

- An arrow returning to the left, above the main line, indicates an item that can be repeated.

▶▶ *required_item* ▶▶

If the repeat arrow contains a comma, you must separate repeated items with a comma.



A repeat arrow above a stack indicates that you can repeat the items in the stack.

- Keywords, and their minimum abbreviations if applicable, appear in uppercase. They must be spelled exactly as shown. Variables appear in all lowercase italic letters (for example, *column-name*). They represent user-supplied names or values.
- Separate keywords and parameters by at least one space if no intervening punctuation is shown in the diagram.
- Enter punctuation marks, parentheses, arithmetic operators, and other symbols exactly as shown in the diagram.
- Footnotes are shown by a number in parentheses; for example, (1).

Part 7. Troubleshooting

These topics provide technical references to help you troubleshoot and diagnose IMS HP Fast Path Utilities problems.

Topics:

- [Chapter 50, “Messages and codes of IMS HP Fast Path Utilities,” on page 1211](#)
- [Chapter 51, “Diagnostics aid,” on page 1433](#)

Chapter 50. Messages and codes of IMS HP Fast Path Utilities

These reference topics provide you detailed information of IMS HP Fast Path Utilities messages.

Topics:

- [“Return codes” on page 1211](#)
- [“Abend codes” on page 1212](#)
- [“Messages” on page 1218](#)
- [“Gathering diagnostic information” on page 1431](#)

Return codes

These reference topics describe the return codes that are issued by the tools of IMS HP Fast Path Utilities.

Return codes of FPA

This topic provides detailed information about the return codes that are issued by the processes of FPA.

The return codes from FPA are:

- 0** Job successfully ended.
- 4** Job ended with a warning message.
- 8** Job ended with an error message.
- 12** Job ended with an abend message.

Return codes of FPO

This topic provides detailed information about the return codes that are issued by the utilities of FPO. OPC, ODE, and OER support a return code user exit routine that can change the return codes.

The return codes from FPO are:

- 0** Job successfully ended, or job ended with a warning message that was issued during parameter analysis.
- 4** Job ended with a warning message, or OPC job ended with a pointer error message when pointer errors were detected during the fast scan of database integrity.
- 8** Job ended with an error message, or OPC job ended with a pointer error message when pointer errors were detected during the cross-reference pointer checking.
- 12** Job of ODE run-level control using SYSIN and FPXCTL DD ended with an error message that was issued during parameter analysis.

Related reference

[Return code user exit routine \(FPXRCEX0\)](#)

IMS Fast Path Online Tools supports a return code user exit routine that can be used to change the return codes of OPC, ODE, or OER.

Return codes of OSM

This topic provides detailed information about the return codes that are issued by the Online Space Management utility.

The return codes from the Online Space Management utility are:

- 0** Job successfully ended.
- 4** Job ended with one or more warning messages.
- 8** Job ended with one or more error messages.
- 12** Job ended abnormally. An abend code is issued.

Return codes of diagnostics aid (FABADIAG)

This topic provides detailed information about the return codes that are issued by the diagnostics aid program (FABADIAG).

FABADIAG

FABADIAG returns one of the following return codes:

- 0** The program has been completed successfully.
- 4** Warning messages were issued, but the requested operation was completed.
- 8** Error messages were issued, but the requested operation was completed.

Abend codes

These reference topics describe the abend codes that are issued by the tools of IMS HP Fast Path Utilities.

Abend codes of FPA

This topic provides detailed information about the abend codes that are issued by FPA.

Every 3xxx user abend code has an associated HFPC3xxxE message. To determine the meaning of the 3xxx user abend, read the description of the associated HFPC3xxxE message.

Abend codes of FPO

This topic provides detailed information about the user abend codes that are issued by FPO.

For each abend code, the following accompanying information is provided where applicable:

Explanation

This explains what the abend code means, what caused the abend to be issued, and what its variable entry fields are (if any).

System action

This explains what the system will do next.

User response

This describes whether a response is necessary, what the appropriate response should be, and what the resulting effect is on the system or program.

The following explanations pertain to FPO abend codes:

3500**Explanation**

This abend code is issued by FPO. An unexpected IMS status code was returned during a DL/I call.

System action

The IMS IFP utility ends abnormally. This abend code is accompanied by message FPX0011E.

User response

Examine message FPX0011E for further information. This message is described in [“FPX messages” on page 1223](#). Then see *IMS Messages and Codes* for the appropriate response to this abend code. Then resubmit the job.

3501**Explanation**

This abend code is issued by FPO. During buffer processing, an invalid buffer was found.

System action

The IMS IFP utility ends abnormally.

User response

This problem might be an ODE internal error. Contact IBM Software Support.

3502**Explanation**

This abend code is issued by FPO. During processing of the root addressable area, a loop in an independent twin chain was encountered. The segment RBA must be either within the current UOW or within the IOVF part of the area.

System action

The IMS IFP utility ends abnormally.

User response

This situation could indicate a database pointer problem. It could also be the result of an ODE internal

error. Check pointers in the area. If no errors are found, contact IBM Software Support.

3503**Explanation**

This abend code is issued by FPO. The segment code at the RBA being evaluated did not match the expected segment code, or the CI prefix failed validation.

System action

The IMS IFP utility ends abnormally.

User response

This situation could indicate a broken database. It could also be caused by an ODE internal error. Pointer check the area. If no errors are found, contact IBM Software Support.

3504**Explanation**

This abend code is issued by FPO. An abend or unexpected end of the extraction subtask occurred. This abend occurs when the main task is waiting for the subtask and must be ended. The actual problem is in the abend or premature termination of the extraction subtask. This is an internal error of ODE or OPC.

System action

The IMS IFP utility ends abnormally.

User response

Contact IBM Software Support.

3505**Explanation**

This abend code is issued by FPO. CSVQUERY service for module FPXMBLK0 or FPXBMBL0 failed.

System action

The IMS IFP utility ends abnormally.

User response

This is an FPO internal error. Contact IBM Software Support.

3506

Explanation

This abend code is issued by FPO. CSVQUERY service for module FPXTRCT0 failed.

System action

The IMS IFP utility ends abnormally.

User response

This is an ODE internal error. Contact IBM Software Support.

3507

Explanation

This abend code is issued by FPO. I/O errors were encountered reading a data set.

System action

The IMS IFP utility ends abnormally. This abend code is accompanied by messages FPX0015E, FPX0018E, and FPX0130E.

User response

Determine the nature of the I/O error, and correct the problem.

3508

Explanation

This abend code is issued by FPO. Module FPXEWMV0 was passed a zero data address or a zero data length, or FPXEWMV0 was unable to locate the next extract work buffer.

System action

The IMS IFP utility ends abnormally.

User response

This is an ODE internal error. Contact IBM Software Support.

3509

Explanation

This abend code is issued by FPO. A data management function failed. The relevant DCBs are snapped prior to the abend.

- Reason code 1 = RDJFCB failure
- Reason code 2 = OPEN failure
- Reason code 3 = CLOSE failure
- Reason code 4 = DEVTYPE failure
- Reason code 5 = DYNALLOC failure

System action

Processing ends with abend code 3509.

User response

This is an OPC internal error. Contact IBM Software Support.

3510

Explanation

This abend code is issued by FPO. An error was encountered during pointer checking. The program ended with an abend because ERRORACT was given a value of ABEND, and the MAXERROR value was reached.

System action

The IMS IFP utility ends abnormally.

User response

Determine the reason for the number of pointer problems.

If these are true pointer problems, increase the MAXERROR threshold.

3511

Explanation

This abend code is issued by FPO. An unrecognized record was encountered during processing.

System action

The IMS IFP utility ends abnormally. This abend code is accompanied by message FPX0059E.

User response

This is an ODE internal error. Contact IBM Software Support.

3512**Explanation**

This abend code is issued by FPO. A compressed segment was referenced; however, the compression exit cannot be loaded. This is a problem because a compressed segment was encountered, and the segment needs to be expanded for evaluation.

System action

The IMS IFP utility ends abnormally.

User response

Make sure that the compression exit is available to the IFP utility in the JOBLIB or STEPLIB DD statement concatenation.

3513**Explanation**

This abend code is issued by FPO. Fatal DBRC errors were encountered registering an image copy with DBRC, as requested by the DBRC=Y keyword. This abend is accompanied by DBRC messages identifying the problem.

System action

The IMS IFP utility ends abnormally.

User response

Correct the problem specified by the DBRC messages.

3514**Explanation**

This abend code is issued by FPO. A segment contains more than 8 sub-segment pointers. As this situation is impossible, it is indicative of a problem that prevents pointer checking from continuing.

System action

The IMS IFP utility ends abnormally with abend code 3514.

User response

Correct the specified pointer problem. The database might need to be recovered.

3515**Explanation**

This abend code is issued by FPO. The Child Segment Block does not have a pointer to a parent. As this situation is impossible, it is indicative of a problem that prevents pointer checking from continuing.

System action

Processing ends with abend code 3515.

User response

Correct the specified pointer problem. The database might need to be recovered.

3516**Explanation**

This abend code is issued by FPO. The length of the output record is greater than the DCBLRECL value of FPXOFIL that is specified by the OLRECL keyword.

System action

Processing ends with abend code 3516.

User response

Increase the OLRECL value.

3517**Explanation**

This abend code is issued by FPO. The program could not get sufficient virtual storage.

System action

Processing ends with abend code 3517.

User response

If the region size specified is too small, increase the REGION size on the JOB statement in the JCL, and resubmit the job.

3518**Explanation**

This abend code is issued by FPO. OAE could not stop the temporary area data set.

System action

Processing ends with abend code 3518.

User response

Correct the specified problem and resubmit the job.

3519

Explanation

This abend code is issued by FPO. Access method service process or DBRC process called by OAE ended abnormally.

System action

Processing ends with abend code 3519.

User response

Correct the specified problem and resubmit the job.

3520

Explanation

This abend code is issued by FPO. Catalog search interface (IGGCSI0) called by OAE ended abnormally.

System action

Processing ends with abend code 3520.

User response

See 'Reason code' and 'Return code' in message FPX0095E. Correct the specified problem and resubmit the job.

3521

Explanation

This abend code is issued by FPO. Load failure occurred. See message FPX0123E for details.

System action

Processing ends with abend code 3521.

User response

See message FPX0123E. Correct the specified problem and resubmit the job.

3522

Explanation

This abend code is issued by FPO. The address of a control block cannot be obtained. The reason code shows the reason of the error:

RSN=1 CSVQUERY for FPXMBLK0 failed
RSN=2 LOAD of FPXMBLK0 failed
RSN=3 LOAD of a sub-control block failed

System action

Processing ends with abend code 3522.

User response

This situation could be caused by an internal error. Contact IBM Software Support.

3523

Explanation

This abend code is issued by FPO. MAXFAILS was reached while UOWs were being reorganized, and ERRORACT=ABEND had been specified.

System action

Message FPX0322E is issued and processing ends with abend code 3523.

User response

See the explanation for messages FPX0317E and check the reasons of the errors.

3524

Explanation

This abend code is issued by FPO. Primary setup failed.

System action

Processing ends with abend code 3524.

User response

This situation could be caused by an internal error. Contact IBM Software Support.

3525

Explanation

This abend code is issued by FPO. Module load failure.

System action

Processing ends with abend code 3525.

User response

See the explanation for message FPX0172E.

3526**Explanation**

This abend code is issued by FPO. IMS /RM command failed.

System action

Processing ends with abend code U3526.

User response

See the explanation for message FPX0199E.

3527**Explanation**

This abend code is issued by FPO. Load module library is not APF-authorized.

System action

Processing ends with abend code U3527.

User response

See the explanation for message FPX0188E.

3528**Explanation**

This abend code is issued by FPO. TOSI API returned abnormally.

System action

Processing ends with abend code U3528.

User response

See the explanation for message FPX0190E.

3529**Explanation**

This abend code is issued by FPO. Exceeded the specified execution time that is set for TOSI REQUEST.

System action

Processing ends with abend code U3529.

User response

See the explanation for message FPX0191E.

3530**Explanation**

This abend code is issued by FPO. Storage was being obtained by the GETMAIN or the STORAGE macro. The action completed unsuccessfully with a nonzero return code.

System action

Processing ends with abend code U3530.

User response

See the explanation for message FPX0017E.

3531**Explanation**

This abend code is issued by FPO. Incorrect FPXBINTP file was specified.

System action

Processing ends with abend code U3531.

User response

See the explanation for message FPX0151E.

3532**Explanation**

This abend code is issued by FPO. Program FPXBTOAO found that there is no active IMS system to communicate.

System action

Processing ends with abend code U3532.

User response

See the explanation for message FPX0015E.

3533**Explanation**

This abend code is issued by FPO. The number of members that joined the XCF group exceeded the maximum.

System action

Processing ends with abend code U3533.

User response

See the explanation for message FPX0400E.

3598

Explanation

This abend code is issued by FPO. The exit encountered an unexpected, non-recoverable condition in ODE data interface.

System action

Processing ends. This abend code is accompanied by one of the following messages. The messages are written to the file associated with the XDRPRINT DD statement.

- FPXG001E
- FPXG002E
- FPXG003E

Abend codes of OSM

The Online Space Management utility internally calls FPA, FPO, and IMS utility processes. Therefore, abend codes of FPA, FPO, and IMS utilities are issued by the Online Space Management utility.

Messages

Use the information in these topics to help you diagnose and solve IMS HP Fast Path Utilities problems.

For each message, the following accompanying information is provided where applicable:

Explanation:

This explains what the message text means, what caused the message to be issued, and what its variable entry fields are (if any).

System action:

This explains what the system will do next

User response:

This describes whether a response is necessary, what the appropriate response should be, and what the resulting effect is on the system or program.

Message prefixes

The following table shows the prefixes of messages and the utility or the process that issues the messages.

Table 160. Message prefixes

Prefix	Utility or process
FABU	Diagnostics Aid utilities of FPB
FPX	OPC, ODE, OAE, OER, or ODM of FPO
HFPO	OSC driver job

- FPXG005E
- FPXG006E
- FPXG010E
- FPXG011E

User response

Correct the specified problem and resubmit the job.

3599

Explanation

This abend code is issued by FPO. This is the default user abend code. This abend code is unexpected, and the reason for it can vary.

System action

The IMS IFP utility ends abnormally.

User response

This problem might be an ODE internal error. Contact IBM Software Support.

Table 160. Message prefixes (continued)

Prefix	Utility or process
HFP1	OSC internal job generator
HFP2	Policy evaluation by using Policy Services
HFP4	OSC internal apply job
HFPA	Analyze process and Tune process of FPA
HFPB	Build Index process and Analyze process (index verify process) of FPA
HFPC	Command process of FPA
HFPD	Data set management process of FPA
HFPF	Function task process of FPA
HFPG	IMS Tools KB support
HFPH	DMAC Print process of FPA
HFPI	Image copy process of FPA
HFPJ	Extract process and Unload process of FPA
HFPK	Reload process of FPA
HFPL	Report process of FPA
HFPM	Main process of FPA
HFPO	DB Sensor process
HFPP	Checkpoint restart process of FPA
HFPQ	Online pointer check process of FPO
HFPR	Resource control process of FPA
HFPS	DMB process of FPA
HFPT	Thread task process of FPA
HFPU	Common routine process of FPA
HFPV	Area I/O process of FPA
HFPX	DBRC interface process of FPA
No prefix	Resolve Pointers report messages of FPA
No prefix	Process Summary of Analysis report messages of FPA
No prefix	Online Pointer Scan report messages of FPO

Note: The Online Space Management utility internally calls FPA, FPO, and IMS utility processes. Therefore, the Online Space Management utility issues messages that have prefix of FPA, FPO, and IMS utilities messages.

For FABA, FABB, FABC, and FABD messages, see the *IMS Fast Path Solution Pack: IMS Fast Path Basic Tools User's Guide*. For FABA4xxx messages, see the *IMS Fast Path Solution Pack: Supplementary Utilities User's Guide*.

Message suffixes

Some messages provide additional information by including the following suffixes:

A

Indicates that operator intervention is required before processing can continue.

I

Indicates that the message is informational only.

W

Indicates that the message is a warning to alert you to a possible error condition.

E

Indicates that an error occurred, which might or might not require operator intervention.

FABU messages

The following information is about messages and codes that begin with FABU.

FABU1001I DIAG ENDED NORMALLY

Explanation

This message is generated when Diagnostics Aid has completed successfully.

System action

Diagnostics Aid completes the job successfully with a return code of 0.

User response

None. This message is informational.

FABU1002W DIAG ENDED WITH WARNINGS

Explanation

This message is generated when trivial error conditions are encountered by Diagnostics Aid.

System action

Diagnostics Aid ends with a return code of 4.

User response

See other messages generated by Diagnostics Aid to determine the nature and the cause of the detected errors. Correct the problem, and rerun the job.

FABU1003E DIAG ENDED WITH ERRORS

Explanation

This message is generated when severe error conditions are encountered by Diagnostics Aid.

System action

Diagnostics Aid ends with a return code of 8.

User response

See other messages generated by Diagnostics Aid to determine the nature and the cause of the detected errors. Correct the problem, and rerun the job.

FABU1005W [SHPSLMD | SHPSMAC] DD STATEMENT NOT FOUND

Explanation

Diagnostics Aid could not find the SHPSLMD/SHPSMAC DD statement.

System action

Diagnostics Aid sets an end-of-job return code of 4 and continues processing. Diagnostics Aid does not generate a report for the load module or the macro.

User response

If you intended to specify the specified DD statement, correct the error and rerun the job.

FABU1006W DUPLICATE *member name* IN LIBRARY DDNAME *ddname*

Explanation

Diagnostics Aid found a duplicated member in the concatenated libraries.

System action

Diagnostics Aid uses the member which is first found in the concatenated libraries. Diagnostics Aid sets an end-of-job return code of 4 and continues processing.

User response

Make sure that the appropriate libraries have correct module/macro libraries. Correct the error, and rerun the job if necessary.

FABU1007W **DUMMY SPECIFIED FOR
[SHPSLMD | SHPSMAC] DD
STATEMENT**

Explanation

DUMMY was specified for the SHPSLMD/SHPSMAC DD statement.

System action

Diagnostics Aid sets an end-of-job return code of 4 and continues processing. Diagnostics Aid does not generate a report for the load module or the macro.

User response

If you did not intend to specify the dummy DD statement, correct the error and rerun the job.

FABU1008W **NO [MODULE | MACRO] MEMBERS
FOUND IN DDNAME [SHPSLMD |
SHPSMAC]**

Explanation

Diagnostics Aid could not find any utility modules or macros members from the DD *ddname* data set.

System action

Diagnostics Aid sets an end-of-job return code of 4 and continues processing.

User response

Make sure that the libraries have correct utility module or macro libraries. Correct the error, and rerun the job.

FABU2001E **LOAD FAILED FOR DDNAME
ddname MODULE *member***

Explanation

Diagnostics Aid could not load a member name from *ddname*.

System action

Diagnostics Aid sets an end-of-job return code of 8 and continues processing.

User response

Make sure that the member specified exists in the data set specified for the specified *ddname*. Correct the error, and rerun the job.

FABU3600E **OPEN FAILED FOR DDNAME
*ddname***

Explanation

The named DCB could not be opened.

System action

Diagnostics Aid ends with an abend code of U3600.

User response

Make sure that a *ddname* DD statement exists, and that it specifies the correct DD parameter. Correct any errors, and rerun the job.

FABU3601E **GET FAILED FOR DDNAME *ddname***

Explanation

The GET failed for a directory from the DD *ddname* data set.

System action

Diagnostics Aid ends with an abend code of U3601.

User response

See the MVS system message and programmer response. Correct the error, and rerun Diagnostics Aid. If the error persists, contact IBM Software Support.

FABU3602E **READ FAILED FOR DDNAME
ddname MEMBER *member***

Explanation

The READ failed for a *member* from the DD *ddname* data set.

System action

Diagnostics Aid ends with an abend code of U3602.

User response

See the MVS system message and programmer response. Correct the error, and rerun Diagnostic Aids. If the error persists, contact IBM Software Support.

FABU3603E **BLDL FAILED FOR DDNAME
ddname MEMBER *member***

Explanation

The *member* was not found when the BLDL macro searched the PDS directory for the *ddname*.

System action

Diagnostics Aid ends with an abend code of U3603.

User response

Make sure that the member specified exists in the data set specified for the specified ddname. Correct the error and rerun the job. If the error persists, contact IBM Software Support.

FABU3604E LOAD FAILED FOR DDNAME *ddname MODULE member*

Explanation

Diagnostics Aid could not load the member name from the *ddname*.

System action

Diagnostics Aid ends with an abend code of U3604.

User response

See the MVS system message and programmer response. Correct the error, and rerun Diagnostics Aid. If the error persists, contact IBM Software Support.

FABU3605E DELETE FAILED FOR MODULE *member*

Explanation

Diagnostics Aid could not delete a *member name*.

System action

Diagnostics Aid ends with an abend code of U3605.

User response

Contact IBM Software Support.

FABU3606E PUT FAILED FOR SYSPRINT

Explanation

Diagnostics Aid could not put report data in SYSPRINT.

System action

Diagnostics Aid ends with an abend code of U3606.

User response

See the MVS system message and programmer response. Correct the error and rerun Diagnostics Aid. If the error persists, contact IBM Software Support.

FABU3607E OPEN FAILED FOR SYSPRINT

Explanation

SYSPRINT DCB could not be opened.

System action

Diagnostics Aid ends with an abend code of U3607.

User response

Make sure that a *ddname* SYSPRINT DD statement exists, and that it specifies the correct DD parameter. Correct any errors, and rerun the job.

FABU3608E FIND FAILED FOR DDNAME *ddname MEMBER member*

Explanation

The FIND failed for a *member* from DDNAME *ddname* data set.

System action

Diagnostics Aid ends with an abend code of U3608.

User response

Make sure that the member specified exists in the data set specified for the specified ddname. Correct the error, and rerun the job. If the error persists, contact IBM Software Support.

FABU3609E DEVTYPE FAILED FOR DDNAME *ddname*

Explanation

The DEVTYPE failed for a DDNAME *ddname* data set.

System action

Diagnostics Aid ends with an abend code of U3609.

User response

Contact IBM Software Support.

FABU3610E RDJFCB FAILED FOR DDNAME *ddname*

Explanation

The READJFCB failed for a DDNAME *ddname* data set.

System action

Diagnostics Aid ends with an abend code of U3610.

User response

Contact IBM Software Support.

FABU3611E GETMAIN FAILED. INSUFFICIENT STORAGE TO RUN THE JOB

Explanation

Work space for Diagnostics Aid could not be obtained.

System action

Diagnostics Aid ends with an abend code of U3611.

User response

Increase the region size, and rerun the job.

FPX messages

The following information is about messages and codes that begin with FPX.

FPX0000E Unable to load message table FPXMSG00.

Explanation

Message table FPXMSG00 was being loaded. The action completed unsuccessfully with a nonzero return code. This error is normally associated with either the STEPLIB DD concatenation or virtual storage constraints.

System action

Processing ends with a return code of 8.

User response

Make sure that the load library containing object code is not damaged and that module FPXMSG00 is present in the library. Also, make sure that sufficient virtual storage is allocated to the utility region being executed so that MVS can bring FPXMSG00 into virtual storage.

FPX0001E Invalid EXIT parameter for IBMTOOL function.

Explanation

An invalid parameter was specified in the SYSIN input file.

FABU3612E TOO MANY [MODULE | MACRO] MEMBERS DETECTED IN DDNAME [SHPSMOD | SHPSMAC]

Explanation

There are too many IMS DBT members in the SHPSMOD/SHPSMAC DD data set.

System action

Diagnostics Aid ends with an abend code of U3612.

User response

Specify the correct data set for the specified DD statement, and rerun the job.

System action

Processing ends with a return code of 8.

User response

Make sure that the correct EXIT control statement is included in the SYSIN input stream.

FPX0002E Specified DD statement not found.

Explanation

The DD statement associated with an area was not found.

This message might be accompanied by the following text identifying the ddname associated with the DD statement.

- DD name = *ddname*

System action

Processing ends with a return code of 4.

User response

Supply the specified DD statement, and resubmit the job.

FPX0003E Error(s) processing dataset.

Explanation

This message is issued when the required DD name is not allocated to the job or identified data set could not be opened.

This message might be accompanied by the following additional text:

Function = *typ* DD name = *ddname* MVS RET and RSN codes are: *rc* + *rsn*

Explanation

The accompanying text identifies the following associated with the failure:

- *typ*
The failing operation
- *ddname*
The DDNAME associated with the file
- *rc*
The MVS return code associated with the failure
- *rsn*
The MVS reason code associated with the failure

System action

Processing ends with an abend code of 3509.

User response

If the error was due to a DD name not allocated to the job, supply the appropriate DD statement and resubmit the job. For other errors, determine the nature of the problem with the data set, correct it, and resubmit the job.

FPX0004E **Invalid or inconsistent parameters in DCB.**

Explanation

DCB parameters were being evaluated for a DD statement. The action completed unsuccessfully with a nonzero return code. When the DD name is DAR13SO, DAR3CSO, or DAR12O, LRECL and BLKSIZE (or both) might be too small.

This message might be accompanied by one of the following additional texts:

- *parm* = *dcbparm*
The parameters associated with the DCB for the DD statement.
- DD name = *ddname*
The *ddname* associated with the DD statement.

System action

Processing ends with a return code of 8.

User response

Correct the DCB parameters, and resubmit the job.

FPX0005E **Authorization failure ENTITY = <imsid.dbname.areaname> SAF RC = aa RACF RC =bb RACF RSN =dd**

Explanation

A security check to validate access failed.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Make sure that you have appropriate authorization to the specified entity. For details, see Chapter 29, “RACF considerations for FPO,” on page 951.

The high-level qualifier of the ADS does not need to be the same as the ENTITYX parameter '*imsid.dbname.areaname*'.

FPX0006W **IMS Version 5 SDEP CI encountered, CI was skipped.**

Explanation

A control interval containing IMS 5.1 SDEP segments was encountered but was not processed. ODE does not support IMS 5.1 SDEP segments, even when running under IMS 6.1.

The segments might have already been logically deleted, or they might represent valid segments inserted under IMS 5.1.

System action

Processing continues, but the IMS 5.1 SDEP data will be skipped on this run, potentially resulting in an incomplete extraction of data.

User response

Take appropriate steps to remove IMS 5.1 SDEP data from the area, such as:

- Delete the IMS 5.1 data and let IMS 6.1 data overwrite it.

- Unload and reload the area under a higher release than IMS 5.1.

FPX0007I **SDEP Logical Begin Time Stamp is:**
DBRC FORMAT =
ccyy.ddd.hh.mm.ss.t **STCK format**
= 16-
digit_hexadecimal_STCK_timestam
p

Explanation

The specified timestamp was selected for processing SDEPs.

System action

Processing continues.

User response

None. This message is informational. It is issued *once* per run when ODE detects that SDEPs are defined to the DEDB and the DMAC reflects a nonzero LBTS value.

FPX0008W **Selected SDEP timestamp is prior**
to the IMS SDEP Logical Begin
Time Stamp: SDEP LBTS: DBRC
FORMAT = *ccyy.ddd.hh.mm.ss.t* **STCK**
FORMAT = 16-
digit_hexadecimal_STCK_timestam
p
AFTERTIME: DBRC FORMAT =
ccyy.ddd.hh.mm.ss.t **STCK**
FORMAT = 16-
digit_hexadecimal_STCK_timestam
p

Explanation

A timestamp was selected for processing SDEPs. However, a time prior to the SDEP logical begin timestamp was selected. This warning message is issued every time a valid AFTERTIME timestamp is processed that is earlier than the IMS SDEP LBTS value.

System action

Processing continues.

User response

Verify that the selected SDEP timestamp reflects the desired time.

FPX0009E **AIB INQY call failed, AIBRETRN**
xxxxxxxx AIBREASN xxxxxxxx

Explanation

An AIB INQY ENVIRON call was issued to get the current IMS release level and IMSID. The call completed unsuccessfully.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

See *IMS Messages and Codes* for the appropriate response to this return code. Then resubmit the job.

FPX0010E **Unsupported IMS release.**

Explanation

The IMS release level returned by the AIB INQY ENVIRON call is not supported by ODE.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Upgrade IMS to one of the supported releases.

FPX0011E **DL/I #FPU call call error, status =**
SC, module = *modname*
(cont) First RBA of requested CI is
rba, **number of CIs is** *nnnn*

Explanation

An unexpected IMS status code was returned during a DL/I call. The second line is issued when FPO issues a DL/I call to read CIs of the DEDB area. The message identifies the following items:

call

The 4-character DL/I call type

SC

The 2-character IMS status code

modname

The FPO module that issued the call

rba

The RBA of the first requested CI

nnnn

The number of requested CIs

System action

Processing ends with an abend code of 3500.

User response

See *IMS Messages and Codes* for the appropriate response to these return codes. Then resubmit the job.

FPX0012I **FPXTRACT resource cleanup complete.**

Explanation

ODE is expected to be re-invoked by FP Utility Control, but was not. Resource cleanup was invoked to free storage and end the extract subtask. This situation can be caused by errors in the FP utility SYSIN stream, or by other parameter errors. The status of any previous processing is not affected, and some or all of this processing might have been successful. This message might be accompanied by additional ODE messages.

System action

Processing varies with the situation.

User response

Correct any errors in SYSIN, FPXCTL, or area-level control statements. Provide appropriate actions associated with any additional messages.

FPX0013I **Default output file //FPXOFIL detected.**

Explanation

The DD statement for the run-level control output file (FPXOFIL) was detected in the JCL.

System action

Processing continues.

User response

None. This message is informational.

FPX0014I **Input file //FPXCTL detected: Run-level control established.**

Explanation

The DD statement for run-level control input (FPXCTL) was detected in the JCL.

System action

Processing continues.

User response

None. This message is informational.

FPX0015E **No active IMS.**

Explanation

Program FPXBTOA0 found that there is no active IMS system to communicate.

System action

Processing ends with an abend code of U3532.

User response

Follow the description in [“Recovering and restarting OAE” on page 758.](#)

FPX0016E **Terminating errors during parameter analysis. Run aborted.**

Explanation

Errors were detected during parameter analysis.

This message is accompanied by additional messages that describe the detailed parameter errors.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the parameter errors, then resubmit the job.

FPX0017E **Failure to acquire storage.**

Explanation

Storage was being obtained by the GETMAIN or the STORAGE macro. The action completed unsuccessfully with a nonzero return code.

This message might be accompanied by one of the following additional texts:

Requested for *tbl* MVS RET and RSN codes are: *rc* + *rsn*

This accompanying text identifies the following associated with the storage failure:

- *tbl*
The failing internal table
- *rc*

The MVS return code associated with the storage failure

- *rsn*

The MVS reason code associated with the storage failure

System action

Processing ends with a return code of 8. The OAE Rename program (FPXBTOA0) ends with abend code U3530.

User response

Provide the appropriate action for the specified return code and reason code, as described in *MVS/ESA System Programming Library*.

FPX0018E **Error(s) processing dataset: DCB and DCBE snapped.**

Explanation

A file operation was being performed. The action completed unsuccessfully with a nonzero return code.

This message might be accompanied by the following additional text:

Function = *typ* DD name = *ddname* MVS RET and RSN codes are: *rc* + *rsn*.

Explanation

The accompanying text identifies the following associated with the DCB failure:

- *typ*

One of the following operations that failed:

- OPEN
- CLOSE
- GET
- PUT

- *ddname*

The DDNAME associated with the file

- *rc*

The MVS return code associated with the failure

- *rsn*

The MVS reason code associated with the failure

System action

Processing generally ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD

ends with a return code of 12. In some instances, processing ends with an abend code of 3507 or 3509.

User response

Determine the nature of the problem with the data set, and correct the problem.

FPX0019W **In-doubt SDEP segment(s) skipped. First at RBA nnnnnnnn**

Explanation

An in-doubt SDEP segment was encountered.

System action

The segment is skipped, and processing continues.

User response

This message is issued when in-doubt SDEP segment is found and the INDOUBT=NO option is specified, and the processing ends with a return code of 4. Note that the relevant segment is skipped. If extraction or selection of in-doubt SDEP segments is desired, set the INDOUBT=YES option, and resubmit the job.

FPX0020E **Invalid characters detected in control card.**

Explanation

A card image was validity-checked against an internal table of displayable characters. A non-displayable character was encountered.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the non-displayable character, and resubmit the job.

FPX0021E **Control statement expected. Must start in column 1.**

Explanation

One of the following situations can cause this problem when analyzing ODE FPXCTL DD, *areaname* DD, or a site default member:

- A control statement did not begin in column 1.

- The syntax of the statement displayed in the card image was incorrect and was erroneously interpreted as a control statement.

System action

Processing ends with a return code of 8, and the ODE run-level control ends with a return code of 12.

User response

Correct the invalid statement, and resubmit the job.

FPX0022E **At least 1 blank after control statement required.**

Explanation

A control statement was encountered, followed by a non-blank character.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the invalid statement, and resubmit the job.

FPX0023E **Continuation not found for previous card.**

Explanation

Data contained on the previous card did not conclude with a continuation character.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the invalid statement, and resubmit the job.

FPX0024E **Invalid keyword.**

Explanation

A keyword operator was expected. However, the data contained on the card image did not match the valid syntax for keywords.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the incorrect statement, and resubmit the job.

FPX0025W **Keyword specification exceeds range. Default value substituted.**

Explanation

Data associated with a keyword was evaluated for a specific range of numeric values. The value encountered was outside the acceptable range of values.

System action

The default value for the specified keyword is used, and the user coded value is ignored. Processing continues.

User response

To eliminate this warning message for subsequent executions of the job, correct the keyword value to fall within the specified range.

FPX0026E **Keyword specification NOT within range.**

Explanation

Data associated with a keyword was evaluated for a specific range of values. The value encountered was outside the acceptable range of values.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the keyword value, and resubmit the job.

FPX0027E **Invalid data type in keyword value specification.**

Explanation

A data type value was expected. However, the data contained on the card image was not a valid syntax for data types.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the incorrect data type, and resubmit the job.

FPX0028E **Invalid boolean operator specified.**

Explanation

A Boolean operator was expected. However, the data contained on the card image was not a valid syntax for Boolean operators.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the invalid Boolean operator, and resubmit the job.

FPX0029E **Parameter values not properly enclosed by parenthesis.**

Explanation

A parameter value that should be enclosed in parenthesis was encountered.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the invalid statement, and resubmit the job.

FPX0030E **Invalid time-stamp specification.**

Explanation

A timestamp specification was expected. However, the data contained on the card image was not a valid syntax for timestamp specifications.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the invalid timestamp specification, and resubmit the job.

FPX0031E **Invalid control statement.**

Explanation

A control statement was expected. However, the data contained on the card image was not a valid syntax for control statements.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the incorrect data type, and resubmit the job.

FPX0032E **No keyword or continuation specified.**

Explanation

A keyword or continuation character was expected. However, the data contained on the card image was not a valid syntax for keywords or continuation characters.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the statement, and resubmit the job.

FPX0033I **No overriding control cards. All defaults apply.**

Explanation

No input parameters were specified.

System action

Processing continues using default parameter values.

User response

None. This message is informational.

FPX0034E Segment *xxxxxxx* does not exist.

Explanation

A segment name was specified; however, the segment name did not match any segments in the DBD for this database.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the segment name, and resubmit the job.

FPX0035E Cannot specify both **SEG=ALL** and **SEG=seg_name** for the same area / run.

Explanation

Mutually exclusive segment selection criteria were specified.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the incompatible specification, and resubmit the job.

FPX0036W # of **SELECT** statements limit exceeded. Card ignored.

Explanation

More than 2048 valid **SELECT** statements were specified. The limit is 2048 or fewer.

System action

The selection is ignored, and processing continues. However, if there are **FIELD** or **ANDFIELD** qualifications that follow the discarded **SELECT** **SEG** statement, an error condition results and message **FPX0038E** is issued. In such cases, processing ends with a return code of 8, and the ODE run-level control

using **SYSIN** and **FPXCTL** DD ends with a return code of 12.

User response

Correct the specification, and resubmit the job.

FPX0037W Selection of **SEG=ALL** in effect. Additional keywords were ignored.

Explanation

The keyword **SEG=ALL** was specified. Additional keywords were encountered that would impose more restrictive selection criteria than that allowed with the **SEG=ALL** keyword. These additional keywords were ignored.

All segment qualification keywords (such as **FIELD** and **AFTERTIME**) are ignored when **SEG=ALL** is in effect.

System action

Processing continues.

User response

Verify whether the additional keywords are desired.

FPX0038E Not preceded with required keyword.

Explanation

A segment qualification keyword (for example, **FIELD**) was specified without specifying the segment name, that is, without being preceded by a valid **SEG=** keyword.

System action

Processing ends with a return code of 8, and the ODE run-level control using **SYSIN** and **FPXCTL** DD ends with a return code of 12.

User response

Code a valid segment name.

FPX0039I Starting Online Fast Path utility processing: **AREA name = areaname** **AREA number = area#**

Explanation

This message indicates that an Online Fast Path utility has begun processing. It identifies the area name and area number for which the processing is being performed.

System action

Processing continues.

User response

None. This message is informational.

FPX0040E **OUTPUT=UR | TFMT mutually exclusive to SORT=YES.**

Explanation

DEDB TFMT or UR unload/reload format was selected for OUTPUT. An unload-formatted file cannot be sorted; this is because a sort might create a record sequence that is invalid for reload.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the incompatible specification, and resubmit the job.

FPX0041E **OUTPUT=UR | DBT | TFMT mutually exclusive to FIELDS=ss, ll**

Explanation

DEDB DBT, TFMT, or UR Unload/Reload format was selected for OUTPUT. The output records must contain all of the data from the segment; specific fields cannot be selected.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the incompatible specification, and resubmit the job.

FPX0042E **Invalid sub-parameter. Should be "KEY".**

Explanation

The KEY subparameter must be associated with the FIELD= keyword, but KEY was either missing or invalid.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the subparameters on the FIELD keyword, and resubmit the job.

FPX0043E **Data comparison length/position exceeds range.**

Explanation

The value of the position *ss* or length *ll* associated with the FIELD keyword was invalid.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the invalid value, and resubmit the job.

FPX0044E **Sub-parameter of NULL value specified.**

Explanation

A subparameter associated with the FIELD keyword was missing or contained a null value.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the subparameter, and resubmit the job.

FPX0045E **Too many / un-even # of characters.**

Explanation

The number of characters specified in a subparameter of the FIELD keyword exceeds the allowable maximum.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

See the description of the FIELD keyword for a description of the maximum lengths that can be specified in a subparameter. Correct the subparameter, and resubmit the job.

FPX0046E # of FIELD= / ANDFIELD= / FIELDS= keywords limit exceeded.

Explanation

Too many of one of the indicated keywords for the current segment qualification were encountered.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Consult the description of the appropriate keyword to find the maximum number of specifications allowed. Correct the specification, and resubmit the job.

FPX0047E Comparison data not properly enclosed by apostrophes.

Explanation

The value associated with subparameter data of the FIELD keyword was not properly bounded by apostrophes (').

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Enclose the value with apostrophes, and resubmit the job.

FPX0048E Invalid KEY specification. DBD does not have key defined.

Explanation

The KEY subparameter of the FIELD keyword was specified; however, the segment was defined in the DBD without a key.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the incompatibility between the FIELD keyword and the segment description in the DBD. Then resubmit the job.

FPX0049I (FPXIN, FPXCTL, or DASYSIN input card image)

Explanation

The message shows the FPXIN, FPXCTL, *areaname* or DASYSIN control statements used in this execution of the utility.

System action

Processing continues.

User response

None. This message is informational.

FPX0050W OUTPUT data specification is beyond segment length. SEGMENT *seg_name* RBA *rba* actual length is *seg_length*

Explanation

The FIELDS keyword of the OUTPUT control statement specified a relative starting position *ss* and length *ll* that exceeded the length of the segment.

System action

The record being processed is not included in the output.

User response

Correct the starting position *ss* and/or length *ll* in the control statement, and resubmit the job.

FPX0051E Neither default nor explicit output file specification detected. Cannot process this area.

Explanation

No output file specification exists. Neither the FPXOFIL DD statement nor the OFIL= keyword and its accompanying DD statement were found.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the error, and resubmit the job.

FPX0052W	# of SELECT block limit exceeded. Selection/Extraction specification discarded.
-----------------	--

Explanation

ODE tried to generate internally a SELECT statement for an OUTPUT SEG= specification for which no corresponding SELECT SEG= specification was provided. The generation failed because preceding SELECT and/or OUTPUT statements exceeded the limit of 2048 segment selections.

System action

All selections for the relevant segment are discarded. Processing continues.

User response

Reduce the number of SELECT statements (see [“Utility control statements for ODE”](#) on page 737), and resubmit the job.

FPX0053W	Invalid keyword for non-SDEP segment. Keyword ignored.
-----------------	---

Explanation

An SDEP-only keyword was specified for a non-SDEP type segment.

System action

The keyword is ignored. Processing continues.

User response

To eliminate this message from future executions of the job, do not use SDEP-only keywords on non-SDEP segment specifications.

FPX0054I	Area level statistics: <i>statistics data</i>
-----------------	--

Explanation

The statistics following this message are associated with the specified area. Message FPX0039I, which precedes this message, identifies the area.

System action

Processing continues.

User response

None. This message is informational.

FPX0055I	Extract process completed successfully.
-----------------	--

Explanation

This message indicates that ODE has successfully completed.

System action

Processing continues.

User response

None. This message is informational.

FPX0056I	Database level statistics: DEDB NAME = <i>dbdname</i>
-----------------	--

Explanation

The statistics following this message are associated with the specified database.

System action

Processing continues.

User response

None. This message is informational.

FPX0057E	Inconsistent EXIT parameter for IBMTOOL function.
-----------------	--

Explanation

A single execution of the IFP utility contains multiple "TYPE" control statements where more than one utility has been requested. Different utilities cannot be mixed in the same input stream to the IFP utility.

System action

Processing ends with a return code of 8.

User response

Restrict the control statements to a single utility type.

FPX0058E **Cannot process more than 1 Area per run with ANALYZE function.**

Explanation

A single execution of the IFP utility requests multiple runs of the ANALYZE function by containing multiple "EXIT" control statements specifying this function. The IFP utility supports only a single run of the ANALYZE function.

System action

Processing ends with a return code of 8.

User response

Restrict the control statements to a single utility type.

FPX0059E **Invalid record type found by FPXAEPRO.**

Explanation

During processing a record was not recognized.

System action

Processing ends with an abend code of 3511.

User response

This is an internal error. Contact IBM Software Support.

FPX0060E **Mandatory DD statements not allocated.**

Explanation

The JCL stream for the IFP utility is missing some of the required DD statements.

System action

Processing ends with return a return code of 8.

User response

Make sure that all of the mandatory DD statements are present in the JCL.

FPX0062E **MAXERROR Threshold has been reached.**

Explanation

The number of pointer errors encountered exceeded the threshold allowable limit of these errors as specified on the MAXERROR utility control statement.

System action

Processing generally ends with return code 8, but it is dependent on the values of ERRORACT and MAXERROR.

If ERRORACT is ABEND and the MAXERROR value is reached, the program ends with an abend code of 3510.

Return code 4 is returned in some cases.

User response

Determine the reason for the number of pointer problems.

Either increase the MAXERROR threshold or take the area offline and do a full analysis.

FPX0063E **Compression exit *exit_name* not available for segment *segment_name***

Explanation

The EXPAND keyword has been specified in the utility control statement, but the identified compression exit is not available to the IFP utility. This is a problem because a compressed segment was encountered, and the segment needs to be expanded for evaluation.

System action

Processing ends with an abend code of 3512.

User response

Make sure that the compression exit is available to the IFP utility in the JOBLIB or STEPLIB DD statement concatenation.

FPX0064W **Randomizer is not loaded. Full area scan will be used.**

Explanation

The randomizer has not been loaded. A full area scan will be performed.

System action

Processing does a full area scan and ends with a return code of 4.

User response

This is a warning message. Check that the randomizer is included in the load module in STEPLIB.

FPX0065W **Randomizer returned correct area # but invalid RAP number. Full area scan will be used.**

Explanation

The randomizer was successfully loaded and selected the proper DMAC; however, the RAP number was not within the RAA range.

System action

Processing does a full area scan and ends with a return code of 4.

User response

Determine the reason why this RAP number was not within the RAA range.

FPX0066I **All root segment SELECTs are full key EQ, optimized access using randomizer is in effect.**

Explanation

The randomizer has been successfully loaded. Only the required UOW's needed will be read; a full area scan is not performed.

ODE uses optimized access via the randomizer to read only the UOWs required instead of scanning the entire area.

This message informs you that optimized access is in effect.

System action

Processing continues.

User response

None. This message is informational.

FPX0067W **Randomizer is AMODE(24). Must be AMODE(31).**

Explanation

The randomizer has been compiled/linked with AMODE(24). IMS requires all randomizers to be 31-bit capable.

System action

The utility detected that the randomizer is not 31-bit capable. Processing builds the performance optimized UOW list using the randomizer and ends with a return code of 4.

User response

This is a warning message. If possible, determine the reason the randomizer is AMODE (24) and correct the problem.

FPX0068E **IC requested, but no DD statement for primary IC output found.**

Explanation

An image copy has been requested via specification of the IC=YES keyword in the utility control statements. However, neither the default DD statement (DFSUDUMP) nor a DD statement matching the name on the optional IC1OUTDD= utility control statement was included in the IFP utility JCL stream.

System action

Processing ends with a return code of 8.

User response

Make sure that a DD statement for the image copy is included in the JCL stream.

FPX0069W **No IC requested. DBRC registration suppressed.**

Explanation

An image copy was not requested in the utility control statements. DBRC is active on the IMS system; however DBRC registration of the area being processed is not performed.

System action

Processing continues.

User response

None required.

FPX0070E **Required DD statements for run type not found.**

Explanation

The parameter associated with the TYPRUN utility control statement requires additional DD statements

in the JCL stream in addition to those always required by the IFP utility. The JCL stream does not contain DD statements for these additional data sets. For more information about the required DD statements, see [“Data sets required for each operating mode” on page 680.](#)

System action

Processing ends with a return code of 8.

User response

Make sure that all the mandatory DD statements are present in the JCL.

FPX0071E **Fatal DBRC error during DBRC [INIT | EOD] call.**

Explanation

This message indicates that a problem was encountered during the DBRC INIT call or the DBRC EOD call. This message is accompanied by a DBRC message (prefix DSP) giving more details pertaining to the failure.

System action

Processing ends with an abend code of 3513. The DBRC return code from the DFSBRLSC call is set to register 15.

User response

Consult the IMS Messages and Codes manual for an explanation of the DBRC message and take the appropriate actions.

FPX0071E **The image copy of AREA name = areaname not registered with DBRC due to pointer errors or premature utility termination.**

Explanation

The image copy data set that is created for the indicated area is not registered to DBRC because pointer errors were detected or errors occurred before the completion of the image copy process.

System action

Processing ends with a return code of 8.

User response

Correct the problems and create another image copy for recovery purposes.

FPX0072W **CI SNAP limit reached, no further CIs will be snapped.**

Explanation

This message indicates that more than 10 CIs containing serious errors have been encountered.

System action

A SNAP of the first 10 CIs in error, along with a brief detailed description of the error, is created. Processing continues and error detail descriptions of additional CIs containing severe errors are produced, but no more CIs are snapped.

User response

This area has more than 10 bad CIs. A full offline analysis is recommended. Alternatively, use IDCAMS to print any RBAs that are reported as errors in the Online Pointer Scan report.

FPX0073W **Area has pointer errors that are not related to orphan SDEP segments.**

Explanation

One or more pointer errors that are not related to orphan SDEP segments are detected in the area. This message is reported only when the return code exit FPXRCEXT is used.

System action

Processing ends with a return code of 4.

User response

Examine the Online Pointer Scan report and investigate the cause of the error. See [“Online Pointer Scan report” on page 704](#) to interpret the report.

FPX0074I **Image Copy [using compression_routine] completes. [Record count nnnnnnnn]**

Explanation

The image copy processing completed.

compression_routine is shown only when an IMS HP Image Copy compression routine is specified by the IC=(YES, *comprtn*) keyword. Record count is not shown when the IC=(YES,ICEIN) keyword is specified.

System action

Processing continues.

User response

None. This message is informational.

FPX0075E **Segment twin chain loop. Fatal Error. R15 at abend= last segment RBA processed.**

Explanation

A segment was encountered with a pointer error such that the segment twin pointer points to itself, or the number of twin segments exceeded the threshold value that is defined by the MAXTWINCTR keyword. The default value for the MAXTWINCTR keyword is 2000000.

System action

Processing ends with an abend code of 3502.

User response

Run OPC with the SNAPPIT DD statement, then check the CI SNAP dump in the SNAPPIT data set to see if the area does have the wrong pointer that is causing the twin chain loop, or the area does have twin chains more than the threshold value. If the number of twin segments exceeds the threshold value, specify a larger value for the MAXTWINCTR keyword and rerun the job.

FPX0076W **Extraction of SDEP segments suppressed: Output is in UNLOAD/RELOAD format.**

Explanation

Output was requested in unload/reload format via the OUTPUT=DBT, the OUTPUT=TFMT, or the OUTPUT=UR control statements. SDEP segments are not included in the output file when unload/reload format has been selected.

System action

Processing continues.

User response

If OUTPUT=DBT is specified, you can extract SDEP segments by specifying the SDEPHIER=YES keyword.

FPX0077E **IC1OUTDD and IC2OUTDD refer to the same DD name.**

Explanation

The input parameters IC1OUTDD and IC2OUTDD both indicate the same DD name for the image copy data set. These two parameters must reference unique DD names, which, in turn, must reference unique data set names.

System action

Processing ends with a return code of 8.

User response

Make sure that the IC1OUTDD and IC2OUTDD refer to different DD names.

FPX0078E **Not all segments in hierarchic path selected with UNLOAD/RELOAD format.**

Explanation

Output was requested in an unload/reload format, but not all segments in the hierarchical path were selected. The selected segment and all of its parent segments must be selected for output. If the output file is not used for reload, specify the OUTPUT keyword as STD or USER.

System action

Processing ends with a return code of 8.

User response

Correct the selection criteria to include all segments in the hierarchical path.

FPX0079W **STARTUOW and/or STOPUOW utility statement detected: Request for IC ignored.**

Explanation

An image copy was requested, but the extract process is limited by STARTUOW and/or STOPUOW utility statements.

System action

Processing continues without taking an image copy.

User response

If an image copy is required, remove the STARTUOW and/or STOPUOW utility statements.

FPX0080E **The image copy data set of AREA name = *areaname* contains pointer errors.**

Explanation

Image copy processing has not completed successfully. The image copy data set produced is unusable. This message is accompanied by additional messages further identifying the problem.

System action

The job ends. The return code depends on the preceding message. If message FPX0074I is issued, the return code is 4; if message FPX0269E is issued, the return code is 8.

User response

Correct the specified problem. The database might need to be recovered.

FPX0081E **New IOVF size is greater than area size.**

Explanation

As specified in the ROOT parameter of the DBD definition associated with the target area, the IOVF size will exceed the VSAM allocation for the target area data set.

System action

Processing ends with a return code of 8.

User response

One of the following actions should be taken:

- Increase the VSAM definitions for the target area data set.
- Request a smaller increase of IOVF, as defined in the ROOT parameter of the DBD definition associated with the target area, to allow more room for the SDEP area. Perform the DBD/ACB gens.

Resubmit the utility after taking one of these actions.

FPX0082E **SDEP portion of the area will decrease by *xxxxxxx***

Explanation

As a result of utility processing, the SDEP area will decrease by the specified amount. Via the SDEPWARN control keyword, the user specified that OAE should end rather than decrease the SDEP area.

System action

Processing ends with a return code of 8.

User response

One of the following actions should be taken:

- Increase the VSAM definitions for the target area data sets by the amount specified in the message.
- Request a smaller increase of IOVF, as defined in the ROOT parameter of the DBD definition for the target area, to allow more room for the SDEP area. Perform the DBD/ACB gens.
- Remove the SDEPWARN control keyword if a decrease in the SDEP area is allowable.

Resubmit the utility after taking one of these actions.

FPX0083I **SDEP portion of the area will decrease by *xxxxxxx***

Explanation

As a result of utility processing, the SDEP area will decrease by the specified amount. The user did NOT include the SDEPWARN control keyword, and did not want to flag this occurrence as an error.

System action

Processing continues.

User response

None. This message is informational.

FPX0084E **Changes that alter RAP portion of area not permitted.**

Explanation

As a result of the ROOT parameter of the DBD definition associated with the target area, the RAP portion of the area will be of a size different than in the original area. The RAP portion cannot be increased or decreased by the OAE utility.

System action

Processing ends with a return code of 8.

User response

Change the ROOT parameter of the DBD for the DEDB such that no changes to the RAP are introduced and perform DBD/ACB gen. Or change the ROOT= control keyword parameter such that no changes to the RAP are introduced.

FPX0085E IOVF expansion requires IOVF size to be increased via DBD source.

Explanation

The ROOT parameter of the DBD definition associated with the target area are specified in one of the following ways:

- Root parameter in the database descriptors (DBDs) on the ACBLIB DD
- Root parameter in the EXIT control statement of SYSIN DD
- Root parameter in FPXIN DD

The size of the IOVF portion of the area will not increase from that of the original area. The IOVF portion must be increased by the OAE utility.

System action

Processing ends with a return code of 8.

User response

Change the ROOT parameter of the DBD for the DEDB such that the IOVF will not decrease and perform DBD/ACB gen. Or change the ROOT= control keyword parameter such that the IOVF will not decrease.

FPX0086E Area is RSR covered.

Explanation

The area is currently authorized as RSR covered.

System action

Processing ends with a return code of 8.

User response

Rerun the OAE utility after the state of the area becomes non-RSR covered.

FPX0087E VSO Area not allowed.

Explanation

The area is currently loaded into VSO (either data space or Coupling Facility), and it must be unloaded prior to utility execution.

System action

Processing ends with a return code of 8.

User response

Issue the /VUNLOAD IMS command and rerun the OAE utility.

FPX0088E IOVF and SDEP keywords cannot both be specified.

Explanation

Both the IOVF and SDEP control keywords were specified for the OAE utility. These keywords are mutually exclusive.

System action

Processing ends with a return code of 8.

User response

Specify either the IOVF or SDEP control keywords, but not both.

FPX0089E EXIT EXTEND statement must specify IOVF on this restart.

Explanation

The prior occurrence of utility processing specified the IOVF control keyword. The current occurrence of the utility did not specify IOVF. The restarted utility must indicate the same type of processing as the prior failed occurrence.

System action

Processing ends with a return code of 8.

User response

Specify the IOVF control keyword and resubmit the job.

FPX0090I OAE has been restarted from a prior failure.

Explanation

The RESTART control keyword was included in the current occurrence of the utility. Processing began at the point of failure in the previous occurrence of the utility.

System action

Processing continues.

User response

None. This message is informational.

FPX0091E **OAE restart not processed. Follow the RESTART procedure in the OAE User's Guide.**

Explanation

OAE requires user's action to restart the job.

System action

Processing ends with a return code of 8.

User response

Follow the description in [“Recovering and restarting OAE”](#) on page 758.

FPX0092E **OAE Restart is required. Include RESTART parameter on the EXIT EXTEND statement.**

Explanation

A prior occurrence of the OAE utility had ended prematurely. The utility cannot be executed to extend a given area until a restart completes the prior failed attempt.

System action

Processing ends with a return code of 8.

User response

Specify the RESTART parameter and resubmit the utility.

FPX0093E **Either IOVF or SDEP keyword must be specified.**

Explanation

Neither the IOVF nor SDEP control keywords were specified for the OAE utility. One of these keywords must be specified.

System action

Processing ends with a return code of 8.

User response

Specify either the IOVF or SDEP control keywords.

FPX0094E **Active SDEP segments in the CSC portion of area.**

Explanation

An IOVF extension was requested for an area which also supports SDEP segments. Values coded for the number2 and overflow2 parameters on the ROOT= parameter for the new DBD result in an IOVF extension into CIs which currently contain SDEP segments or are available for the insertion of the next new SDEP segment. These CIs are referred to as the CSC portion of the area.

While such an extension is supported by OAE, one of the following situations might be causing a problem:

- The CSC portion of the area contains SDEP segments which are currently active.
- Even after the SDEP Delete Utility has been run, one or the following RBAs lies within the CSC portion:
 - The Oldest Active SDEP RBA
 - The Next Available SDEP RBA

System action

Processing ends with a return code of 8.

User response

Make sure that the pointer to the Oldest SDEP RBA and the Next Available RBA are both greater than the address of the new First SDEP RBA. The values for all three of these RBAs are specified on the statistics report. To make sure that these values are adequate relative to each other for the extension, take one of the following actions:

- Insert additional SDEP segments until the value for the Next Available SDEP RBA is larger than the new value for First SDEP RBA, and then run the SDEP Delete Utility to delete segments until the Oldest SDEP RBA is also greater than the new value for First SDEP RBA.

On a system with a high volume of SDEP activity, you might not need to take any action other than to wait until normal system activity has inserted sufficient additional SDEPs. You might need to monitor the SDEP usage to determine when sufficient activity has occurred.

Additional options:

- On a less active system, you might need to insert SDEPs via an application and then subsequently delete them.
- Reduce the new values for the number2 and overflow2 parameters of ROOT= parameter to request a smaller IOVF extension, one that does not go beyond the specified oldest SDEP RBA value.

FPX0095E **Failure reading data set information from catalog.**
Reason code: xxx Return code: yyy
Dataset: data set name

Explanation

The catalog was queried for information related to the VSAM allocation for the data sets associated with the area, as defined to DBRC. Information for one or more of the data sets was not successfully extracted. This message is accompanied by additional messages which help determine the nature of the problem. Common problems included are as follows:

- The data set name identified to DBRC does not match the data set name used in the VSAM allocation.
- The data set resides in a catalog different than the one used at utility run time.

System action

Processing ends with a return code of 8. When the original expansion area extender is used, processing might end with an abend code of 3520.

User response

Correct the problem and resubmit the OAE utility.

FPX0096E **INVALID FPXIN keyword: xxxxxxxx**

Explanation

One or more of the control keywords or keyword parameters passed to the OAE utility were invalid. The error message identifies the invalid control keyword.

System action

Processing ends with a return code of 8.

User response

Correct the invalid control keyword or parameters and resubmit the utility.

FPX0096E **DUPLICATE FPXIN keyword:**
xxxxxxx

Explanation

One or more of the control keywords or keyword parameters passed to the OAE utility were invalid. The error message identifies the invalid control keyword.

System action

Processing ends with a return code of 8.

User response

Correct the invalid control keyword or parameters and resubmit the utility.

FPX0096E **FPXIN keyword parameter error:**
xxxxxxxxxx

Explanation

One or more of the control keywords or keyword parameters passed to the OAE utility were invalid. The error message identifies the invalid control keyword.

System action

Processing ends with a return code of 8.

User response

Correct the invalid control keyword or parameters and resubmit the utility.

FPX0097W **Utility completed successfully in TEST mode with no database updates.**

Explanation

The TEST control keyword was included in the current execution of the OAE utility. The inputs to the utility were validated and the statistical report was generated, but the area was not extended.

System action

Processing ends with a return code of 4.

User response

This message is informational; however, make sure that only TEST processing was desired.

FPX0098W **Statistics may be incomplete due to processing errors.**

Explanation

The utility was prematurely ended due to an error condition. At the point of termination, not all of the data required to generate a complete statistical report was available. The report was generated, but some of the data might be incomplete or inaccurate.

System action

Processing continues.

User response

This message is a warning that, although the data on the report is representative of the processing, it might not be entirely accurate.

FPX0099I **Online Area Extender statistics:
statistics data**

Explanation

This line indicates the beginning of the statistical report.

System action

Processing continues.

User response

None. This message is informational.

FPX0100E **The DBD name specified in the
PARM does not match the member
indicated by the ACBLIB DD
statement.**

Explanation

A valid ACB member was specified on the ACBLIB DD statement; however, this member did not match the DEDB specified in the second position on the PARM portion on the EXEC FPUTIL statement.

System action

Processing ends with a return code of 8.

User response

Make sure that the DEDB specified by the PARM statement matches the member specified on the ACBLIB DD statement.

FPX0101E **Unable to open the ACBLIB DD
statement.**

Explanation

One of the following might have occurred in the JCL for the OAE utility:

- The ACBLIB DD statement might be missing
- The data set associated with the ACBLIB DD statement might be invalid.

- The DEDB member name might be missing from the DSN= parameter of the DD statement.

System action

Processing ends with a return code of 8.

User response

Determine the problem with the ACBLIB DD statement and resubmit the utility.

FPX0103E **CREATE utility did not complete
successfully. See accompanying
DFS messages for additional
information.**

Explanation

OAE invoked the CREATE utility internally; however, the utility did not complete successfully. Additional messages with prefix DFS further explain the failure.

System action

Processing ends with a return code of 8.

User response

Determine and correct the problem and resubmit the utility.

FPX0104E **Changes that alter CIs per UOW
not permitted.**

Explanation

As a result of the UOW parameter of the DBD definition associated with the target area, the number of CIs per UOW will be different from that in the original area. The CIs per UOW cannot be changed by the OAE utility.

System action

Processing ends with a return code of 8.

User response

Change the UOW parameter of the DBD for the DEDB such that no changes to the CIs per UOW are introduced and perform DBD/ACB gens.

FPX0105E **Changes that alter DOVF CIs per
UOW not permitted.**

Explanation

As a result of the UOW parameter of the DBD definition associated with the target area, the number of DOVF

CI per UOW will be different from that in the original area. The DOVF CIs per UOW cannot be changed by the OAE utility.

System action

Processing ends with a return code of 8.

User response

Change the UOW parameter of the DBD for the DEDB such that no changes to the DOVF CIs per UOW are introduced and perform DBD/ACB gens.

FPX0106E **Changes that alter CI size not permitted.**

Explanation

As a result of the SIZE parameter of the DBD definition associated with the target area, the CI size will be different from that in the original area. The CI size cannot be changed by the OAE utility.

System action

Processing ends with a return code of 8.

User response

Change the SIZE parameter of the DBD for the DEDB such that no change to the CI size is introduced and perform DBD/ACB gens.

FPX0107E **SDEPHIER keyword is specified with mutually exclusive keyword value or against no SDEP defined DB.**

Explanation

The SDEPHIER keyword is specified with one of the following specifications:

- OUTPUT=TFMT | UR (this keyword and SDEPHIER are mutually exclusive)
- SORT=YES and ODESORT DD are not specified in JCL
- A database that has no SDEP defined in the DBD definition

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

SDEPHIER keyword cannot be specified when any one of the above three conditions is met. Correct the incompatible specification, and resubmit the job.

FPX0109E **SDEPHIER=YES is not supported by current IMS APAR level.**

Explanation

The current IMS APAR level is below PQ78840.

System action

Processing ends with an abend code of 3500.

User response

Install IMS APAR PQ78840, and resubmit the job.

FPX0110I **CI formatted by Create Utility was encountered, CI was skipped. RBA of CI: nnnnnnnn**

Explanation

A control interval that contains dummy data formatted by the DEDB Area Data Set Create Utility was encountered, but it was not processed. The dummy data CI that is specified by the system ID "CREATFMT" is in the system name filed in the SDEP CI prefix, and the CI was formatted during the copy phase of the DEDB Area Data Set Create Utility.

This CI implies that it was preallocated and will be overlaid by either of the following SDEP CI images:

- SDEP CI image that contains real SDEP segments
- SDEP preallocated dummy CI image that contains the CIs that were extended by Online Area Extender (OAE) SDEP extension

System action

Processing continues, but the "CREATFMT" dummy CI will be skipped on this run.

User response

None. This message is informational.

FPX0111E **Incorrect keyword for SDEP segment.**

Explanation

SDEP type segment has no dependent segment.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Do not use DEPENDENTS or BOTH on SDEP segment specification. Correct the incompatible specification, and resubmit the job.

FPX0112E **Selection of DEPENDENTS in effect. No optional keyword permitted.**

Explanation

Only the dependent segments of this segment are to be included in the output file. Optional keyword specification for this segment is not permitted.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Do not use OUTPUT optional keywords FIELDS, STOPAFT, SKIP, EVERY, or AFTERTIME if DEPENDENTS is specified. Correct the incompatible specification, and resubmit the job.

FPX0113E **Incorrect DEPENDENTS/BOTH specification. This segment has no dependent segment.**

Explanation

Because this segment has no dependent segment, you cannot specify DEPENDENTS or BOTH.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the incompatible specification, and resubmit the job.

FPX0114I **SDEP segment ignored for SORT=YES option: AREA NAME=*areaname* AREA NUMBER=*area#***

Explanation

SDEP segment was not extracted because SORT=YES option was specified but SDEPHIER=YES was not specified.

System action

Processing continues.

User response

If you need SDEP segments for the extract, you can rerun the job by specifying SDEPHIER=YES to extract SDEP segments with SORT=YES.

FPX0115W **No SDEP segments found -SDEP processing bypassed for area.**

Explanation

The SDEP keyword was specified on the ANALYZE control statement. However, OPC determined that the SDEP part that is associated with the area was empty. This condition is detected when the *logical beginning* and the *logical end* that are contained in DMAC have the same value, or if their values differ by 4.

System action

OPC sets an end-of-job return code of 4 and continues processing. The SDEP processing is bypassed.

User response

Proceed with the remainder of the job steps in the report generation procedure.

FPX0116W **Area may have orphan SDEP segments.**

Explanation

The number of 4,P records is less than the number of 4,S records that were generated by OPC. The root segment that contains SDEP segments might have been deleted, or there were some SDEP segments within the LB-LE range that were deleted by the SDEP Delete Utility. This message is reported only when the return code exit FPXRCEXT is provided.

System action

OPC sets an end-of-job return code of 4 and continues processing.

User response

Proceed with the remaining job steps in the report generation procedure.

FPX0117I **Return code user exit FPRCEX0 is called.**

Explanation

Return code user exit routine FPRCEX0 is called.

System action

Processing continues.

User response

None. This message is informational.

FPX0118I **Return code user exit FPRCEX0 completed. Original RC = aa, returned RC = bb**

Explanation

Return code user exit routine FPRCEX0 has completed. The original return code might be changed to the returned return code.

System action

Processing continues.

User response

None. This message is informational.

FPX0119I **Return code user exit FPRCEX0 is not called because of message table GETMAIN failure.**

Explanation

This message is generated when the return code user exit routine FABARCEX is not called due to GETMAIN failure of the message and return code table. The return code is not changed even if FPRCEX0 exists.

System action

Processing continues.

User response

Increase the region size and rerun the utility.

FPX0120W **-Warning only:**

Explanation

An SDEP Segment checksum error occurred during pointer checking.

System action

OPC sets an end-of-job return code of 4 and continues processing.

User response

Run FABADA3 and FABADA4 routines of FP DEDB Pointer Checker to obtain the details of the error, then analyze the error from the generated reports.

FPX0120W **Segment checksum error in SDEP portion.**

Explanation

An SDEP Segment checksum error occurred during pointer checking.

System action

OPC sets an end-of-job return code of 4 and continues processing.

User response

Run FABADA3 and FABADA4 routines of FP DEDB Pointer Checker to obtain the details of the error, then analyze the error from the generated reports.

FPX0120W **Segment code: 02 Segment name: xxxxxxxx Checksum value: xxxxxxxxxxxxxxxx**

Explanation

An SDEP Segment checksum error occurred during pointer checking.

System action

OPC sets an end-of-job return code of 4 and continues processing.

User response

Run FABADA3 and FABADA4 routines of FP DEDB Pointer Checker to obtain the details of the error, then analyze the error from the generated reports.

FPX0121I **OAE replaced ROOT= parameter from (aaaaa,bbbb) to (cccc,ffff)**

Explanation

The ROOT= control keyword was specified and OAE replaced ROOT= parameter of DBD definition in ACBLIB.

System action

Processing continues.

User response

None. This message is informational.

FPX0122W **Incorrect OLRECL xxxxx specification. Default value xxxxx substituted.**

Explanation

FPXOFIL LRECL is too large. It must be less than or equal to BLKSIZE-4.

System action

ODE determines the default value for FPXOFIL LRECL as BLKSIZE-4 (OLRECL specification is ignored). ODE sets an end-of-job return code of 4 and continues processing.

User response

To eliminate this warning message for subsequent runs of the job, correct the OLRECL value.

FPX0123E **module is not loaded. Abend code: Sxxxx reason code: yyyy.**

Explanation

module has not been loaded. *module* is one of the following modules:

- Compression routine *compression_routine*
- DBRC module DSPURXRT
- Access Method Service program IDCAMS

System action

If it is a compression routine, processing ends with a return code of 8. If it is a DBRC module DSPURXRT or an Access Method Service program IDCAMS, processing ends with an abend code of U3521.

User response

Correct the incompatible specification, and resubmit the job. If the compression routine is not loaded, the

IMS HP Image Copy (HPIC) library might not be concatenated to the STEPLIB library.

FPX0124I **(One of the texts listed in Explanation)**

Explanation

A discrepancy was encountered during pointer checking. This message contains one of the following text that describes the discrepancy encountered:

- CI allocation information was discrepant between the IOVF CI and the IOVF Space Map element.

These messages might be accompanied by additional data to help you identify the discrepancy:

- RBA of IOVF space map CI: *nnnnnnnn*
- RBA of target IOVF CI: *nnnnnnnn*
- CI prefix: *nnnnnnnn*
- Space map element offset: *nnnnnnnn*
- Space map element: *nnnnnnnn*

System action

Processing continues.

User response

This message is informational. If necessary, consult your database administrator to correct the problems with the database.

FPX0125I **OAE created a temporary ADS. DDNAME *ddname*, DSNAME *dsname***

Explanation

A temporary ADS was created dynamically for extension of the original ADS.

System action

Processing continues.

User response

None. This message is informational.

FPX0125I **OAE deleted a temporary ADS. DDNAME *ddname*, DSNAME *dsname***

Explanation

The temporary ADS that was created dynamically was deleted.

System action

Processing continues.

User response

None. This message is informational.

FPX0126I **An ADS was registered with DBRC.**
DDNAME *ddname*, DSNAME
dsname

Explanation

The temporary ADS that was created dynamically was registered with DBRC for processing.

System action

Processing continues.

User response

None. This message is informational.

FPX0126I **An ADS was deleted with DBRC.**
DDNAME *ddname*, DSNAME
dsname

Explanation

The temporary ADS that was created dynamically was deleted with DBRC for processing.

System action

Processing continues.

User response

None. This message is informational.

FPX0127E **DUMMY or NULLFILE is specified**
for *ddname* DD statement.

Explanation

DUMMY or NULLFILE is not allowed for the *ddname* DD statement.

System action

Processing ends with a return code of 8.

User response

Correct the specified problem and rerun the job.

FPX0128I ***parameter* parameter of FPXIN DD**
is ignored.

Explanation

The SMS parameter, DATACLAS, STORCLAS or MGMTCLAS are specified in the non-SMS environment.

System action

Processing continues.

User response

None. This message is informational.

FPX0129I **OAE deleted and created an**
original ADS by extension.

Explanation

The original ADS was extended by using the specified parameters.

System action

Processing continues.

User response

None. This message is informational.

FPX0130E **Dynamic allocation failed for**
(cont) DDNAME: *ddname*,
DSNAME: *dsname*
(cont) RC: *rc*, RSN:
S99ERROR(*eeee*), S99INFO(*iiii*),
S99ERSN(*rrrr*).

Explanation

Dynamic allocation for *ddname* and *dsname* failed. *rc* is the return code from SVC99. *eeee* is the S99 ERROR contents. *iiii* is the S99 INFO contents. *rrrr* is the ERROR reason.

System action

Processing ends with an abend code of U3507.

User response

Look up the dynamic allocation (SVC99) code in the *MVS Programming: Assembler Services Reference*. Correct the problem, then rerun the job.

FPX0131I **Status of original ADS was**
changed into AVAIL. DDNAME:
***ddname*, AREA: *areaname*,**
DBDNAME: *dbdname*

Explanation

Program FPXBTOA0 renamed the temporary ADS to the original one and made it available with DBRC successfully.

System action

FPXBTOA0 ends with a return code of 0.

User response

None. This message is informational.

FPX0132I LISTCAT information of the area data set before extending:
(cont) - AREA: *areaname*
(cont) - ADSDD: *ddname* ADSDS:
dsname

Explanation

This message shows the LISTCAT report of the original area data set before extension.

System action

Processing continues.

User response

None. This message is informational. This message provides you information when you run OAE with the RESTART option.

FPX0133E One or both of *ddname* specified by DDNAME parameter and *dsname* specified by DSNNAME parameter are already registered with DBRC.
(cont) - AREA: *areaname*
(cont) - ADSDDN(registered): *ddname*
(cont) - ADSDSN(registered): *dsname*

Explanation

One or both of the *ddnames* specified by the DDNAME parameter and data set name specified by the DSNNAME parameter are already registered with DBRC.

System action

Processing ends with a return code of 8.

User response

Specify *ddname* or *dsname* which is not registered with DBRC. Correct the specified problem and rerun the job.

FPX0134E Either IOVF_UOW, IOVF_ROOT, or SDEP_CI keyword must be specified.

Explanation

None of the keywords IOVF_UOW, IOVF_ROOT, and SDEP_CI is specified in the FPXIN DD statement. One of these keywords must be specified for extension.

System action

Processing ends with a return code of 8.

User response

Specify the IOVF_UOW keyword, the IOVF_ROOT keyword, or the SDEP_CI keyword.

FPX0135E The number of [VOLUME | VOLUME_EXT] keyword limit exceeded.

Explanation

More than 59 volumes are specified for a temporary area data set or an extended original area data set. The limit is 59 or fewer.

System action

Processing ends with a return code of 8.

User response

Correct the specification, and resubmit the job.

FPX0136W The OAE restart process extended the temporary ADS. It did not copy the temporary ADS back to the original ADS.

Explanation

The OAE restart process extended the temporary ADS, but it did not copy the temporary ADS back to the original ADS. All area data sets except the temporary area data set are unavailable with DBRC.

System action

Processing ends with a return code of 4.

User response

See “Recovering and restarting OAE” on page 758 for detailed procedure to complete the OAE.

FPX0137I **OAE RESTART process determines the original area data set with the DBRC available ADS.**
(cont) - AREA: *areaname*
(cont) - AVAIL ADS
(cont) - ADSDD: *ddname* ADSDS:
dsname
(cont) - ADSDD: *ddname* ADSDS:
dsname
:
(cont) - UNAVAIL ADS
(cont) - ADSDD: *ddname* ADSDS:
dsname
:

Explanation

OAE restart process determines the original area data set from the DBRC. The temporary area data set is reported as one of the available ADSs. FPX0125I and FPX0126I messages will follow this message to tell that the temporary area data set is deleted with DBRC, and the temporary area data set cluster is deleted as a result of the OAE restart process.

System action

Processing continues.

User response

None. This message is informational.

FPX0138E **Incorrect RC received from Access method service process called by FPXBTOA0.**

Explanation

Access method service process called by OAE or by FPXBTOA0 ended abnormally.

System action

Processing ends with an abend code of U3519.

User response

See the message issued by access method service.

FPX0139E **Fatal DBRC error occurred during OAE.**

Explanation

DBRC process called by OAE ended abnormally.

System action

Processing ends with an abend code of U3519.

User response

See the message issued by DBRC.

FPX0140E **Mandatory keyword keyword is not specified in FPXIN DD.**

Explanation

The NAME, DDNAME, or AREA control keywords were not specified for the OAE utility. All of these keywords must be specified.

System action

Processing ends with a return code of 8.

User response

Specify both the NAME and DDNAME control keywords.

FPX0141E **Extended ADS size specified with IOVF_UOW, IOVF_ROOT, or SDEP_CI keyword exceeds the maximum ADS size.**

Explanation

The extended ADS size specified with the IOVF_UOW, IOVF_ROOT, or SDEP_CI keyword exceeds the maximum ADS size, which is 4 GB.

System action

Processing ends with a return code of 8.

User response

Correct the specification, and resubmit the job.

FPX0142E **AREA=ALL is not allowed for xxxxxxxx control statement in FPXIN DD.**

Explanation

AREA=ALL for xxxxxxxx control statement in FPXIN DD is not supported.

System action

Processing ends with a return code of 8.

User response

Correct the AREA parameter value.

FPX0143E	No extension specified with SDEPCI=0 and no IOVF_ROOT in FPXIN DD.
-----------------	---

Explanation

Extension size is not specified. SDEPCI=0 without the IOVF keyword is specified.

System action

Processing ends with a return code of 8.

User response

Specify the IOVF parameter or correct the SDEPCI= value.

FPX0144E	FPXIN command must be specified at first except comment statements.
-----------------	--

Explanation

A keyword is found instead of a command. Only the comment statements can precede the command in the FPXIN DD statement.

System action

Processing ends with a return code of 8.

User response

Correct the control statements, and rerun the job.

FPX0145E	FPXIN DD and SYSIN DD are mutually exclusive.
-----------------	--

Explanation

Both the FPXIN DD and SYSIN DD statements were specified in the FPXIN DD statement for the OAE utility. These statements are mutually exclusive.

System action

Processing ends with a return code of 8.

User response

Correct the control statements, and rerun the job.

FPX0146I	DEDB UTILITY xxxxxxxx STARTED, DB dbname AREA areaname
-----------------	---

Explanation

This is the same message as message DFS2601I of IMS.

System action

Processing continues.

User response

See *IMS Messages and Codes*.

FPX0147E	INVALID VALIDATION CALL, CODE=xx
-----------------	---

Explanation

This is the same message as message DFS2661A of IMS.

System action

Processing ends with a return code of 8.

User response

See *IMS Messages and Codes*.

FPX0148E	UTILITY STOPPED PRIOR TO COMPLETION DUE TO STOP REGION OR SYSTEM TERMINATION COMMAND
-----------------	---

Explanation

This is the same message as message DFS2688A of IMS.

System action

Processing ends with a return code of 4.

User response

See *IMS Messages and Codes*.

FPX0149E	CREATE UTILITY FAILED DUE TO IN-DOUBT SEGMENT IN AREA DATA SET
-----------------	---

Explanation

This is the same message as message DFS3140A of IMS.

System action

Processing ends with a return code of 8.

User response

See *IMS Messages and Codes*.

FPX0150E **VALIDATION CALL FAILED FOR
AREA=areaname DD=ddname
CODE=xx**

Explanation

This is the same message as message DFS3736A of IMS.

System action

Processing ends with a return code of 8.

User response

See *IMS Messages and Codes*.

FPX0151E **Incorrect FPXBINTP file was
specified. incorrect interface item**

Explanation

Program FPXBTOA0 found that there was at least one of the following incorrect interface items in the FPXBINTP file:

- Used FPXBINTP file
- Incorrect ID
- Incorrect OAE level
- Incorrect DBDNAME, AREANAME, or IMSID
- Incorrect temporary ADS information
- Incorrect original ADS information
- Exceeded the FPXBINTP creation time limit, which is 3 minutes
- No information data
- Incorrect creation date or time

Subsequent message FPX0187I reports the content of the interface information in the FPXBINTP file when interface item is not "Incorrect ID" or "No information data".

System action

FPXBTOA0 ends with an abend code of U3531.

User response

Specify the correct FPXBINTP file, and rerun the job.

FPX0152E **AREA areaname DOES NOT EXIST**

Explanation

This is the same message as message DFS2650A of IMS.

System action

ODM continues the process with a return code of 4, other utilities end the process with a return code of 8.

User response

See *IMS Messages and Codes*.

FPX0153E **OTHER UTILITY ACTIVE IN AREA
areaname**

Explanation

This is the same message as message DFS2651A of IMS.

System action

ODM continues the process with a return code of 4, other utilities end the process with a return code of 12.

User response

See *IMS Messages and Codes*.

FPX0154E **CHANGE OF UTILITY FOR THIS
AREA**

Explanation

This is the same message as message DFS2654A of IMS.

System action

ODM continues the process with a return code of 4, other utilities end the process with a return code of 28.

User response

See *IMS Messages and Codes*.

FPX0155E **OPEN FAILED FOR
AREA=areaname**

Explanation

This is the same message as message DFS2655A of IMS.

System action

ODM continues the process with a return code of 4, other utilities end the process with a return code of 20.

User response

See *IMS Messages and Codes*.

FPX0156E INVALID DISCONNECT CALL

Explanation

This is the same message as message DFS2656A of IMS. The message might have been issued after the FPO process for the area had been completed.

System action

Processing ends with a return code of 8.

User response

See *IMS Messages and Codes*. If FPX0055I is issued before this message, the ODE process for the area has been processed as expected.

FPX0157E WRONG PROGRAM NAME

Explanation

This is the same message as message DFS2658A of IMS.

System action

ODM continues the process with a return code of 4, other utilities end the process with a return code of 28.

User response

See *IMS Messages and Codes*.

FPX0158E ROOT SEGMENT HAS NO SEQUENTIAL DEPENDENT SEGMENTS

Explanation

This is the same message as message DFS2660A of IMS.

System action

ODM continues the process with a return code of 4, other utilities end the process with a return code of 8.

User response

See *IMS Messages and Codes*.

FPX0159E XX STATUS FROM I/O PROCESS IN AREA

Explanation

This is the same message as message DFS2667A of IMS.

System action

ODM continues the process with a return code of 4, other utilities end the process with a return code of 20.

User response

See *IMS Messages and Codes*.

FPX0160E GET AREA LOCK FAILED, AREA=areaname

Explanation

This is the same message as message DFS2669A of IMS.

System action

ODM continues the process with a return code of 4, other utilities end the process with a return code of 28.

User response

See *IMS Messages and Codes*.

FPX0161E GETMAIN FAILED, AREA=areaname

Explanation

This is the same message as message DFS2687A of IMS.

System action

ODM continues the process with a return code of 4, other utilities end the process with a return code of 28.

User response

See *IMS Messages and Codes*.

FPX0162E FAILED TO CONNECT THE AREA

Explanation

This is the same message as message DFS2712A of IMS.

System action

ODM continues the process with a return code of 4, other utilities end the process with a return code of 16.

User response

See *IMS Messages and Codes*.

FPX0163E	UNABLE TO OBTAIN AUTHORIZATION FOR DATA BASE=xxxxxxx AREA=areaname. REASON CODE=xx
-----------------	---

Explanation

This is the same message as message DFS3710A of IMS.

System action

ODM continues the process with a return code of 4, other utilities end the process with a return code of 12.

User response

See *IMS Messages and Codes*.

FPX0164E	DMAC SYNCHRONIZATION FAILED FOR AREA=areaname
-----------------	--

Explanation

This is the same message as message DFS3737A of IMS.

System action

ODM continues the process with a return code of 4, other utilities end the process with a return code of 24.

User response

See *IMS Messages and Codes*.

FPX0165E	UTILITY=xxxxxxx TERMINATED BECAUSE AREA=areaname STOPPED
-----------------	---

Explanation

This is the same message as message DFS3754A of IMS.

System action

ODM continues the process with a return code of 4, other utilities end the process with a return code of 20.

User response

See *IMS Messages and Codes*.

FPX0165E	UTILITY=xxxxxxx SKIPPED PROCESSING FOR AREA=areaname BECAUSE AREA IS STOPPED
-----------------	---

Explanation

This is the same message as message DFS3754A of IMS.

System action

ODM continues the process with a return code of 4, other utilities end the process with a return code of 20.

User response

See *IMS Messages and Codes*.

FPX0166E	AREA=xxxxxxx NOT REGISTERED IN RECON D/S
-----------------	---

Explanation

This is the same message as message DFS3760A of IMS.

System action

ODM continues the process with a return code of 4, other utilities end the process with a return code of 20.

User response

See *IMS Messages and Codes*.

FPX0167E	The name of the IMS Tools KB server XCF group must be specified on the ITKBSVR keyword when [SENSOR=YES ITKBDATA=UTILHIST] is specified.
-----------------	---

Explanation

When you specify SENSOR=YES or ITKBDATA=UTILHIST, you must also specify the name of the IMS Tools KB server.

System action

Processing ends with a return code of 8.

User response

If you want to store the sensor data or the utility history member in the IMS Tools KB repository, specify

the name of the IMS Tools KB server XCF group on the ITKBSRVR keyword and resubmit the job.

FPX0168E [Original | Temporary] ADS could not be [created | deleted].
DDNAME: adsdd DSNAME: dsname

Explanation

Failed to create or delete the original ADS or the temporary ADS.

System action

Processing ends with a return code of 8 or with an abend code of 3519.

User response

Check the DSNAME, VOLUMES, STORCLAS, MGMTCLAS, or DATACLAS keyword value in the FPXIN DD statement. If the process ends with an abend code of U3519, see the message of IDCAMS, and remove the cause.

FPX0169E [Original | Temporary] ADS
DDNAME: ddname DSNAME: dsname
could not be stopped.

Explanation

Failed to stop the original ADS or the temporary ADS.

System action

Processing ends with abend code 3518.

User response

Specify the cause from the status code shown in FPX0011E, and correct the error.

FPX0170E Maximum number of area data sets has been reached in DBRC.
AREA=areaname

Explanation

An EXTEND command was specified for the area for which the maximum number of the area data set has been reached in DBRC. Up to seven area data sets can be defined for one area. The total number of area data sets in DBRC exceeds the maximum number, which is seven. The temporary ADS could not be registered.

System action

Processing ends with a return code of 8.

User response

Make sure that there are seven area data sets already registered, unregister one of unavailable area data set to make space for a temporary area data set which will be used for OAE, and rerun the job.

FPX0171W -> Warning:
Possible loop in segment twin chain.
Source SC seg-code Name seg-name RBA x'xxxxxxxx'
Target SC seg-code Name seg-name RBA x'xxxxxxxx'
Root SC seg-code Name seg-name RBA x'xxxxxxxx'
Root key value:
xxxxxxxxxxxx
Maximum 255 bytes of contents of source segment:
xxxxxxxxxxxx

Explanation

OPC detected that the number of twin segments exceeded the threshold value that is defined by the MAXTWINCTR keyword. The default value for the MAXTWINCTR keyword is 2000000. OPC assumes that the twin chain might be looped.

System action

OPC sets an end-of-job return code of 4 and continues processing, but stops checking the next twin segments.

User response

Check the CI SNAP dump in the SNAPPIT data set to see if the area does have the wrong pointer that is causing the twin chain loop, or the area does have twin chains more than the threshold value. If the number of twin segments exceeds the threshold value, specify a larger value for the MAXTWINCTR keyword and rerun the job.

FPX0172E Failed to load module: xxxxxxxx (Sccc-rr).

Explanation

FPO failed to load a module. xxxxxxxx shows the name of the module that failed to be loaded, ccc is the system completion code, and rr is the reason code associated with the system completion code that is returned from the LOAD macro.

System action

If xxxxxxxx is an FPO control block, user abend 3522 occurs. If xxxxxxxx is an FPO module, user abend 3525 occurs. Otherwise, processing ends with a return code of 8.

User response

See *MVS System Codes*, and take an appropriate action.

FPX0173I *process_name* process started.

Explanation

The Online Pointer Checker (OPC), Online Data Extract (ODE), Online Area Extender (OAE), Online Expert Reorganization (OER), or Online DMAC Print (ODM) process started.

System action

Processing continues.

User response

None. This message is informational.

FPX0174I *proces_name* process ended with
RC=00. (Elapsed Time:
hh:mm:ss.tt)

Explanation

The Online Pointer Checker (OPC), Online Data Extract (ODE), Online Area Extender (OAE), Online Expert Reorganization (OER), or Online DMAC Print (ODM) process ended normally. Elapsed Time shows the time elapsed for the entire processing.

System action

Processing continues.

User response

None. This message is informational.

FPX0175E *process_name* process ended with
RC=nn. (Elapsed Time:
hh:mm:ss.tt)

Explanation

The Online Pointer Checker (OPC), Online Data Extract (ODE), Online Area Extender (OAE), Online Expert Reorganization (OER), or Online DMAC Print (ODM)

process ended with an error. Elapsed Time shows the time elapsed for the entire processing.

System action

Processing ends with a return code of *nn*.

User response

Locate FPXnnnnW and HFPnnnnW messages or review the reports that were generated by the utility to determine the cause of the error. Correct the error, and rerun the job.

FPX0176E **The FPO utility could not process
because no areas were available.**

Explanation

FPO tried to connect to an area that is defined in the DEDB to collect database information but all of the areas were in unavailable status.

System action

Processing ends with a return code of 8.

User response

Correct the error, and rerun the job.

FPX0177W **Area xxxxxxxx is stopped. Process
for the area will be bypassed.**

Explanation

The area specified in the control statement is stopped.

System action

Process for the area is bypassed and processing continues with a return code of 4.

User response

This is a warning message. If the result is not in an expected condition, issue the /STA AREA command to start the area, and rerun the job.

FPX0178E **[ICEXTEND | ICEXTEND=YES] is
not supported by the current IMS
APAR level.**

Explanation

The current maintenance level of IMS does not support the ICEXTEND keyword for IOVF extensions. To have the ICEXTEND keyword in effect for IOVF

extensions, you must be on IMS 14 with APAR PI43928 or on IMS 15 or later.

System action

Processing ends with a return code of 8.

User response

If you want to specify the ICEXTEND keyword for an IOVF extension, apply APAR PI12801 to IMS and then resubmit the job. If the problem persists even after the APAR is applied, contact IBM Software Support.

FPX0181I **The *process_name* process for AREA name = *areaname* completed successfully.**

Explanation

The ANALYZE, EXTRACT, REORG, or DMACPRT process for the indicated area ended.

System action

Processing continues.

User response

None. This message is informational.

FPX0182W **The *process_name* process for AREA name = *areaname* ended with warnings.**

Explanation

The ANALYZE, EXTRACT, REORG, or DMACPRT process for the indicated area ended with a warning.

System action

Processing ends with a return code of *nn*.

User response

Locate FPX*nnnn*W and HFP*nnnn*W messages or review the reports that were generated by the utility to determine the cause of the error. If this is not the expected result, correct the error, and rerun the job.

FPX0183E **The *process_name* process for AREA name = *areaname* ended with errors.**

Explanation

The ANALYZE, EXTRACT, REORG, or DMACPRT process for the indicated area ended with errors.

System action

Processing ends with a return code of 8.

User response

Locate FPX*nnnn*W and HFP*nnnn*W messages or review the reports that were generated by the utility to determine the cause of the error. Correct the error, and rerun the job.

FPX0184E ***process_name* process ended abnormally.**

Explanation

The Online Pointer Checker (OPC), Online Data Extract (ODE), Online Expert Reorganization (OER), or Online DMAC Print (ODM) process ended abnormally.

System action

Processing ends with a system or user abend code.

User response

Locate FPX*nnnn*W and HFP*nnnn*W messages or review the reports that were generated by the utility to determine the cause of the error. Correct the error, and rerun the job.

FPX0185W ***process_name* process ended with RC=*nn*. (Elapsed Time: *hh:mm:ss.tt*)**

Explanation

The Online Pointer Checker (OPC), Online Data Extract (ODE), Online Area Extender (OAE), Online Expert Reorganization (OER), or Online DMAC Print (ODM) process ended with a warning. Elapsed Time shows the time elapsed for the entire processing.

System action

Processing ends with a return code of *nn*.

User response

Locate FPX*nnnn*W and HFP*nnnn*W messages or review the reports that were generated by the utility to determine the cause of the error. If this is not the expected result, correct the error and rerun the job.

FPX0186E **Number of total CIs per UOW exceeded the maximum value *nnnn*.**

Explanation

The AREA definition for the number of CIs for each UOW that exceeds *nnnn* is not valid because the IMS buffer set number is a half word.

System action

Processing ends with a return code of 8.

User response

Correct the AREA definition, and rerun the job.

FPX0187I **OAE interface parameter file was generated as follows:
interface data**

Explanation

The message shows the contents of the FPXBINTP DD data set generated by OAE.

System action

Processing continues.

User response

None. This message is informational.

FPX0188E **Load module library is not APF-authorized.**

Explanation

Program FPXBTOA0 found that the load module library data set that was specified on the JOBLIB/STEPLIB DD statement was not authorized by APF.

System action

FPXBTOA0 ends with an abend code of U3527.

User response

Make the load module library APF-authorized for future processing, and rerun the job.

FPX0189E **FPXIN control statement error.
No keywords specified
Unknown keyword
Invalid xxxxxxxx= control
statement
xxxxxxx= control statement
previously processed**

Explanation

Program FPXBTOA0 found an invalid control statement on the FPXIN DD statement.

System action

FPXBTOA0 ends with a return code of 8.

User response

Correct or remove the specified control statement, and rerun the job.

FPX0190E **TOSI API failed. FUNCTION:
xxxxxxx, RC: rc, RSN: rsn**

Explanation

TOSI API returned an abnormal return code.

System action

Processing ends with an abend code of U3528.

User response

Look up the RC and RSN. Correct the problem.

FPX0191E **TOITIME time over. REQUEST
command=xxxxxxx**

Explanation

Exceeded the specified running time set for TOSI REQUEST.

System action

Processing ends with an abend code of U3529.

User response

Follow the description in [“Recovering and restarting OAE”](#) on page 758.

FPX0192I **AREA stopped. AREA: areaname,
IMSID: imsid**

Explanation

The area stopped and was deallocated from the *imsid* IMS online system to delete the VSAM cluster of the original ADS and to rename the temporary ADS to the original one. This message is issued per IMS system that opens the area.

System action

Processing continues.

User response

None. This message is informational.

FPX0193I **Original ADS deleted. DDNAME:**
ddname, DSNAME: dsname

Explanation

The VSAM cluster of the original ADS was deleted in order to rename the temporary ADS to the original one.

System action

Processing continues.

User response

None. This message is informational.

FPX0194I **Temporary ADS renamed as**
original ADS (RENAME target).
DDNAME: ddname
CLUSTER NAME: cluster name
DATA NAME: data name

Explanation

The temporary ADS was renamed to the original one.

System action

Processing continues.

User response

None. This message is informational.

FPX0195I **The following IMS command**
issued:
content of the command

Explanation

Program FPXBTOA0 issued the IMS command through the IMS Tools Online System Interface.

System action

Processing continues.

User response

None. This message is informational.

FPX0197I **Active IMS list as follows:**
imsid
:

Explanation

This is the list of IMS systems of the IMSPLEX group to which the IMS Tools Online System Interface is connected.

System action

Processing continues.

User response

None. This message is informational.

FPX0198I **/RMxxxxxx command completed.**

Explanation

The IMS /RM command that program FPXBTOA0 issued completed successfully.

System action

Processing continues.

User response

None. This message is informational.

FPX0199E **/RMxxxxxx command failed with**
RC=xx

Explanation

The IMS /RM command that program FPXBTOA0 issued failed with the highest return code xx.

System action

Processing ends with an abend code of U3526.

User response

See the explanation for the DBRC message DSPnnnnI that precedes this message.

FPX0200I **Temporary ADS in RECON is**
deleted. DDNAME: ddname, AREA:
areaname, DBDNAME: dbdname

Explanation

The temporary ADS indicated by *ddname* was deleted from the RECON record structure.

System action

Processing continues.

User response

None. This message is informational.

FPX0201E **SORT ended with internal error.
SORT may issue the message to
SYSOUT DD. SORT return code is:
nnnn**

Explanation

ODE ended with an error during the internal sort process. *nnnn* shows the hexadecimal SORT return code. SORT might also issue the message to SYSOUT DD.

System action

Processing ends with an abend code of 3504.

User response

For further information, see SYSOUT DD in “DD statements for OPC” on page 685. Correct errors, and rerun the job.

FPX0202I **Site default options are specified.**

Explanation

Site default options are specified.

System action

Processing continues.

User response

None. This message is informational.

FPX0203I **xxx processing options in effect:**

Explanation

The keyword values that follow this message are the processing options in effect. *xxx* shows the tool name.

System action

Processing continues.

User response

None. This message is informational.

FPX0204E **PDS processing failed for
xxxxxxx: FUNC=yyyyyy, RC=nnnn,
RSN=nnnn.**

Explanation

The accompanying text identifies the following information associated with the PDS processing failure.

xxxxxxx shows:

- DDNAME: OPEN/GETDSN/CLOSE error
- member name: FIND error

yyyyyy shows the function that failed, which will be one of the following functions:

- OPEN: open a PDS
- FIND: find a member in the PDS
- GETDSN: get the data set name of the PDS where the current member resides

RC=*nnnn* and RSN=*nnnn* are as follows:

- FUNCTION=OPEN RC=0004,RSN=0000: DD statement no found.
- FUNCTION=FIND RC=0004,RSN=0000: Member not found.

If you receive other return codes or reason codes, see the user response section.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the specification, and resubmit the job. The return code or the reason code that is not in the preceding list is for an internal error. If such a return code or reason code is returned, contact IBM Software Support.

FPX0205E **Site default options cannot be
specified for keywords of SELECT
and OUTPUT control statements.**

Explanation

For the ODE site default options, only the FPXCTL control statement can be specified.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the specification, and resubmit the job.

FPX0206I **Global level site default values have been read from the following member: PROCLIB: *dsn* MEMBER: *member name***

Explanation

This message informs that the global level site default values were read from the PROCLIB member *member name*. *dsn* shows the data set in which the member was found.

System action

Processing continues.

User response

None. This message is informational.

FPX0207I ***xxxx*-level site default values have been read from the following member: *yyy*PARMLIB: *dsn* MEMBER: *mbr_name***

Explanation

This message informs the data set name (*dsn*) of the parameter library, and the name of the area-level site default member (*mbr_name*) or the DB-level site default member. *xxxx* shows the level of site default options, which is AREA or DB. *yyy* shows the tool name.

System action

Processing continues.

User response

None. This message is informational.

FPX0208I **Keywords are specified as follows:**

Explanation

The keyword values that follow this message are those specified in JCL, AREA/DB level site default options, global level site default options, and the system default.

System action

Processing continues.

User response

None. This message is informational.

FPX0209I **See message FPX0049I for the following control statements:**

Explanation

This message is issued when a SELECT, an OUTPUT, a THRESHOLD, or a USER_RECORD control statement is specified.

System action

Processing continues.

User response

None. This message is informational.

FPX0210I **Parameter analysis for the global site default options started ...**

Explanation

This message informs the start of the parameter analysis for global level site default options.

System action

Processing continues.

User response

None. This message is informational.

FPX0211I **Parameter analysis for the AREA/DB level site default options started ...**

Explanation

This message informs the start of the parameter analysis for AREA or DB level site default options.

System action

Processing continues.

User response

None. This message is informational.

FPX0212I **Parameter analysis for control statements specified in the JCL stream started ...**

Explanation

This message informs the start of the parameter analysis for the control statements specified in the JCL stream.

System action

Processing continues.

User response

None. This message is informational.

FPX0213E **DYNALLOC failed for DSNAME:**
dsname, MVS RET. RSN codes are:
rc + rsn

Explanation

Dynamic allocation failed for *dsname*. *rc* is the return code from SVC99 and *rsn* is the S99 ERROR contents.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the specification, and resubmit the job.

FPX0214I **Processing ends because of**
PCHKONLY=YES.

Explanation

Processing ends after evaluating the site default options and the utility control statements because PCHKONLY=YES was specified.

System action

Processing continues.

User response

None. This message is informational.

FPX0215E **Length error for xxxxxxxxxxxx**
detected in FPXCFG00.

Explanation

There is a length error in keyword xxxxxxxxxxxx.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the specification, and resubmit the job.

FPX0216E **Incorrect specification detected in**
FPXCFG00.

Explanation

Incorrect control keyword is specified in FPXCFG00.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the specification, and resubmit the job.

FPX0217I **Starting keyword analysis: AREA**
name = areaname, AREA
number=area#

Explanation

The keyword analysis has started for *areaname* that has the area number of *area#*. For OPC with FPXIN DD, ODE in run-level control, OER and ODM, the AREA name and the AREA number are not printed.

System action

Processing continues.

User response

None. This message is informational.

FPX0218I **Completed keyword analysis:**
AREA name = areaname, AREA
number=area#

Explanation

The keyword analysis has completed for *areaname*, which has the area number of *area#*. For OPC with FPXIN DD, ODE in run-level control, OER and ODM, the AREA name and the AREA number are not printed.

System action

Processing continues.

User response

None. This message is informational.

FPX0219I **AREA level site default values are**
not read because of run-level
control.

Explanation

This message informs that the AREA level site default values are not processed because of the run-level control.

System action

Processing continues.

User response

None. This message is informational.

FPX0220E *keyword1=value1 and keyword2=value2 are mutually exclusive.*

Explanation

The *keyword1=value1* is mutually exclusive with *keyword2=value2*.

System action

Processing ends with a return code of 8.

User response

Correct the incompatible specification, and resubmit the job.

FPX0221E **AREA must be specified in FPXIN.**

Explanation

AREA is not specified in FPXIN.

System action

Processing ends with a return code of 8.

User response

Specify AREA with the area name that needed to be reorganized, and resubmit the job.

FPX0222E **IC was not requested, but IC1OUTDD or IC2OUTDD was specified.**

Explanation

An image copy was not requested in the utility control statements, but image copy data set IC1OUTDD or IC2OUTDD was specified.

System action

Processing ends with a return code of 8.

User response

Correct the incompatible specification, and resubmit the job.

FPX0223E **PDS processing failed for site default.**

Explanation

The accompanying FPX0204E message identifies the detailed PDS processing failure.

System action

Processing ends with a return code of 8.

User response

Correct the specification, and resubmit the job.

FPX0224E **Site default options cannot be specified for the keyword keyword.**

Explanation

The indicated keyword can be specified only in FPXIN control statements.

System action

Processing ends with a return code of 8.

User response

Remove the indicated keyword from the site default, and resubmit the job.

FPX0225E **Global site default options exceed nnnn lines.**

Explanation

The global site default options exceed the maximum number of lines.

System action

Processing ends with a return code of 8.

User response

Correct the specification, and resubmit the job.

FPX0226E **Area/DB site default options exceed nnnn lines.**

Explanation

The area or the DB site default options exceed the maximum number of lines.

System action

Processing ends with a return code of 8.

User response

Correct the specification, and resubmit the job.

FPX0228E	Errors were detected in XXXXXXXX parameter analysis for global site default.
-----------------	---

Explanation

Errors were detected during the parameter analysis for the global site default options. Messages that follow explain the details of the errors.

System action

Processing ends with a return code of 8.

User response

Correct the specification, and resubmit the job.

FPX0229E	Errors were detected in XXXXXXXX parameter analysis for area or DB site default.
-----------------	---

Explanation

Errors were detected during the parameter analysis for the DB-level or the area-level site default options. Messages that follow explain the details of the errors.

System action

Processing ends with a return code of 8.

User response

Correct the specification, and resubmit the job.

FPX0230E	Errors were detected in XXXXXXXX parameter analysis for FPXIN JCL stream.
-----------------	--

Explanation

Errors were detected during parameter analysis for FPXIN JCL stream. Messages that follow explain the details of the errors.

System action

Processing ends with a return code of 8.

User response

Correct the specification, and resubmit the job.

FPX0231E	A syntax error was detected in the global site default.
-----------------	--

Explanation

A syntax error was detected in the global site default. Messages that follow show reasons of the error in detail.

System action

Processing ends with a return code of 8.

User response

Correct the specification, and resubmit the job.

FPX0232E	A syntax error was detected in area or DB site default.
-----------------	--

Explanation

A syntax error was detected in the area or the DB site default. Messages that follow show reasons of the error in detail.

System action

Processing ends with a return code of 8.

User response

Correct the specification, and resubmit the job.

FPX0233E	A syntax error was detected in FPXIN JCL stream.
-----------------	---

Explanation

A syntax error was detected in FPXIN JCL stream. Messages that follow shows reasons of the error in detail.

System action

Processing ends with a return code of 8.

User response

Correct the specification, and resubmit the job.

FPX0234I **The following control statements were specified in //FPXIN:**

Explanation

This message is accompanied by message FPX0049I. See message FPX0049I for the complete control statements specified in the JCL stream.

System action

Processing continues.

User response

None. This message is informational.

FPX0235E **The site default configuration in FPXCFG00 must start in column 1.**

Explanation

The site default configuration specified in FPXCFG00 must start in column 1.

System action

Processing ends with a return code of 8, and the ODE run-level control using SYSIN and FPXCTL DD ends with a return code of 12.

User response

Correct the incorrect specification, and resubmit the job.

FPX0236E **Duplicate value: xxxxxxxx is specified in the xxxxxxxx keyword.**

Explanation

Duplicate value is specified for the keyword. This type of keyword cannot be specified more than once.

System action

Processing ends with a return code of 8.

User response

Correct the error, and rerun the job.

FPX0237E **EXTRACT is not supported by the current IMS APAR level.**

Explanation

The current IMS APAR level is below PK25289 (IMS V8) or below PK25294 (IMS V9).

System action

Processing ends with a return code of 8.

User response

Install the IMS APAR, and resubmit the job. If the APAR is installed and the problem occurs, contact IBM Software Support.

FPX0238E **Duplicate specification: ALL and area names.**

Explanation

ALL and area names are mutually exclusive as the parameter of the AREA keyword.

System action

Processing ends with a return code of 8.

User response

Correct the error, and rerun the job.

FPX0239W **SDEP_ORPHAN_MSGLVL is ignored because SDEP=NO was specified.**

Explanation

If no SDEP pointer integrity checking is specified, the SDEP_ORPHAN_MSGLVL keyword is ignored.

System action

Processing continues.

User response

None required.

FPX0240E **The first non-comment line of a site default member must start with the xxxxxxxx command**

Explanation

The command indicated by xxxxxxxx must be specified in the first non-comment line of a site default member for the utility.

System action

Processing ends with a return code of 8.

User response

Correct the control statements specified in the site default member, and resubmit the job.

FPX0241E **The error is at the position marked with '*' or in the preceding keyword parameter.**

Explanation

The FPXIN syntax error was found at the position marked with '*' or in the preceding keyword parameter.

System action

Processing ends with a return code of 8.

User response

Correct the error, and rerun the job.

FPX0242E **The following comment line is specified in parentheses.**

Explanation

The comment line at the position marked with '*' is specified in parentheses.

System action

Processing ends with a return code of 8.

User response

Correct the error, and rerun the job.

FPX0243E **The right parenthesis that corresponds to the left parenthesis is not found.**

Explanation

The right parenthesis is specified for the value at the position marked with '*' but the left parenthesis for the value is not found.

System action

Processing ends with a return code of 8.

User response

Correct the error, and rerun the job.

FPX0244E **A comma or a blank is expected at the position marked with '*'.**

Explanation

A comma or a blank must be specified after the right parenthesis.

System action

Processing ends with a return code of 8.

User response

Correct the error, and rerun the job.

FPX0245E **The value for a keyword parameter must immediately follow the equal sign.**

Explanation

The value for a keyword parameter must immediately follow the equal sign.

System action

Processing ends with a return code of 8.

User response

Correct the error, and rerun the job.

FPX0246E **A value, not null, must be specified within the parentheses.**

Explanation

A value, not null, must be specified within the parentheses.

System action

Processing ends with a return code of 8.

User response

Correct the error, and rerun the job.

FPX0247I **ERRORACT=ABEND will not be effective for the reorganization step.**

Explanation

If RFSSCAN=(NO,YES) or RFSSCAN=(NO,NO) is specified, ERRORACT=ABEND will not be effective for the UOW reorganization. When the number of errors that occurred during the UOW reorganization has exceeded the value that is specified in the MAXFAILS keyword, the processing always stops with a return code of 8 regardless of what is specified for the ERRORACT parameter. If RFSSCAN=(NO,YES) or RFSSCAN=(NO,NO) is specified, ERRORACT=ABEND applies to errors that might occur during free space scanning and to the MAXERROR condition that might

occur in OPC. For details, see [“ERRORACT keyword”](#) on page 870.

System action

Processing continues.

User response

None. This message is informational.

FPX0248E **Incorrect duplicate keyword specification.**

Explanation

The indicated keyword cannot be specified more than once in a control statement.

System action

Processing ends with a return code of 8.

User response

Correct the error, and rerun the job.

FPX0249E **AREA=ALL is not supported by the current IMS APAR level.**

Explanation

The current IMS APAR level is below PK25289 (IMS V8) or below PK25294 (IMS V9).

System action

Processing ends with a return code of 8.

User response

To use AREA=ALL, install the IMS APAR, and resubmit the job. If the APAR is installed and the problem occurs, contact IBM Software Support.

FPX0250I **The following control statements were specified in //FPXOPTS:**

Explanation

This message shows the list of all control statements that are specified in FPXOPTS DD.

System action

Processing continues.

User response

None. This message is informational.

FPX0251I **Suppressed messages:**

Explanation

This message shows the message ID, the number of messages that are issued, and the number of messages that are suppressed.

System action

Processing continues.

User response

None. This message is informational.

FPX0252E **Command or keyword xxxxxxxx is missing for FPXOPTS DD.**

Explanation

A mandatory command or keyword is not specified in FPXOPTS DD.

System action

Processing ends with a return code of 8.

User response

Correct the control statement, and rerun the job.

FPX0253E **Duplicate specification is not allowed for FPXOPTS DD.**

Explanation

Duplicate keyword or keyword value was specified in FPXOPTS DD.

System action

Processing ends with a return code of 8.

User response

Correct the incorrect statement, and rerun the job.

FPX0254E **Incorrect keyword was specified for FPXOPTS DD.**

Explanation

The specified keyword is incorrect.

System action

Processing ends with a return code of 8.

User response

Correct the incorrect statement, and rerun the job.

FPX0255E	Keyword specification was NOT within range for FPXOPTS DD.
-----------------	---

Explanation

The specified value was outside the acceptable range.

System action

Processing ends with a return code of 8.

User response

Correct the incorrect value, and rerun the job.

FPX0256E	Incorrect data type for keyword value specification was detected in FPXOPTS DD.
-----------------	--

Explanation

The data type of the specified value is incorrect.

System action

Processing ends with a return code of 8.

User response

Correct the incorrect data type, and rerun the job.

FPX0257E	A syntax error was detected in FPXOPTS DD.
-----------------	---

Explanation

A syntax error was detected in FPXOPTS DD.

System action

Processing ends with a return code of 8.

User response

Correct the incorrect statement, and rerun the job.

FPX0258E	The operating mode specified is not supported in multiple area processing.
-----------------	---

Explanation

The operating modes that are supported in multiple area processing are the following modes:

- TYPRUN=PTRSCAN
- TYPRUN=FS and FULLSTEP=YES
- TYPRUN=PTRALL and FULLSTEP=YES
- TYPRUN=RPT and FULLSTEP=YES

System action

Processing ends with a return code of 8.

User response

Correct the specifications and resubmit the job.

FPX0259I	PROC_AFT_ERROR=NEXTAREA is ignored because only one area is specified.
-----------------	---

Explanation

PROC_AFT_ERROR=NEXTAREA is specified in FPXIN DD, but only one area is specified for processing.

System action

Processing continues.

User response

None. This message is informational.

FPX0260E	Image copy output function cannot be used. - IMS HP Image Copy module: <i>modulename</i> is not found. - IMS HP Image Copy version check module ended with error: <i>nn</i> - The PTF level of the IMS HP Image Copy module is not high enough. - Product information: <i>product_information</i>
-----------------	--

Explanation

IMS HP Image Copy function cannot be used. The reason is described in the subtext of this message.

System action

Processing ends with a return code of 8.

User response

Correct the problems and resubmit the job.

FPX0261W **An error was detected in the IMS HP Image Copy processing.**
- Area name: *areaname*, Area number: *area_number*, Function: *function*, Return code: *return_code*, Reason code: *reason_code*.

Explanation

IMS HP Image Copy processing failed. The reason is described in the subtext of this message. The subtext shows the function that was used.

System action

Processing ends with a return code of 4.

User response

Correct the problems and resubmit the job.

FPX0262W **Failed to take an image copy of this area. Processing continues with the next area.**

Explanation

IMS HP Image Copy processing failed for this area, but it continues with the next area.

System action

Processing ends with a return code of 4.

User response

Correct the problems. Create another image copy for this area for recovery purposes.

FPX0263W **Processing continues, but no image copy is taken in the subsequent processing because image copy processing has failed.**

Explanation

IMS HP Image Copy processing has failed, no image copy is taken in the subsequent processing.

System action

FPO sets an end-of-job return code of 4, and continues processing. Associated error message might be issued before this message. No image copy is taken in the subsequent processing.

User response

Correct the problems. Create image copies for recovery purposes.

FPX0264E **Processing ended because errors were detected in ICEIN DD of IMS HP Image Copy.**

Explanation

The specifications in ICEIN DD are incorrect.

System action

Processing ends with a return code of 8.

User response

Correct the specifications and resubmit the job.

FPX0265I **The image copy data set generated within this job cannot be used for database recovery.**
- ICNOTIFY=NO is specified
- Area is not defined in RECON:
DBDNAME=*dbdname*
AREANAME=*areaname*

Explanation

The image copy data set cannot be used for database recovery because it is not registered to DBRC.

System action

Processing continues.

User response

None. This message is informational.

FPX0266W **Concurrent image copy of a non-recoverable database is not supported.**

Explanation

Concurrent image copy is not supported for a non-recoverable database.

System action

Processing continues, but no image copy is taken in the subsequent processing. When the processing ends, it returns a return code of 4.

User response

Correct the problems and create another image copy for recovery purposes.

FPX0267W **RECON access failed:**
FUNC=function RC=return_code
RSN=reason_code

Explanation

An error was detected in the DBRC API call.

System action

Processing continues, but no image copy is taken in the subsequent processing when FUNC=STARTDBRC or FUNC=COMMAND. When the processing ends, it returns a return code of 4.

User response

Check the DBRC message that precedes this message.

FPX0269E **Image Copy dataset is incomplete.**

Explanation

Image copy data set is incomplete due to premature utility termination.

System action

Processing ends with a return code of 8.

User response

Correct the problems and create another image copy for recovery purposes.

FPX0270E **Incorrect UOW range specified:**
STARTUOW=nnnnn,
STOPUOW=mmmmm

Explanation

An incorrect UOW range was specified. STARTUOW must not be greater than STOPUOW.

System action

Processing ends with a return code of 8.

User response

Correct the keyword value, and resubmit the job.

FPX0271E **Mandatory keyword keyword is not specified in statement name statement.**

Explanation

The *keyword* is not specified in FPXIN. This keyword is required for the statement.

System action

Processing ends with a return code of 8.

User response

Correct the incompatible specification, and resubmit the job.

FPX0272E **GLOBAL command must be specified as a first command.**

Explanation

FPO found that the GLOBAL command was not specified on the first line of FPXIN.

System action

Processing ends with a return code of 8.

User response

Specify the GLOBAL command on the first line of FPXIN, and rerun the job.

FPX0273I **The ITKBLOAD parameter was ignored because no server name was specified.**

Explanation

An IMS Tools KB load module library was specified for the ITKBLOAD keyword parameter, but the specification was ignored because no server XCF group name was specified for the ITKBSRVR keyword parameter.

System action

Processing continues, but IMS HP Fast Path Utilities does not store any reports to the IMS Tools KB Output repository.

User response

If you want to store the reports to the IMS Tools KB Output repository, specify the IMS Tools KB server XCF group name for the ITKBSRVR keyword parameter.

FPX0274I **SORT=YES was internally changed to SORT=NO because OUTPUT=USER was specified.**

Explanation

SDEP=YES is not allowed when OUTPUT=USER is specified. ODE changes the SORT=YES option to the SORT=NO option internally.

System action

Processing continues.

User response

None. This message is informational.

FPX0275E	The built-in variable ROOT_KEY or SEGMENT_CKEY is not valid for SDEP segments when SDEPHIER=NO is specified.
-----------------	---

Explanation

The built-in variables ROOT_KEY and SEGMENT_CKEY cannot be specified for SDEP segments when SDEPHIER=NO is specified.

System action

Processing ends with a return code of 8.

User response

Correct the incompatible specification, and resubmit the job.

FPX0276E	Image copy dataset (ddname DD) cannot be defined as SYSOUT when ICNOTIFY=YES is specified.
-----------------	---

Explanation

When ICNOTIFY=YES is specified to register the image copy data set to DBRC, the image copy data set must not be defined as SYSOUT.

System action

Processing ends with a return code of 8.

User response

Correct the specifications and resubmit the job.

FPX0277I	The REPORT control statement is ignored.
-----------------	---

Explanation

A REPORT control statement was found, but it was ignored because analysis reports can be generated only in the following operating modes:

- TYPRUN=RPT and FULLSTEP=YES
- TYPRUN=FS and FULLSTEP=YES

System action

Processing continues.

User response

None. This message is informational.

FPX0278I	FULLSTEP=YES must be specified in order to generate the analysis reports.
-----------------	--

Explanation

A REPORT control statement was found, but because FULLSTEP=YES is not specified, analysis reports are not generated.

System action

Processing continues. No analysis reports are generated.

User response

None. This message is informational.

FPX0279I	DUMMY or NULLFILE or SYSOUT is specified for the ddname DD statement.
-----------------	--

Explanation

DUMMY, NULLFILE, or SYSOUT cannot be specified for the indicated DD statement.

System action

Processing continues. OAE does not extend the image copy whose target extended image copy data set is specified as DUMMY, NULLFILE, or SYSOUT, and invalidates the image copy in the RECON data sets.

User response

- If an FPX0287W message is issued, create an image copy of the extended DEDB area as soon as possible.
- If an FPX0328E message is issued, check the image copy data set that is registered in the RECON data sets, correct the incompatible error, and rerun the job.

FPX0280W	Due to errors, no image copy data set was extended.
-----------------	--

Explanation

OAE did not extend the image copy due to errors.

System action

- If the ICERR_CONT option is in effect, OAE continues to extend the DEDB area. The return code is set to 4 and processing continues.
- If the ICERR_CONT option is not in effect, OAE stops extending the DEDB area and ends the process with a return code of 8.

User response

Check the preceding messages. Follow the User Response for those messages, and, if necessary, rerun the job.

FPX0281W	The output data set for the extended image copy is not specified. Specify the <i>ddname</i> DD statement.
-----------------	--

Explanation

Although the image copy is registered in the RECON data sets, the corresponding output data set for the extended image copy is not specified in the JCL stream.

System action

The return code is set to 4 and processing continues. OAE does not extend the image copy whose target extended image copy data set is not specified, and invalidates the image copy in the RECON data sets.

User response

- If an FPX0287W message is issued, create an image copy of the extended DEDB area as soon as possible.
- If an FPX0328E message is issued, specify the indicated DD statement in JCL, and rerun the job.

FPX0282W	The image copy record is not found in RECON.
-----------------	---

Explanation

No valid image copy is registered in the RECON data sets.

System action

The return code is set to 4 and processing continues. No image copy data set is extended.

User response

- If an FPX0287W message is issued, create an image copy of the extended DEDB area as soon as possible.
- If an FPX0328E message is issued, ensure that valid image copies are registered in the RECON data sets and rerun the job.

FPX0283W	The image copy data set is incorrect. [IC1 IC2] DSN=<i>data_set_name</i> - The format is incorrect. - The DBD name is incorrect. DBD name: <i>dbd_name</i> - The area name is incorrect. AREA name: <i>area_name</i> - The CI size is incorrect. CI size: <i>xxxxx</i> - An unsupported compression routine is used. Routine: <i>routine_name</i> - The timestamps that are recorded in RECON and image copy header record do not match. (cont) Primary image copy data set: (cont) RECON <i>yyyy/ddd HH:MM:DD thmiju +x</i> (cont) IC Header <i>yyyy/ddd HH:MM:DD thmiju +x</i> (cont) Secondary image copy data set: (cont) RECON <i>yyyy/ddd HH:MM:DD thmiju +x</i> (cont) IC Header <i>yyyy/ddd HH:MM:DD thmiju +x</i>
-----------------	---

Explanation

The image copy data set that is registered in the RECON data sets cannot be extended. The reason is described in the subtext of this message.

System action

The return code is set to 4 and processing continues. OAE does not extend this image copy data set, and invalidates the image copy in the RECON data sets.

User response

- If an FPX0287W message is issued, create an image copy of the extended DEDB area as soon as possible.
- If an FPX0328E message is issued, ensure that valid image copies are registered in the RECON data sets and rerun the job.

FPX0283W **The image copy data set is incorrect.**
- The numbers of records are different.
(cont) IC1 DSN= *data_set_name*
(*x,xxx,xxx,xxx*)
(cont) IC2 DSN= *data_set_name*
(*x,xxx,xxx,xxx*)

Explanation

The number of the primary image copy records is different from that of the secondary image copy records.

System action

The return code is set to 4 and processing continues. OAE does not extend the image copy data sets.

User response

- If an FPX0287W message is issued, create an image copy of the extended area as soon as possible.
- If an FPX0328E message is issued, ensure that valid image copies are registered in the RECON data sets and rerun the job.

FPX0284I **Image copy [IC1 | IC2] is not registered in RECON. The following specification is ignored:**
(cont) DDN=FPXBIC0x DSN=
data_set_name

Explanation

The indicated FPXBIC0x DD statement is ignored because the corresponding image copy is not registered in the RECON data sets.

System action

Processing continues.

User response

None. This message is informational.

FPX0285W **The following error occurred while extending the image copy data set:**
(cont) - The image copy data set that is registered in RECON does not exist. [IC1 | IC2] DSN=
data_set_name
(cont) - Function = *typ* DD name =
ddname* MVS RET and RSN codes are: *rc + rsn

(cont) - I/O error. [IC1 | IC2 | FPXBIC01 | FPXBIC02] DSN=
data_set_name

Explanation

An error occurred while extending the image copy data set that is registered in the RECON data sets. The reason is described in the subtext of this message.

System action

The return code is set to 4 and processing continues. OAE does not extend this image copy data set, and invalidates the image copy in the RECON data sets.

User response

- If an FPX0287W message is issued, create an image copy of the extended DEDB area as soon as possible.
- If an FPX0328E message is issued, correct the invalid specification and rerun the job.

FPX0286I **OAE invalidated the entry for the following image copy data set in RECON:**
(cont) [IC1 | IC2] DSN=
data_set_name

Explanation

For consistency, OAE invalidated the specified image copy data set in the RECON data sets.

System action

Processing continues.

User response

None. This message is informational.

FPX0287W **Create an image copy of the extended DEDB area immediately.**

Explanation

Although the image copy extension option is specified, no image copy data set was extended because an error was detected or the RESTART keyword is specified.

Important: Because the image copy that was created before this OAE run cannot be used to recover the extended DEDB area, create an image copy of the extended area as soon as possible.

System action

The return code is set to 4 and processing continues.

User response

Create an image copy of the extended area as soon as possible.

FPX0288I **OAE extended the image copy data set.**

Explanation

The OAE process ended normally. Both the area and the image copy data sets are extended by OAE. The extended area can be recovered by using the extended image copy data set.

System action

Processing continues.

User response

None. This message is informational.

FPX0289I **Create an image copy of the extended DEDB area immediately.**

Explanation

The OAE process ended normally.

Important: Because the image copy that was created before this OAE run cannot be used to recover the extended DEDB area, create an image copy of the extended area as soon as possible.

System action

Processing continues.

User response

Create an image copy of the extended area as soon as possible.

FPX0291W **The data set names on the FPXBIC01 DD, the FPXBIC02 DD, the IC1 and the IC2 fields in RECON must be different.**
(cont) DSN= data_set_name

Explanation

OAE detected the same data set names. The data set names must be different.

System action

The return code is set to 4 and processing continues. OAE does not extend the corresponding image copy

data set, and invalidates the image copy in the RECON data sets.

User response

- If an FPX0287W message is issued, create an image copy of the extended DEDB area as soon as possible.
- If an FPX0328E message is issued, correct the invalid specification and rerun the job.

FPX0292E **OAE restart is not required.**
Remove the RESTART parameter from the EXTEND statement.

Explanation

Although a previous run of the OAE utility ended prematurely, OAE can be rerun without the RESTART parameter.

System action

The return code is set to 8.

User response

Remove the RESTART parameter from the EXTEND statement and rerun the job.

FPX0293I **The *process_name* process ended.**

Explanation

One of the following OAE processes ended normally.

- Initialization
- Environment setup
- ADS copy
- SDEP expansion
- CSC checking
- IOVF expansion
- ADS copy back
- Termination

System action

Processing continues.

User response

None. This message is informational.

FPX0295W **The *function* function of compression exit routine *exit-routine-name* failed with RC= *nn*.**

Explanation

The compression exit routine *exit-routine-name* returned an error return code. *function* shows the function that failed:

INITIALIZATION

Initialization processing

BLOCK WRITE

Block writing processing

TERMINATION

Termination processing

System action

The return code is set to 4 and processing continues. OAE does not extend the corresponding image copy data set, and invalidates the image copy in the RECON data sets.

User response

- If an FPX0287W message is issued, create an image copy of the extended DEDB area as soon as possible.
- If an FPX0328E message is issued, locate the error message from IMS HP Image Copy in the job log, correct the error, and rerun the job.

FPX0296W The image copy that is registered in RECON cannot be supported.

Explanation

OAE cannot extend an IC2 image copy that was created by the IMS Database Image Copy 2 Utility (DFSUDMT0). Also, OAE cannot extend an image copy that is created by the flash copy process.

System action

The return code is set to 4 and processing continues. No image copy is extended.

User response

- If an FPX0287W message is issued, create an image copy of the extended DEDB area as soon as possible.
- If an FPX0328E message is issued, ensure that valid image copies are registered in the RECON data sets and rerun the job.

FPX0297W The image copy data set was extended, but the DBRC RECON data sets were not updated.

Explanation

OAE extended the image copy data set, but failed to update the RECON data sets.

System action

Processing continues. OAE did not update the last image record in the RECON data set with the information of the extended image copy data set.

User response

See message FPX0267W and messages issued by DBRC. Correct the error, and issue the CHANGE.IC command.

FPX0298W ICEXTEND keyword is ignored because RESTART keyword is specified.

Explanation

OAE cannot extend image copies during restart processing. The ICEXTEND keyword is ignored.

System action

Processing ends with a return code of 4.

User response

Create an image copy of the extended DEDB area as soon as possible.

FPX0299E The *process_name* process ended with errors.

Explanation

One of the following OAE processes ended with errors:

- Initialization
- Environment setup
- ADS copy
- SDEP expansion
- CSC checking
- IOVF expansion
- ADS copy back
- Termination

System action

Processing ends with a return code of 8.

User response

See the preceding error messages, correct any errors, and rerun the job with the RESTART keyword.

FPX0300I **Area Sensor process started.**

Explanation

The Area Sensor process started.

System action

Processing continues.

User response

None. This message is informational.

FPX0301I **Area Sensor process ended normally. (Elapsed Time: hh:mm:ss.tt)**

Explanation

The Area Sensor process ended normally. Elapsed Time shows the time elapsed for the Area Sensor processing.

System action

Processing continues.

User response

None. This message is informational.

FPX0302E **Area Sensor process ended with errors**

Explanation

The OER Area Sensor Process ended with errors.

System action

Processing ends with a return code of 8.

User response

See message FPX0148E or FPX0303E, correct the problem, and rerun the job.

FPX0303E **A serious error was detected in the CI at RBA: xxxxxxxx during the Area Sensor process.**

Explanation

The OER Area Sensor found a pointer error.

System action

If Area Sensor is running stand-alone, OER issues message FPX0302W and ends with a return code of 8. If Area Sensor is running under OPC, the processing continues until OPC process ends.

User response

If Area Sensor was running stand-alone, run FPB DEDB Pointer Checker or OPC to identify errors in the DEDB area. If Area Sensor was running with OPC, correct the error reported by OPC and run the OER job again.

FPX0308I **UOW xxxxxxxxxx process started.**

Explanation

The UOW Evaluation process or the UOW Ranking process has started. String xxxxxxxxxx is either Evaluation or Ranking.

System action

Processing continues.

User response

This is an information message. No action is required.

FPX0309I **UOW xxxxxxxxxx process ended normally. (Elapsed Time: hh:mm:ss.tt)**

Explanation

The UOW Evaluation process or the UOW Ranking process ended normally. String xxxxxxxxxx is either Evaluation or Ranking. Elapsed Time shows the time elapsed for the processing.

System action

Processing continues.

User response

This is an information message. No action is required.

FPX0310E **UOW Ranking process ended with an error.**

Explanation

The UOW Ranking process ended with an error.

System action

Processing ends with a return code of 8.

User response

Correct the error, and rerun the job.

FPX0311E **Name/Token service xxxxxxxx failed. Name: name RC=nn.**

Explanation

The process failed in the z/OS MVS Name/Token Service. String xxxxxxxx shows the service name. String name shows the name used to get access to the token. Value nn is the return code of the Name or the Token service.

System action

Processing ends with a return code of 8.

User response

Correct the error, and rerun the job.

FPX0312W **UOW Ranking process ended normally, but the number of UOWs to be reorganized is zero.**

Explanation

The UOW Ranking process ended normally, but the number of UOWs to be reorganized is zero.

System action

Processing continues.

User response

None.

FPX0313I **Reorganization process starts.**

Explanation

This message shows that the reorganization of selected UOWs has started.

System action

Processing continues.

User response

This is an information message. No action is required.

FPX0314I **Reorg was attempted for UOW nnnnnnnn (resulttxt); No. of free IOVF CIs after reorg: xxxxxxxx (yyy % free)**

Explanation

The UOW Reorganizer of OER attempted to reorganize the UOW of the UOW number nnnnnnnn. String resulttxt shows the result of the reorganization attempt, which is either 'succeeded' or 'skipped.' Value xxxxxxxx shows the number of free IOVF control intervals after a successful reorganization, and value yyy shows the percentage of the number of free IOVF CIs against the total number of IOVF CIs.

System action

Processing continues.

User response

This is an information message. No action is required.

FPX0315I **Reorganization attempted from UOW nnnnnnnn to UOW mmmmmmmm (Number of UOWs: xxxxxxxx).**

Explanation

The UOW Reorganizer of OER attempted to reorganize UOWs starting from UOW number nnnnnnnn and ending at UOW number mmmmmmmm. Value xxxxxxxx shows the total number of UOWs that are processed.

System action

Processing continues.

User response

This is an information message. No action is required.

FPX0316I **(Several lines that indicate reorganization statistics)**

Explanation

Statistics that are collected during an HSR call are printed with this message number. The statistics include the following items:

UOW reorganization activity

Number of UOWs succeeded in reorg..... nnnnnnnn
The number of UOWs that were actually reorganized.

Number of UOWs skipped in reorg..... nnnnnnnn (all RAP CIs were empty)
The number of UOWs that were skipped because anchor points were empty.

Number of UOWs failed in reorg..... *nnnnnnnn*
The number of UOWs that failed to be reorganized.

Private buffer set usage

Number of buffer sets - (initial, extension).... *nnn,nnn*
The number of buffer sets that were initially allocated, and the number of times the private buffer pool was extended.

Number of buffer sets - (input, output/IOVF).... *nnn,nnn*
The number of buffer sets that were used for the input for the root addressable portion, and the number of buffer sets that are used for output. The latter *nnn* includes the number of buffer sets that were used for the root addressable portion and IOVF data CIs.

Highest usage of buffer sets for IOVF I/Os..... *nnnnnnnn* (used by UOW *nnnnnnnn*)
The highest number of buffer sets that were used and the UOW number of the UOW for which the highest number of buffer sets were allocated.

Space reclamation activity

Number of IOVF CIs finally freed..... *nnnnnnnn* (freed: *nnnnnnnn*, reused: *nnnnnnnn*)
The number of IOVF data CIs that were freed and the number that were reused.

Number of IOVF CIs finally allocated..... *nnnnnnnn* (new: *nnnnnnnn*, reused: *nnnnnnnn*)
The number of IOVF data CIs that were newly allocated and the number that were reused.

Number of free IOVF CIs after reorg..... *nnnnnnnn* (*nnn%* free)
The total number of free IOVF data CIs that are in the area, and the number of free CIs as the percentage of the total number of CIs.

Other statistics

Number of asynchronous read ahead I/Os..... *nnnnnnnn*
The number of asynchronous read ahead I/Os that were performed.

Number of waits for UOW locks..... *nnnnnnnn*
The number of lock waits that occurred when UOW locks were requested.

Number of waits for private buffers..... *nnnnnnnn*
The number of waits that occurred when private buffers were requested to be allocated for HSR.

System action

Processing continues.

User response

This is an information message. No action is required.

FPX0317E **Reorganization failed in UOW**
xxxxxxx; reason: cc, reason_text

Explanation

Reorganization of a UOW failed. The number *xxxxxxx* shows the UOW number. The value *cc* shows the reason code of the error, and *reason_text* shows the textual explanation of the reason of the error.

System action

Processing continues until the error count reaches the MAXFAILS value. The system action depends on the ERRORACT parameter that is specified in the REORG control statement.

User response

Check the reason of the error and correct the cause of the error.

FPX0318E **Reorganization ended at UOW**
xxxxxxx; reason: reason_text

Explanation

Reorganization processing ended while OER was processing UOW number *xxxxxxx*. String *reason_text* shows the reason of the error.

System action

Reorganization processing stops. The next action depends on the ERRORACT parameter that is specified in the REORG control statement.

User response

Check the reason of the error and check if it is acceptable. If not, correct the cause of the error.

FPX0319E **Unexpected status code: xx,**
reason_text

Explanation

Unexpected DL/I status code *xx* was returned when OER called the IMS DEDB High-Speed Direct Reorganization. String *reason_text* shows the textual explanation of the reason of the error.

System action

Reorganization processing stops. The next action depends on the ERRORACT parameter that is specified in the REORG control statement.

User response

Check the reason of the error and correct the cause of the error. If the cause of the error indicates an internal error in OER, call IBM Software Support.

FPX0320W	Previous reorg process failed - reorg will be restarted from the UOW that failed in the last reorg.
-----------------	--

Explanation

The reorganization has been started from the UOW that failed in the last online reorganization.

System action

Check the UOW number that is shown in a message that follows.

User response

None.

FPX0321I	Reorganization process ended with RC=<i>ccc</i>. (Elapsed Time: <i>hh:mm:ss.tt</i>)
-----------------	--

Explanation

UOW Reorganization process ended with a return code of *ccc*, which can be one of 0, 4, or 8. Elapsed Time shows the time elapsed for the UOW Reorganization processing for all UOWs that are selected.

System action

If the return code is either 0 or 4, processing continues. If the return code *ccc* is larger than 4, OER processing ends with a return code larger than or equal to *ccc*.

User response

If the return code is either 4 or 8, see the messages that precede this message and take appropriate action.

FPX0322E	Failure count has exceeded the MAXFAILS number.
-----------------	--

Explanation

The total number of failures that occurred in the UOW Reorganization processing has exceeded the value that is specified in the MAXFAILS parameter of the REORG control statement.

System action

Reorganization processing stops. The next action depends on the ERRORACT parameter that is specified in the REORG control statement.

User response

Check reasons of the errors shown in messages FPX0317E and check if they are acceptable. If not, correct the cause of each error.

FPX0323I	The online pointer scan process for AREA name = <i>areaname</i> started.
-----------------	---

Explanation

The online pointer scan process started.

System action

Processing continues.

User response

None. This message is informational.

FPX0324I	The online pointer scan process for AREA name = <i>areaname</i> completed successfully. (Elapsed Time: <i>hh:mm:ss.tt</i>)
-----------------	---

Explanation

The online pointer scan process ended normally. Elapsed Time shows the time elapsed for the online pointer scan processing.

System action

Processing continues.

User response

None. This message is informational.

FPX0325E	The online pointer scan process for AREA name = <i>areaname</i> ended with errors. (Elapsed Time: <i>hh:mm:ss.tt</i>)
-----------------	--

Explanation

The online pointer scan process ended with errors. Elapsed Time shows the time elapsed for the online pointer scan processing.

System action

Processing ends with a return code of 8.

User response

Correct the error, and rerun the job.

FPX0326W	The online pointer scan process for AREA name = <i>areaname</i> ended with warnings. (Elapsed Time: <i>hh:mm:ss.tt</i>)
-----------------	--

Explanation

The online pointer scan process detected pointer errors, or the in-depth pointer checker and the report generator ended with a warning message. Elapsed Time shows the time elapsed for the online pointer scan processing.

System action

Processing ends with a return code of 4.

User response

Correct any errors, and rerun the job.

FPX0327E	The number of pointer errors that are detected in AREA name = <i>areaname</i> is <i>nnn</i>.
-----------------	---

Explanation

The number of pointer errors that are detected during the OPC process for the indicated area is shown.

If FULLSTEP=NO is specified, it is the number of pointer errors that are detected during online pointer checking.

If FULLSTEP=YES is specified, it is the number of pointer errors that are detected during both online pointer checking and in-depth pointer checking.

System action

Processing ends with a return code of 4 when FULLSTEP=NO is specified, 8 when FULLSTEP=YES is specified.

User response

Consult your database administrator to correct the problems with the database.

FPX0328E	OAE process stopped due to ICERR_CONT=NO option.
-----------------	---

Explanation

OAE stopped expanding the DEDB area because the ICERR_CON=NO option is specified and the attempt to extend the image copy failed.

System action

Processing ends with a return code of 8.

User response

See the preceding error messages, correct any errors, and rerun the job.

If you want OAE to extend the DEDB area only, specify the ICERR_CONT keyword in SYSIN DD for shadow expansion or the ICERR_CONT=YES keyword in FPXIN DD for original expansion. After OAE ends normally, create an image copy of the extended DEDB area as soon as possible.

FPX0329W	The <i>process_name</i> process ended with warnings.
-----------------	---

Explanation

One of the following OAE processes ended with warnings:

- Initialization
- Environment setup
- ADS copy
- SDEP expansion
- CSC checking
- IOVF expansion
- ADS copy back
- Termination

System action

Processing ends with a return code of 4.

User response

See the preceding error messages.

FPX0330I	LISTCAT information of the temporary area data set after extending:
-----------------	--

(cont) - AREA: *areaname*
(cont) - ADSDD: *ddname* ADSDS:
dsname

Explanation

This message shows the LISTCAT report of the temporary area data set after extension.

This message provides you information when you run OAE with the RESTART option or when you recover the area after OAE.

System action

Processing continues.

User response

None. This message is informational.

FPX0331I **LISTCAT information of the area data set after extending:**
(cont) - AREA: *areaname*
(cont) - ADSDD: *ddname* ADSDS:
dsname

Explanation

This message shows the LISTCAT report of the original area data set after extension.

This message provides you information when you recover the area after OAE.

System action

Processing continues.

User response

None. This message is informational.

FPX0332I **The main process completed for all the specified areas. Subordinate processes are still in progress.**

Explanation

The main process of OPC or OER completed for all the specified areas.

System action

Processing continues.

User response

None. This message is informational.

FPX0334E **Data set name error.**

Explanation

The specified data set name is incorrect. On the continued line (cont), an asterisk (*) points to the beginning of the part in error.

System action

Processing ends with a return code of 8.

User response

Ensure that the data set name is specified correctly. Correct any errors, and rerun the job.

FPX0335E **Specify a mask variable for the TARDSNMASK keyword parameter.**

Explanation

A mask variable, such as &AREA, must be specified for the TARDSNMASK keyword parameter when you process multiples areas to generate multiple tuning aid record data sets. On the continued line (cont), an asterisk (*) points to the beginning of the part in error.

System action

Processing ends with a return code of 8.

User response

Ensure that the TARDSNMASK keyword parameter is specified correctly. Correct any errors, and rerun the job.

FPX0337I **OLRECL=SEGMENT is ignored because OUTPUT= parameter was specified.**

Explanation

ODE supports OLRECL=SEGMENT only when STD or DBT format is specified.

System action

Processing continues.

User response

None. This message is informational.

FPX0338I **statement control statement is ignored because keyword=value is specified.**

Explanation

The indicated control statement and all the associated keywords are ignored because the indicated keyword value is specified.

System action

Processing continues.

User response

None. This message is informational.

FPX0339E **Incorrect RBA value in CI at RBA
xxxxxxx.**
**(cont) RBA value: yyyyyyyy Offset:
zzzz**

Explanation

The indicated RBA value (yyyyyyy), which is at offset zzzz of a CI at RBA xxxxxxx, is incorrect.

System action

Processing ends with an abend code of U3500.

User response

Run the FPA Analyze function to check the integrity of the CI. Correct the reported problem, and rerun the job.

FPX0340E **[OPEN | GET] failed for VSAM data
set.**
(cont) - DSNAME: dsname
(cont) - Return code is rc (hex_rc),
[ACB error flag code | RPL
feedback area] is rsn (hex_rsn).

Explanation

The utility received a nonzero return code from VSAM when attempting to access the VSAM data set with the OPEN or GET macro. The return code and the reason code are shown in both decimal (*rc*, *rsn*) and hexadecimal (*hex_rc*, *hex_rsn*) formats.

System action

Processing ends with an abend code of U3500.

User response

See *DFSMS Macro Instructions for Data Sets*, which describes VSAM administration macros. If this situation persists, contact IBM Software Support.

FPX0341E **TRKCALC macro failed. RC=rc**

Explanation

The TRKCALC macro failed. *rc* shows the return code.

System action

Processing ends with a return code of 8.

User response

See the *z/OS MVS Authorized Assembler Services Guide* to determine the meaning of the return code. Correct the error and rerun the job. If the problem persists, contact IBM Software Support.

FPX0342E **The maximum number of volumes
for the [original | temporary] ADS
is 1, but the size of the extended
ADS exceeds the total number of
CIs available for the device.**

Explanation

The extended ADS requires more volumes for allocation.

System action

Processing ends with a return code of 8.

User response

For temporary ADS, use a device with larger capacity. For extended original ADS, increase the number of volumes on the MAXVOLS keyword (for a SMS environment) or on the VOLUME_EXT keyword (for a non-SMS environment). Then, rerun the job.

FPX0343E **The VOLUME keyword must be
supplied in a non-SMS
environment.**

Explanation

The VOLUME keyword is a mandatory keyword in a non-SMS environment, but it is not specified.

System action

Processing ends with a return code of 8.

User response

Specify the VOLUME keyword and rerun the job.

FPX0344E **The VOLUME_EXT keyword is not supported when the original ADS is a MADS.**

Explanation

The VOLUME_EXT keyword is specified, but the original ADS is a MADS (multiple area data sets).

System action

Processing ends with a return code of 8.

User response

Remove the VOLUME_EXT keyword, or change the original ADS to a SADS (single area data set), and rerun the job.

FPX0345E **Specify a volume with sufficient space to contain the temporary ADS.**

Explanation

The volume that is specified on the VOLUME keyword has insufficient space to contain the temporary ADS.

System action

Processing ends with an abend code of U3519.

User response

Specify a volume with sufficient space to contain the temporary ADS on the VOLUME keyword, and rerun the job.

FPX0346I **64 bit storage used for utility private buffers.**

Explanation

OER enabled the Fast Path 64-bit buffer manager for the IMS High-Speed DEDB Direct Reorganization utility (DBFUHDR0).

System action

Processing continues.

User response

None. This message is informational.

FPX0347I **FPBP64U=YES is not supported by the current IMS APAR level.**

Explanation

OER supports the Fast Path 64-bit buffer manager for the IMS High-Speed DEDB Direct Reorganization utility (DBFUHDR0) only for IMS 14 and later.

System action

Processing continues.

User response

None. This message is informational.

FPX0348I **The following control statements were specified in //INDD:**

Explanation

This message is accompanied by message FPX0049I. See message FPX0049I for a set of control statements that are specified in the INDD data set.

System action

Processing continues.

User response

None. This message is informational.

FPX0349E **INDD DD card error:
error_message_text
ROW: nnnn - error_card_image**

Explanation

One or more errors were detected in the INDD card.

System action

Processing ends with a return code of 8.

User response

Correct the statements and rerun the job.

FPX0350I **ERRACTN=*parm* was internally changed to ERRACTN=TERM.**

Explanation

The OER utility internally applied ERRACTN=TERM because the ERRACTN keyword was not specified or the parameter of ERRACTN was EXIT or CONT.

System action

Processing continues.

User response

None. This message is informational.

FPX0351E **BLDL failed for dynamic allocation member: *member_name***

Explanation

The BLDL macro was issued for the indicated dynamic allocation member that was specified in the STEPLIB, but the macro failed.

System action

Processing ends with a return code of 8.

User response

To determine the cause of the error, see *DFSMS Macro Instructions for Data Sets*. Correct any errors, and rerun the job. If this situation persists, contact IBM Software Support.

FPX0352E **DFSMDA member *member_name* was not found.**

Explanation

The indicated DFSMDA member was not found in the STEPLIB.

System action

Processing ends with a return code of 8.

User response

Ensure that the specified member exists in the STEPLIB. Correct any errors, and rerun the job. If this situation persists, contact IBM Software Support.

FPX0353E **Incorrect dynamic allocation member: *member_name***

Explanation

The indicated dynamic allocation member does not conform to the expected DFSMDA structure.

System action

Processing ends with a return code of 8.

User response

Verify the integrity of the dynamic allocation member. Correct the problem, then rerun the job.

FPX0354E **Dynamic allocation failed for DDNAME: *ddname*, DSNAME: *dsname*, RC= *code*, RSN=*eeeeiiii***

Explanation

Dynamic allocation for *ddname* and *dsname* failed. *code* is the return code from SVC99. *eeee* is the S99 ERROR contents. *iiii* is the S99 INFO contents.

System action

Processing ends with a return code of 8.

User response

Look up the dynamic allocation code (SVC99) in the *MVS Programming: Assembler Services Reference*. Correct the problem, then rerun the job.

FPX0356I **SDEP segment data is not extracted because of OUTPUT, SORT, and SDEPHIER keyword specifications.**

Explanation

SDEP segment data was not extracted due to the combination of OUTPUT, SORT, and SDEPHIER keyword parameters. For the combination of these keyword parameters, see [Table 111 on page 900](#) in “OUTPUT keyword” on [page 899](#). This message is issued because SDEP_BYPASS_MSGLVL=INFORMATIONAL is specified.

System action

Processing continues.

User response

None. This message is informational.

FPX0357W **SDEP segment data is not extracted because of OUTPUT, SORT, and SDEPHIER keyword specifications.**

Explanation

SDEP segment data was not extracted due to the combination of OUTPUT, SORT, and SDEPHIER keyword parameters. For the combination of these keyword parameters, see [Table 111 on page 900](#) in “OUTPUT keyword” on [page 899](#). This message is issued because SDEP_BYPASS_MSGLVL=WARNING is specified.

System action

The return code is set to 4 and processing continues.

User response

To have SDEP segment data extracted, change the parameters on the OUTPUT, SORT, and SDEPHIER keywords and rerun the job.

FPX0358I **AREA *areaname* is flagged as empty.**

Explanation

The empty area flag DMACEMPT is set in the DMAC.

System action

Processing continues.

User response

None. This message is informational.

FPX0359I **Found SDEP pointers that point to the IOVF portion.**

Explanation

OPC detected an SPCF or an SPTF pointer that points to the IOVF portion. This message is issued because SDEP_PTRIOVF_MSGLVL=INFORMATIONAL is specified.

System action

Processing continues.

User response

None. This message is informational.

FPX0360I **The number of informational messages in the Online Pointer Scan report exceeded the limit.**

Explanation

The number of informational messages issued to the Online Pointer Scan report exceeded 100. No further informational messages are written to the Online Pointer Scan report.

System action

Processing continues.

User response

None. This message is informational.

FPX0361E **Compression exit *exit_name* returned an error for segment *segment_name* at RBA X'xxxxxxxx' Abend code = Unnnn Reason code = X'xxxxxxxx'**

Explanation

The specified edit/compression routine was called, but it returned an error. The segment RBA is shown only when the error occurs in the EXPAND call.

System action

The job ends with an abend code of U3512.

User response

Correct the errors, and rerun the job.

FPX0400E **The number of members in the XCF group exceeded the maximum.**

Explanation

The number of members that joined the XCF group exceeded the maximum.

System action

Processing ends with abend code U3533.

User response

Check the XCF group name and the number of members that joined the XCF group.

FPX0401W **Error in call to randomizer routine *rmodname* FUNCTION: *function_name*. Full area scan will be used. (cont) RETURN CODE: xxx REASON CODE: X'xxxxxxxx'**

Explanation

Specified randomizer module returned with an error. FUNCTION and REASON CODE are issued only when XCI randomizer is called.

System action

Processing does a full area scan and ends with a return code of 4.

User response

Check the randomizer that is loaded by IMS.

FPX0402W **The RBA of the current overflow CI in the first DOVF CI is null. RBA: rba, UOW#: nnnnn**

Explanation

The DOVFNULPTR_MSGLVL=WARNING option is enabled and the utility detected a null pointer (RBA) value for the current overflow CI in the indicated DOVF CI. The RBA and the UOW number in the message identify the first DOVF CI that contains the null pointer value.

System action

The return code is set to 4 and processing continues.

User response

To update the RBA value in the first DOVF CI, use the Online Expert Reorganization utility or the High-Speed DEDB Direct Reorganization utility and reorganize the area.

FPX1001E **Invalid function code function_code passed to FPXGXDR.**

Explanation

The user-written program passed the specified invalid function code.

System action

Processing ends with an abend code of 3599.

User response

Make sure that a valid function code is passed to the exit before attempting any other function.

FPX1002E **GET call prior to INI call to FPXGXDR.**

Explanation

The user-written program passed a GET-type function to the exit without previously passing a corresponding INI-type function for the input file.

System action

Processing ends with an abend code of 3599.

User response

Make sure that an INI-type function code is passed to the exit before attempting any other function.

FPX1003E **Invalid suffix on function_code call to FPXGXDR (SUFFIX = suffix).**

Explanation

The user-written program passed a function code with an invalid suffix to the exit. The invalid suffix is specified in the message.

System action

Processing ends with an abend code of 3599.

User response

Make sure that a valid suffix is appended to the function code. Valid suffixes are '1' through '9'.

FPX1004W **Duplicate INIT or INIx call to FPXGXDR.**

Explanation

The user-written program passed multiple INI-type function codes for the same input file.

System action

Processing continues and returns a blank (' ') code.

User response

Although processing continues, there might be a logic error in the user-written program with caused this situation. The program should be verified to make sure that only one INI-type function is passed for each input file.

FPX1005E **Unable to open file DDname**

Explanation

The exit was unable to open the specified file. One of the following might be in error:

- The XDRDATAx DD statement has been omitted from the JCL
- The suffix associated with a function passed by the user-written program does not match the prefix on the XDRDATAx DD statement contained in the JCL.

System action

Processing ends with an abend code of 3599.

User response

Correct the problem and resubmit the job.

FPX1006E **Unrecognized file format for file
DDname**

Explanation

The exit was unable to determine the format of the input file. One of the following might be in error:

- The file is not in either the SORT or STD format created by the ODE utility.
- The first record in the file is not in the format of a control record.

System action

Processing ends with an abend code of 3599.

User response

Make sure that the correct file is in use.

FPX1007W **EOJ or EOJx call without prior INIT
or INIx call to FPXGXDR.**

Explanation

The exit was passed an EOJ-type function against a file for which INI-type functionality was not active. One of the following might be in error:

- The exit was passed an EOJ-type function for an input file that was never opened.
- The exit was passed an EOJ-type function for an input file that had already been closed.

As a result, there might not be a correct flow of function for one or more of the input files.

System action

Processing continues and returns a blank (' ') code.

User response

Although processing continues, there might be a logic error in the user-written program with caused this situation. The program should be verified for the correct logic. Make sure the proper flow of INI-type and EOJ-type functions for each input file.

FPX1008I **Statistics for DDNAME DDNAME**

Explanation

The message indicates that subsequent messages contain processing statistics for the specified input file.

System action

Processing continues.

User response

None. This message is informational.

FPX1010E **Parameter 2 or 3 missing on GET
or GETx call to FPXGXDR.**

Explanation

Parameter 2 (key-related parms) and/or Parameter 3 (data-related parms) were omitted from a GET-type function passed to the exit.

System action

Processing ends with an abend code of 3599.

User response

Make sure that the user-written program includes these parameters in the call to the exit.

FPX1011E **End of file encountered during call
processing before end of control
records.**

Explanation

End-of-file was reach for an input file before the control records had been logically processed.

System action

Processing ends with an abend code of 3599.

User response

Make sure that the input file is correct.

HFPO messages

The following information is about messages and codes that begin with HFPO.

HFP00001I *processname* **PROCESS STARTED.**

Explanation

The indicated process, XCF CONNECT, MCS CONSOLE, or TIMER process, has started.

System action

Processing continues.

User response

None. This message is informational.

HFP00002I *processname* **PROCESS COMPLETED.**

Explanation

The indicated process, XCF CONNECT, MCS CONSOLE, or TIMER process, has completed.

System action

Processing continues.

User response

None. This message is informational.

HFP00004I **OSC INTERNAL APPLY JOB STARTED. JOB: *int_jobname*, OWNER: *driver_jobname***

Explanation

The indicated internal apply job, which is submitted by the indicated OSC driver job, has started.

System action

Processing continues.

User response

None. This message is informational.

HFP00005I **OSC INTERNAL APPLY JOB COMPLETED. JOB: *int_jobname*, OWNER: *driver_jobname***

Explanation

The indicated internal apply job, which is submitted by the indicated OSC driver job, has completed.

System action

Processing continues.

User response

None. This message is informational.

HFP00006I **TAKEOVER WILL START AT THE SPECIFIED TIME (*hh:mm*).**

Explanation

The takeover process will start at the indicated start time, which is specified on the TAKEOVER_TIME keyword.

System action

Processing continues.

User response

None. This message is informational.

HFP00103I **(MCS) CONSOLE HAS BEEN ACTIVATED.**

Explanation

The MCS console has been activated.

System action

Processing continues.

User response

None. This message is informational.

HFP00105I **(MCS) CONSOLE HAS BEEN DEACTIVATED.**

Explanation

The MCS console has been deactivated.

System action

Processing continues.

User response

None. This message is informational.

HFP00108I **(MCS) NOW ACCEPTING MODIFY COMMANDS.**

Explanation

The Online Structure Change utility is accepting modify commands.

System action

Processing continues.

User response

None. This message is informational.

HFP00114I *message text*

Explanation

Shows OSC internal job messages written to the SYSLOG.

System action

Processing continues.

User response

None. This message is informational.

HFP00201I (XCF) XCF CONNECTION COMPLETED (JOB: *jobname*, XCFGROUP: *xcfgroupname*).

Explanation

The XCF connection was established successfully.

System action

Processing continues.

User response

None. This message is informational.

HFP00202I (XCF) CHANGE CAPTURE STARTED SUCCESSFULLY FOR DBD *dbdname*

Explanation

The change capture process for the indicated DEDB has started.

System action

Processing continues.

User response

None. This message is informational.

HFP00203I (XCF) CHANGE CAPTURE STOPPED SUCCESSFULLY FOR DBD *dbdname*

Explanation

The change capture process for the indicated DEDB has stopped.

System action

Processing continues.

User response

None. This message is informational.

HFP00204I (XCF) XCF DISCONNECT COMPLETED (JOB: *driver_jobname*, XCFGROUP: *xcfgroupname*).

Explanation

The XCF connection was disconnected successfully.

System action

Processing continues.

User response

None. This message is informational.

HFP00206I (XCF) APPLY RECORD DD REQUESTED FOR DBD *dbdname*

Explanation

The internal apply job is requesting for the online updates data set for the indicated DEDB.

System action

Processing continues.

User response

None. This message is informational.

HFP00207I (XCF) APPLY PROCESS COMPLETED FOR DBD *dbdname*

Explanation

The change apply process for the indicated DEDB has completed successfully.

System action

Processing continues.

User response

None. This message is informational.

HFP00209I (XCF) APPLY PROCESS STARTED FOR DBD *dbdname*

Explanation

The change apply process for the indicated DEDB has started.

System action

Processing continues.

User response

None. This message is informational.

HFP01000I LOGGER EXIT [INITIALIZING | INITIALIZED | TERMINATING | TERMINATED].

Explanation

Shows the status of logger exit.

System action

Processing continues.

User response

None. This message is informational.

HFP01001I OSC LOGGER EXIT CANNOT BE INITIALIZED. THE IMS RELEASE LEVEL IS NOT SUPPORTED

Explanation

The OSC logger exit routine could not be initialized. The OSC logger exit routine can be used only with IMS 15.

System action

Processing continues.

User response

Use IMS 15 if you want to use the Online Structure Change utility. If you do not intend to use the Online Structure Change utility, ignore this message or remove the configuration of the OSC logger exit

routine. For more information, see the topic "Configuring the OSC logger exit routine" in *IMS Fast Path Solution Pack: Overview and Customization*.

HFP01002I OSC LOGGER EXIT CANNOT BE INITIALIZED. DEDB CALL ANALYSIS INITIALIZATION FAILURE

Explanation

The OSC logger exit routine could not be initialized.

System action

Processing continues.

User response

If this message was issued while an IBM Security Guardium S-TAP for IMS on z/OS job was running in the same IMS environment, configure the OSC logger exit routine by referring to the topic "Configuring the OSC logger exit routine" in *IMS Fast Path Solution Pack: Overview and Customization*. Otherwise, collect the job output and contact IBM Software Support.

HFP01040I DATA SET *dsname* NOT FOUND.

Explanation

Dynamic allocation returned an error when allocating the data set named *dsname*. The data set was not found.

System action

If the data set is essential, the process terminates with error.

User response

The data set name comes from the MDALIB or from the RECON. Correct the data set name in those places and rerun the job.

HFP01041I DATA SET *dsname* IN USE.

Explanation

Dynamic allocation returned an error when allocating the data set. The data set is allocated in an exclusive state by a different job.

System action

The process terminates.

User response

Wait until the other job releases the data set and rerun the job.

HFP01042E **SUBTASK *taskid* HAS
TERMINATED ABNORMALLY.**

Explanation

The indicated subtask did not complete normally.

System action

The job ends with an abend code of U9999.

User response

Collect the job output, including the dump, and contact IBM Software Support.

HFP01043E **ABEND CODE IS *code***

Explanation

This message follows message HFP01042E and displays the abend code.

System action

The job ends with an abend code of U9999.

User response

Collect the job output, including the dump, and contact IBM Software Support.

HFP01108I **[CONNECTED TO | DISCONNECTED
FROM] TARGET MEMBER
*membername***

Explanation

The Online Structure Change utility job connected to or disconnected from the target XCF connection group member.

System action

Processing continues.

User response

None. This message is informational.

HFP01178E **DYNAMIC ALLOCATION LIMIT
REACHED. SPECIFY LARGER
DYNAMNBR.**

Explanation

The maximum dynamic allocation limit has been reached.

System action

The process terminates.

User response

Specify the DYNAMNBR parameter on the EXEC statement. A value of DYNAMNBR=999 typically resolves this problem.

HFP01301W **DYNAMIC ALLOCATION FAILED,
SVC 99 REASON CODE '*xxxx*'.**

Explanation

The Online Structure Change utility was unable to dynamically allocate a data set. You can find information about the SVC 99 reason codes in the *z/OS MVS Authorized Assembler Services Guide*.

System action

If the data set is required, processing terminates. Otherwise, processing continues.

User response

Correct the allocation failure and rerun the job.

HFP01400E **XCF CONNECTION TO
xcfgroupmember WAS LOST.**

Explanation

The XCF connection with the indicated XCF group member was lost.

System action

The process terminates.

User response

Collect the job output, including the dump, and contact IBM Software Support.

HFP01501W **DEBUG SETUP FAILED. *reason*,
RC=*rc***

Explanation

The Online Structure Change utility could not activate additional debugging options. *reason* explains why setup failed.

System action

Processing continues with DEBUG off.

User response

This error is likely an internal system error. Contact IBM Software Support.

HFP01502I OSC DEBUG [STARTED | ENDED].

Explanation

This informational message indicates the status of OSC DEBUG.

System action

Processing continues.

User response

None. This message is informational.

**HFP01503I [JOINED | LEFT] XCF GROUP
 xcfgroupname AS MEMBER
 *membername***

Explanation

The job joined or left the indicated XCF group.

System action

Processing continues.

User response

Verify that *xcfgroupname* is the same group name as other address spaces in order to be able to communicate with them.

**HFP01505E OSC IS NOT ACTIVE IN TARGET
 IMS SYSTEM.**

Explanation

The OSC subtask that runs on the target IMS system is not activated. Either the specified IMS system is incorrect or the load module library of IMS Fast Path Solution Pack is not included in the STEPLIB concatenation of the IMS control region job.

System action

The process terminates.

User response

Verify that the target IMS system is correct and that the load module library of IMS Fast Path Solution Pack is included in the STEPLIB concatenation of the IMS control region job. The OSC subtasks must initially be activated in the IMS system with a restart of the IMS control region.

**HFP01511E TARGET MEMBER *membername*
 [NOT ACTIVE | NOT FOUND].**

Explanation

Failed to connect to the Online Structure Change utility component or IMS Tools Online System Interface component in the IMS control region address space.

System action

The process terminates.

User response

If the IMS system is down, it must either be restarted or the SUBSYS record must be removed from DBRC. If the IMS system is active, verify that either the Online Structure Change utility or IMS Tools Online System Interface component was successfully initialized in that system and that the component joined the same XCF group.

See messages HFPX0005I and HFPX0101I for the IMSIDs that are registered to the production DBRC and that have joined the TOSI XCF group.

**HFP01512I CONNECTED WITH TARGET
 MEMBER *membername***

Explanation

The job successfully connected to the corresponding member.

System action

Processing continues.

User response

None. This message is informational.

HFP01513E OSC SUBTASK FAILED RC=*rc*

Explanation

An OSC subtask in the IMS control region terminated unexpectedly.

System action

Online Structure Change processing in the IMS control region is no longer available.

User response

Restart the IMS system to activate the OSC subtasks, and rerun the job. If the problem persists, contact IBM Software Support.

HFP01514I **CHANGE CAPTURE FOR DBD**
(dbdname) action

Explanation

This informational message indicates that capturing log or change records for the DBD has either been started or has just ended.

System action

If capture has been activated, the captured records will begin being sent to the remote OSC subtask. If capture has been stopped, no more change records for the DBD will be sent to the remote OSC subtask.

User response

None. This message is informational.

HFP01515W *message_type* FROM MEMBER
membername

Explanation

An unexpected XCF message was received from the corresponding member. *message_type* shows one of the following values:

NEWMBR

The member that newly joined the XCF group. This type of message is typically received when an IMS system is started during the structure change.

LOSTCONNT

The connection to the member was lost. This type of message is typically received when an IMS system terminates during the structure change or the structure change job abnormally terminates.

System action

If the OSC driver job receives the message, the job terminates. If the message is received in the IMS control region, change capture ends.

User response

None.

HFP01517E **ERROR ENCOUNTERED DURING**
capturetype CAPTURE FOR DBD
dbdname: FUNC=function RC=rc
RSN=rsn

Explanation

The Online Structure Change utility was unable to capture a change record for the indicated DEDB.

System action

Record capture is terminated for the indicated DEDB. The OSC driver job is notified of the records that were being captured. The job ends with an abend code of U1517.

User response

If the parameters in the message are FUNC=IXCMSGO, RC=000C, RSN=0004, an XCF send error, which is caused by an XCF message buffer shortage, occurred. In this case, consider increasing the MAXMSG value on the CLASSDEF or PATHOUT definition. You can find information about XCF message buffers in *z/OS MVS Setting Up a Sysplex*. If the parameters in the message indicate other errors, contact IBM Software Support.

HFP01519I **NUMBER OF RECORDS CAPTURED**
WAS nnnn, TOTAL RECORDS SENT
WAS nnnn

Explanation

This informational message indicates the numbers of records that have been captured and sent.

System action

Processing continues.

User response

None. This message is informational.

HFP01521W **ERROR FROM modulename:**
FUNC=function RC=rc RSN=rsn

Explanation

An error was encountered while capturing records in the IMS control region.

System action

The job ends with an abend code of U1521.

User response

This error is likely an internal system error. Contact IBM Software Support.

HFP01537E **CHANGE CAPTURE ALREADY
ACTIVE FOR DBD *dbdname* ON
*imsid***

Explanation

An OSC driver job was submitted to process the indicated DBD. However, another OSC driver job is already processing the indicated DBD in the IMS system. A DBD cannot be processed by more than two OSC driver jobs at the same time.

System action

The job ends with an abend code of U1537.

User response

Wait for the current OSC driver job to end. Then, if necessary, rerun the other OSC job.

HFP01572E **WAITING FOR DATABASE PAUSE
HAS EXPIRED.**

Explanation

An attempt to pause the database has timed out.

System action

The job ends with an abend code of U1572.

User response

Rerun the OSC driver job. If the error persists, contact IBM Software Support.

HFP01576E **INTERNAL MESSAGE BUFFERS
AREA EXHAUSTED.**

Explanation

The internal message buffers that receive XCF messages are full.

System action

The job ends with an abend code of U1576.

User response

This error is likely an internal system error. Contact IBM Software Support.

HFP01581E **NUMBER OF ONLINE IMS
SUBSYSTEMS HAS CHANGED.**

Explanation

The OSC driver job detected a change in the number of online IMS systems during structure change. See the preceding HFP01515W message to determine whether IMS system was started or terminated.

System action

The job ends with an abend code of U1581.

User response

If you do not plan to start or shutdown any online IMS systems, rerun the OSC driver job. If you received this message as a result of an error or a sudden termination of the IMS control region, correct the problem in the IMS control region and rerun the job.

HFP03713E **OSC CANNOT CONTINUE THE
PROCESSING DUE TO XCF
CONNECT TASK ERRORS.
- FOR DETAILS, SEE THE
HFP3XMSG DD IN THE JOBLOG OF
OSC DRIVER JOB OR OSC
INTERNAL APPLY JOB.**

Explanation

The Online Structure Change utility ended abnormally due to XCF connect task errors.

System action

The job ends with an abend code of U3713.

User response

Review the messages written to the HFP3XMSG DD. Correct the error and rerun the job. If the error persists, contact IBM Software Support.

HFP03714E **TAKEOVER WINDOW HAS
EXPIRED.**

Explanation

The end time that was specified by the TAKEOVER_TIME keyword has been reached but the Online Structure Change utility is not yet ready to perform takeover processing for the job.

System action

The job ends with an abend code of U3714.

User response

See “[Troubleshooting Online Structure Change utility problems](#)” on page 1023 to determine the next action.

HFP1 messages

The following information is about messages and codes that begin with HFP1.

HFP10001I **OSC INTERNAL JOBS ARE BEING GENERATED.**

Explanation

The Online Structure Change utility has started generating OSC internal jobs.

System action

Processing continues.

User response

None. This message is informational.

HFP10050E ***dsname* CANNOT BE SPECIFIED IN THE JCL.**

Explanation

The indicated data set is specified, but it cannot be specified in the JCL.

System action

The return code is set to 8.

User response

Correct the error and rerun the job.

HFP10051E **DYNAMIC ALLOCATION FAILED FOR THE *keyword* KEYWORD.
- DSNAME: *dsname***

Explanation

An attempt to dynamically allocate the data set whose name is specified by the *keyword* failed.

System action

The job ends with an abend code of U3711.

User response

This error is likely an internal system error. Contact IBM Software Support.

HFP10052I **THE DATA SET SPECIFIED BY THE ENQDSN KEYWORD IS BEING**

USED BY ANOTHER JOB. WAITING FOR THE DATA SET TO BE RELEASED.

**- CURRENT RETRY COUNT: *n*
- DSNAME: *dsname***

Explanation

The ENQ data set is being used by another OSC job. The Online Structure Change utility cannot generate internal SYSIN files in the ENQ data set. The Online Structure Change utility is waiting for the other OSC job to release the ENQ data set.

System action

The Online Structure Change utility waits for 20 seconds and attempts to dynamically allocate the ENQ data set. If the ENQ data set is still not released on the fifth attempt, the utility issues message HFP10053E and terminates.

User response

None. This message is informational.

HFP10053E **THE DATA SET SPECIFIED BY THE ENQDSN KEYWORD CANNOT BE ACCESSED.**

**- THE MAX RETRY COUNT (*n*) EXCEEDED.
- DSNAME: *dsname***

Explanation

Failed to dynamically allocate the ENQ data set whose name is specified by the ENQDSN keyword. Another OSC job is using the ENQ data set exclusively.

System action

The job ends with an abend code of U3711.

User response

Rerun the job after the other OSC job completes online change and releases the ENQ data set.

HFP13701E **IMSPLEX NAME OR DBRCGRP NAME IS NOT SPECIFIED IN EXEC PARAMETER.**

Explanation

The name of the IMSplex or DBRCGRP is not found in the EXEC statement. IMSplex name or DBRCGRP name must be supplied as an EXEC statement parameter.

System action

The job ends with an abend code of U3701.

User response

Specify the IMSplex name or the DBRCGRP name in the EXEC statement and rerun the job.

HFP13702E RECON DD CANNOT BE SPECIFIED IN THE JCL.

Explanation

DD statements to specify RECON data sets (RECON DDs) must not be specified in the JCL.

System action

The job ends with an abend code of U3702.

User response

Specify the RECON DD statements as follows:

- Use the RECONx_DSN keywords to specify the RECON data sets used by the production IMS systems.
- Use the RECONx_DSN_APPLY keywords to specify the RECON data sets used by the IMS DBCTL system.

HFP13703E RECON DYNAMIC [ALLOCATION | DEALLOCATION] FAILED.

Explanation

An attempt to dynamically allocate or deallocate the RECON data set failed.

System action

The job ends with an abend code of U3703.

User response

This error is likely an internal system error. Contact IBM Software Support.

HFP13704E RECON ADSLIST REQUEST FAILED.

Explanation

An attempt to issue an ADSLIST request to DBRC failed.

System action

The job ends with an abend code of U3704.

User response

This error is likely an internal system error. Contact IBM Software Support.

HFP13705E THE [FIRST | LAST] ADS IS NOT AVAILABLE.

Explanation

The first or the last ADS is not available in DBRC.

System action

The job ends with an abend code of U3705.

User response

Check the status of ADS in DBRC and ensure that the status of the ADS is available. Correct the error and rerun the job.

HFP13707E THE ORIGINAL ADS HAS 7 COPIES.

Explanation

The original ADS is MADS that has seven copies. The Online Structure Change utility could not register the shadow ADS to DBRC.

System action

The job ends with an abend code of U3707.

User response

Reduce the number of MADS copies and rerun the job.

HFP13708E NO IMSID(SSID) WERE FOUND IN THE SPECIFIED RECON.

Explanation

There is no SSID registered in the specified RECON data set.

System action

The job ends with an abend code of U3708.

User response

Ensure that the SSID is registered to DBRC. Correct the error and rerun the job.

HFP13709E THE IMSID(SSID) FOR PROCESSING WAS NOT FOUND IN THE SPECIFIED RECON.

Explanation

The SSID specified on the IMSID keyword is not registered in the specified RECON data set.

System action

The job ends with an abend code of U3709.

User response

Ensure that the parameter specified for the IMSID keyword is correct. Correct the error and rerun the job.

HFP13710E ERROR OCCURRED WHEN GETTING IMSID(SSID) FROM THE SPECIFIED RECON.

Explanation

An attempt to obtain the SSID from the specified RECON data set failed.

System action

The job ends with an abend code of U3710.

User response

This error is likely an internal system error. Contact IBM Software Support.

HFP13711E PROCESS ENDED ABNORMALLY DUE TO PREVIOUS ERRORS.

HFP2 messages

The following information is about messages and codes that begin with HFP2.

HFP2900I *area_name* IN RECONID=*recon_id* HAS BEEN [REORGANIZED | IOVF EXTENDED | SDEP EXTENDED], AND NO EXCEPTIONS WERE DETECTED AFTER THE [REORG | IOVF EXTEND | SDEP EXTEND].

Explanation

The Online Space Management utility performed the action that is indicated in the message on the DEDB area *area_name*. No exception was detected after the

Explanation

The Online Structure Change utility could not continue the processing because of some errors that occurred in previous processing.

System action

The job ends with an abend code of U3711.

User response

Locate the error messages that are issued before this message and correct the error. Also refer to “Troubleshooting Online Structure Change utility problems” on page 1023 to determine the next action.

HFP13715E RENAME VALIDATION CHECK FAILED.
- THE ORIGINAL ADS NAME IS TOO LONG.
- THE TEMPORARY DATA SET (**.T) ALREADY EXISTS.
- DSNAME: *dsname*

Explanation

Because ORIGADS_PROC=RENAME is specified, the Online Structure Change utility attempted to rename the original ADS but it failed.

System action

The job ends with an abend code of U3715.

User response

Correct the error. Also refer to “Troubleshooting Online Structure Change utility problems” on page 1023 to determine the next action.

System action

Policy Services continues its processing.

User response

No user action is needed because all the detected exceptions were removed by the action indicated in the message.

HFP2901I ***area_name* IN RECONID=*recon_id*
HAS BEEN [REORGANIZED | IOVF
EXTENDED | SDEP EXTENDED],
BUT HAS REMAINING WARNING
EXCEPTIONS.**

Explanation

The Online Space Management utility performed the action that is indicated in the message on the DEDB area *area_name*. Some warning-level exceptions remain after the action. *recon_id* shows the 8-byte RECON ID of the RECON environment to which the area belongs.

System action

Policy Services continues its processing.

User response

Check the exceptions that remain after the action. If you identify that any of those exceptions need to be addressed, follow the guideline in the topic "Analyzing and tuning IMS data entry databases" in the *IMS Solution Packs IMS Database Space Tuning Guide*.

HFP2902I ***area_name* IN RECONID=*recon_id*
HAS WARNING EXCEPTIONS IN
THE REORG DOMAIN.**

Explanation

Policy Services detected exceptions in the DEDB area *area_name*. All the exceptions were warning-level exceptions. *recon_id* shows the 8-byte RECON ID of the RECON environment to which the area belongs.

System action

Policy Services continues its processing.

User response

Check the exceptions that were reported by Policy Services. If you determine that any of those exceptions need to be addressed, follow the guidelines in the topic "Analyzing and tuning IMS data entry databases" in the *IMS Solution Packs IMS Database Space Tuning Guide*.

HFP2903I ***area_name* IN RECONID=*recon_id*
HAS BEEN [REORGANIZED | IOVF
EXTENDED | SDEP EXTENDED],
BUT IS STILL IN A
"remaining_action NEEDED"
STATE.**

Explanation

The Online Space Management utility performed the action that is indicated in the message on the DEDB area *area_name*. However, the area is still in need of the action that is indicated by *remaining_action*. *recon_id* shows the 8-byte RECON ID of the RECON environment to which the area belongs.

System action

Policy Services continues its processing.

User response

Check the critical-level exceptions that remain after the action and any accompanying exceptions that are in lower severity. Then, follow the guideline in the topic "Analyzing and tuning IMS data entry databases" in the *IMS Solution Packs IMS Database Space Tuning Guide*.

HFP2904I ***area_name* IN RECONID=*recon_id*
HAS BEEN [REORGANIZED | IOVF
EXTENDED | SDEP EXTENDED],
BUT HAS REMAINING CRITICAL
EXCEPTIONS.**

Explanation

The Online Space Management utility performed the action that is indicated in the message on the DEDB area *area_name*. However, the area is still in critical state. *recon_id* shows the 8-byte RECON ID of the RECON environment to which the database resource belongs.

System action

Policy Services continues its processing.

User response

Check the critical-level exceptions that remain after the action and any accompanying exceptions that are in lower severity. Then, follow the guideline in the topic "Analyzing and tuning IMS data entry databases" in the *IMS Solution Packs IMS Database Space Tuning Guide*.

HFP2905I ***area_name* IN RECONID=*recon_id*
HAS CRITICAL EXCEPTION IN THE
REORG DOMAIN. USER ACTION IS
REQUIRED.**

Explanation

Policy Services detected a critical-level exception in the DEDB area *area_name*. *recon_id* shows the 8-byte RECON ID of the RECON environment to which the area belongs.

System action

Policy Services continues its processing.

User response

Check the critical-level exceptions that were reported and any accompanying exceptions that are in lower severity. Then, follow the guidelines in the topic "Analyzing and tuning IMS data entry databases" in the *IMS Solution Packs IMS Database Space Tuning Guide* to resolve the critical-level exceptions.

HFP2906I THE [REORGANIZATION | IOVF EXTENSION | SDEP EXTENSION] OF *area_name* IN RECONID=*recon_id* WAS ATTEMPTED AND FAILED.

Explanation

The Online Space Management utility detected one or more critical-level exceptions in the area *area_name*. The utility attempted the action indicated in the message, but it failed. *recon_id* shows the 8-byte RECON ID of the RECON environment to which the area belongs.

System action

Policy Services continues its processing.

User response

Because the action failed and the detected exceptions remain, identify the cause of the error and then run the job that performs the failed process.

HFP2907I *area_name* IN RECONID=*recon_id* HAS BEEN [REORGANIZED | IOVF EXTENDED | SDEP EXTENDED], BUT HAS REMAINING SEVERE EXCEPTIONS.

Explanation

The Online Space Management utility detected one or more critical-level exceptions in the area *area_name*. The utility attempted the action indicated in the message, but one or more severe-level exceptions remain. *recon_id* shows the 8-byte RECON ID of the RECON environment to which the area belongs.

System action

Policy Services continues its processing.

User response

Check the severe-level exceptions that remain after the action and any accompanying warning-level exceptions. If you identify that any of those exceptions need to be addressed, follow the guideline in the topic "Analyzing and tuning IMS data entry databases" in the *IMS Solution Packs IMS Database Space Tuning Guide*.

HFP2908I *area_name* IN RECONID=*recon_id* HAS SEVERE EXCEPTIONS IN THE REORG DOMAIN.

Explanation

Policy Services detected one or more severe-level exceptions in the DEDB area *area_name*. However, no critical level exception was reported. *recon_id* shows the 8-byte RECON ID of the RECON environment to which the area belongs.

System action

Policy Services continues its processing.

User response

Check the severe-level exceptions that were detected and any accompanying warning-level exceptions. If you determine that any of those exceptions need to be addressed, follow the guidelines in the topic "Analyzing and tuning IMS data entry databases" in the *IMS Solution Packs IMS Database Space Tuning Guide*.

HFP2909I *area_name* IN RECONID=*recon_id* HAS BEEN [REORGANIZED | IOVF EXTENDED | SDEP EXTENDED], BUT ITS STATUS IS UNAVAILABLE.

Explanation

The Online Space Management utility performed the action indicated in the message on the DEDB area *area_name*. However, the sensor data of the area after the action is not available for some reason.

System action

Policy Services continues its processing.

User response

Identify the cause of the error why the sensor data is not available. Then, correct the error and run a DB Sensor job to collect and store the latest sensor data.

HFP2910I ACTION "[REORG | EXTENDIOVF | EXTENDSDEP]" IS

**RECOMMENDED FOR *area_name*
IN RECONID=*recon_id*.**

Explanation

Policy Services detected one or more critical-level exceptions in the DEDB area *area_name* and recommends the action that is indicated in the message. *recon_id* shows the 8-byte RECON ID of the RECON environment to which the area belongs.

HFP4 messages

The following information is about messages and codes that begin with HFP4.

HFP40001I *processname* **PROCESS STARTED
FOR DBD *dbdname*.**

Explanation

The indicated process has started for the indicated DEDB.

System action

Processing continues.

User response

None. This message is informational.

HFP40002I *processname* **PROCESS
COMPLETED FOR DBD *dbdname*.**

Explanation

The indicated process has completed for the indicated DEDB.

System action

Processing continues.

User response

None. This message is informational.

HFP40004E **ASMTDLI CALL ERROR FOR DBD
dbdname
- FUNC CODE: *funccode*, RET
CODE: *rc***

Explanation

An ASMTDLI call error occurred during the apply process.

System action

The return code is set to 8.

System action

Policy Services continues its processing.

User response

Follow the guideline in the topic "Analyzing and tuning IMS data entry databases" in the *IMS Solution Packs IMS Database Space Tuning Guide* to resolve the reported exceptions.

User response

This error is likely an internal system error. Contact IBM Software Support.

HFP40005E **DBD *dbdname* WAS NOT FOUND IN
ACB MEMBER.**

Explanation

The indicated DEDB was not found in the ACB member that is referred to by the IMS DBCTL system.

System action

The return code is set to 8.

User response

Ensure that the ACB member used in the IMS DBCTL system is correct. Correct the error and rerun the job.

HFP40006I **RECEIVED LAST USER LOG
RECORD FOR DBD *dbdname*, IMS
OLDEST TIMESTAMP IS *timestamp***

Explanation

The OSC internal apply job received the last user log record.

System action

Processing continues.

User response

None. This message is informational.

HFP40008E **SOME DB RECORDS WERE NOT
RANDOMIZED TO RESTRUCTURED
ADS. (INPUT AREA NO: *areanum1*,
OUTPUT AREA NO: *areanum2*)**

Explanation

Some database records were randomized to an area that is not specified with the OAREA keyword.

System action

The return code is set to 8 and processing continues.

User response

Specify, for the OAREA keyword, the output area number that is shown in this message (*areanum2*).

HFP40009I **RECEIVED USER LOG RECORD
DATA SET *ddname*. IMS OLDEST
TIMESTAMP: *timestamp*, COUNT:
*nnnnn***

Explanation

The OSC internal apply job opened the user log data set.

System action

Processing continues.

User response

None. This message is informational.

HFP40010I **TYPE OF SORT INPUT DATA.
TOTAL RECORD COUNT: *nnnnn*
- *type* LATEST TIMESTAMP:
timestamp
- TOTAL COUNT: *nnnnn*, APPLY
AVAILABLE: *nnnnn*, APPLY
UNAVAILABLE: *nnnnn*
- OTHER RECORDS: *nnnnn***

Explanation

Shows the number of each user log record type.

System action

Processing continues.

User response

None. This message is informational.

HFP40011I **COMPLETED READING USER LOG
RECORD DATA SET *ddname***

Explanation

The OSC internal apply job has completed reading the indicated user log record data set.

System action

Processing continues.

User response

None. This message is informational.

HFP40050E **ERROR IN CALL TO RANDOMIZER:
*modulename***

Explanation

An error occurred while calling the indicated randomizing module.

System action

The return code is set to 8.

User response

This error is likely an internal system error. Contact IBM Software Support.

HFP40100I **UNEXPECTED USER LOG RECORD
WAS FOUND FOR DBD *dbdname*
- FUNC CODE: *nnnn***

Explanation

Detected an unexpected user log record for the indicated DEDB. This user log record is ignored.

System action

Processing continues.

User response

None. This message is informational.

HFP43717E **INVALID USER LOG RECORD WAS
FOUND.**

Explanation

A user log record, whose corresponding SYNC call or ROLB call has not been issued, was found.

System action

The job ends with an abend code of U3717.

User response

This error is likely an internal system error. Contact IBM Software Support.

HFP43718E DETECTED AN ERROR USER LOG FOR DBD *dbdname*

Explanation

Detected an error user log record that cannot be applied to the restructured ADS.

System action

The job ends with an abend code of U3718.

User response

This error is likely an internal system error. Contact IBM Software Support.

HFPA messages

The following information is about messages and codes that begin with HFPA.

HFPA0001I ANALYZE PROCESSING STARTED FOR DATABASE: *databasename*, AREA NO: *nnnn*, AREANAME: *areaname*

Explanation

The Analyze process for area number *nnnn*, area *areaname* has started.

System action

Processing continues.

User response

None. This message is informational.

HFPA0002I NO POINTER ERRORS WERE DETECTED IN DATABASE: *databasename*, AREA NO: *nnnn*, AREANAME: *areaname*

Explanation

The Analyze process for area number *nnnn*, area *areaname* has completed successfully.

System action

Processing continues.

User response

None. This message is informational.

HFPA0003E POINTER ERRORS WERE DETECTED IN DATABASE: *databasename*, AREA NO: *nnnn*, AREANAME: *areaname*

Explanation

The Analyze process for area number *nnnn*, area *areaname* has completed, but some pointer errors were detected.

System action

The return code is set to 8 and processing continues.

User response

For further details, see “Resolve Pointers report” on page 585. If this message was issued in an OPC job, also see “Online Pointer Scan report” on page 704.

HFPA0004I ANALYZE PROCESSING COMPLETED FOR DATABASE: *databasename*, AREA NO: *nnnn*, AREANAME: *areaname*

Explanation

The Analyze process for area number *nnnn*, area *areaname* has completed.

System action

Processing continues.

User response

None. This message is informational.

HFPA0005E ANALYZE PROCESSING CANCELED FOR DATABASE: *databasename*, AREA NO: *nnnn*, AREANAME: *areaname*

Explanation

The Analyze process for area number *nnnn*, area *areaname* has been canceled because the number of pointer errors or bad CIs is beyond the limit.

System action

The return code is set to 8 and processing continues.

User response

For further details, see “Resolve Pointers report” on page 585.

HFPA0008E THE NUMBER OF ERRORS IS BEYOND THE LIMIT IN DATABASE: *databasename*, AREA NO: *nnnn*, AREANAME: *areaname*

Explanation

The number of pointer errors is beyond the limit for area number *nnnn*, area *areaname*.

System action

The return code is set to 8 and processing continues.

User response

For further details, see [“Resolve Pointers report”](#) on page 585.

HFPA0009E STORAGE RELEASE FAILED FOR ANALYZE PROCESSING - DATABASE: *databasename*, AREA NO: *nnnn*, AREANAME: *areaname*

Explanation

Program FREEMAIN failed in the Analyze process.

System action

The return code is set to 8 and processing continues.

User response

Correct the errors, and rerun the job. If this situation persists, contact IBM Software Support.

HFPA0010E STORAGE ALLOCATION FAILED FOR ANALYZE PROCESSING - DATABASE: *databasename*, AREA NO: *nnnn*, AREANAME: *areaname*

Explanation

Program GETMAIN failed in the Analyze process.

System action

The return code is set to 8 and processing continues.

User response

If the region size specified is too small, increase the REGION size on the JOB statement in the JCL, and rerun the utility.

HFPA0011E DYNAMIC ALLOCATION FAILED FOR DDNAME: *ddname*

- DATABASE: *databasename*, AREA NO: *nnnn*, AREANAME: *areaname*

Explanation

An attempt to dynamically allocate the *ddname* data set failed.

System action

The return code is set to 8 and processing continues.

User response

This error is likely an internal system error. Contact IBM Software Support.

HFPA0012E OPEN FAILED FOR DDNAME: *ddname* - DATABASE: *databasename*, AREA NO: *nnnn*, AREANAME: *areaname*

Explanation

OPEN for the *ddname* data set failed.

System action

The return code is set to 8 and processing continues.

User response

Correct the errors, and rerun the job. If this situation persists, contact IBM Software Support.

HFPA0013W STORAGE ALLOCATION FAILED FOR CISNAP REQUEST - DATABASE: *databasename*, AREA NO: *nnnn*, AREANAME: *areaname*

Explanation

Program GETMAIN failed to obtain space for the CISNAP request. The Analyze process detected some pointer errors in the area but the CI Map/CI Dump report was not provided.

System action

The return code is set to 4 and processing continues.

User response

If the region size specified is too small, increase the REGION size on the JOB statement in the JCL. Then rerun the utility.

HFPA0014W CI MAP/DUMP FUNCTION IS INTERRUPTED. REASON CODE *nn*

- **DATABASE:** *dbname*, **AREA NO:** *nnnn*, **AREANAME:** *areaname*

Explanation

The CI map/CI dump function is interrupted because of one of the following reasons:

Reason code <i>nn</i>	Explanation
1	GETMAIN failed to obtain space for internal blocks.
2	50 CI maps/dumps are already generated.
3	GET failed for the area data set or the STOP REGION command was issued.
4	All entries in block map table (internal table) have been consumed.

System action

The return code is set to 4 and CI map/CI dump function for the area no longer works.

User response

Correct any errors, and rerun the job.

HFFPA0015I **SOME ORPHAN SDEP SEGMENTS WERE DETECTED IN DATABASE: *dbname*, AREA NO: *nnnn*, AREANAME: *areaname***

Explanation

The Analyze process detected some orphan SDEP segments and issued an information message because SDEP_ORPHAN_MSGLVL=INFORMATIONAL is specified.

System action

Processing continues.

User response

None. This message is informational.

HFFPA0016W **SOME ORPHAN SDEP SEGMENTS WERE DETECTED IN DATABASE: *dbname*, AREA NO: *nnnn*, AREANAME: *areaname***

Explanation

The Analyze process detected some orphan SDEP segments and issued a warning message because SDEP_ORPHAN_MSGLVL=WARNING is specified.

System action

The return code is set to 4 and processing continues.

User response

For further details, see [“Resolve Pointers report”](#) on page 585.

HFFPA0017E **SOME ORPHAN SDEP SEGMENTS WERE DETECTED IN DATABASE: *dbname*, AREA NO: *nnnn*, AREANAME: *areaname***

Explanation

The Analyze process detected some orphan SDEP segments and issued an error message because SDEP_ORPHAN_MSGLVL=ERROR is specified.

System action

The return code is set to 8 and processing continues.

User response

For further details, see [“Resolve Pointers report”](#) on page 585.

HFFPA0018W **ANALYSIS REPORT PROCESSING ENDED WITH WARNINGS FOR DATABASE: *dbname*, AREA NO: *nnnn*, AREANAME: *areaname***

Explanation

The analysis report process for the database *dbname*, area number *nnnn*, area *areaname* has completed, but some warning messages were issued.

System action

The return code is set to 4 and processing continues.

User response

For further details, see [“Process Summary of Analysis report”](#) on page 590.

HFFPA0019E **ANALYSIS REPORT PROCESSING ENDED WITH ERRORS FOR DATABASE: *dbname*, AREA NO: *nnnn*, AREANAME: *areaname***

Explanation

The analysis report process for the database *dbname*, area number *nnnn*, area *areaname* has completed, but some error messages were issued.

System action

The return code is set to 8 and processing continues.

User response

For further details, see “Process Summary of Analysis report” on page 590.

HFPAA0020I	ANALYZE PROCESSING STARTED FOR SECONDARY INDEX DATABASE: <i>dbname</i>
-------------------	---

Explanation

The Analyze process for secondary index database *dbname* has started.

System action

Processing continues.

User response

None. This message is informational.

HFPAA0021I	NO ERRORS WERE DETECTED IN SECONDARY INDEX DATABASE: <i>dbname</i>
-------------------	---

Explanation

The Analyze process for secondary index database *dbname* has completed successfully.

System action

Processing continues.

User response

None. This message is informational.

HFPAA0022E	ERRORS WERE DETECTED IN SECONDARY INDEX DATABASE: <i>dbname</i>
-------------------	--

Explanation

The Analyze process for secondary index database *dbname* has completed, but some errors were detected.

System action

The return code is set to 8 and processing continues.

User response

For further details, see HFPAA0024E message and HFPAA0025E message.

HFPAA0023E	ANALYZE PROCESSING WAS CANCELED FOR SECONDARY INDEX DATABASE: <i>dbname</i> - ANALYZE ENDED WITH ERRORS IN DATABASE: <i>dbname</i>
-------------------	---

Explanation

The Analyze process for secondary index database *dbname* has been canceled because the number of errors has exceeded the internal limit.

System action

The return code is set to 8 and processing continues.

User response

None.

HFPAA0024E	INDEX POINTER SEGMENT THAT POINTS TO THE CORRESPONDING TARGET SEGMENT IS NOT FOUND. - AREA NO. <i>nnnn</i>, AREA NAME: <i>areaname</i> - SECONDARY INDEX DATABASE: <i>index_dbname</i>, DDNAME: <i>ddname</i> - SECONDARY INDEX DATABASE: <i>index_dbname</i>, DDNAME: <i>ddname</i>, OVERFLOW DDNAME: <i>overflow_ddname</i>
-------------------	--

Explanation

The index pointer segment that points to the corresponding target segment is not found.

System action

The return code is set to 8 and processing continues.

User response

The secondary index database must be rebuilt or recovered.

HFPAA0025E	TARGET SEGMENT THAT IS POINTED BY THE CORRESPONDING INDEX
-------------------	--

POINTER SEGMENT IS NOT FOUND.
 - **SECONDARY INDEX DATABASE:** *index_database*, **DDNAME:** *ddname*
 - **SECONDARY INDEX DATABASE:** *index_database*, **DDNAME:** *ddname*, **OVERFLOW DDNAME:** *overflow_ddname*

Explanation

The target segment that is pointed by the corresponding index pointer segment is not found.

System action

The return code is set to 8 and processing continues.

User response

The secondary index database must be rebuilt or recovered.

HFPAA0026I **ANALYZE PROCESSING WAS CANCELED FOR SECONDARY INDEX DATABASE:** *database*
 - **INPUT DATA SET COULD NOT BE OPENED**
 - **THE NUMBER OF POINTER SEGMENTS THAT SHOULD BE IN area IS ZERO. SOURCE SEGMENT:** *source_segment*
 - **THE NUMBER OF POINTER SEGMENTS THAT SHOULD BE IN area IS NOT ZERO. SOURCE SEGMENT:** *source_segment*

Explanation

Analyze processing for the indicated secondary index database was canceled because the input data set could not be opened.

System action

Processing continues.

User response

If the data set has never been accessed, the data set cannot be opened because the data set is empty. If this is the case, you can ignore this message. In other cases, this message indicates that an open error has occurred. Correct the error condition and rerun the job for the secondary index database.

HFPAA0027W **POINTER SEGMENT DUMP PROCESSING FOR SECONDARY**

INDEX DATABASE: database WAS SKIPPED.
 - **GETMAIN FAILED.**
 - **DYNAMIC ALLOCATION FAILED.**
 - **OPEN FAILED.**
 - **SORT ENDED ABNORMALLY. (RC: code, TCT: tctname, MSGDDN: msg_ddname)**

Explanation

Pointer segment dump processing was skipped.

System action

Processing continues.

User response

Correct any errors, rerun the job.

HFPAA0028I **TUNE function PROCESSING STARTED FOR DATABASE:** *database*, **AREA NO:** *nnnn*, **AREANAME:** *areaname*
 - **INPUT DATA SET IS IMAGE COPY.**

Explanation

The Tune function process for the indicated area started. If the input data set is an image copy, subtext INPUT DATA SET IS IMAGE COPY is also printed. Otherwise, it is not printed.

function indicates one of the following functions:

SCAN

Scans the database.

READ

Reads the tuning aid record generated by SCAN function.

SIMULATE

Simulates the database with the specified conditions.

System action

Processing continues.

User response

None. This message is informational.

HFPAA0029I **TUNE function PROCESSING COMPLETED FOR DATABASE:** *database*, **AREA NO:** *nnnn*, **AREANAME:** *areaname*

Explanation

The Tune function process for the indicated area completed successfully.

function indicates one of the following functions:

SCAN

Scans the database.

READ

Reads the tuning aid record generated by SCAN function.

SIMULATE

Simulates the database with the specified conditions.

System action

Processing continues.

User response

None. This message is informational.

HFPAA0030E TUNE SCAN PROCESSING
CANCELED FOR DATABASE:
database, AREA NO: *nnnn*,
AREANAME: *areaname*

Explanation

The SCAN process of the Tune function for the indicated area was canceled because the number of pointer errors or bad CIs is beyond the limit.

System action

The return code is set to 8 and processing continues.

User response

Examine the Resolve Pointers report and investigate the cause of the error. See [“Resolve Pointers report” on page 585](#) to interpret the report.

HFPAA0031I NO ROOT SEGMENT IS ASSIGNED
TO AREA *areaname*

Explanation

As a result of the Tune function, FPA detected that no root segment is assigned to the indicated area. This area is specified by the OAREA keyword of the SIMULATE subcommand.

System action

Processing continues.

User response

None. This message is informational.

HFPAA0032I SYNTBL=*nn* IN HFPAPARM DD IS
CHANGED TO *mm*, WHICH IS THE
LOWER_BOUND VALUE OF
RAP_LONGSYN_PCT PLUS 1.

Explanation

Because the value of the SYNTBL keyword in the HFPAPARM DD or the default value of the SYNTBL keyword is less than or equal to the value of the LOWER_BOUND value that you specified with the RAP_LONGSYN_PCT keyword, FPA replaced the SYNTBL keyword value with the value of LOWER_BOUND plus 1.

System action

FPA replaces the SYNTBL keyword value and continues processing.

User response

None. This message is informational.

HFPAA0033W NULL RBA VALUE DETECTED IN
ONE OR MORE FIRST DOVF CIS.
DATABASE: *datasename*, AREA
NO: *nnnn*, AREANAME: *areaname*

Explanation

The DOVFNULLPTR_MSGLVL=WARNING option is enabled and the Analyze process detected a null pointer (RBA) value for the current overflow CI in the first DOVF CI in one or more UOWs.

System action

The return code is set to 4 and processing continues.

User response

To update the RBA value in the first DOVF CI, use the Online Expert Reorganization utility or the High-Speed DEDB Direct Reorganization utility and reorganize the area.

HFPAA0034I FOUND SDEP POINTERS THAT
POINT TO THE IOVF PORTION.
DATABASE: *datasename*, AREA
NO: *nnnn*, AREANAME: *areaname*

Explanation

The Analyze process detected an SPCF or an SPTF pointer that points to the IOVF portion. This message is issued because SDEP_PTRIOVF_MSGLVL=INFORMATIONAL is specified.

System action

Processing continues.

User response

None. This message is informational.

**HFFPA3400E ANALYZE SUB TASK ENDED
 ABNORMALLY.**

Explanation

The Analyze subtask ended abnormally.

System action

The system action differs between functions:

- For the Change, the Reload, and the Reorg functions with the PTRCHKLVL=FULL option, the return code is set to 12 and processing continues.
- For other functions including the OPC utility, the job ends with an abend code of U3003 or U3006.

User response

Check the preceding message and determine the cause of the analyze subtask error. Correct any errors, and rerun the job.

**HFFPA3401E SORT ENDED ABNORMALLY. (RC:
 code, TCT: tctname, MSGDDN:
 msg_ddname)**

Explanation

The SORT program that was internally invoked ended with an error. *code* is the return code from the SORT program.

System action

The system action differs between functions:

- For the Change, the Reload, and the Reorg functions with the PTRCHKLVL=FULL option, the return code is set to 12 and processing continues.
- For other functions including the OPC utility, the job ends with an abend code of U3003 or U3006.

User response

Check the message and return code from the SORT program. Correct the problem, then rerun the job. If the SORT program issued message ICE046A, specify the SORTOPT_FILSZ=AUTO option or DD statements for the intermediate storage data sets for the SORT program and rerun the job. For more information about intermediate storage data sets for the SORT program, see the DD statements topic of the function that you are using.

**HFFPA3402E TUNING AID RECORD DATA SET IS
 EMPTY. DDNAME: ddname**

Explanation

Either the tuning aid record data set that is specified by the indicated DD is empty or an incorrect data set is specified as the tuning aid record data set.

System action

The job ends with an abend code of U3003.

User response

Specify the correct tuning aid record data set and rerun the job.

**HFFPA3403E INCORRECT TUNING AID RECORD
 DATA SET IS SPECIFIED.
 DDNAME: ddname
 - THE HEADER RECORD MUST
 CONTAIN THE FOLLOWING
 DATABASE NAME: database
 - THE HEADER RECORD MUST
 CONTAIN THE FOLLOWING AREA
 NAME: areaname**

Explanation

The tuning aid record data set that is specified by the indicated DD is not the correct tuning aid record data set for the indicated resource.

System action

The job ends with an abend code of U3003.

User response

Specify the correct tuning aid record data set and rerun the job.

**HFFPA3404E SEQUENCE ERROR DETECTED IN
 TUNING AID RECORD DATA SET.
 DDNAME: ddname**

Explanation

The tuning aid record data set that is specified by the indicated DD does not contain the correct tuning aid record.

System action

The job ends with an abend code of U3003.

User response

Specify the correct tuning aid record data set and rerun the job.

HFPA3405E	OUT OF SPACE WHILE ATTEMPTING TO ALLOCATE IOVF CI. - AREA NO: nnnn, AREANAME: areaname
------------------	---

Explanation

The Tune function attempted a pseudo segment insertion for the area model. The UOW that is being processed required an additional IOVF CI, but none were available because the area is full.

System action

The job ends with an abend code of U3003.

User response

Modify the tuning parameters so that all segments are inserted into the area model, and then rerun the job.

HFPA3406E	TUNE PROCESSING INITIALIZATION FAILED. - GETMAIN FAILED. - FILE OPEN ERROR. - HFPAPARM ERROR. - DYNAMIC ALLOCATION FAILED.
------------------	---

HFPB messages

The following information is about messages and codes that begin with HFPB.

HFPB0001I	INDEXBLD SCAN PROCESSING STARTED FOR AREA NO: nnnn, AREANAME: areaname - INPUT DATA SET IS IMAGE COPY.
------------------	---

Explanation

The scan phase of the Build Index process for the secondary index database has started. When the input data set is an image copy, message INPUT DATA SET IS IMAGE COPY is issued.

Explanation

This message consists of multiple parts. These parts are:

- **GETMAIN FAILED.**
GETMAIN to obtain space for internal blocks failed.
- **FILE OPEN ERROR.**
OPEN processing for the file that is associated with the Analyze process failed.
- **HFPAPARM ERROR.**
Some errors were detected in the HFPAPARM data set.
- **DYNAMIC ALLOCATION FAILED.**
An attempt to dynamically allocate the internal work data set failed.

System action

The job ends with an abend code of U3003.

User response

- **GETMAIN FAILED.**
If the region size specified is too small, increase the REGION size on the JOB statement in the JCL, and rerun the utility.
- **FILE OPEN ERROR.**
Correct any errors, and rerun the job. If this situation persists, contact IBM Software Support.
- **HFPAPARM ERROR.**
Correct the keyword specification in the HFPAPARM data set.
- **DYNAMIC ALLOCATION FAILED.**
This error is likely an internal system error. Contact IBM Software Support.

If the tuning aid records exist in the tuning aid record data set, then you can run the TUNE function with AREASCAN=NO after you correct the errors.

System action

Processing continues.

User response

None. This message is informational.

HFPB0002I	INDEXBLD SCAN PROCESSING COMPLETED FOR AREA NO: nnnn, AREANAME: areaname (ELAPSED TIME: hh:mm:ss.tt)
------------------	---

- THE AREA WAS NOT SCANNED BECAUSE THE AREA IS EMPTY.
- THE AREA WAS SCANNED. THE AREA IS EMPTY.

Explanation

The scan phase of the Build Index process for the secondary index database has completed. ELAPSED TIME shows the time spent for the processing.

- THE AREA WAS NOT SCANNED BECAUSE THE AREA IS EMPTY.

The Build Index function did not scan the area because the area is flagged as empty.

- THE AREA WAS SCANNED. THE AREA IS EMPTY.

The Build Index function scanned the area because the area is not flagged as empty. But the area contained no segments.

System action

Processing continues.

User response

None. This message is informational.

HFPB0003I INDEXBLD [LOAD | COMPARE | UPDATE | FS_RECLAIM] PROCESSING STARTED FOR SECONDARY INDEX DBD: *index_dbdname*

Explanation

The load phase, the compare phase, the update phase, or the freespace reclaim phase of the Build Index process for the indicated secondary index database started.

System action

Processing continues.

User response

None. This message is informational.

HFPB0004I INDEXBLD [LOAD | COMPARE | UPDATE | FS_RECLAIM] PROCESSING COMPLETED FOR SECONDARY INDEX DBD: *index_dbdname* (ELAPSED TIME: *hh:mm:ss.tt*)
- THE SECONDARY INDEX DATABASE DID NOT NEED TO BE UPDATED. ALREADY IN SYNC WITH THE DEDB.

Explanation

The load phase, the compare phase, the update phase, or the freespace reclaim phase of the Build Index process for the indicated secondary index database completed. ELAPSED TIME shows the time spent for the processing. When the secondary index database is already in sync with the DEDB, the subtext is printed.

System action

Processing continues.

User response

None. This message is informational.

HFPB0005I DSNAME FOR *ddname* DD IS NOT REGISTERED TO DBRC.

Explanation

The specified secondary index database is not registered to DBRC.

System action

Processing continues.

User response

None. This message is informational.

HFPB0006W RESYNC POINTER SEGMENT DUMP PROCESSING FOR SECONDARY INDEX DATABASE: *index_dbdname* WAS SKIPPED.
- OPEN FAILED FOR DDNAME: *ddname*

Explanation

Failed to open the data set that is indicated by *ddname*. This DD points to the resync pointer segment record data set for the indicated secondary index database.

System action

The program skips processing of the indicated secondary index, sets the end-of-job return code to 4, and continues processing the next secondary index database.

User response

Correct the errors, and rerun the job. If this situation persists, contact IBM Software Support.

HFPB0007E **IMS IS NOT AT THE REQUIRED MAINTENANCE LEVEL TO SUPPORT [TOICTL=NONE | IDXPROC=FS_RECLAIM].**

Explanation

The IMS system is not at the required maintenance level to support the TOICTL=NONE option or the IDXPROC=FS_RECLAIM option.

System action

FPA issues message HFPT3309E and ends the job with an abend code of U3309.

User response

To activate the TOICTL=NONE option, apply the PTF for APAR PI39873 to IMS 14.

To activate the IDXPROC=FS_RECLAIM option, apply the PTF for APAR PI62621 to IMS 14.

HFPB0008E **OPEN FAILED IN THE INDEX RESYNC PROCESS. DDNAME: *ddname***

Explanation

FPA could not open the data set that the indicated DD statement points to.

System action

FPA issues message HFPT3309E and ends the job with an abend code of U3309.

User response

This error is likely an internal error. Contact IBM Software Support.

HFPB0009E **GETMAIN FAILED IN THE INDEX RESYNC PROCESS.**

Explanation

GETMAIN failed while processing the DL/I task.

System action

FPA issues message HFPT3309E and ends the job with an abend code of U3309.

User response

If the region size that is specified is too small, increase the REGION size on the JOB statement in the JCL. Then rerun the utility.

HFPB0010E **A POINTER SEGMENT FOR DLET WAS NOT FOUND. SECONDARY INDEX DATABASE: *index_dbdname***

Explanation

A pointer segment to be deleted was not found in the indicated secondary index database.

System action

The program skips processing of the indicated secondary index, sets the end-of-job return code to 8, and continues processing the next secondary index database.

User response

If RESYNCDUMP=YES is specified, identify the failed segment in the Resync Pointer Segment Dump report. Correct the errors, and rerun the job.

HFPB0011E **INCORRECT RESYNC POINTER SEGMENT RECORD DATA SET IS SPECIFIED.**
- SPECIFIED RESYNC POINTER SEGMENT RECORD DATA SET NAME: *dsname*
- THE HEADER RECORD MUST CONTAIN THE FOLLOWING SECONDARY INDEX DATABASE NAME: *index_dbdname*
- THE HEADER RECORD MUST CONTAIN THE FOLLOWING RESYNC POINTER SEGMENT RECORD DATA SET DDNAME: *S0nnnnnR*
- THE RECORD TYPE IS NOT OF A RESYNC POINTER SEGMENT RECORD.

Explanation

An incorrect resync pointer segment record data set is specified as input. To run the job in update mode (RESYNCMODE=UPDATE), a valid resync pointer segment record data set must be provided.

System action

The program skips processing of the indicated secondary index, sets the end-of-job return code to 8,

and continues processing the next secondary index database.

User response

Specify the correct resync pointer segment record data set and rerun the job.

**HFPB0012E DUPLICATE KEYS FOUND IN
SECONDARY INDEX DATABASE
dbdname**

Explanation

One or more duplicate keys were found in the indicated secondary index database during the Build Index process.

System action

The return code is set to 8 and processing continues.

User response

Duplicate keys are reported in the Secondary Index Duplicate Key report. Identify the duplicate keys from the report and remove the records with duplicate keys. Then, rerun the job. To interpret the report, see “Secondary Index Duplicate Key report” on page 627.

**HFPB0013E NUMBER OF DUPLICATE KEYS
REACHED THE LIMIT DEFINED BY
DUPKEYMAX. SECONDARY INDEX
DATABASE: dbdname**

Explanation

The number of duplicate keys that were found in the indicated secondary index database reached the maximum allowable number of duplicate keys that is defined by the DUPKEYMAX keyword.

System action

If PROC_AFT_DUPKEY=CONT is specified, FPA sets the return code to 8 and continues processing the remaining secondary index databases. If PROC_AFT_DUPKEY=STOP is specified, FPA ends the job with a return code of 8.

User response

Identify the cause of the key duplication. Correct the errors and rerun the job. For more information about duplicate keys, see “Secondary Index Duplicate Key report” on page 627.

**HFPB0014I PROC_AFT_DUPKEY=STOP IS
SPECIFIED. INDEXBLD**

**PROCESSING IS CANCELED FOR
ALL SECONDARY INDEX
DATABASES.**

Explanation

PROC_AFT_DUPKEY=STOP is specified and the number of duplicate keys in a secondary index database reached the maximum allowable number of duplicate keys that is defined by the DUPKEYMAX keyword. The Build Index function or the Resync function cancels the processing for the remaining secondary index databases.

System action

Processing continues.

User response

None. This message is informational.

**HFPB0015E SECONDARY INDEX DUPLICATE
KEY REPORT FOR dbdname IS NOT
GENERATED BECAUSE OF THE
FOLLOWING REASON:
- DYNAMIC ALLOCATION FAILED
FOR DDNAME ddname, RC=rc,
RSN=rsn**

Explanation

The Build Index function or the Resync function could not obtain the information that is required to generate the Secondary Index Duplicate Key report for the indicated secondary index database. The reason is described in the subtext.

System action

The Build Index function or the Resync function does not generate the Secondary Index Duplicate Key report for the indicated secondary index database. Processing continues.

User response

Look up the dynamic allocation code (SVC99) in the *MVS Programming: Assembler Services Reference*. Correct the problem, then rerun the job.

**HFPB0016I INDEXBLD FS_RECLAIM
PROCESSING SKIPPED FOR
SECONDARY INDEX DBD:
index_dbdname
- THE DATABASE ORGANIZATION
TYPE IS HISAM AND THE
OVERFLOW DATA SET IS DEFINED.**

- THE DATABASE ORGANIZATION TYPE IS SHISAM.

Explanation

The freespace reclaim phase of the Build Index process for the indicated secondary index database was skipped. Subtext shows the reason why the resource was skipped.

System action

Processing continues.

User response

None. This message is informational.

HFPB3500E AN ERROR WAS RETURNED FROM USER PARTITION SELECTION EXIT ROUTINE: *routine*, FUNCTION: *function*, RC: *rc*
- TARGET SEGMENT: *segment*, XDFLD: *xdfld*
- AREANAME: *areaname*, SOURCE SEGMENT: *source_segment*, RBA: *nnnnnnnn*

Explanation

An error was returned from the indicated user partition selection exit routine.

System action

The job ends with an abend code of U3003.

User response

Check the indicated user partition selection exit, the function, and the return code. Correct the errors and rerun the job.

HFPB3501E THE DBDNAME RETURNED FROM USER PARTITION SELECTION EXIT ROUTINE: *routine* IS NOT FOUND IN THE DBD DEFINITION.
- AREA NO: *areanum*, AREANAME: *areaname*, SECONDARY INDEX DBD: *dbdname*, XDLFD: *xdfld*
- TARGET SEGMENT: *target_segment*, SOURCE SEGMENT: *source_segment*

Explanation

The *dbdname* that is returned from the indicated user partition selection exit is not found in the primary DEDB DBD definition.

System action

The job ends with an abend code of U3003.

User response

Correct the errors and rerun the job.

HFPB3510E KEYLEN VALUE IN VSAM CATALOG DIFFERS FROM DBD DEFINITION. SECONDARY INDEX DATABASE: *database*, DDNAME: *ddname*

Explanation

The KEYLEN value in the VSAM catalog is not the same as the corresponding value in the DBD for the secondary index database data set.

System action

The job ends with an abend code of U3003.

User response

Correct the error and rerun the job.

HFPB3511E THE RECORD SIZE IS TOO SMALL TO STORE THE POINTER SEGMENT. SECONDARY INDEX DATABASE: *database*, DDNAME: *ddname*

Explanation

The record size is too small to store the pointer segment.

System action

The job ends with an abend code of U3003.

User response

Correct the error and rerun the job.

HFPB3512E CATALOG INFORMATION COULD NOT BE OBTAINED. SECONDARY INDEX DATABASE: *database*, DDNAME: *ddname*

Explanation

Catalog information could not be obtained.

System action

The job ends with an abend code of U3003.

User response

Correct the error and rerun the job.

HFPB3513E **SECONDARY INDEX DBDS IS NOT EMPTY. SECONDARY INDEX DATABASE: *database*, DDNAME: *ddname***

Explanation

Secondary index DBDS is not empty.

System action

The job ends with an abend code of U3003.

User response

Prepare an empty DBDS for the secondary index database and rerun the job.

HFPB3514E ***function* CALL FOR SECONDARY INDEX DATABASE FAILED. SECONDARY INDEX DATABASE: *database*, DDNAME: *ddname* - RETURN CODE IS *rc* (X'*hex_rc*'). RPL FEEDBACK AREA IS *rsn* (X'*hex_rsn*'). - RETURN CODE IS *rc* (X'*hex_rc*'). ACB ERROR FLAG CODE IS *rsn* (X'*hex_rsn*').**

Explanation

The utility received a nonzero return code from VSAM when it attempted to access the VSAM data set for the secondary index database. *function* indicates the name of macro that was issued to access the VSAM data set. The return code and the reason code are shown both in decimal (*rc*, *rsn*) and hexadecimal (*hex_rc*, *hex_rsn*) formats.

System action

The job ends with an abend code of U3003.

User response

See *DFSMS Macro Instructions for Data Sets*, which describes VSAM administration macros. If this situation persists, contact IBM Software Support.

HFPB3515E **SYNAD EXIT WAS INVOKED FOR A VSAM DATA SET. SECONDARY INDEX DATABASE: *database*, DDNAME: *ddname***

Explanation

The SYNAD exit was invoked for the VSAM data set.

System action

The job ends with an abend code of U3003.

User response

Contact IBM Software Support.

HFPB3516E **OVERFLOW RECORD SIZE IS SMALLER THAN PRIME RECORD SIZE. SECONDARY INDEX DATABASE: *database*, DDNAME: *ddname***

Explanation

The overflow record size is smaller than the prime record size.

System action

The job ends with an abend code of U3003.

User response

Correct the error and rerun the job.

HFPB3517E **DATA SIZE EXCEEDED THE TOTAL SIZE OF ALLOCATED STRIPE FILES. *nnnnnnnnnn*K BYTES USED IN DD *Snnnnnnn*0**

Explanation

The data size has exceeded the size of the allocated stripe file.

System action

The job ends with an abend code of U3003.

User response

Increase the size of the indicated DD or specify the indicated DD with a larger size, and rerun the job.

HFPB3518E **FOUND A DUPLICATE KEY IN POINTER SEGMENT. OVERFLOW DATA SET MUST BE DEFINED. SECONDARY INDEX DATABASE: *dbdname***

Explanation

A duplicate key was found in a pointer segment. However, an overflow data set was not defined.

System action

The job ends with an abend code of U3003.

User response

Correct the error and rerun the job.

**HFPB3519E THE RECORD SIZE IS ODD-SIZE.
SECONDARY INDEX DATABASE:
database, DDNAME: ddname**

Explanation

The record has an odd size. The size must be an even.

System action

The job ends with an abend code of U3003.

User response

Correct the error and rerun the job.

**HFPB3520E KEY OFFSET IN VSAM CATALOG
DIFFERS FROM DBD DEFINITION.
SECONDARY INDEX DATABASE:
database, DDNAME: ddname**

Explanation

The KEY OFFSET value in the VSAM catalog is not the same as the corresponding value in the DBD for the secondary index database data set.

System action

The job ends with an abend code of U3003.

User response

Correct the error and rerun the job.

**HFPB3521E ERROR IN CALL TO RANDOMIZER:
rmodname WHILE ANALYZING
SECONDARY INDEX DATABASE:
database_name
- RETURN CODE: nnnn**

Explanation

The Analyze function invoked the randomizer routine that was specified to calculate the area number and the RAP RBA information. The return code from the randomizer routine shown in the error message was not zero.

System action

The job ends with an abend code of U3003.

User response

Make sure that the RMODLIB DD statement is identified in the correct data set, and that the randomizer routine has been correctly added, assembled, and link-edited. Correct any errors and rerun the job.

**HFPB3522E INCORRECT BUILD POINTER
SEGMENT RECORD DATA SET IS
SPECIFIED. DDNAME: S0nnnnn0
- HEADER RECORD IS NOT FOUND.
- INVALID RECORD TYPE IS
FOUND.
- THE HEADER RECORD MUST
CONTAIN THE FOLLOWING
SECONDARY INDEX DATABASE
NAME: indexdbd
- THE HEADER RECORD MUST
CONTAIN THE FOLLOWING BUILD
POINTER SEGMENT RECORD DATA
SET DDNAME: S0nnnnn0
- RECORD LENGTH IS TOO SHORT
(xxxxx BYTES SPECIFIED, xxxxx
BYTES REQUIRED)**

Explanation

An incorrect build pointer segment record data set is specified as input. To run the job in load mode (BUILD MODE=LOAD), a valid build pointer segment record data set must be provided.

System action

The job ends with an abend code of U3003.

User response

Specify the correct build pointer segment record data set and rerun the job.

**HFPB3523E [BUILD | UNVERIFIED RESYNC]
POINTER SEGMENT RECORDS ARE
NOT SORTED. DDNAME: S0nnnnn0**

Explanation

The supplied pointer segment record data set contains unsorted records. When the Build Index process runs with one of the following keyword-parameter combinations, the supplied record data sets must be sorted in advance.

- IDXPROC=BUILD, BUILD MODE=LOAD, and SORT=NO
- IDXPROC=RESYNC, RESYNCMODE=VERIFYUPDATE, and AREASCAN=NO

System action

The job ends with an abend code of U3003.

User response

Sort the supplied pointer segment record data set, and then rerun the job.

HFPB3524E **INCORRECT UNVERIFIED RESYNC POINTER SEGMENT RECORD DATA SET IS SPECIFIED.**
- SPECIFIED UNVERIFIED RESYNC POINTER SEGMENT RECORD DATA SET NAME: *dsname*
- THE HEADER RECORD MUST CONTAIN THE FOLLOWING SECONDARY INDEX DATABASE NAME: *index_dbdname*
- THE HEADER RECORD MUST CONTAIN THE FOLLOWING UNVERIFIED RESYNC POINTER SEGMENT RECORD DATA SET DDNAME: *S0nnnnn0*
- THE RECORD TYPE IS NOT OF AN UNVERIFIED RESYNC POINTER SEGMENT RECORD.

- THE AREA NUMBER *nnnn* IS INCORRECT.
- THE DATA SET CONTAINS DUPLICATE RECORDS FOR AREA NUMBER *nnnn*

Explanation

An incorrect unverified resync pointer segment record data set is specified as input. To run the job in verify update mode (RESYNCMODE=VERIFYUPDATE) or direct update mode (RESYNCMODE=DIRECTUPDATE) with the AREASCAN=NO option, a valid unverified resync pointer segment record data set must be provided.

System action

The job ends with an abend code of U3003.

User response

Specify the correct unverified resync pointer segment record data set and rerun the job.

HFPC messages

The following information is about messages and codes that begin with HFPC.

HFPC0001I **COMMAND ANALYZE PROCESSING STARTED.**

Explanation

The command analysis process has started to analyze HFPSYSIN.

System action

Processing continues.

User response

None. This message is informational.

HFPC0002I **COMMAND ANALYZE PROCESSING COMPLETED.**

Explanation

The command analysis process has completed successfully.

System action

Processing continues.

User response

None. This message is informational.

HFPC0003W **COMMAND ANALYZE PROCESSING COMPLETED WITH A WARNING.**

Explanation

The command analysis process detected a warning in HFPSYSIN and completed with a warning message. Review the other generated message, the HFPC message.

System action

The return code is set to 4 and processing continues.

User response

Correct the command set in HFPSYSIN, and if necessary, rerun the job.

HFPC0004E **COMMAND ANALYZE PROCESSING COMPLETED WITH AN ERROR.**

Explanation

The command analysis process detected an error in HFPSYSIN and has completed with an error message. Review the other generated message, the HFPC message.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0005I DBRC=xxx IS SPECIFIED.

Explanation

DBRC=xxx is specified in HFPSYSIN. One of the following subtexts is printed:

- EEQE DETECTION IS PERFORMED.

DBRC=YES is specified. The program establishes a DBRC interface, checks the existence of EEQEs, and obtains area information from DBRC.

- EEQE DETECTION IS NOT PERFORMED.

DBRC=NO is specified. The program does not establish DBRC interface and does not check the existence of EEQEs.

- EEQE DETECTION IS PERFORMED BECAUSE AREADSN=RECON IS SPECIFIED.

DBRC=NO and AREADSN=RECON are specified. The program establishes a DBRC interface, checks the existence of EEQEs, and obtains area information from DBRC.

System action

Processing continues.

User response

When DBRC=NO is specified and AREADSN=RECON is not specified, and the area is registered with DBRC, obtain the DBRC RECON list and make sure there are no EEQEs registered with DBRC for the area. If there is an EEQE for the area, recover the area and rerun the job.

HFPC0006E NO FUNCTIONAL COMMANDS FOUND IN HFPSYSIN.

Explanation

The program could not find functional commands in HFPSYSIN.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify functional commands in HFPSYSIN, and rerun the job.

HFPC0007W DBRC=YES IS FORCED DUE TO IMS SYSTEM DEFINITION OF DBRC=FORCE.

Explanation

DBRC=FORCE was specified at IMS system definition time, whereas DBRC=NO was specified in HFPSYSIN.

System action

The return code is set to 4 and processing continues.

User response

Correct the command set in HFPSYSIN and, if necessary, rerun the job.

HFPC0008I keyword=nnnn IS REPLACED BY mmmm BASED ON MAXIMUM NUMBER OF PROCESS TASKS.

Explanation

keyword=nnnn is specified in HFPSYSIN. However, the *maximum number* of attached process tasks is *mmmm* (*nnnn*>*mmmm*).

System action

keyword=nnnn is replaced by *mmmm*, and processing continues.

User response

None. This message is informational.

HFPC0009E MORE THAN ONE FUNCTIONAL COMMANDS ARE SPECIFIED IN HFPSYSIN.

Explanation

More than one functional command is specified in HFPSYSIN.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify only one functional command in the HFPSYSIN, and rerun the job.

HFPC0010E **ROW: *nn*, COLUMN: *mm*, GLOBAL COMMAND MUST BE SPECIFIED AS FIRST COMMAND**

Explanation

The GLOBAL command is not specified on the first line of HFPSYSIN.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify the GLOBAL command on the first line of HFPSYSIN, and rerun the job.

HFPC0011E **ROW: *nn*, COLUMN: *mm*, QUOTATION MARK IS MISSING.**

Explanation

Data set name is specified without quotation marks.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify quotation marks around the data set name, and rerun the job.

HFPC0012E **ROW: *nn*, COLUMN: *mm*, INCORRECT [SINGLE or DOUBLE]**

QUOTATION MARK EXISTS IN A PARAMETER.

Explanation

A parameter is specified with single or double quotation marks. The parameter cannot be specified with single or double quotation marks.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify the parameter without single or double quotation marks, and rerun the job.

HFPC0013E **MANDATORY KEYWORD: *keyword* IS NOT SPECIFIED FOR SUBCOMMAND: *subcommand(nn)*. - WITH NON-SMS ENVIRONMENT.**

Explanation

The program could not find *keyword* in HFPSYSIN. The *keyword* is required for the *subcommand*.

WITH NON-SMS ENVIRONMENT

The indicated keyword must be specified when the job is run in a non-SMS environment.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify *keyword* in *subcommand*, and rerun the job.

HFPC0014E **MANDATORY KEYWORD: *keyword* IS NOT SPECIFIED FOR COMMAND: *command*.**

Explanation

The program could not find *keyword* in HFPSYSIN. The *keyword* is required for the *command*.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is

specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify keyword for the command, and rerun the job.

**HFPC0015I MIGRATION MODULE WAS FOUND
 IN JOBLIB OR STEPLIB.**

Explanation

The default value for keyword was overwritten by the migration module.

System action

Processing continues.

User response

None. This message is informational.

**HFPC0016E LRECL=SEGTFMT NEEDS
 FORMAT=TFMT.**

Explanation

LRECL=SEGTFMT is specified in HFPSYSIN or the site default. However, the FORMAT keyword is not TFMT.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

If you want to process LRECL=SEGTFMT, specify the correct FORMAT keyword.

**HFPC0017I KEYWORD=*keyword1* IS
 REPLACED BY
 KEYWORD=*keyword2***

Explanation

If you specified OCHECK=DEFER, the parameter is internally replaced with OCHECK=NODEFER because OCHECK=DEFER for the UNLOAD command is ignored when the jobs run in non-REORG mode (restructure). If you specified ICNOTIFY=COND, the parameter is internally replaced with ICNOTIFY=YES because ICNOTIFY=COND is ignored when this keyword is specified for the following commands:

- The GLOBAL command for the RELOAD, CHANGE, ANALYZE, and REORG functions

- The RELOAD, CHANGE, ANALYZE, and REORG commands

System action

Processing continues.

User response

None. This message is informational.

**HFPC0018I THE ITKBLOAD PARAMETER WAS
 IGNORED BECAUSE NO SERVER
 NAME WAS SPECIFIED.**

Explanation

An IMS Tools KB load module library was specified for the ITKBLOAD keyword parameter, but the specification was ignored because no server XCF group name was specified for the ITKBSRVR keyword parameter.

System action

Processing continues, but IMS HP Fast Path Utilities does not store any reports to the IMS Tools KB Output repository.

User response

If you want to store the reports to the IMS Tools KB Output repository, specify the IMS Tools KB server XCF group name for the ITKBSRVR keyword parameter.

**HFPC0019I THE [*keyword* |
 keyword=parameter]
 SPECIFICATION ON THE *command*
 COMMAND IS IGNORED.**

Explanation

The *keyword* or the *keyword=parameter* specification was ignored because the indicated keyword or the keyword and parameter combination is not supported for the indicated command.

System action

Processing continues.

User response

None. This message is informational.

**HFPC0020E AREA: *areaname* IS NOT DEFINED
 IN DBD: *dbdname***

Explanation

areaname is specified for the parameter of the IAREA or the OAREA keyword in HFPSYSIN. However, the *areaname* is not defined in DBD: *dbdname*, which is specified by the DBD keyword.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify the correct area name for the IAREA or the OAREA keyword, and rerun the job.

HFPC0021E **ROW: *nn*, COLUMN: *mm*, *keyword1* IS NOT ALLOWED. *keyword1* AND *keyword2* ARE MUTUALLY EXCLUSIVE.**

Explanation

Keywords *keyword1* and *keyword2* cannot be specified together.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Remove one of the keywords in HFPSYSIN, and rerun the job.

HFPC0022E **AREA: *areaname* IS DUPLICATED FOR KEYWORD: *keyword***

Explanation

areaname is specified more than once for *keyword*.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

HFPC0023E **ROW: *nn*, COLUMN: *mm*, PARAMETER: *parameter* IS DUPLICATED FOR KEYWORD: *keyword***

Explanation

parameter is specified more than once for *keyword*.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

HFPC0024E **MIGRATION MODULE WAS SPECIFIED WITH KEYWORD VALUES OTHER THAN GLOBAL PTRCHKLVL=NONE.**

Explanation

Incorrect keyword values are specified with the migration module (HFPCZZ00). The only keyword that can be specified with the migration module (HFPCZZ00) is PTRCHKLVL=NONE of the GLOBAL command.

System action

If SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the definition of HFPCZZ00, then create the load module, and rerun the job.

HFPC0025E **DO NOT SPECIFY BOTH HFPCSITE DD AND SITE_DSNAME IN HFPSYSIN DD.**

Explanation

The data set name of the site default table is specified on both the SITE_DSNAME keyword of the HFPSYSIN DD statement and the HFPCSITE DD statement. You can specify this data set name only once.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0026E DO NOT SPECIFY A MEMBER NAME FOR HFPCSITE DD.

Explanation

A member name is specified on the HFPCSITE DD statement. You can specify only DSNAME on the HFPCSITE DD statement.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the HFPCSITE DD statement of the JCL, and rerun the job.

HFPC0027I THE SITE DEFAULT TABLE THAT IS SPECIFIED IN THE HFPCSITE MODULE WAS NOT FOUND.
- **DSNAME:** *dsname*
- **MEMBER:** *member*

Explanation

The site default table with the data set name and member name that are specified for the HFPCSITE module does not exist.

System action

Processing continues.

User response

None. This message is informational.

HFPC0028E THE SECOND OPERAND OF THE SITE_MEMBER KEYWORD CAN BE SPECIFIED ONLY WITH SCAN=SITE_CREATE.

Explanation

The second operand of the SITE_MEMBER keyword is specified without SCAN=SITE_CREATE. The second operand can be specified only with SCAN=SITE_CREATE.

System action

If SCAN=YES or SCAN=SITE_REPORT is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0029E command IS ALREADY SPECIFIED DURING SITE DEFAULT [TABLE | MODULE] CREATION.

Explanation

A command that was already specified was found while the indicated site default table or module was being created.

System action

The return code is set to 8, and the job ends.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0030I A SITE DEFAULT [TABLE | MODULE] WAS CREATED.
- **DSNAME:** *dsname*
- **MEMBER:** *member*

Explanation

A site default table or module was created with *dsname* and *member*.

System action

Processing continues.

User response

None. This message is informational.

HFPC0031I THE SITE DEFAULT [TABLE | MODULE] WAS USED.
- **DSNAME:** *dsname*
- **MEMBER:** *member*

Explanation

The site default table or module that was determined with *dsname* and *member* is used.

System action

Processing continues.

User response

None. This message is informational.

HFPC0032I **SCAN=SITE_REPORT WAS FOUND,
AND THE REST OF THE CONTROL
CARDS ARE SKIPPED.**

Explanation

Because the SCAN=SITE_REPORT keyword is specified, the rest of the control cards are ignored.

System action

Processing continues.

User response

None. This message is informational.

HFPC0033E **THE SITE DEFAULT TABLE ARE
NOT SPECIFIED.
- subtext**

Explanation

An error was found when using the site default table. Either of the following subtexts is issued:

**- THOUGH [DSNAME | MEMBER] IS SPECIFIED,
[DSNAME | MEMBER] IS NOT FOUND.**

Only the data set name or the member name is defined.

**- BOTH DSNAME AND MEMBER ARE REQUIRED
WHEN SCAN=[SITE_CREATE | SITE_REPORT] IS
SPECIFIED.**

If SCAN=SITE_CREATE or SCAN=SITE_REPORT is specified, you need to specify DSNAME and MEMBER.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0035E **SITE DEFAULT TABLE MEMBER:
site member ALREADY EXISTS.
- DSNAME: dsname**

Explanation

The site default member to be created already exists.

System action

The return code is set to 8, and the job ends.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0036E **THE SITE DEFAULT [TABLE |
MODULE] IS INCORRECT.
- subtext
- DSNAME: dsname
- MEMBER: member**

Explanation

The content of the specified site default table or module is incorrect. Either of the following subtexts is issued:

- THE SITE DEFAULT [TABLE | MODULE] IS EMPTY.

The content is empty.

**- THE SITE DEFAULT [TABLE | MODULE] HAS A
FORMAT ERROR.**

The format is incorrect.

System action

For HFPSNSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0037E **SITE DEFAULT ENTRIES ARE NOT
CREATED.
- SPECIFIED KEYWORD VALUES
ARE THE SAME AS THE SYSTEM
DEFAULT VALUE
- DSNAME: dsname
- MEMBER: member**

Explanation

SCAN=SITE_CREATE was specified. However, content to be written was not found. The same values as the SYSTEM DEFAULT was specified.

If you overwrite the current member with the REUSE option, that member will be an empty member.

System action

The return code is set to 8, and the job ends.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0037E **SITE DEFAULT ENTRIES ARE NOT CREATED.**
- **KEYWORD ALLOWED AS SITE DEFAULT IS NOT SPECIFIED.**
- **DSNAME: *dsname***
- **MEMBER: *member***

Explanation

SCAN=SITE_CREATE was specified. However, content to be written was not found. No keywords that can be specified for site default was defined.

If you overwrite the current member with the REUSE option, that member will be an empty member.

System action

The return code is set to 8, and the job ends.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0037E **SITE DEFAULT ENTRIES ARE NOT CREATED.**
- **THE TABLE WAS OVERRIDDEN DUE TO THE REUSE OPTION AND EMPTY SITE DEFAULT TABLE IS CREATED.**
- **SPECIFIED KEYWORD VALUES ARE THE SAME AS THE SYSTEM DEFAULT VALUE.**
- **DSNAME: *dsname***
- **MEMBER: *member***

Explanation

SCAN=SITE_CREATE was specified. However, content to be written was not found. The reason can be one of the following occurrences:

- The same values as the SYSTEM DEFAULT were specified.
- No keywords that can be specified for site default was defined.

If you overwrite the current member with the REUSE option, that member will be an empty member.

System action

The return code is set to 8, and the job ends.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0038E **THE SITE DEFAULT TABLE IS NOT FOUND.**
- **DSNAME: *dsname***
- **MEMBER: *member***

Explanation

The specified member of the site default table was not found.

System action

If SCAN=YES or SCAN=SITE_REPORT is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0039E **DSNAME MUST BE SPECIFIED.**

Explanation

The DSNAME parameter must be specified for the HFPCSITM macro.

System action

The assemble step ends with a return code of 12.

User response

Correct the error, and rerun the job.

HFPC0040E **ROW: *nn*, COLUMN: *mm*, SEPARATOR: *x* IS INCORRECT IN THIS COMMAND SET.**

Explanation

An incorrect separator is specified for keyword.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify the correct separator for the keyword, and rerun the job.

HFPC0042E **ROW: *nn*, COLUMN: *mm*,
PARENTHESES ARE NOT
BALANCED.**

Explanation

The parameter of a keyword is specified with parentheses. However, the parentheses are not closed.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

HFPC0043E **ROW: *nn*, COLUMN: *mm*, TOO
MANY SUBPARAMETER VALUES
ARE SPECIFIED FOR KEYWORD:
*keyword***

Explanation

Multiple parameters are specified for *keyword*. The number of the parameters exceeded the maximum number for *keyword*.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

HFPC0044E **ROW: *nn*, REQUIRED
SUBPARAMETER VALUE IS
MISSING FOR KEYWORD: *keyword***

Explanation

The program could not find the required subparameter for *keyword*.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

HFPC0045E **ROW: *nn*, COLUMN: *mm*,
PARAMETER VALUE: *parameter* IS
INCORRECT FOR KEYWORD:
*keyword***

Explanation

An incorrect parameter is specified for *keyword*.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify the correct parameter for the keyword, and rerun the job.

HFPC0046E **ROW: *nn*, COLUMN: *mm*,
PARENTHESES ARE MISSING.**

Explanation

The program could not find the parentheses used to specify the parameter of the keyword. Enclose the parameter in parentheses.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Enclose the parameter in parentheses, and rerun the job.

HFPC0047E **ROW: *nn*, COLUMN: *mm*,
parameter1 IS SPECIFIED.
parameter2 IS REQUIRED AS
POSITIONAL PARAMETER FOR
*parameter3***

Explanation

parameter1 is specified as the parameter that follows *parameter3*. However, *parameter2* must follow *parameter3*.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

HFPC0048E **MANDATORY SUBCOMMAND
subcommand1 IS NOT SPECIFIED
FOR THE *subcommand2*
SUBCOMMAND.**

Explanation

When subcommand *subcommand2* is specified, subcommand *subcommand1* must also be specified.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify the missing subcommand and rerun the job.

HFPC0049E **THE *subcommand1* SUBCOMMAND
MUST PRECEDE *subcommand2*
SUBCOMMANDS.**

Explanation

When one or more *subcommand2* subcommands are specified, the *subcommand1* subcommand must be specified in advance.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0050E **INCORRECT VALUE IS FOUND IN
command COMMAND (*nnn*).
- PARAMETER VALUE: *parameter*
IS INCORRECT FOR KEYWORD:
keyword
- THE PARAMETER VALUE
parameter OF KEYWORD: *keyword*
CONFLICTS WITH THE DEFAULT
VALUE OF REORGLIM OR
RFSSCAN
- SPECIFY A MASK VARIABLE FOR
THE TARDSNMASK KEYWORD
PARAMETER.**

Explanation

An incorrect parameter is specified for *keyword*.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify the correct parameter or default for the keyword, and rerun the job.

HFPC0051E **ROW: *nn*, COLUMN: *mm*,
DUPLICATE KEYWORD: *keyword*
IS SPECIFIED.**

Explanation

keyword is specified more than once in HFPSYSIN.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is

specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

HFPC0052I **EDITCOMP_SKIP=YES IS SPECIFIED, BUT FPA CANNOT SKIP THE EDIT/COMPRESSION PROCESS.**
- A USER EXIT IS SPECIFIED.
- THE SEGMENT DEFINITION HAS CHANGED.
- COMPRESS=NO IS SPECIFIED, BUT A COMPRTN KEYWORD EXISTS IN IMSACB.

Explanation

EDITCOMP_SKIP=YES is specified but the segment edit/compression routine process was not skipped. The second line of the message explains the reason. For the Unload function, FPA compares IMSACB with NEWACB. For the Change function, FPA compares OLDACB with IMSACB.

System action

Processing continues.

User response

None. This message is informational.

HFPC0053E **WHEN SDEP=PHYSICAL, COMPRTN KEYWORD PARAMETERS MUST BE THE SAME BETWEEN [IMSACB | OLDACB] AND [NEWACB | IMSACB] FOR THE SDEP SEGMENT.**

Explanation

The edit/compression routine name or the INIT parameter on the COMPRTN keyword in NEWACB or OLDACB is different from that of IMSACB for the SDEP segment.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

If the COMPRTN keyword parameters are different between IMSACB and NEWACB for the Unload function or between OLDACB and IMSACB for the Change function, run the job with the SDEP=LOGICAL option.

HFPC0054I **SDEP=value IS REPLACED WITH SDEP=NO BECAUSE SDEP IS NOT DEFINED IN THE DBD.**

Explanation

SDEP=PHYSICAL or SDEP=LOGICAL is specified in HFPSYSIN. However, there is no SDEP segment defined in the DBD.

System action

Processing continues.

User response

None. This message is informational.

HFPC0061E **[IN CASE OF REORG MODE,] AN INCONSISTENCY IS FOUND BETWEEN IAREA AND OAREA VALUES.**
[- *command* | *subcommand*]
[IAREA=ALL | OAREA=ALL] (DEFAULT) IS APPLIED.]
- *areaname* IS MISSING IN THE **[*command* | *subcommand*] [IAREA | OAREA] PARAMETER.**

Explanation

An inconsistency is detected between the IAREA and the OAREA keywords. IAREA and OAREA must be consistent if specified for one of the following commands:

- EXTRACT command
- Reorganization mode of UNLOAD command, CHANGE command, or TUNE command

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the area names in the IAREA and the OAREA keywords, and rerun the job.

HFPC0062E REORG MODE IS NECESSARY FOR SDEP=PHYSICAL WHEN RELOCATE_SDEP=YES IS NOT SPECIFIED.

Explanation

SDEP=PHYSICAL is specified in HFPSYSIN. However, the Unload process is not in reorganization mode.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

If you want to process unload in the reorganization mode, specify the correct IMSACB and NEWACB DD statements. Otherwise, specify SDEP=LOGICAL for the UNLOAD command in HFPSYSIN.

HFPC0063E SEGMENT: *seg-name* IS NOT DEFINED IN DBD FOR SUBCOMMAND: *subcommand(nn)*

Explanation

The segment *seg-name* is specified in the SEGMENT= keyword parameter for the specified subcommand, but it is not defined in the DBD that is specified in the DBD= keyword parameter of the associated command.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0064E SEGMENT: *seg-name* IS NOT SUPPORTED FOR SUBCOMMAND: *subcommand(nn)* DUE TO *seg-type* SEGMENT.

Explanation

The segment type of *seg-name* is not supported for the specified subcommand.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0065E PARAMETER: *xxxx* IS NOT ALLOWED FOR SEGMENT: *segment* FOR SUBCOMMAND: *subcommand(nn)* DUE TO NO CHILD SEGMENT.

Explanation

The parameter *xxxx* is not allowed for the specified segment because no children are defined for the segment.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0066E PARAMETER: *xxxx* IS NOT ALLOWED FOR SEGMENT: *segment* FOR SUBCOMMAND: LOADCTL(*nn*) DUE TO ROOT SEGMENT.

Explanation

The parameter *xxxx* is not allowed for the specified segment because it is a root segment. The root segment itself cannot be the target of LOADCTL. Only its dependent segments can be targets.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0070E **ROW: *nn*, COLUMN: *mm*,
INCORRECT (SUB)COMMAND:
(*sub*)command IS SPECIFIED.**

Explanation

An incorrect command or subcommand (*command* or *subcommand*) specified in HFPSYSIN.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

HFPC0071E **ROW: *nn*, COLUMN: *mm*, CLOSING
QUOTATION MARK IS MISSING.**

Explanation

Data set name is specified without the closing quotation mark.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify the closing quotation mark for data set name, and rerun the job.

HFPC0073E **ROW: *nn*, COLUMN: *mm*,
INCORRECT CHARACTER: *xxxxx* IS
SPECIFIED.**

Explanation

Incorrect characters in HFPSYSIN. Do not specify the character that is shown in the message in a (*sub*)command.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

HFPC0074E **ROW: *nn*, COLUMN: *mm*,
KEYWORD: *keyword* IS TOO LONG.**

Explanation

Keyword *keyword* is too long.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

HFPC0076E **ROW: *nn*, COLUMN: *mm*,
KEYWORD: *keyword* IS
UNDEFINED OR INCORRECT IN
THIS COMMAND SET.**

Explanation

An unknown keyword is found in HFPSYSIN. The keyword cannot be used for this command or subcommand.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

HFPC0079E **ROW: *nn*, COLUMN: *mm*,
KEYWORD: *keyword* IS
AMBIGUOUS IN THIS COMMAND
SET.**

Explanation

The shortened keyword *keyword* specified in HFPSYSIN is ambiguous; the full keyword name could not be determined based on the shortened version that is specified.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

HFPC0084E **ANALYSIS OF KEYWORD: *keyword* IS INTERRUPTED BY EOF.**

Explanation

The program encountered an EOF in HFPSYSIN during the keyword analysis; the keyword analysis was interrupted.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

HFPC0091E **PTRCHKLVL=FULL IS REQUIRED TO SPECIFY *text***

Explanation

text indicates the keyword that must be specified.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

HFPC0099E **ROW: *nn*, COLUMN: *mm*, INTERNAL ERROR OCCURRED IN *xxxxxxx***
- UNKNOWN SEPARATOR
- NOT SUPPORTED KEYWORD

Explanation

An internal error occurred during command analysis of HFPSYSIN.

System action

For HFPSENSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

This problem might be an internal error in the program. Contact IBM Software Support.

HFPC0100E **INCORRECT KEYWORD VALUE IS FOUND IN *subcommand* SUBCOMMAND (*nn*).**
- MANDATORY KEYWORD "DDNAME" OR "DSNAME" IS NOT SPECIFIED.
- SAME AREANAME IS SPECIFIED FOR MULTIPLE *subcommand* [IAREA | OAREA] KEYWORDS.
- THE NUMBER OF USR FILE IS LARGER THAN THE NUMBER OF OAREA.
- DISP=NEW IS NOT SPECIFIED TO BE A PDS FILE.
- AREA: *areaname* IS NOT SPECIFIED IN *command* [IAREA | OAREA] ALTHOUGH IT IS SPECIFIED IN *subcommand* [IAREA | OAREA].
- AREA: *areaname* IS ALREADY SPECIFIED FOR THE OTHER *subcommand* [IAREA | OAREA] KEYWORD.
- THERE IS NO SPECIFIED AREANAME IN *command* [IAREA | OAREA] BUT *subcommand* [IAREA | OAREA]=ALL IS SPECIFIED.
- *keyword1* AND *keyword2* ARE MUTUALLY EXCLUSIVE.
- MANDATORY KEYWORD "ROOT", "UOW", OR "CISIZE" IS NOT SPECIFIED.
- [VALUE OF CISIZE | VALUE OF ROOT | VALUE OF UOW | SIZE OF THE NEW ADS] IS NOT APPROPRIATE.
- THE NUMBER OF [DSNAME | DDNAME] OPERANDS IS NOT SAME AS THE NUMBER OF IAREA OPERANDS.

- SEGMENT DEFINITION IS DIFFERENT.
- IMSACB : [*data_set_name* | IMS CATALOG (CURRENT)]
- ACBLIB : [*data_set_name* | IMS CATALOG (PENDING)]
- IN REORGANIZATION MODE, IAREA AND OAREA VALUES MUST BE THE SAME.
- OAREA=ALL SPECIFIED ON THE AREADEF SUBCOMMAND, BUT THE OAREA SPECIFICATION IS DIFFERENT ON THE SIMULATE SUBCOMMAND.
- SECONDARY INDEX DATABASE *index_dbdname* IS NOT SPECIFIED IN INDEXBLD INDEXDBD ALTHOUGH IT IS SPECIFIED IN FILECTL INDEXDBD.
- THE PARTITION GROUP NAME FOR INDEXDBD KEYWORD MUST BE THE FIRST DBDNAME IN THE GROUP. SPECIFIED DBDNAME: *index_dbdname*
- SECONDARY INDEX DATABASE *index_dbdname* IS DUPLICATED FOR INDEXDBD KEYWORD.
- THE &INDEX PARAMETER MUST BE SPECIFIED FOR THE DSNAME KEYWORD.
- THE DSNAME KEYWORD IS MISSING.
- MULTIPLE FILECTL SPECIFICATIONS ARE NOT ALLOWED WHEN INDEXDBD=_INHERIT IS SPECIFIED.
- SYMBOLIC VARIABLE &INDEX IS USED INCORRECTLY IN THE *keyword* KEYWORD.
- SPECIFY A MASK VARIABLE FOR THE *keyword* KEYWORD PARAMETER.
- JOBCARD1 KEYWORD MUST BE SPECIFIED BEFORE JOBCARD*n*.
- ERROR IN ONE OF THE JOBCARD KEYWORDS. COMMENT STATEMENT MUST BE AFTER THE JOB STATEMENT.
- *keyword1* KEYWORD MUST BE SPECIFIED WITH *keyword2*.

Explanation

An incorrect value for a keyword was found in *subcommand*.

If the subtext is SEGMENT DEFINITION IS DIFFERENT . , FPA detected a segment definition change between the original ACB and the new ACB. The FPA Tune function does not support changes in the segment definition.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

If the subtext is SEGMENT DEFINITION IS DIFFERENT . , correct the new ACB to restore the changes made to the segment definition.

HFPC0101E	INCORRECT DEFINITION IS FOUND IN TABLE: xxxxxxxx WHEN MODIFYING DEFAULT VALUE. - COMMAND: <i>command</i> , SUBCOMMAND: <i>subcommand</i> - DEFAULT VALUE MODIFICATION IS NOT ALLOWED FOR KEYWORD: <i>keyword</i>
------------------	---

Explanation

An attempt was made to change the default value of a keyword that should not be changed. The keyword is in table xxxxxxxx.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Edit table xxxxxxxx, delete the keyword definition that is specified in the message, and rerun the job.

HFPC0101E	INCORRECT DEFINITION IS FOUND IN TABLE: xxxxxxxx WHEN MODIFYING DEFAULT VALUE. - COMMAND: <i>command</i> , SUBCOMMAND: <i>subcommand</i> - KEYWORD: <i>keyword</i> IS NOT FOUND.
------------------	---

Explanation

An error was detected while changing the default value in table xxxxxxxx.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Check that the combination of the commands, keywords, and subcommands in table xxxxxxxx are correct, and that the definitions are specified in ascending order, and rerun the job.

HFPC0102E MORE THAN ONE *subcommand* SUBCOMMANDS ARE SPECIFIED IN HFPSYSIN.

Explanation

More than one subcommand is specified in HFPSYSIN.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify only one functional command in HFPSYSIN, and rerun the job.

HFPC0103E INCORRECT KEYWORD VALUE IS FOUND IN *command* COMMAND. - AREA: *areaname* IS NOT SPECIFIED IN *subcommand* [IAREA|OAREA] ALTHOUGH IT IS SPECIFIED IN *command* [IAREA|OAREA].

Explanation

The area names that are specified on the indicated command and the indicated subcommand are different. The indicated command and subcommand must specify the same area name.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the area names specified for the IAREA and the OAREA keywords, and rerun the job.

HFPC0106E IAREA HAS TWO OR MORE AREANAMES WHILE IDSNMASK IS SPECIFIED FOR ONE DATA SET.

Explanation

Although IAREA has more than one area name, only one data set is specified for IDSNMASK.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify one area name for IAREA, or specify the same number of data sets in IDSNMASK; rerun the job.

HFPC0107E THE NUMBER OF USR FILE (IAREA) IS LARGER THAN THE NUMBER OF OAREA. - THE NUMBER OF IAREA: *nn*, THE NUMBER OF OAREA: *mm*

Explanation

The number of IAREA is larger than the number of OAREA.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the number of area names for IAREA, OAREA, or both; rerun the job.

HFPC0114E HFPCHKPT DD MUST BE SPECIFIED WHEN RESTART=YES IS SPECIFIED.

Explanation

RESTART=YES is specified in HFPSYSIN or the site default. However, the HFPCHKPT DD is not specified.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

If you want to process with the RESTART=YES option, specify the HFPCHKPT DD in the Unload JCL.

HFPC0120E **ROW: *nn*, COLUMN: *nn*, AN INCORRECT VALUE IS SPECIFIED FOR THE EXPRESSIONS.**

- AN OPERAND CANNOT BE NUMERIC ONLY.
- AN UNKNOWN STRING IS FOUND.
- THE OFFSET VALUE OF THE FIELD VARIABLE IS NOT CORRECTLY SPECIFIED.
- THE [OFFSET|LENGTH] VALUE OF THE FIELD VARIABLE EXCEEDS THE SEGMENT MAXIMUM.
- KEY SEQUENCE FIELD IS NOT SPECIFIED IN SEGMENT: *segment*.
- THE LENGTH VALUE OF THE FIELD VARIABLE IS NOT CORRECTLY SPECIFIED.
- THE SUM OF OFFSET AND LENGTH EXCEEDS THE SEGMENT MAXIMUM.
- AN INCORRECT VALUE IS SPECIFIED BETWEEN THE QUOTATIONS.
- THE LENGTH OF THE HEXADECIMAL VALUE IS AN ODD BYTE.
- THE NUMBER OF CHARACTERS EXCEEDS THE MAXIMUM LENGTH.
- THIS VALUE EXCEEDS THE MAXIMUM.
- AN INCORRECT VALUE IS SPECIFIED [BEFORE|AFTER] THE QUOTATION.
- NO VALUE IS SPECIFIED BETWEEN THE QUOTATIONS.
- THE EXPLICIT LENGTH EXCEEDS THE MAXIMUM LENGTH.
- THE EXPLICIT LENGTH IS SHORTER THAN THE IMPLICIT LENGTH.
- A CLOSING QUOTATION MARK IS MISSING FOR THE LITERAL VALUE.
- AN EXTRA QUOTATION MARK EXISTS AFTER THE LITERAL VALUE.
- THE EXPLICIT LENGTH IS ZERO.
- NO VALUE IS SPECIFIED FOR THE EXPRESSIONS.
- BUILT-IN VARIABLE: *built-in_variable* CANNOT BE

SPECIFIED FOR THIS STATEMENT SPECIFICATION.

- THE FIELD VARIABLE CANNOT BE SPECIFIED FOR THIS STATEMENT SPECIFICATION.

Explanation

An incorrect value is specified for expressions. For details, see the subtext of this message.

System action

For HFPSNSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0121E **ROW: *nn*, COLUMN: *nn*, AN EXPRESSION SYNTAX ERROR IS FOUND.**

- *string* CANNOT BE SPECIFIED AT THIS POSITION.
- THE SECOND OPERAND LENGTH EXCEEDS THE KEY LENGTH.
- *string* CANNOT BE SPECIFIED WHEN *keyword=value* IS SPECIFIED.
- THIS VALUE IS NOT ALLOWED AS *location* FOR CRITERIA DEFINITION.
- THE LENGTH OF THE FIRST OPERAND EXCEEDS THE MAXIMUM LENGTH FOR COMPARISON.
- THE CRITERIA DEFINITION IS NOT COMPLETED DUE TO LACK OF [OPERATOR|SECOND OPERAND].
- THE CRITERIA DEFINITION AFTER THE BOOLEAN OPERATOR IS MISSING.
- '*' CANNOT BE SPECIFIED IN A CRITERIA DEFINITION.
- COMPARED WITH PACKED DECIMAL, THE FIRST OPERAND LENGTH MUST NOT EXCEED *nn* BYTES.
- A COMMA IS EXPECTED AT THIS POSITION.
- NO VALUE CAN BE SPECIFIED AFTER THE VL= DEFINITION.

**- THIS VALUE IS UNAVAILABLE IN EXPRESSIONS FOR FIELD DEFINITION.
- THE FIELD DEFINITION AFTER THE COMMA IS MISSING.**

Explanation

The control statement in expressions is not correctly specified. For details, see the subtext of this message.

System action

For HFPSNSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0122E *keyword1=USER IS REQUIRED TO SPECIFY KEYWORD: keyword2.*

Explanation

keyword2 is specified in HFPSYSIN. However, *keyword2* must be specified with *keyword1=USER*.

System action

If SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0123E *keyword=USER IS REQUIRED TO SPECIFY SUBCOMMAND: subcommand.*

Explanation

subcommand is specified in HFPSYSIN. However, *subcommand* must be specified with *keyword=USER*.

System action

If SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0124E *keyword VALUE IS DUPLICATED FOR SUBCOMMAND: subcommand (n).*

Explanation

The same keyword value is specified more than once for *subcommand*.

System action

If SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0125E **KEYWORD: keyword AND COMPRESS=YES ARE SPECIFIED FOR THE COMPRESSED SEGMENT. - THE COMPRESS ROUTINE IS SPECIFIED FOR SEGMENT: segment.**

Explanation

Both keywords *keyword* and COMPRESS=YES are specified for the compressed segment.

System action

If SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0126E **INCORRECT UOW RANGE SPECIFIED: STARTUOW=nnnnn, STOPUOW=mmmmm**

Explanation

An incorrect UOW range was specified. STARTUOW must not be greater than STOPUOW.

System action

If SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command set in HFPSYSIN, and rerun the job.

HFPC0127E **INPUT FILE CANNOT BE RELOADED.**
- **SEGREC_PREFIX IS NOT SPECIFIED.** - **BUILT-IN VARIABLE: *built-in_variable* IS NOT SPECIFIED IN SEGREC_PREFIX.**
- **NEITHER SEGMENT_NAME NOR SEGMENT_CODE IS SPECIFIED IN SEGREC_PREFIX.**
- ***built-in_variable1* MUST BE SPECIFIED BEFORE *built-in_variable2* IN SEGREC_PREFIX.**

Explanation

The unloaded segment records data set, which is in user-specified format, cannot be reloaded because the file format is not valid for the Reload function.

System action

If SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, an abend occurs with a code of U3003.

User response

Correct the unloaded segment records data set, and rerun the job.

HFPC0128E **IN CASE OF NON-REORG MODE, AREA CANNOT BE SPECIFIED AS THE FIRST PARAMETER OF THE BREAK KEYWORD.**

Explanation

While processing the UNLOAD command in the non-REORG mode, incorrect value, AREA, was found on the BREAK keyword.

System action

If SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, an abend occurs with a code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

HFPC0130E **ROW: *nn*, COLUMN: *nn*, INCORRECT TIMESTAMP SPECIFICATION.**

Explanation

A timestamp specification was expected. However, the data contained in HFPSYSIN was not a correct syntax for timestamp specification.

System action

If SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the timestamp specification, and rerun the job.

HFPC0131E ***keyword1* MUST BE SPECIFIED FOR *keyword2***

Explanation

keyword1 is found in the HFPSYSIN or the site default table. However, *keyword2* is not specified.

System action

For HFPSENSOR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

If you want to process *keyword1*, specify *keyword2* correctly.

HFPC0132E **THE NAME OF THE IMS TOOLS KB SERVER MUST BE SPECIFIED ON THE ITKBSRVR KEYWORD WHEN [SENSOR=YES | ITKBDATA=UTILHIST] IS SPECIFIED.**

Explanation

When SENSOR=YES or ITKBDATA=UTILHIST is specified, the name of the IMS Tools KB server must be specified.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

If you want to store the sensor data or the utility history member in the IMS Tools KB repository, specify the name of the IMS Tools KB server XCF group on the ITKBSRVR keyword and resubmit the job.

HFPC0133E **SYMBOLIC VARIABLE *variable* IS USED INCORRECTLY IN THE keyword KEYWORD.**

Explanation

The symbolic variable *variable* is specified for the value of the indicated keyword, but the usage of the symbolic variable is incorrect.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify the symbolic variable correctly, and then rerun the job.

HFPC0141E **SECONDARY INDEX DATABASE *dbname* IS DUPLICATED FOR keyword KEYWORD.**

Explanation

Secondary index database *dbname* is specified more than once for the indicated keyword.

System action

If SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

HFPC0142E **THE PARTITION GROUP NAME FOR *keyword* KEYWORD MUST BE THE FIRST DBDNAME IN THE GROUP. SPECIFIED DBDNAME: *dbname***

Explanation

The specified DBDNAME *dbname* is not the first DBD name in the group.

System action

If SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

HFPC0143E **THE DATABASE MUST HAVE A SECONDARY INDEX DATABASE DEFINITION FOR THE *command*.**

Explanation

The database that was specified in DBD does not have a secondary index database definition. The indicated command process cannot process.

System action

If SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify the database that has the secondary index database definition, and rerun the job.

HFPC0144E **SECONDARY INDEX DBD *index_dbname* IS NOT DEFINED IN DBD *dbname*.**

Explanation

The secondary index DBD *index_dbname* is not defined in DEDB DBD *dbname*.

System action

If SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the definition of DEDB DBD, and rerun the job.

HFPC0145E **SPECIFY EITHER PARTITION GROUP *groupname* OR SECONDARY INDEX DATABASE**

***index_dbdname* FOR keyword
KEYWORD.**

Explanation

Partition group *groupname* contains secondary index database *index_dbdname*.

System action

If SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

**HFPC0150I ACBCHK=YES IS IGNORED
 BECAUSE FORMAT=USER IS
 SPECIFIED.**

Explanation

ACBCHK=YES is specified in HFPSYSIN or the site default. However, ACBCHK=YES is ignored because the FORMAT keyword specifies USER.

System action

Processing continues.

User response

None. This message is informational.

**HFPC0151E [*keyword1* | *keyword1=value1*] IS
 NOT ALLOWED. [*keyword1* |
 keyword1=value1] AND
 keyword2=value2 ARE MUTUALLY
 EXCLUSIVE.**

Explanation

The *keyword1* or *keyword1=value1* specification is not supported when *keyword2=value2* is specified.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

**HFPC0152E AN ERROR OCCURRED WHILE
 ANALYZING THE [IAREA | OAREA]
 KEYWORD PARAMETER.**

Explanation

An incorrect parameter is specified on the IAREA or OAREA keyword.

System action

For HFPSNSR or HFPSTGEN, the job ends with a return code of 8. For other programs, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command set in HFPSYSIN, and rerun the job.

**HFPC0155E THE SPECIFICATION THAT
 CORRESPONDS TO THE ACTION
 LIST IS NOT FOUND.
 - [EXTENDIOVF | EXTENDSDEP] IS
 SPECIFIED, BUT THE [IOVF_UOW |
 SDEP_CI] KEYWORD IS NOT
 SPECIFIED ON THE
 SETEXTENDPARM SUBCOMMAND.
 - EXTENDIOVF, EXTENDSDEP, OR
 BOTH ARE SPECIFIED, BUT THE
 SETEXTENDPARM SUBCOMMAND
 IS NOT SPECIFIED.**

Explanation

The keyword or the subcommand that corresponds to the action specified for the ACTION keyword is not found in the HFPSYSIN data set.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command set in HFPSYSIN, and then rerun the job.

**HFPC0156I THE AREA IS NOT REORGANIZED
 BECAUSE REORGATIOVFEXT=NO
 IS SPECIFIED.**

Explanation

Action REORG is specified for the ACTION keyword. However, because REORGATIOVFEXT=NO is specified, the area is not reorganized.

System action

Processing continues.

User response

None. This message is informational.

HFPC0157E **KEYS MUST BE UNIQUE WHEN BOTH TOICTL=NONE AND RESYNCMODE=*parameter* ARE SPECIFIED.**
- SECONDARY INDEX DATABASE: *index_dbdname*

Explanation

The indicated secondary index database contains non-unique keys. Secondary index databases with non-unique keys are not supported in verify mode, update mode, or online verify update mode.

System action

If SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Do not specify a secondary index that contains non-unique keys.

HFPC0158E **[*command* | *keyword1=parameter1*] CANNOT BE SPECIFIED WHEN [*keyword2=parameter2* | *keyword2=parameter2* AND *keyword3=parameter3*]**

Explanation

This command or keyword combination is not supported.

System action

If SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the keyword or the keyword parameter, and then rerun the job.

HFPC0159I **SDEP SEGMENT DATA IS NOT UNLOADED BECAUSE SDEP=NO IS SPECIFIED.**

Explanation

Because SDEP=NO is specified, SDEP segment data is not written to the new area or the unloaded file. This message is issued because SDEP_BYPASS_MSGLVL=INFORMATIONAL is specified.

System action

Processing continues.

User response

None. This message is informational.

HFPC0160W **SDEP SEGMENT DATA IS NOT UNLOADED BECAUSE SDEP=NO IS SPECIFIED.**

Explanation

Because SDEP=NO is specified, SDEP segment data is not written to the new area or the unloaded file. This message is issued because SDEP_BYPASS_MSGLVL=WARNING is specified.

System action

The return code is set to 4 and processing continues.

User response

To have SDEP segment data written to the new area or the unloaded file, specify a parameter other than NO on the SDEP keyword, and rerun the job.

HFPC0161E **AN INCONSISTENCY IS FOUND BETWEEN IAREA AND OAREA VALUES.**
- WHEN CHANGING THE EDIT/ COMPRESSION ROUTINE, PARAMETER ALL MUST BE SPECIFIED FOR IAREA AND OAREA.
- DELETED AREA MUST BE SPECIFIED FOR IAREA KEYWORD.
- NEW AREA MUST BE SPECIFIED FOR OAREA KEYWORD.

- AREA : xxxxxxxx MUST BE SPECIFIED FOR [IAREA | OAREA] KEYWORD.

Explanation

An inconsistency is detected between the IAREA and the OAREA keyword values. Additional lines are printed to describe the details of the error.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the area names specified for the IAREA and the OAREA keywords, and rerun the job.

HFPC0162E THE LENGTH OF THE OSC DRIVER JOB NAME IS TOO SHORT TO GENERATE OSC INTERNAL JOB NAMES.

Explanation

The length of the OSC driver job name is too short to generate OSC internal job names. The name of the OSC driver job must be more than 6 characters, or the JOBN_PREFIX keyword must be provided.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the error and rerun the job.

HFPC0163E START AND END TIME OF THE TAKEOVER_TIME KEYWORD MUST BE DIFFERENT VALUES.

Explanation

The start and end time specified on the TAKEOVER_TIME keyword must be different.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify different start and end time on the TAKEOVER_TIME keyword and rerun the job.

HFPC0164E BOTH START AND END TIME MUST BE SPECIFIED IN THE TAKEOVER_TIME KEYWORD.

Explanation

When TAKEOVER_TIME=(*hhmm, hhmm*) is specified, both the start and end time must be specified.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify start and end time on the TAKEOVER_TIME keyword and rerun the job.

HFPC0165E THE CLUSTER NAME IS TOO LONG TO DEFINE A DATA COMPONENT WITH THE SPECIFIED SUFFIX.
- DSNMASK_[SHADOW | NEW | UIC]: *dsnmask_[shadow | new | uic]*
- CLUSTER DSNAME: *dsname*
- SUFFIX: *suffix*

Explanation

The data component name exceeds 44 bytes because the cluster name is too long.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the cluster name and rerun the job.

HFPC0166E VALUES FOR *keyword1* AND *keyword2* MUST BE DIFFERENT.

Explanation

The same value is specified for both *keyword1* and *keyword2*. The values must be different.

System action

If SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify the keywords correctly and rerun the job.

HFPC0167E	AREA LEVEL CHANGE IS REQUIRED WHEN REPLRAND_AREALVL=YES IS SPECIFIED.
------------------	--

Explanation

When REPLRAND_AREALVL=YES is specified, the database-level change, which stops all areas of the DEDB, is not allowed.

System action

If SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify the keyword correctly and rerun the job.

HFPC0168E	IMSRESLB_RND_DSN AND NEWRESLB_RND_DSN MUST HAVE DIFFERENT VALUES WHEN REPLRAND_AREALVL=YES IS SPECIFIED.
------------------	---

Explanation

When REPLRAND_AREALVL=YES is specified, which requests to replace the randomizer, the data set names specified for the IMSRESLB_RND_DSN keyword and the NEWRESLB_RND_DSN must be different.

System action

If SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify the keywords correctly and rerun the job.

HFPC0169I	REPLRAND_AREALVL=YES IS IGNORED BECAUSE IAREA=ALL AND OAREA=ALL ARE SPECIFIED.
------------------	---

Explanation

When both IAREA=ALL and OAREA=ALL are specified, a database-level change is performed. REPLRAND_AREALVL=YES, which is supported only for area-level change, is ignored.

System action

Processing continues.

User response

None. This message is informational.

HFPC0170E	SKIP_OLC=YES IS NOT ALLOWED WHEN THE VALUES FOR IMSRESLB_EDC_DSN AND NEWRESLB_EDC_DSN ARE DIFFERENT.
------------------	---

Explanation

When the values specified for the IMSRESLB_EDC_DSN keyword and the NEWRESLB_EDC_DSN keyword are different, the Online Structure Change utility copies the edit/compression routines from the data set specified by the NEWRESLB_EDC_DSN keyword to the production IMS system data set specified by the IMSRESLB_EDC_DSN keyword. To enable the replaced edit/compression routines in the production IMS system region, IMS online change must be performed. Therefore, SKIP_OLC=YES, which skips IMS online change, is not allowed when replacing edit/compression routines.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

HFPC0171E	VALUES FOR THE FOLLOWING KEYWORDS MUST MATCH WHEN THE EDIT/COMP ROUTINE DEFINITIONS IN THE OLD AND NEW DBD DEFINITIONS ARE THE SAME. - IMSRESLB_EDC_DSN - NEWRESLB_EDC_DSN
------------------	---

Explanation

When the definitions of the old and the new edit/compression routines are the same, the Online Structure Change utility cannot replace the edit/compression routines. This is because IMS online change, which is a required process to replace edit/compression routines, does not support reloading new edit/compression routines to the production IMS system region if the definitions of old and new edit/compression routines are the same.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3016.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

HFPC0172E	[IMSRESLB_EDC_DSN NEWRESLB_EDC_DSN] KEYWORD IS REQUIRED WHEN THE EDIT/COMP ROUTINE IS DEFINED IN THE [OLD NEW] DEFINITION.
------------------	---

Explanation

If edit/compression routines are defined to the DEDB, the library that contains the edit/compression routines must be supplied with the IMSRESLB_EDC_DSN keyword or the NEWRESLB_EDC_DSN keyword.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3016.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

HFPC0173E	IMSRESLB_EDC_DSN KEYWORD IS REQUIRED WHEN NEWRESLB_EDC_DSN KEYWORD IS SPECIFIED.
------------------	---

Explanation

When the NEWRESLB_EDC_DSN keyword is specified, the Online Structure Change utility requires the IMSRESLB_EDC_DSN keyword so it can copy the new edit/compression routines to the production IMS systems.

If you do not want the edit/compression routines copied to the production IMS systems, specify the same library on both the IMSRESLB_EDC_DSN keyword and the NEWRESLB_EDC_DSN keyword.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Correct the command syntax of HFPSYSIN, and rerun the job.

HFPC0174E	ENQDSN KEYWORD IS REQUIRED FOR OLC OR IMPORT INTERNAL JOB WHEN SKIP_OLC=YES IS NOT SPECIFIED.
------------------	--

Explanation

ENQDSN keyword is not supplied. When SKIP_OLC=YES (skips online change) is not specified, the ENQDSN keyword is required. The Online Structure Change utility uses the ENQDSN keyword to generate internal SYSIN files, which the batch SPOC utility uses in the online change process.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Supply the ENQDSN keyword and rerun the job.

HFPC0175E	THE JCLDSN KEYWORD AND THE ENQDSN KEYWORD MUST NOT SPECIFY THE SAME DATA SET.
------------------	--

Explanation

The JCLDSN keyword and the ENQDSN keyword must not specify the same data set.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3003.

User response

Specify different data sets for the JCLDSN keyword and the ENQDSN keyword, and rerun the job.

HFPC3150E **IMS TOOLS CATALOG INTERFACE
CANNOT BE USED.
- UNSUPPORTED IMS RELEASE.**

Explanation

The IMSCATHLQ=*bsdshlq* parameter is specified on HFPSYSIN, but FPA could not use IMS Tools Catalog Interface to read the ACB from the IMS directory because the version of IMS is lower than 14.

HFPD messages

The following information is about messages and codes that begin with HFPD.

HFPD0001I ***ddname* DYNAMICALLY
ALLOCATED TO
- *dsname***

Explanation

The specified data set *dsname* has been dynamically allocated to the DD statement *ddname*.

System action

Processing continues.

User response

None. This message is informational.

HFPD0002I ***ddname* DYNAMICALLY
DEALLOCATED.**

Explanation

The specified DD statement *ddname* has been dynamically deallocated.

System action

Processing continues.

User response

None. This message is informational.

HFPD0003A **WAITING MAX *mm* MINUTES FOR
TAPE MOUNT. DSNAME: *dsname***

Explanation

The program retries dynamic allocation on the tape unit or the tape label.

System action

The job ends with an abend code of U3150.

User response

Rerun the job using a supported version of IMS.

System action

The job continues. If the tape is not mounted before the time limit, the job will end with an abend code of U3003.

Operator response

Mount the specified tape data set.

User response

None.

HFPD0005I **VSAM DATA SET: *dsname*
SUCCESSFULLY DEFINED.**

Explanation

IDCAMS DEFINE CLUSTER has completed successfully by the ALLOCATE subcommand.

System action

Processing continues.

User response

None. This message is informational.

HFPD0006I **NUMBER OF MESSAGES REACHED
UPPER LIMIT: *nn*, ALL
SUBSEQUENT MESSAGES ARE
SUPPRESSED.**

Explanation

The number of messages that was issued for the failure of a data set allocation has reached the upper threshold value.

System action

The program stops issuing further error messages, and continues processing.

User response

None. This message is informational.

HFPD0007W **IDCAMS RETURNED ABNORMAL
CONDITION CODE FOR DATA SET:
dsname
** IDCAMS MESSAGES END *****

Explanation

An error occurred during processing of the output allocation that was requested by the ATTRIBUTE keyword of the ALLOCATE subcommand.

System action

If EROPT=ABORT was specified in the ATTRIBUTE keyword of the ALLOCATE subcommand, an HFPD3033 message was issued, and the job ends with an abend code of U3003. Or if EROPT=IGNORE was specified, processing continues.

User response

Determine the cause of the ALLOCATE subcommand failure, correct any errors, and rerun the job.

HFPD3020E **BLDL FAILED FOR DYNAMIC
ALLOCATION MEMBER:
*member_name***

Explanation

The BLDL macro is issued for the dynamic allocation member *member_name* that was specified in the STEPLIB. The macro failed.

System action

The job ends with an abend code of U3003 or U3020.

User response

To determine the cause of the error, see *DFSMS Macro Instructions for Data Sets*. Correct any errors, and rerun the job. If this situation persists, contact IBM Software Support.

HFPD3021E **DFSMDA MEMBER:*member_name*
NOT FOUND.**

Explanation

DFSMDA member *member_name* was not found in the STEPLIB.

System action

The job ends with an abend code of U3003 or U3021.

User response

Make sure the specified member exists in the STEPLIB. Correct any errors, and rerun the job. If this situation persists, contact IBM Software Support.

HFPD3022E **INCORRECT DYNAMIC
ALLOCATION
MEMBER:*member_name***

Explanation

The dynamic allocation member *member_name* does not conform to the expected DFSMDA structure.

System action

The job ends with an abend code of U3003 or U3022.

User response

Verify the integrity of the dynamic allocation member. Correct the problem, then rerun the job.

HFPD3023E **DYNAMIC ALLOCATION FAILED
FOR
DDNAME:*ddname*,
DSNAME:*dsname*,
RC=*code*, RSN=*eeeeiiii***

Explanation

Dynamic allocation for *ddname* and *dsname* failed. *code* is the return code from SVC99. *eeee* is the S99 ERROR contents. *iiii* is the S99 INFO contents.

System action

The job ends with an abend code of U3003, U3006, or U3023.

User response

Look up the dynamic allocation (SVC99) code in the *MVS Programming: Assembler Services Reference*. Correct the problem, then rerun the job.

HFPD3024E **DYNAMIC DEALLOCATION FAILED
FOR DDNAME:*ddname*,
RC=*code*, RSN=*eeeeiiii***

Explanation

Dynamic deallocation for *ddname* failed. *code* is the return code from SVC99. *eeee* is the S99 ERROR contents. *iiii* is the S99 INFO contents.

System action

The job ends with an abend code of U3003 or U3024.

User response

Look up the dynamic deallocation (SVC99) code in the *MVS Programming: Assembler Services Reference*. Correct the problem, then rerun the job.

**HFPD3025E TIMEOUT OCCURRED FOR
WAITING
DSNAME:*dsname*,
RC=*code*, RSN=*eeeeiiii***

Explanation

Dynamic allocation for *dsname* on the tape unit failed because the operation exceeded the wait time.

System action

The job ends with an abend code of U3003 or U3025.

User response

Rerun the job.

**HFPD3026E *ddname* DD INFORMATION IS NOT
FOUND IN THE DFSMDA MEMBER
member_name.**

Explanation

ddname DD information is not found in the DFSMDA member *member_name*.

System action

The job ends with an abend code of U3003.

User response

Correct the error and rerun the job.

**HFPD3027E DFSMDA MEMBER *member_name*
IS NOT FOUND AND THE
FOLLOWING DATA SETS ARE NOT
PROVIDED.
- KSDS IS NOT PROVIDED FOR
PRIMARY DD: *ddname*
- ESDS IS NOT PROVIDED FOR
OVERFLOW DD: *ddname***

Explanation

DFSMDA member *member_name* is not found and the database data set for the secondary index database was not provided for the output DD.

System action

The job ends with an abend code of U3003.

User response

Correct the error, and rerun the job.

**HFPD3032E ALLOCATION FAILED FOR DATA
SET:*dsname*
(*IDCAMS messages*) *** IDCAMS
MESSAGES END *****

Explanation

IDCAMS DEFINE CLUSTER failed. The reason and code is shown by *IDCAMS messages*.

System action

The job ends with an abend code of U3003.

User response

Check *IDCAMS messages*, Correct any errors, and rerun the job.

**HFPD3033E ALLOCATION PROCESS ENDED BY
EROPT=ABORT OPTION.**

Explanation

An error occurred during processing of the output allocation that was requested by the ATTRIBUTE keyword of the ALLOCATE subcommand. EROPT=ABORT was specified in the ATTRIBUTE keyword.

System action

The job ends with an abend code of U3003.

User response

Determine the cause of the ALLOCATE subcommand failure, correct any errors, and rerun the job.

**HFPD3035E ALLOCATION DSNAME IS NOT
PROVIDED.**

Explanation

Dynamic allocation failed because no data set name was specified in the JCL statement.

System action

The job ends with an abend code of U3003.

HFPF messages

The following information is about messages and codes that begin with HFPF.

HFPF0001I *command* **PROCESSING STARTED
FOR DATABASE: *database***

Explanation

The specified command processing for the specified database has started.

System action

Processing continues.

User response

None. This message is informational.

HFPF0002I *command* **PROCESSING
COMPLETED FOR DATABASE:
*database***

Explanation

The specified command processing has completed successfully.

System action

Processing continues.

User response

None. This message is informational.

HFPF0003W **[LRECL | BLKSIZE] FOR DDNAME:
ddname IS OVERRIDDEN BY
SYSTEM DETERMINED VALUE.
- SPECIFIED [LRECL | BLKSIZE]
TOO SMALL OR TOO LARGE
- [LRECL | BLKSIZE] SPECIFIED:
nnnnn
- [LRECL | BLKSIZE] OVERRIDDEN:
*nnnnn***

Explanation

The LRECL or BLKSIZE for the USR file or the sequence error segment records file that is specified in *ddname* is incorrect. The program overrides the LRECL or BLKSIZE with the system value.

User response

Specify a data set name for dynamic allocation, and rerun the job.

System action

The return code is set to 4 and processing continues.

User response

None.

HFPF0004I **CISNAP=YES IS REPLACED BY
CISNAP=NO**

Explanation

CI map/CI dump request was ignored because the HFPASNAP DD statement was not provided in the JCL.

System action

Processing continues.

User response

Provide an HFPASNAP DD statement in the JCL, and if necessary, rerun the job.

HFPF0005E **[ANALYZE | DB SENSOR | TUNE |
MANAGE] PROCESSING
INITIALIZATION FAILED
- GETMAIN FAILED.
- FILE OPEN ERROR.
- HFPAPARM ERROR.
- DYNAMIC ALLOCATION FAILED.
- DDNAME: *ddname* WAS NOT
FOUND.
- SR BUFFER POOL CREATION
FAILED.
- DDNAME: *ddname* CANNOT BE
SPECIFIED.
- DDNAME: *ddname* MUST
SPECIFY SYSOUT.
- DDNAME: *ddname* CANNOT
SPECIFY SYSOUT.
- RECON INFORMATION IS NOT
FOUND.
- DDNAME *ddname* MUST NOT BE
ASSIGNED TO DUMMY OR
NULLFILE.**

Explanation

This message consists of multiple parts, which include:

- **GETMAIN FAILED.**
GETMAIN to obtain space for internal blocks failed.
- **FILE OPEN ERROR.**
OPEN processing for the file that is associated with the Analyze process failed.
- **HFPAPARM ERROR.**
Some errors were detected in the HFPAPARM data set.
- **DYNAMIC ALLOCATION FAILED.**
An attempt to dynamically allocate the internal work data set failed.
- **DDNAME: *ddname* WAS NOT FOUND.**
The Analyze process could not find DD statement *ddname* in the JCL statement.
- **SR BUFFER POOL CREATION FAILED.**
GETMAIN to obtain space for SR buffer pool failed.
- **DDNAME: *ddname* CANNOT BE SPECIFIED.**
The indicated DD statement is specified in the JCL stream, but the statement must not be specified because the data set is allocated dynamically.
- **DDNAME: *ddname* MUST SPECIFY SYSOUT.**
The indicated DD statement must specify SYSOUT.
- **DDNAME: *ddname* CANNOT SPECIFY SYSOUT.**
The indicated DD must not specify SYSOUT.
- **RECON INFORMATION IS NOT FOUND.**
RECON data set information is not found.
- **DDNAME *ddname* MUST NOT BE ASSIGNED TO DUMMY OR NULLFILE.**
The indicated DD statement must not be assigned to DUMMY or NULLFILE.

System action

The return code is set to 8 and processing continues.

User response

- **GETMAIN FAILED.**
If the region size specified is too small, increase the REGION size on the JOB statement in the JCL, and rerun the utility.
- **FILE OPEN ERROR.**
Correct any errors, and rerun the job. If this situation persists, contact IBM Software Support.
- **HFPAPARM ERROR.**
For further details, see [“Process Summary of Analysis report” on page 590](#) or previous error messages.

- DYNAMIC ALLOCATION FAILED.

This error is likely an internal system error. Contact IBM Software Support.

- DDNAME: *ddname* WAS NOT FOUND.

Specify the *ddname* DD statement in the JCL statement, and rerun the job.

- SR BUFFER POOL CREATION FAILED.

If the region size that is specified is too small, increase the REGION size on the JOB statement in the JCL, and rerun the utility.

- DDNAME: *ddname* CANNOT BE SPECIFIED.

Delete the indicated DD statement from the JCL statements.

- DDNAME: *ddname* MUST SPECIFY SYSOUT.

Specify SYSOUT for the indicated DD statement.

- DDNAME: *ddname* CANNOT SPECIFY SYSOUT.

Specify a data set for the indicated DD statement.

- RECON INFORMATION IS NOT FOUND.

Specify the RECONx DD statements, or specify the DFSMDA library that contains the RECON information in the JOBLIB, the STEPLIB, or the IMSDALIB DD statements.

- DDNAME *ddname* MUST NOT BE ASSIGNED TO DUMMY OR NULLFILE.

Specify a data set or SYSOUT stream for the indicated DD statement.

HFPF0006E STORAGE OBTAIN FAILURE FOR ANALYZE PROCESS

- ACC CREATION FAILED.
- GETMAIN FOR HFPASNAP DCB FAILED.
- GETMAIN FOR TCT LIST FAILED.
- SR BUFFER POOL CREATION FAILED.
- GETMAIN FOR ADLIST FAILED.

Explanation

GETMAIN to obtain space for internal blocks failed.

System action

The return code is set to 8 and processing continues.

User response

If the region size that is specified is too small, increase the REGION size on the JOB statement in the JCL, and rerun the utility.

HFPF0007E *command* PROCESSING COMPLETED, BUT ANALYZE PROCESS COMPLETED WITH AN ERROR.

Explanation

The specified command processing has completed successfully, but the Analyze process detected an error.

System action

The return code is set to 12 and processing continues.

User response

Check the preceding message and determine the cause of the Analyze process error. Correct any errors, and issue the ANALYZE command.

HFPF0008I **IRENAME PROCESS COMPLETED FOR AREA NO: *nnnn* AREA NAME: *areaname***
- **INPUT DSNAME: *dsname***
- **RENAMED DSNAME: *dsname***
- **INPUT CLUSTER NAME: *cluster name***
- **RENAMED CLUSTER NAME: *cluster name***
- **INPUT DATA NAME: *data component dsname***
- **RENAMED DATA NAME: *data component dsname***

Explanation

Renaming of ADS with IRENAME subcommand has completed successfully.

System action

Processing continues.

User response

None. This message is informational.

HFPF0009I **INCORRECT SDEP LOGICAL BEGIN RBA: *xxxxxxxx***

Explanation

SDEP logical begin is not at a CI boundary.

System action

Processing continues.

User response

None. This message is informational.

HFPF0010E **[INPUT | OUTPUT] THREAD TASK ENDED ABNORMALLY.**
- **AREA NO: *nnnn*, AREANAME: *areaname***

Explanation

One of the input or the output thread task ended abnormally during the function task processing.

System action

The return code is set to 12 and processing continues.

User response

Check the preceding message and determine the cause of the thread task error. Correct any errors, and rerun the job.

HFPF0011I **ONE OR MORE SEQUENCE ERROR SEGMENTS WERE WRITTEN TO DATA SET.**
- **DDNAME: *HFPSEQR*, DSNAME: *dsname***

Explanation

One or more sequence error segments were written to the data set *dsname* that was specified in the HFPSEQR DD statement.

System action

Processing continues.

User response

None. This message is informational.

HFPF0012I **POINTER ERROR INFORMATION IS NOT PROVIDED BECAUSE THE INPUT DATA SET IS AN IMAGE COPY.**
- **AREA NO: *nnnn*, AREANAME: *areaname***

Explanation

When an image copy data set is provided as an input data set, PTRCHK=DETAIL option is treated as PTRCHK=BYPASS. Therefore, the pointer error information is not provided.

System action

Processing continues.

User response

None. This message is informational.

HFPF0013E [OAE | OER | OPC | SCAN | DELETE] PROCESSING ENDED ABNORMALLY FOR DATABASE: *databasename*, AREANAME: *areaname*

Explanation

The OAE, OER, OPC, SDEP SCAN, or SDEP DELETE utility ended abnormally while processing the indicated area.

System action

Processing ends with a return code of 8.

User response

Correct the error, and then rerun the job.

HFPF0014I SORTOPT_FILSZ=AUTO FOR *command* PROCESSING WAS IGNORED BECAUSE OF THE FOLLOWING REASON:
- ICHECK=DEFER IS SPECIFIED AND SDEP IS DEFINED.
- INPUT DATA SET IS IMAGE COPY AND SDEP IS DEFINED.
- THE NUMBER OF INPUT AREAS IS NOT THE SAME AS THE NUMBER OF OUTPUT AREAS.

Explanation

SORTOPT_FILSZ=AUTO is specified but FPA did not apply this option to the indicated command. The reason is described in the subtext.

System action

Processing continues.

User response

None. This message is informational.

HFPF0015E AREA WAS NOT RELOADED DUE TO NO USR RECORDS IN THE USR FILE(S) OR SOME USR FILE PROBLEMS.
- AREA NO: *nnnn*, AREANAME: *areaname*

Explanation

The indicated area was not reloaded because FPA found no USR records in the USR files or encountered USR file problems.

System action

Message HFPF3373E is issued after this message. The job ends with an abend code of U3003.

User response

Correct the error in the USR files, and then rerun the job.

HFPF0501I *phasename* PHASE PROCESS STARTED.

Explanation

The indicated process phase has started.

System action

Processing continues.

User response

None. This message is informational.

HFPF0502I *phasename* PHASE PROCESS COMPLETED.

Explanation

The indicated process phase has completed.

System action

Processing continues.

User response

None. This message is informational.

HFPF0507I ALL PHASES COMPLETED. (JOB : *driver_jobname*).

Explanation

All the process phases have completed.

System action

Processing continues.

User response

None. This message is informational.

HFPP0601I **OSC INTERNAL JOB STARTED.**
JOB : *int_jobname* **OWNER:**
driver_jobname

Explanation

The indicated OSC internal job, which is submitted by the indicated OSC driver job, has started.

System action

Processing continues.

User response

None. This message is informational.

HFPP0602I **OSC INTERNAL JOB COMPLETED.**
JOB: *int_jobname* **OWNER:**
driver_jobname

Explanation

The indicated OSC internal job, which is submitted by the indicated OSC driver job, has completed.

System action

Processing continues.

User response

None. This message is informational.

HFPP0603W **OSC INTERNAL JOB ENDED WITH**
WARNINGS. JOB: *int_jobname*
OWNER: *driver_jobname*

Explanation

The indicated OSC internal job, which is submitted by the indicated OSC driver job, ended with warnings.

System action

The return code is set to 4 and processing continues.

User response

None. This message is informational.

HFPP0604E **OSC INTERNAL JOB ENDED WITH**
ERRORS. JOB: *int_jobname*
OWNER: *driver_jobname*

Explanation

The indicated OSC internal job, which is submitted by the indicated OSC driver job, ended with errors.

System action

If ERRORACT=WTOR is specified, the Online Structure Change utility issues message HFPT0512A to query the user for the action to take. Otherwise, the OSC driver job ends abnormally.

User response

If ERRORACT=WTOR is specified, correct the failed OSC internal job and rerun the OSC internal job. If the job ends normally, reply to the HFPT0512A message with C so that the OSC driver job continues processing from where it was paused. Otherwise, correct the error, and rerun the OSC driver job.

HFPP3006E **[INPUT, OUTPUT, or ANALYZE]**
THREAD TASK ENDED
ABNORMALLY.

Explanation

One of the input, the output, or the analyze thread task ended abnormally during the function task processing.

System action

The job ends with an abend code of U3003 or U3006.

User response

Check the preceding message and determine the cause of the thread task error. Correct any errors, and rerun the job.

HFPP3011E **THE SAME DSNAME IN *command***
COMMAND IS SPECIFIED.
- DSNAME: *dsname*
- INPUT AREA NO: *nnnn*,
AREaname: *areaname*, **DDNAME:**
ddname
- OUTPUT AREA NO: *nnnn*,
AREaname: *areaname*, **DDNAME:**
ddname

Explanation

The same data set *dsname* was used for input and output.

System action

The job ends with an abend code of U3003.

User response

Check the data sets used for input and output, correct the data set names and, if necessary, rerun the job.

HFPF3012E IRENAME PROCESS FAILED FOR
AREA NO: *nnnn* AREA NAME:
areaname
- INPUT DATASET TO BE RENAMED
MUST BE ADS (VSAM ESDS).
- DDNAME: *ddname*, DSNAMES:
dsname

Explanation

The data set *dsname* that was specified by the IRENAME subcommand was not ADS.

System action

The job ends with an abend code of U3003.

User response

Check the data set used for input, correct the data set names and, if necessary, rerun the job.

HFPF3012E IRENAME PROCESS FAILED FOR
AREA NO: *nnnn* AREA NAME:
areaname
- RENAME FAILED.
- INPUT DSNAMES: *dsname*
- RENAMED DSNAMES: *dsname*

Explanation

The data set *dsname* that was specified by the IRENAME subcommand was not renamed.

System action

The job ends with an abend code of U3003.

User response

The reason for the failure is shown in message HFPD0007W. Check HFPD0007W and correct the problem and, if necessary, rerun the job.

HFPF3350E DDNAME: *ddname* IS SPECIFIED
IN FILECTL SUBCOMMAND, BUT
THERE IS NO DD STATEMENT IN
JCL.

Explanation

The program could not find DD statement *ddname* in the JCL statement; the *ddname* was specified on the FILECTL subcommand in HFPSYSIN.

System action

The job ends with an abend code of U3003.

User response

Specify the *ddname* DD statement in the JCL statement, and rerun the job.

HFPF3351E NO AVAILABLE ADS IN INPUT
AREANAME: *areaname*

Explanation

During the Unload process, no area data sets were available in the input area *areaname*.

System action

The job ends with an abend code of U3003.

User response

Check the preceding message and determine the cause of the thread task error. Correct any errors, and rerun the job.

HFPF3352E NO DBRC REGISTERED ADS IN
AREANAME: *areaname*

Explanation

DBRC=YES was specified in HFPSYSIN during the command processing. However, there was no area data set registered with DBRC for area *areaname*.

System action

The job ends with an abend code of U3003.

User response

Get a LIST.RECON output report, and specify an unused area data set name. Specify the name in the *adsname* DD statement, and rerun the job.

HFPF3353E [INPUT or OUTPUT] DATA SET IS
NOT A VSAM CLUSTER FOR ADS.
DDNAME: *ddname*, DSNAMES:
dsname

Explanation

The specified ADS, for the input data set in the Unload process or for the output data set in the Reload process, was not a VSAM cluster.

System action

The job ends with an abend code of U3003.

User response

Specify a VSAM data set for the ADS, and rerun the job.

HFPF3355E **INCONSISTENT DSNAME
BETWEEN DBRC AND JCL DD.
DDNAME: *ddname*
- DBRC: *dsname1*
- JCL: *dsname2***

Explanation

Ddname *ddname* for the database dsname *dsname2* that was specified in the JCL DD statement did not match the *ddname* for the database *dsname1* in the DBRC.

System action

The job ends with an abend code of U3003.

User response

Specify the correct data set name for the JCL statement or in the DBRC, and rerun the job.

HFPF3356E **ADS: *dsname* SPECIFIED BY
[IDSNMASK or ODSNMASK] IS
NOT DEFINED IN DBRC.**

Explanation

The area data set that was specified by the IDSNMASK or the ODSNMASK keyword in HFPSYSIN was not registered with DBRC.

System action

The job ends with an abend code of U3003.

User response

Get a LIST.RECON output report, and find an available area data set name. Then specify the name in the IDSNMASK or the ODSNMASK keyword, and rerun the job.

HFPF3357E **OPEN FAILED FOR DDNAME:
ddname
- INCORRECT RECORD FORMAT
(RECFM)
- [IAREAnnn/IARnnnnn or
OAREAnnn/OARnnnnn] DD
STATEMENTS NOT FOUND
- DD STATEMENT NOT FOUND**

Explanation

The program issued OPEN for the DD statement *ddname* specified. The OPEN failed.

System action

The job ends with an abend code of U3003.

User response

Make sure that a DD statement is present for the specified *ddname*, and that it properly identifies the correct data set. Correct any errors, and rerun the job.

HFPF3358E **DDNAME: *ddname* FOR
subcommand SUBCOMMAND
DSNAME IS SPECIFIED ON THE
JCL DD.**

Explanation

The program dynamically allocated a data set, which was specified on the indicated subcommand, to DD statement *ddname*. However, the *ddname* for the data set is also specified in the JCL DD statement.

System action

The job ends with an abend code of U3003.

User response

Remove *ddname* from the JCL DD statement, and rerun the job.

HFPF3359E **THERE ARE NO AVAILABLE ADS IN
DBRC FOR AREA: *areaname*
- ALL ADS ARE UNAVAILABLE.
- ADS SPECIFIED ON JCL DD IS
UNAVAILABLE.
- ADS SPECIFIED BY IDSNMASK
IS UNAVAILABLE.**

Explanation

The data set for area *areaname* that is specified in the JCL DD statement or in HFPSYSIN is unavailable in DBRC.

System action

The job ends with an abend code of U3003.

User response

Get a LIST.RECON output report, and find a valid area and an available area data set name. Then specify the name in the JCL DD statement or in HFPSYSIN, and rerun the job.

HFPF3361E **ESDS DATA SET FOR AREANAME:
areaname HAS NO ENOUGH
SPACE.**

**HI-ALLOCATED RBA: xxxxxxxx,
SDEP FIRST CI RBA: xxxxxxxx**

Explanation

The ESDS data set for the indicated area does not have enough space. The program detected one of the following errors:

- High-allocated RBA of the ESDS data set is smaller than the ending RBA of the last CI in REORG UOW (the first CI RBA of SDEP).
- Although the database definition was changed in the Unload process, the Reload process is using the original ACBLIB.

System action

The job ends with an abend code of U3003.

User response

Based on the size of the area in the database definition, define enough space for the ESDS data set and rerun the job.

If you changed the database definition in the Unload process, you must provide the updated ACBLIB for the Reload process. Ensure that the IMSACB DD statement of the Reload job specifies the updated ACBLIB, and rerun the job.

**HFPF3362E DDNAME: *ddname* FOR
[IDSNMASK | ODSNMASK |
TARDSNMASK |
RESYNCDSNMASK] IS SPECIFIED
ON THE JCL DD.**

Explanation

The IDSNMASK, the ODSNMASK, the TARDSNMASK, or the RESYNCDSNMASK keyword is specified in HFPSYSIN for DD statement *ddname*. However, the *ddname* is also specified in the JCL DD statement.

System action

The job ends with an abend code of U3003.

User response

Remove the *ddname* from the JCL DD statement or remove the IDSNMASK, the ODSNMASK, the TARDSNMASK, or the RESYNCDSNMASK keyword from HFPSYSIN, and rerun the job.

**HFPF3363E AT LEAST ONE ADS IS AVAILABLE
IN DBRC FOR [RELOAD |**

**INITIALIZE] AREANAME:
*areaname***

Explanation

The program processed the Reload function or the Initialize function with DBRC=YES. However, the program found at least one available ADS for area *areaname* in DBRC. All ADSs of the area must be in the UNAVAILABLE status in DBRC.

System action

The job ends with an abend code of U3003.

User response

Get a LIST.RECON output report, and find a valid area and an unavailable area data set name. Then specify the name in the JCL DD statement or in HFPSYSIN and rerun the job.

**HFPF3364E [ADS | ESDS | KSDS] IS NOT
PROVIDED FOR [INPUT | OUTPUT]
[AREANAME | DDNAME]: *ddname***

Explanation

The area data set for area *ddname* or the database data set for secondary index database *ddname* was not provided to process the command that is specified in HFPSYSIN.

System action

The job ends with an abend code of U3003.

User response

Correct any errors, and rerun the job. If this situation persists, contact IBM Software Support.

**HFPF3365E AN IMAGE COPY DATASET INPUT
AND ICOUT=YES ARE MUTUALLY
EXCLUSIVE.
- DDNAME: *ddname*, DSNAME:
*dsname***

Explanation

When an image copy data set is provided as an input data set, ICOUT=YES cannot be specified in HFPSYSIN.

System action

The job ends with an abend code of U3003.

User response

Remove the ICOUT=YES keyword in HFPSYSIN, and rerun the job.

HFPF3366E **IMAGE COPY DATA SET CANNOT
BE PROCESSED BY REORG
COMMAND. AREANAME: xxxxxxxx**

Explanation

The image copy could not be processed as an input data set by the REORG command.

System action

The job ends with an abend code of U3003.

User response

Correct the errors, and rerun the job.

HFPF3367E **ANOTHER TOOL IS UPDATING
DMAC AREANAME: xxxxxxxx**

Explanation

Another tool is updating DMAC of area xxxxxxxx.

System action

The job ends with an abend code of U3003.

User response

Correct the errors, and rerun the job.

HFPF3368E **FPA IS UPDATING DMAC OF TWO
OR MORE MADS DATA SETS.
AREANAME: xxxxxxxx**

Explanation

FPA is updating DMAC of two or more MADS data set of area xxxxxxxx.

System action

The job ends with an abend code of U3003.

User response

Correct the errors, and rerun the job. If this situation persists, contact IBM Software Support.

HFPF3369E **ERROR IN CALL TO RANDOMIZER:
randomizer, FUNCTION:
functionname
- RETURN CODE: xxxx REASON
CODE: xxxxxxxx**

Explanation

The function task invoked the XCI randomizer routine to perform an initialization or termination call. The return code from the randomizer routine shown in the error message was not zero.

System action

The job ends with an abend code of U3003.

User response

See the description of the return code and reason code for the XCI randomizer specification. Correct any errors, and rerun the job.

HFPF3370E **DDNAME: ddname FOR
SECONDARY INDEX DATABASE:
dbdname IS SPECIFIED ON THE
JCL DD.**

Explanation

The ALLOCATE subcommand for the indicated DD statement is specified in HFPSYSIN; however, the indicated DD name is specified in the JCL DD statement.

System action

The job ends with an abend code of U3003.

User response

Remove the indicated DD from the JCL DD statement, and rerun the job.

HFPF3371E **DUPLICATE TUNING AID RECORD
DATA SET IS SPECIFIED.
- DSNAME: dsname**

Explanation

Identical tuning aid record data set is specified on more than two DD statements. *dsname* indicates the name of the data set.

System action

The program ends with an abend code of 3003.

User response

Specify the correct tuning aid record data sets in the JCL and rerun the job.

HFPF3372E *data_set_name* DATA SET MUST BE PROVIDED TO ENABLE THE *keyword=value* OPTION.
- SECONDARY INDEX DATABASE:
indexdbd

Explanation

When you specify the *keyword=value* option, you must also supply the *data_set_name* data set with JCL DD statements or with HFPSYSIN keywords.

System action

The job ends with an abend code of U3003.

User response

Specify the correct data set and then rerun the job.

HFPF3373E *process* PROCESS ENDED ABNORMALLY DUE TO PREVIOUS ERRORS.

Explanation

The indicated process ended abnormally due to one or more errors. Messages that describe the errors are issued before this message.

HFPG message

The following information is about the message and code that begin with HFPG.

HFPG0001W CONNECTION TO THE ITKB SERVER FAILED.

Explanation

The connection to the IMS Tools KB server failed. This message might be issued for the following reasons:

- The server configuration is incomplete.
- The server is not started.
- The server XCF group name that is specified by the ITKBSRVR keyword is incorrect.
- Insufficient access authority to repository.

System action

Processing continues, but IMS HP Fast Path Utilities does not store any reports to the IMS Tools KB Output repository.

User response

If you want to store the reports to the IMS Tools KB Output repository, complete the following steps:

System action

The job ends with an abend code of U3003.

User response

Locate the error messages that are issued before this message. Correct the errors and rerun the job.

HFPF3374E "PGSER" MACRO FAILED. (RC: xx)

Explanation

The program issued a PGSER macro to do page fix or page free. The return code shows that the attempt was unsuccessful.

System action

The job ends with an abend code of U3003.

User response

Correct any errors, and rerun the job. If this situation persists, contact IBM Software Support.

1. Ensure that the server XCF group name specified on the ITKBSRVR keyword is correct.
2. Ensure that the IMS Tools KB server is configured and started without any errors. For configuration steps, see the topic "Configuring IMS Tools Knowledge Base" in the *Tools Base Configuration Guide for IMS*.

If the problem persists, contact your system administrator to obtain the required level of authorization.

HFPG0002W THE ACCESS TO OUTPUT REPOSITORY WAS CANCELED REASON: *reason*

Explanation

IMS HP Fast Path Utilities canceled its access to the IMS Tools KB Output repository because the initialization process failed. The reason is one of the following:

- Dynamic allocation for the library that is specified with the ITKBLOAD keyword failed

- Failed to open the library specified with the ITKBLOAD keyword
- Failed to load the HKTXXLI module

System action

Processing continues, but IMS HP Fast Path Utilities does not store any reports to the IMS Tools KB Output repository.

User response

If you want to store the reports to the IMS Tools KB Output repository, specify the correct load module library of the IMS Tools KB product.

HFPG0003W HPFPU WAS NOT DEFINED IN ITKB.

Explanation

IMS HP Fast Path Utilities was not defined in the IMS Tools KB information management environment as a product that can store reports to the IMS Tools KB Output repository.

System action

Processing continues, but IMS HP Fast Path Utilities does not store any reports to the IMS Tools KB Output repository.

User response

If you want to store the reports to the IMS Tools KB Output repository, register the IMS HP Fast Path Utilities product by using the IMS Tools KB product administration utility (HKTAPRA0).

HFPG0004W REPORT WAS NOT DEFINED IN ITKB.

Explanation

The report was not defined in the IMS Tools KB information management environment.

System action

Processing continues, but IMS HP Fast Path Utilities does not store the report to the IMS Tools KB Output repository.

User response

If you want to store the report to the IMS Tools KB Output repository, register the report by using the IMS Tools KB product administration utility (HKTAPRA0).

**HFPG0005W ERROR OCCURRED IN ACCESSING OUTPUT REPOSITORY FUNC:
function RC: rc RSN: rsn**

Explanation

An error occurred while getting access to the IMS Tools KB Output repository.

System action

Processing continues. If the return code is equal to or greater than 08, IMS HP Fast Path Utilities does not store its reports to the IMS Tools KB Base Output repository.

User response

If you want to store the reports to the IMS Tools KB Output repository, correct the error. If any of the messages, HFPG0001W, HFPG0003W, HFPG0004W, or HFPG0007W, which describes the cause of the error, are issued following this message, see the explanation for those messages. If these messages are not issued, check the return code and the reason code specified in this message. The codes are in hexadecimal. For the description of the return code and reason code, see the *Tools Base IMS Tools Knowledge Base User's Guide*.

**HFPG0006W DBD: dbdname AREA: areaname
DD: ddname REPORT: reportname**

Explanation

This message is displayed after message HFPG0005W.

System action

Processing continues. If the return code in the message HFPG0005W is equal to or greater than 08, IMS HP Fast Path Utilities does not store the *reportname* report for *dbdname*, *areaname*, and *ddname*.

User response

See the description for message HFPG0005W.

HFPG0007W RECON ENTRY WAS NOT FOUND IN ITKB.

Explanation

The RECON entry was not defined in your IMS Tools KB information management environment.

System action

Processing continues, but IMS HP Fast Path Utilities does not store any reports to the IMS Tools KB Output repository.

User response

If you want to store the reports to the IMS Tools KB Output repository, add a RECON environment. For more information about adding the RECON environment, see the topic "RECON ID (locale) administration" in the *Tools Base Configuration Guide for IMS*.

HFPG0009E	AREA <i>areaname</i> IS SPECIFIED MORE THAN ONCE BY THE FOLLOWING KEYWORD PARAMETERS: - "[<i>areaname1</i> <i>range1</i> <i>wildcard1</i>]" AND "[<i>areaname2</i> <i>range2</i> <i>wildcard2</i>]"
------------------	---

Explanation

The indicated area is specified more than once.

System action

For FPA, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, an abend occurs with a code of U3003. For FPO and FP stand-alone DB Sensor, the job ends with a return code of 8.

User response

Correct the command set in HFPSYSIN for FPA and FP stand-alone DB Sensor, or in FPXIN for FPO, and rerun the job.

HFPG0010E	DBD <i>dbdname</i> CONTAINS NO AREA THAT MATCHES THE WILDCARD CONDITION (<i>xxxxxxxx</i>).
------------------	---

Explanation

No area matches the wild card specification.

System action

For FPA, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, an abend occurs with a code of U3003. For FPO and FP stand-alone DB Sensor, the job ends with a return code of 8.

User response

Correct the command set in HFPSYSIN for FPA and FP stand-alone DB Sensor, or in FPXIN for FPO, and rerun the job.

HFPG0011E	[IAREA OAREA AREA]=[<i>n1-n2</i> <i>area1-areaN</i>] SPECIFICATION IS INVALID. - THE FIRST OPERAND VALUE MUST BE SMALLER THAN OR EQUAL TO THE SECOND OPERAND VALUE. - THE SPECIFIED AREA NUMBER (<i>n1-n2</i>) IS NOT DEFINED IN DBD <i>dbdname</i> . - THE MAXIMUM AREA NUMBER IS <i>nnnn</i> . - THE AREA RANGE IS SPECIFIED IN REVERSE ALPHABETICAL ORDER. - THE SPECIFIED AREA [<i>area1</i> <i>areaN</i>] IS NOT DEFINED IN DBD <i>dbdname</i> . - AN AREA RANGE MUST BE SPECIFIED BY A PAIR OF AREA NUMBERS OR BY A PAIR OF AREA NAMES. - WILDCARD PARAMETERS CANNOT BE USED TO SPECIFY THE AREA RANGE. - FORMAT ERROR - LENGTH ERROR
------------------	---

Explanation

The values specified on the indicated keyword are incorrect.

System action

For FPA, if SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, an abend occurs with a code of U3003. For FPO and FP stand-alone DB Sensor, the job ends with a return code of 8.

User response

Correct the command set in HFPSYSIN for FPA and FP stand-alone DB Sensor, or in FPXIN for FPO, and rerun the job.

HFPH message

The following information is about the message and code that begin with HFPH.

HFPH0001W

**DMACVRT REPORT FOR AREA NO: *nnn*, AREANAME: *areaname* WAS MAPPED BY IMS VERSION: *mmm*, BECAUSE OF THE FOLLOWING REASON:
- DMACVNO WAS NOT RECOGNIZED AS THE CURRENTLY SUPPORTED IMS VERSION**

Explanation

The DMAC block for area number *nnn*, area *areaname* was mapped by using the mapping macro for IMS version *mmm*.

System action

The return code is set to 4 and processing continues.

User response

None.

HFPI messages

The following information is about messages and codes that begin with HFPI.

HFPI0001I

IMAGE COPY OUTPUT PROCESSING FOR *command* STARTED.

HFPI0003E

NOTIFY.IC COMMAND WAS NOT ISSUED, BECAUSE THE *command* PROCESS ENDED ABNORMALLY.

Explanation

The Image copy output process for command *command* has started.

System action

Processing continues.

User response

None. This message is informational.

HFPI0002I

IMAGE COPY OUTPUT PROCESSING FOR *command* COMPLETED.

HFPI0004E

**NOTIFY.IC COMMAND PROCESSING FOR *command* FAILED.
- GETMAIN FAIL.
- THE COMMAND FAILED WITH RETURN CODE: *nn*.**

Explanation

The Image copy output process for command *command* has completed.

System action

Processing continues.

User response

None. This message is informational.

Explanation

FPA did not issue a NOTIFY.IC command because the command *command* process ended abnormally.

System action

The return code is set to 8 and processing continues.

User response

Check the cause of error for the command *command* process.

Explanation

This message consists of multiple parts. These parts are:

- GETMAIN FAIL.

The GETMAIN macro to obtain space for internal blocks failed.

- THE COMMAND FAILED WITH RETURN CODE: nn.

An error occurred when FPA issued a NOTIFY.IC command after the image copy processing completed.

System action

The return code is set to 8 and processing continues.

User response

- GETMAIN FAIL.

If the region size specified is too small, increase the REGION size on the JOB statement in the JCL, and rerun the utility.

- THE COMMAND FAILED WITH RETURN CODE: nn.

See the messages that were issued from the DBRC utility, and identify the cause of the error. The messages are printed in the Image Copy Output report.

**HFPI0005E AN ERROR WAS DETECTED IN THE HPIC PROCESSING FOR *command*.
- FUNCTION: *func*, RETURN CODE: *xx*, REASON CODE: *yyyyyyyy***

Explanation

An error was detected during the IMS HP Image Copy process for command *command*.

System action

The return code is set to 8 and processing continues.

User response

Check the error message from IMS HP Image Copy in the job log. Correct the error, and rerun the job.

**HFPI0006E IMAGE COPY OUTPUT PROCESSING FAILED.
- RECON V.R.M NOT SUPPORT LEVEL: *xxxxxxxx*.
- ADS AVAILABLE FOR IMAGE COPY WAS NOT FOUND.
- THE PTF LEVEL OF THE HPIC MODULE IS NOT HIGH ENOUGH.**

Explanation

The Image copy output process failed. For details, see the subtext of this message.

System action

The return code is set to 8 and processing continues.

User response

Correct the error, and rerun the job.

HFPI0007W IMAGE COPY OUTPUT PROCESSING FOR *command* COMPLETED WITH WARNING.

Explanation

The Image copy output process for command *command* completed with warning.

System action

The return code is set to 4 and processing continues.

User response

Make sure the warning message from IMS HP Image Copy in the job log, and if necessary, rerun the job.

HFPI0008E IMAGE COPY OUTPUT PROCESSING FOR *command* COMPLETED WITH ERROR.

Explanation

The Image copy output process for command *command* completed with error.

System action

The return code is set to 8 and processing continues.

User response

Check the error message from IMS HP Image Copy in the job log. Correct the error, and rerun the job.

HFPI0009I *xxxxxxxx* PROCESS ENDED WITH ERRORS. THE NOTIFY.IC COMMAND WAS NOT ISSUED FOR AREA NO: *nnnn*, AREANAME: *areaname*

Explanation

FPA did not issue a NOTIFY.IC command for the indicated area because the indicated command processing ended with errors.

System action

The return code is set to 8 and processing continues.

User response

Check the cause of error for the indicated area that occurred during the indicated command processing. Correct any errors, and rerun the job.

HFPJ0001E **ERROR RETURN FROM IEANTCR,
RETURN IS *nn*.**

Explanation

FPA called the IEANTCR service, but an error return code was returned from the service. Value *nn* shows the return code of the IEANTCR service.

HFPJ messages

The following information is about messages and codes that begin with HFPJ.

HFPJ0001I **UNLOAD PROCESSING STARTED
FOR AREA NO: *nnnn*, AREANAME:
areaname
- INPUT DATA SET IS IMAGE COPY.**

Explanation

The Unload process for area number *nnnn*, area *areaname* has started. When the input data set is an image copy, message "INPUT DATA SET IS IMAGE COPY" is issued. Otherwise, it is not issued.

System action

Processing continues.

User response

None. This message is informational.

HFPJ0002I **UNLOAD PROCESSING
COMPLETED FOR AREA NO: *nnnn*,
AREANAME: *areaname***

Explanation

The Unload process for area number *nnnn*, area *areaname* has completed.

System action

Processing continues.

User response

None. This message is informational.

HFPJ0003I **UNLOAD PROCESSING
COMPLETED FOR AREA NO: *nnnn*,
AREANAME: *areaname* WITH THE
FOLLOWING CONDITION.**

System action

The job ends with an abend code of U3003.

User response

This error is likely an internal system error. Contact IBM Software Support.

**- SOME RANDOMIZED DB
RECORDS ARE NOT WRITTEN TO
UNLOADED SEGMENT RECORD
FILE.
- AREA IS EMPTY.
- AREA IS EMPTY, OR THE
PROCESSING FOR THE AREA WAS
SKIPPED DUE TO AN ERROR.**

Explanation

The Unload process for area number *nnnn*, area *areaname* completed.

**- SOME RANDOMIZED DB RECORDS ARE NOT
WRITTEN TO UNLOADED SEGMENT RECORD FILE.**

Some of the database records were randomized to another area that is not specified with the OAREA keyword.

- AREA IS EMPTY.

This area is empty.

**- AREA IS EMPTY, OR THE PROCESSING FOR THE
AREA WAS SKIPPED DUE TO AN ERROR.**

This area is empty, or the processing for this area was skipped because an error occurred.

System action

Processing continues.

User response

None. This message is informational.

HFPJ0004W **SUBSET POINTER ERROR IN AREA
NO: *nnnn*, AREANAME: *areaname*
- SEGMENT (SEG-CD: *nnnn*) AT
RBA: *xxxxxxxx* CONTAINS
POINTER(S).
- SUBSET POINTER NO.: *nnnn*,
RBA: *xxxxxxxx***

(- SUBSET POINTER NO.: *nnnn*,
RBA: *xxxxxxxx*
:
:
- SUBSET POINTER NO.: *nnnn*,
RBA: *xxxxxxxx*)
**ABOVE SSP(S) DOESN'T POINT TO
CORRECT SEGMENT(S) (SEG-CD:
nnnn).**

Explanation

Subset pointers in the area number *nnnn*, area name *areaname* are in error. The program issues message HFPJ3104E and ends abnormally unless PTRCHK=BYPASS is specified in HFPSYSIN. The following messages show subset pointers that do not point to correct segments.

System action

If PTRCHK=BYPASS is specified, the return code is set to 4 and processing continues. Or if PTRCHK=ABEND is specified, the job ends with an abend code of U3003.

User response

Consult your database administration personnel about procedures for fixing the SUBSET pointer error. If the error message was issued, rerun the unload job.

**HFPJ0005W DB RECORD(S) RANDOMIZED TO
DIFFERENT AREA FROM INPUT
AREA NO: *nnnn*, AREANAME:
*areaname***

Explanation

As a result of calling the randomizer, the input area no. *nnnn*, area name *areaname* was randomized to be reloaded to another area, though IAREA=ALL was not specified. Without IAREA=ALL, this job step might not be able to unload complete DEDB records to be reloaded to an area. Some DEDB records to be reloaded to an area might be in areas which are not specified as input of this job step.

System action

The return code is set to 4 and processing continues.

User response

If this is not the expected result, check the load library that contains the randomizer, correct the error, and rerun the job.

**HFPJ0006W UOW RANGE ERROR IN AREA NO:
nnnn, AREANAME: *areaname*
- SEG-CD: *nnnn*, RBA: *xxxxxxxx*
WITH PROBLEM
- START RBA: *xxxxxxxx*, LAST RBA:
*xxxxxxxx***

Explanation

Detected an incorrect RAP, PCF, or PTF pointer in area number *nnnn* area name *areaname* during the Unload process. The program issues message HFPJ3104E and ends abnormally unless PTRCHK=BYPASS is specified in HFPSYSIN. The messages that follow this message show which pointers are incorrect.

System action

If PTRCHK=BYPASS is specified, return code is set to 4 and processing continues. Or if PTRCHK=ABEND is specified, the job ends with an abend code of U3003.

User response

Consult your database administration personnel about the procedures for fixing the xxx pointer error. If the error message was issued, rerun the unload job.

**HFPJ0007W INCORRECT [RAP/PCF/PTF]
POINTER IN AREA NO: *nnnn*,
AREANAME: *areaname*
- SOURCE SEGMENT SEG-CD: *nnnn*
- TARGET SEGMENT SEG-CD:
mmmm
- RBA: *xxxxxxxx***

Explanation

Detected an incorrect RAP, PCF, or PTF pointer in area number *nnnn* area name *areaname* during the Unload process. The program issues message HFPJ3104E and ends abnormally unless PTRCHK=BYPASS is specified in HFPSYSIN. The messages that follow this message show which pointers are incorrect.

System action

If PTRCHK=BYPASS is specified, return code is set to 4 and processing continues. Or if PTRCHK=ABEND is specified, the job ends with an abend code of U3003.

User response

Consult your database administration personnel about the procedures for fixing the xxx pointer error. If the error message was issued, rerun the unload job.

**HFPJ0007W INCORRECT *xxx* POINTER IN AREA
NO: *nnnn*, AREANAME: *areaname***

- RAP AT RBA: xxxxxxxx HAS
VALUE OF xxxxxxxx

Explanation

Detected an incorrect RAP, PCF, or PTF pointer in area number *nnnn* area name *areaname* during the Unload process. The program issues message HFPJ3104E and ends abnormally unless PTRCHK=BYPASS is specified in HFPSYSIN. The messages that follow this message show which pointers are incorrect.

System action

If PTRCHK=BYPASS is specified, return code is set to 4 and processing continues. Or if PTRCHK=ABEND is specified, the job ends with an abend code of U3003.

User response

Consult your database administration personnel about the procedures for fixing the xxx pointer error. If the error message was issued, rerun the unload job.

HFPJ0008E	PROCESSING ENDED BECAUSE PTRCHK=ABEND WAS SPECIFIED.
------------------	---

Explanation

Processing ended because a pointer error was detected during unload processing and PTRCHK=ABEND is specified in HFPSYSIN.

System action

The return code is set to 8 and processing continues.

User response

Check preceding messages to determine the cause of the pointer error. Correct any errors, and rerun the job.

HFPJ0009E	SOME RANDOMIZED DB RECORDS ARE NOT WRITTEN TO THE UNLOADED SEGMENT RECORD FILE FOR AREA NO: <i>nnnn</i>, AREANAME: <i>areaname1</i> - TO WRITE ALL DB RECORDS, ADD OAREA=<i>areaname2</i>
------------------	--

Explanation

The Unload process for area number *nnnn*, area *areaname1* completed, but some of the database records were randomized to another area that is not specified with the OAREA keyword. This message is issued only when DBREC_DISCARD_MSGLVL=ERROR is specified.

areaname1 and *areaname2* might be the same area name depending on the missing OAREA parameter.

System action

The return code is set to 8 and processing continues.

User response

Specify, for the OAREA keyword, the area name (*areaname2*) that is shown in the sub text.

HFPJ0010I	DB RECORDS RANDOMIZED TO A DIFFERENT AREA. INPUT AREA WAS AREA NO: <i>nnnn</i>, AREANAME: <i>areaname</i>
------------------	--

Explanation

During online structure change, one or more database records were randomized to an area that is different from the original area.

System action

Processing continues.

User response

None. This message is informational.

HFPJ0011I	UNLOAD PROCESSING STARTED FOR UNLOADED SEGMENT RECORD FILE. [- DDNAME: <i>ddname</i>, DSNAME: <i>dsname</i> - DDNAME: <i>ddname</i>, DSNAME: <i>dsname(member)</i>]
------------------	--

Explanation

The Unload process for the unloaded segment record file that was specified in DD statement *ddname* and data set *dsname* (and member *member*) has started.

System action

Processing continues.

User response

None. This message is informational.

HFPJ0012I	UNLOAD PROCESSING COMPLETED FOR UNLOADED SEGMENT RECORD FILE [THAT CONTAINS THE DBD INFORMATION RECORDS].
------------------	--

[- DDNAME: *ddname*, DSNAME: *dsname*
- DDNAME: *ddname*, DSNAME: *dsname(member)*]

Explanation

The Unload process for the unloaded segment record file that was specified in DD statement *ddname* and data set *dsname* (and member *member*) has completed.

System action

Processing continues.

User response

None. This message is informational.

**HFPJ0013I SORT PROCESS WAS SKIPPED
 FOR DDNAME: *ddname*.**

Explanation

The sort process was skipped for the unloaded segment record file or the extracted output record file that was specified in DD statement *ddname*.

For the UNLOAD command, this message is issued when all of the following conditions are met:

- A single area was unloaded into the file.
- The SDEP=LOGICAL was not specified in HFPSYSIN.
- The Unload function was processed in the REORG mode.

For the EXTRACT command, this message is issued when both of the following conditions are met:

- A single area was extracted into the file.
- The database that was specified in DBD did not have an SDEP segment.

System action

Processing continues.

User response

None. This message is informational.

**HFPJ0014W POINTER ERROR INFORMATION
 IS NOT PROVIDED. REASON: *xxx***
**- AREA NO: *nnnn*, AREANAME:
 *areaname***

Explanation

The pointer error information is not provided because of one of the following reasons:

- STORAGE ALLOCATION FAILED FOR THE UOW TABLE
- DYNAMIC ALLOCATION FAILED FOR THE REPORT DATA SET
- OPEN FAILED FOR THE REPORT DATA SET

System action

The return code is set to 4 and processing continues.

User response

Correct any errors, and rerun the job.

**HFPJ0015W CI MAP/CI DUMP REPORT FOR
 SDEP IS NOT PROVIDED. REASON:
 *xxx***
**- AREA NO: *nnnn*, AREANAME:
 *areaname***

Explanation

The CI map/CI dump report for SDEP is not provided because of one of the following reasons:

- DYNAMIC ALLOCATION FAILED FOR THE REPORT DATA SET
- OPEN FAILED FOR THE REPORT DATA SET

System action

The return code is set to 4 and processing continues.

User response

Correct any errors, and rerun the job.

**HFPJ0021I EXTRACT PROCESSING STARTED
 FOR AREA NO: *nnnn*, AREANAME:
 *areaname***

Explanation

The Extract process for area number *nnnn*, area *areaname* has started.

System action

Processing continues.

User response

None. This message is informational.

HFPJ0022I **EXTRACT PROCESSING
COMPLETED FOR AREA NO: *nnnn*,
AREANAME: *areaname***

Explanation

The Extract process for area number *nnnn*, area *areaname* has completed.

System action

Processing continues.

User response

None. This message is informational.

HFPJ0023I **EXTRACT PROCESSING
COMPLETED FOR AREA NO: *nnnn*,
AREANAME: *areaname* WITH THE
FOLLOWING CONDITION.
- AREA IS EMPTY**

Explanation

The Extract process for area number *nnnn*, area *areaname* has completed but there is no segment in the area.

System action

Processing continues.

User response

None. This message is informational.

HFPJ0024I **EXTRACT PROCESSING STARTED
FOR EXTRACTED OUTPUT RECORD
FILE.
[- DDNAME: *ddname*, DSNAME:
dsname
- DDNAME: *ddname*, DSNAME:
dsname(member)]**

Explanation

The Extract process for the extracted output record file that was specified in the DD statement *ddname* and data set *dsname* (and member *member*) has started.

System action

Processing continues.

User response

None. This message is informational.

HFPJ0025I **EXTRACT PROCESSING
COMPLETED FOR EXTRACTED
OUTPUT RECORD FILE.
[- DDNAME: *ddname*, DSNAME:
dsname
- DDNAME: *ddname*, DSNAME:
dsname(member)]**

Explanation

The Extract process for the extracted output record file that was specified in the DD statement *ddname* and data set *dsname* (and member *member*) has completed.

System action

Processing continues.

User response

None. This message is informational.

HFPJ0026W **OUTPUT DATA SPECIFICATION IS
BEYOND SEGMENT LENGTH IN
AREA NO: *nnnn*, AREANAME:
areaname
- SEG-CD: *nnn*
- DDNAME: *xxxxxxxx***

Explanation

The FIELDS keyword of the OUTPUT subcommand showed a relative starting position *ss* and length *ll*, which exceeded the length of the segment. This message is issued only once per a segment type in an area.

System action

The record being processed is not included in the output. The return code is set to 4 and processing continues.

User response

Correct the starting position *ss*, length *ll*, or both in the FIELDS keyword, and rerun the job.

HFPJ0030I **REORG PROCESSING STARTED
FOR AREA NO: *nnnn*, AREANAME:
*areaname***

Explanation

The Reorg process for area number *nnnn*, area *areaname* has started.

System action

Processing continues.

User response

None. This message is informational.

HFPJ0031I	REORG PROCESSING COMPLETED FOR AREA NO: <i>nnnn</i>, AREANAME: <i>areaname</i>
------------------	---

Explanation

The Reorg process for area number *nnnn*, area *areaname* has completed.

System action

Processing continues.

User response

None. This message is informational.

HFPJ0032W	REORG PROCESSING COMPLETED FOR AREA NO: <i>nnnn</i>, AREANAME: <i>areaname</i> WITH EMPTY UOW PROCESS
------------------	--

Explanation

The Reorg process for area number *nnnn*, area *areaname* has completed, but there is no segment in the area.

System action

The return code is set to 4 and processing continues.

User response

None.

HFPJ0033E	REORG PROCESSING COMPLETED FOR AREA NO: <i>nnnn</i>, AREANAME: <i>areaname</i> WITH THE ABOVE CONDITION.
------------------	---

Explanation

The Reorg process for area number *nnnn*, area *areaname* completed with an error.

System action

The return code is set to 8 and processing continues.

User response

Check the preceding message and rerun the job.

HFPJ0034E	ERROR OCCURRED AND REORG OF UOW WAS CANCELED FOR AREA NO: <i>nnnn</i>, AREANAME: <i>areaname</i> - UOW NO: <i>mmmmm</i>, UOW FIRST RBA: <i>xxxxxxxx</i>
------------------	--

Explanation

The Reorg process of UOW number *mmmmm* for area number *nnnn*, area *areaname* was canceled.

System action

The return code is set to 8 and processing continues.

User response

Check the preceding message and rerun the job.

HFPJ0035E	REORG PROCESSING CANCELED FOR AREA NO: <i>nnnn</i>, AREANAME: <i>areaname</i> WITH THE ABOVE CONDITION.
------------------	--

Explanation

The Reorg process for area number *nnnn*, area *areaname* was canceled because some errors were detected.

System action

The return code is set to 8 and processing continues.

User response

Check the preceding message and rerun the job.

HFPJ0036W	REORG PROCESSING COMPLETED FOR AREA NO: <i>nnnn</i>, AREANAME: <i>areaname</i> WITH WARNING.
------------------	---

Explanation

The Reorg process for area number *nnnn*, area *areaname* completed with warning.

System action

The return code is set to 4 and processing continues.

User response

None.

HFPJ0037E **AREA ANALYZER OR AREA
SENSOR PROCESS ENDED WITH
ERROR CONDITION FOR AREA NO:
nnnn, AREANAME: areaname**

Explanation

The area analyzer or the area sensor process ended with some error messages.

System action

The return code is set to 8 and processing continues.

User response

Check the preceding message and rerun the job.

HFPJ0038E **INITIALIZATION OF REORG
REPORT PROCESSING FAILED FOR
AREA NO: nnnn, AREANAME:
areaname**
- GETMAIN FAILED.
- DYNAMIC ALLOCATION FAILED.

Explanation

Program GETMAIN or dynamic allocation for the data set in the Reorg report process for area number *nnnn*, area *areaname* failed.

System action

The return code is set to 8 and processing continues.

User response

If GETMAIN failed and the specified region size is too small, increase the REGION size on the JOB statement in the JCL, and rerun the utility. If dynamic allocation failed, it is probably an internal system error. Contact IBM Software Support.

HFPJ0039I **REORG PROCESSING SKIPPED
FOR AREA NO: nnnn, AREANAME:
areaname DUE TO COMPLETION
BY PREVIOUS REORG.**

Explanation

The Reorg process for area number *nnnn*, area *areaname* was skipped because the area was successfully reorganized in the previous run.

System action

Processing continues.

User response

None. This message is informational.

HFPJ0039I **REORG PROCESSING SKIPPED
FOR AREA NO: nnnn, AREANAME:
areaname DUE TO
RECOVPT=EMERGENCY OPTION.**

Explanation

The Reorg process for area number *nnnn*, area *areaname* was skipped because RECOVPT=EMERGENCY is specified.

System action

Processing continues.

User response

None. This message is informational.

HFPJ3101E **INCONSISTENCY WAS FOUND IN
DMAC AND DMB.**

Explanation

During a consistency check in the Unload process, an inconsistency was detected between VSAM 2nd CI and DMB. For more information, check message HFPS0001E that follows this message.

System action

The job ends with an abend code of U3003.

User response

Check message HFPS0001E. Correct any errors, and rerun the job.

HFPJ3104E **PROCESSING ENDED FOR
PTRCHK=ABEND.**

Explanation

Detected a pointer error during the Unload process, and PTRCHK=ABEND is specified in HFPSYSIN.

System action

The job ends with an abend code of U3003.

User response

Check preceding messages and determine the cause of the pointer error. Correct any errors, and rerun the job.

HFPJ3105E IAREA=ALL IS REQUIRED FOR CHANGE COMMAND.
- DB RECORD(S) RANDOMIZED TO DIFFERENT AREA FROM INPUT AREA NO: *nnnn*, AREANAME: *areaname*

Explanation

One or more segment records were randomized to different areas even though IAREA=ALL was not specified for the CHANGE command.

System action

The job ends with an abend code of U3003.

User response

Either specify IAREA=ALL or check that the randomizer is correctly specified, and rerun the job.

HFPJ3107E INCORRECT "BLOCK TYPE ID" DETECTED IN AREA NO: *nnnn*, AREANAME: *areaname*
- VALUE EXPECTED: X"xx" ("*cccc*" CI)
- VALUE FOUND: X"xx"
- RBA: *xxxxxxxx*

Explanation

An unload subtask read a CI from the area number *nn*, area name *areaname*. The value in the IMS field known as DBLKBTID was not the value that was expected. *cccc* will be one of DMAC, BASE, FOVF, IOVF, and SDEP.

System action

The job ends with an abend code of U3003.

User response

Consult your database administration personnel about procedures for fixing the incorrect data. When the problem is corrected, rerun the unload job.

HFPJ3108E SEGMENT IN AREA NO; *nnnn* AT RBA: *xxxxxxxx* (SEG-CD: *nnnn*) EXCEEDS MAX DEFINED LENGTH.

Explanation

An unload subtask determined that the data in the specified segment was longer than the value that is allowed by the definition for that segment in the *output* DMB. *nnnn* is the area number.

System action

The job ends with an abend code of U3003.

User response

Correct the errors, and rerun the job.

HFPJ3109E INCONSISTENCY WAS FOUND IN SPACE MAP FOR AREA NO: *nnnn*, AREANAME: *areaname*, RBA: *xxxxxxxx*
- OFFSET: *xxxxxxxx*
- VALUE EXPECTED: X"xx"
- VALUE FOUND: X"xx"

Explanation

The Unload process determined that the space map at RBA *xxxxxxxx* OFFSET *xxxxxxxx* has an inconsistent value.

System action

If the REORG command is specified, the return code is set to 8 and processing continues. Or if the CHANGE, EXTRACT, or UNLOAD command is specified, the job ends with an abend code of U3003.

User response

Correct the errors, and rerun the job.

HFPJ3112E INCORRECT SDEP POINTER IN AREA NO: *nnnn*, AREANAME: *areaname*
- SEG-CD: *nnnn*, RBA: *xxxxxxxx* WITH PROBLEM.
- PHYSICAL BEGIN: *yyyyyyyy*
- PHYSICAL END: *zzzzzzzz*

Explanation

The SDEP pointer value that is contained in the segment at RBA *xxxxxxxx* does not point to the SDEP part of the DEDB area number *nn* area name *areaname*. *yyyyyyyy* is the specified RBA of the first SDEP CI, *zzzzzzzz* is the specified RBA of the first CI that is above the address SDEP.

System action

The job ends with an abend code of U3003.

User response

Run the full pointer checker job to determine the cause of the error, and rerun the job.

HFPJ3113E **DATABASE RECORDS
RANDOMIZED TO DIFFERENT
AREA FROM INPUT AREA NO:
nnnn, AREANAME: areaname**

Explanation

One or more segment records were randomized to different areas even though SDEP=PHYSICAL and RELOCATE_SDEP=YES are specified for the UNLOAD or the CHANGE command.

System action

The job ends with an abend code of U3003.

User response

Ensure that the randomizer is specified correctly, and rerun the job.

HFPJ3120E **NO ADS DDNAME IN DBRC FOR
AREA NO: nnnn, AREANAME:
areaname
- DDNAME: ddname DSNAME:
dsname**

Explanation

There is no area data set registered to DBRC for the specified area.

System action

The job ends with an abend code of 3003.

User response

Get a LIST.RECON output report. Find and specify an unused area data set name. Specify the name for the input ADS, and rerun the job.

HFPJ3121E **ADS DSNAME IS NOT THE SAME
IN DBRC FOR AREA NO: nnnn,
AREANAME: areaname, DDNAME:
ddname
- ALLOCATED DSNAME: dsname
- DBRC DSNAME: dsname**

Explanation

The area data set name that is specified in the *ddname* DD statement is not the same as the one registered to DBRC.

System action

The job ends with an abend code of 3003.

User response

Get a LIST.RECON output report. Find an unused area data set name, then specify the name for the input ADS, and rerun the job.

HFPJ3130E **HASH CHECK CANNOT CONTINUE
PROCESSING IN AREA NO: nnnn,
AREANAME: areaname
- UOW NO:nnnn, UOW FIRST RBA:
XXXXXXXX**

Explanation

Hash check cannot continue processing in area number *nnnn*, area *areaname*.

System action

The job ends with an abend code of U3003.

User response

Correct the errors, and rerun the job.

HFPJ3140E **A POINTER ERROR OCCURRED IN
AREA NO: nnnn, AREANAME:
areaname. INDEX BUILDING
PROCESS CANNOT CONTINUE.**

Explanation

Because a pointer error is found, the Build Index process cannot continue processing in area number *nnnn*, area *areaname*.

System action

The job ends with an abend code of U3003.

User response

Correct the pointer error, and rerun the job.

HFPK messages

The following information is about messages and codes that begin with HFPK.

HFPK0001I [RELOAD | INITIALIZE]
PROCESSING STARTED FOR AREA
NO: *nnnnn*, AREANAME:
areaname[, USR FILE DDNAME:
ddname]

Explanation

This message is generated when the indicated process starts for each area.

System action

Processing continues.

User response

None. This message is informational.

HFPK0002I [RELOAD | INITIALIZE]
PROCESSING COMPLETED FOR
AREA NO: *nnnnn*, AREANAME:
areaname[, USR FILE DDNAME:
ddname]
[- AREA IS EMPTY]

Explanation

This message is generated when the indicated process is successfully completed for an area.

For reload processing, when this message is generated, the area has been successfully reloaded, and the data set has been closed. When there is no segment in the area, subtext, - AREA IS EMPTY, is issued. Otherwise, it is not issued.

System action

Processing continues.

User response

None. This message is informational.

HFPK0003W UNLOADED SEGMENT RECORDS
FILE HAS NO RECORD FOR
RELOADED AREAS.
- DDNAME: *ddname*, DSNAME:
dsname
- DDNAME: *ddname*, DSNAME:
dsname(member)

Explanation

The USR file *dsname* did not contain any records for the reloaded areas.

System action

The return code is set to 4 and processing continues.

User response

Check that the USR file is specified correctly, or that the USR file is created successfully.

HFPK0004E UNLOADED SEGMENT RECORDS
FILE HAS RECORDS FOR THE
AREA ALREADY RELOADED.
- AREA NO: *nnnnn*, AREANAME:
areaname
- ERROR FILE DDNAME: *ddname1*,
DSNAME: *dsname1*
- ERROR FILE DDNAME: *ddname1*,
DSNAME: *dsname1(member1)*
- USED FILE DDNAME: *ddname2*,
DSNAME: *dsname2*
- USED FILE DDNAME: *ddname2*,
DSNAME: *dsname2(member2)*

Explanation

The USR file DD statement *ddname1* contained records that were reloaded from another USR file *ddname2*. USR files *ddname1* and *dsname1* were used for reloading, but the area was already reloaded by USR files *ddname2* and *dsname2*. The process for user file *ddname1* and *dsname1* is skipped.

System action

The job ends with a return code of 08.

User response

Check that the USR file is specified correctly. Correct any errors, and rerun the job.

HFPK0005I HFPK0005I UNLOADED SEGMENT
RECORDS ALREADY PROCESSED
BY ANOTHER INPUT DD.
- DDNAME: *ddname*, DSNAME:
dsname
- DDNAME: *ddname*, DSNAME:
dsname(member)

Explanation

The USR file is specified and has been processed by another DD statement. No records for the reloaded areas are processed for the USR file.

System action

Processing continues.

User response

None. This message is informational.

HFPK0006E	FIRST RECORD FOR AREA NO: <i>nnnnn</i>, AREANAME: <i>areaname</i> IS NOT AN AREA INFORMATION RECORD. - DDNAME: <i>ddname</i> , DSNAME: <i>dsname</i> - DDNAME: <i>ddname</i> , DSNAME: <i>dsname(member)</i> - RECORD NUMBER: <i>nnnnnnnnnn</i>
------------------	---

Explanation

The first record of the USR file is not an area information record.

System action

The job ends with a return code of 08.

User response

Check that the USR file specified is created successfully. Correct any errors, and rerun the job.

HFPK0007E	UNLOADED SEGMENT RECORDS WITH SDEP=PHYSICAL ACCOMPANIED BY DBD CHANGE DETECTED. - AREA NO: <i>nnnnn</i> , AREANAME: <i>areaname</i> - DDNAME: <i>ddname</i> , DSNAME: <i>dsname</i> - DDNAME: <i>ddname</i> , DSNAME: <i>dsname(member)</i> - RECORD NUMBER: <i>nnnnnnnnnn</i>
------------------	---

Explanation

The input data set *dsname* is created with SDEP=PHYSICAL and in a different database structure.

System action

The job ends with a return code of 08.

User response

Run the job by FPB Reload.

HFPK0011I	RELOAD PROCESSING STARTED FOR UNLOADED SEGMENT RECORD FILE. - DDNAME: <i>ddname</i> , DSNAME: <i>dsname</i> - DDNAME: <i>ddname</i> , DSNAME: <i>dsname(member)</i>
------------------	--

Explanation

The Reload process has started for the unloaded segment record file.

System action

Processing continues.

User response

None. This message is informational.

HFPK0012I	RELOAD PROCESSING COMPLETED FOR UNLOADED SEGMENT RECORD FILE. - DDNAME: <i>ddname</i> , DSNAME: <i>dsname</i> - DDNAME: <i>ddname</i> , DSNAME: <i>dsname(member)</i>
------------------	--

Explanation

The Reload process has completed for the unloaded segment record file.

System action

Processing continues.

User response

None. This message is informational.

HFPK0013E	AREA NO: <i>nnnn</i>, AREANAME: <i>areaname</i> IS FULL.
------------------	---

Explanation

The program attempted to allocate an available IOVF CI from an IOVF part, but found that all CIs within the IOVF part were in use.

System action

The return code is set to 8 and processing continues.

User response

Correct the errors, and rerun the job.

HFPK0014I RECOVERY PROCESS STARTED FOR AREANO: *nnnn*, AREANAME: *areaname*, UOWNO: *nnnn*

Explanation

The recovery process for area number *nnnn*, area *areaname*, UOW number *nnnn* has started.

System action

Processing continues.

User response

None. This message is informational.

HFPK0015I RECOVERY PROCESS ENDED SUCCESSFULLY FOR AREANO: *nnnn*, AREANAME: *areaname*, UOWNO: *nnnn*

Explanation

The recovery process for area number *nnnn*, area *areaname*, UOW number *nnnn* has completed.

System action

Processing continues.

User response

None. This message is informational.

HFPK0016E RECOVERY PROCESS ENDED WITH ERROR FOR AREANO: *nnnnn*, AREANAME: *areaname*, UOWNO: *nnnn*

Explanation

The recovery process for area number *nnnn*, area *areaname*, UOW number *nnnn* has completed, but the UOW is not recovered.

System action

The return code is set to 8 and processing continues.

User response

Correct the errors, and rerun the job.

HFPK0030E [AREA SENSOR | HASH CHECK] PROCESS ENDED WITH AN ERROR

CONDITION FOR AREA NO: *nnnn*, AREANAME: *areaname*
- UOW NO:*nnnn*, UOW FIRST RBA: *xxxxxxx*

Explanation

The area sensor or the hash check process for area number *nnnn*, area *areaname* has completed, but some errors were detected.

System action

The return code is set to 8 and processing continues.

User response

Check the preceding message and rerun the job.

HFPK3200E NO ADS DDNAME IN DBRC FOR AREA NO: *nnnnn*, AREANAME: *areaname*
- DDNAME: *ddname*, DSNAME: *dsname*

Explanation

There is no area data set registered to DBRC for the specified area.

System action

The job ends with an abend code of 3003.

User response

Get a LIST.RECON output report, and specify an unused area data set name. Specify the name in the *dsname* DD statement, and rerun the job.

HFPK3201E ADS DSNAME NOT THE SAME IN DBRC FOR AREA NO: *nnnnn*, AREANAME: *areaname*, DDNAME: *ddname*
- ALLOCATED DSNAME: *dsname*
- DBRC DSNAME: *dsname*

Explanation

The area data set name that is specified in the *ddname* DD statement is not the same as the one registered to DBRC.

System action

The job ends with an abend code of 3003.

User response

Get a LIST.RECON output report and find an area data set name that is not used, then specify the name in the DARVSAM DD statement, and rerun the job.

HFPK3202E **SDEP FIRST CI RBA
DISCREPANCY DETECTED FOR
AREA NO: nnnnn, AREANAME:
areaname**

Explanation

The first SDEP CI RBA of the SDEP part of the target area is not the same as the unloaded area with the SDEP=PHYSICAL keyword parameter.

System action

The job ends with an abend code of 3003.

User response

VSAM ESDS definition of the target area data set is not correct. Define the target area data set with the same CI size and space definition of the unloaded area data set, and rerun the job.

HFPK3203E **INSUFFICIENT SPACE DEFINED
FOR AREA NO: nnnnn, AREANAME:
areaname**

Explanation

Insufficient space defined for area *areaname*. DBFUMINO will issue message DFS2510I under VSAM cluster allocation.

System action

The job ends with an abend code of 3003.

User response

Delete the VSAM cluster for the area, then redefine it with more space. Rerun the reload job.

HFPK3204E **FIRST INSERTED SEGMENT IS
NOT A ROOT SEGMENT.
- AREA NO: nnnnn, AREANAME:
areaname, SEG-CD: nnnn,
RAPRBA: nnnnnnnn**

Explanation

The first record to be inserted was not a root segment.

System action

The job ends with an abend code of 3003.

User response

Check that the user file is created successfully, and rerun the job.

HFPK3205E **RECORD SEQUENCE ERROR WAS
DETECTED IN UNLOADED
SEGMENT RECORDS FILE.
- DDNAME: ddname, DSNAME:
dsname
- DDNAME: ddname, DSNAME:
dsname(member)
- RECORD NUMBER: nnnnnnnnnn**

Explanation

Detected a record sequence error in the input segment data that is associated with ddname DURDATA.

System action

The job ends with an abend code of 3003.

User response

Make sure that the sort process for the data set ran successfully. If input to the program is a series of concatenated data sets, make sure that they are concatenated in an ascending area number order. Correct the error, and rerun the reload job.

HFPK3207E **ROOT KEY WAS NOT FOUND IN
UTBL WHEN RELOADING USR
RECORD.
- DDNAME: ddname, DSNAME:
dsname
- DDNAME: ddname, DSNAME:
dsname(member)
- RECORD NUMBER: nnnnnnnnnn**

Explanation

A root key is not found in the UTBLs for processing a segment that had ILC FLAG (USRLCFG) on (X'FF'). The cause of the error might be that there were no segments that had USRPFMT on (X'01') in an unloaded file, and nothing was saved in UTBL.

System action

The job ends with an abend code of 3003.

User response

If you created or modified the unloaded file, verify that the unloaded data set is correct. Rerun the job. If this situation persists, contact IBM Software Support.

HFPK3208E **SEGMENT CODE: *nnnn* IS NOT FOUND IN UTBL WHEN RELOADING USR RECORD.**
- DDNAME: *ddname*, DSNAME: *dsname*
- DDNAME: *ddname*, DSNAME: *dsname(member)*
- RECORD NUMBER: *nnnnnnnnnn*

Explanation

Detected different segment codes between the segment unloaded file that had the USRPFLG1 flag on and the segment in UTBLs that had USRLCFLG on (X'FF'). The segment code in UTBLs is shown in decimal format.

System action

The job ends with an abend code of 3003.

User response

If you created or modified the unloaded file, verify that the unloaded data set is correct. Rerun the job. If this situation persists, contact IBM Software Support.

HFPK3209E **INCORRECT SEGMENT RECORD DETECTED (REC#: *nnnnnnnnnn*)**
- USRPSCD FIELD CONTAINS WRONG VALUE.
- DDNAME: *ddname*, DSNAME: *dsname*
- DDNAME: *ddname*, DSNAME: *dsname(member)*

Explanation

The segment code in the USRPSCD field of the unloaded segment record that is specified by the record number (REC#) is not correct.

System action

The job ends with an abend code of 3003.

User response

Check the content of the unloaded segment record that is specified by the record number (REC#), correct the value of the USRPSCD field, and rerun the job.

HFPK3209E **INCORRECT SEGMENT RECORD DETECTED (REC#: *nnnnnnnnnn*)**
- SDEP=PHYSICAL RECORD SEQUENCE ERROR.
- DDNAME: *ddname*, DSNAME: *dsname*
- DDNAME: *ddname*, DSNAME: *dsname(member)*

Explanation

Detected a record sequence error in the input segment data that is associated with ddname *ddname*.

System action

The job ends with an abend code of 3003.

User response

Verify that the USR file has been created successfully, and rerun the job.

HFPK3210E **AREA NO: *nnnnn*, AREANAME: *areaname* IS FULL.**

Explanation

The program attempted to allocate an available IOVF CI from an IOVF part, but found that all CIs within the IOVF part were in use.

System action

The job ends with an abend code of 3003.

User response

Increase the size of the area, and restart the reload job. Check the UOW and ROOT parameters.

HFPK3211E **USR FILE UNLOADED BY FPB WITH THE LOADCTL OPTION IS NOT ALLOWED FOR RELOAD PROCESS WITH LOADCTL SUBCOMMAND.**

Explanation

The RELOAD command with the LOADCTL subcommand option detected that a USR (unloaded segment records) file that was unloaded by FPB FABCUR1 with the LOADCTL option was specified as input. The Reload process does not allow such a USR file when the LOADCTL subcommand is specified.

System action

The job ends with an abend code of 3003.

User response

Use the Reload process without the LOADCTL subcommand or FPB Reload to reload the unloaded segment records that are generated by FPB Unload with the LOADCTL keyword.

HFPK3212E **ERROR IN CALL TO RANDOMIZER:**
randomizer.
- RETURN CODE: *nn*
- INCORRECT AREA NO: *nnnnn*
- USR RECORD NO: *nnnn*
- DDNAME: *ddname*, DSNAME:
dsname

Explanation

The error was returned from the randomizer module.

System action

The job ends with an abend code of 3003.

User response

Correct the unloaded segment records data set or the randomizer module, and rerun the job.

HFPK3213E **SEGMENT HIERARCHICAL**
SEQUENCE ERROR WAS
DETECTED.
- SDEP OR DDEP MUST BE AFTER
ROOT SEGMENT.
- USR RECORD NO: *nnnn*
- DDNAME: *ddname*, DSNAME:
dsname

Explanation

A segment hierarchical sequence error was found in the unloaded segment records data set.

System action

The job ends with an abend code of 3003.

User response

Correct the unloaded segment records data set, and rerun the job.

HFPK3214E **INCORRECT USER FORMAT**
RECORD WAS DETECTED.

- SEGMENT NAME: *seg-name* IS
INCORRECT.
- SEGMENT CODE: *nn* IS IN
CONFLICT WITH SEGMENT NAME:
seg-name.
- SEGMENT CODE: *nn* IS
INCORRECT.
- SEGMENT DATA LENGTH IS TOO
SHORT.
- SEGMENT DATA LENGTH IS TOO
LONG.
- SEGMENT ATTRIBUTE IS IN
CONFLICT WITH DBD
DEFINITION.
- USR RECORD NO: *nnnn*
- DDNAME: *ddname*, DSNAME:
dsname

Explanation

Incorrect user format unloaded segment record was found while processing the RELOAD command.

System action

The job ends with an abend code of 3003.

User response

Correct the unloaded segment records data set, and rerun the job.

HFPK3215E **USR RAP RBA ISN'T RAP CI RBA**
IN AREA NO: *nnnn*, AREANAME:
areaname* RBA: *nnnnnnnn

Explanation

The RAP RBA in the prefix of the unloaded segment record is not an RBA of a RAP CI.

System action

The job ends with an abend code of 3215.

User response

Check whether the combination of the DEDB definition that is obtained from the IMSACB DD and the unloaded segment record file is correct. Correct the error and rerun the job.

HFPL messages

The following information is about messages and codes that begin with HFPL.

HFPL0001I **REPORT STARTED FOR COMMAND:**
command, **DATABASE:** *database*

Explanation

The report generation process has started for command *command* and database *database*.

System action

Processing continues.

User response

None. This message is informational.

HFPL0002I **REPORT COMPLETED FOR**
COMMAND: *command*, **DATABASE:**
database

Explanation

The report generation process has completed for command *command* and database *database*.

System action

Processing continues.

User response

None. This message is informational.

HFPL0003I **NO SEGMENTS WILL BE**
RELOADED TO AREA NO:
areaname, **AREANAME:** *areaname*.
AREA WILL BE EMPTY.

Explanation

No segments were found in area *areaname* during the Unload process. The area will be empty when the reload is to be processed.

System action

Processing continues.

User response

None. This message is informational.

HFPL0004I **NO SEGMENTS RELOADED TO**
AREA NO: *areaname*, **AREANAME:**
areaname. **AREA IS EMPTY.**

Explanation

No segments were found in area *areaname* during the Reload process. The area was empty.

System action

Processing continues.

User response

None. This message is informational.

HFPL0005W **DUMMY OR NULLFILE IS**
SPECIFIED FOR DDNAME: *ddname*

Explanation

DUMMY or DSN=NULLFILE was specified for the report output data set (HFPRPTS) DD statement.

System action

The return code is set to 4 and processing continues.

User response

If you did not intend to specify the dummy DD statement, correct the error, and rerun the job.

HFPL0006I **DUMMY OR NULLFILE IS**
SPECIFIED FOR DDNAME: *ddname*

Explanation

DUMMY or DSN=NULLFILE was specified for the area totals data set (HFPTOTAL) DD statement.

System action

Processing continues.

User response

If you did not intend to specify the dummy DD statement, correct the error, and rerun the job.

HFPL0007W **AREA WAS NOT RELOADED DUE**
TO NO USR RECORDS IN THE USR
FILE(S) OR SOME USR FILE
PROBLEMS.
- AREA NO: *n*, **AREANAME:**
areaname

Explanation

The area name *areaname* did not contain any records for the reloaded areas.

System action

The return code is set to 4 and processing continues.

User response

Check that the OAREA keyword is specified correctly, or that the USR file has been created successfully.

HFPL0008E *report WAS NOT PROVIDED.*
- DATABASE: *databasename*, AREA
NO: *nnnn*, AREANAME: *areaname*

Explanation

The specified report was not provided because the OPEN processing for an internal work file failed.

System action

The return code is set to 8 and processing continues.

HFPM messages

The following information is about messages and codes that begin with HFPM.

HFPM0001W **INCORRECT COMMAND IS FOUND
IN HFPSYSIN.**

Explanation

Detected an incorrect command during the analysis of the HFPSYSIN DD statement.

System action

Processing continues.

User response

Correct any errors, and rerun the job

HFPM0002E **SEVERE DEDB DEFINITION
DIFFERENCE DETECTED.**

Explanation

Detected one or more severe DEDB definition differences. SCAN=YES is specified in the GLOBAL command.

System action

The return code is set to 8.

User response

This error is likely an internal system error. Contact IBM Software Support.

HFPL0010E **REORG REPORT PROCESSING
FAILED FOR AREA NO : *nnnn*,
AREANAME: *areaname*
- GETMAIN FAILED.**

Explanation

Program GETMAIN failed in the Reorg report process.

System action

The return code is set to 8 and processing continues.

User response

If the specified region size is too small, increase the REGION size on the JOB statement in the JCL, and rerun the utility.

User response

Check the DBD Definition report and error messages to determine the cause of the problem. If error messages are issued, follow the user response for those messages. Correct the errors and rerun the job.

HFPM0003E **STORAGE OBTAIN FAILURE FOR
MAIN PROCESS.
- MESSAGE BUFFER POOL**

Explanation

GETMAIN to obtain space for internal blocks failed.

System action

The return code is set to 8 and processing continues.

User response

If the region size that is specified is too small, increase the REGION size on the JOB statement in the JCL, and rerun the utility.

HFPM0004E **THE CI SIZE CANNOT BE
CHANGED WHEN
RELOCATE_SDEP=YES IS
SPECIFIED.**

Explanation

The Unload or the Change process ran in non-REORG mode with SDEP=PHYSICAL and RELOCATE_SDEP=YES. However, the CI size of the original ACB is different from the new ACB.

System action

If a parameter other than NO is specified for the SCAN keyword, the return code is set to 8. If SCAN=NO is specified, the job ends with an abend code of U3016.

User response

If you want to process Unload or Change in non-REORG mode with SDEP=PHYSICAL and RELOCATE_SDEP=YES, specify the correct IMSACB and NEWACB DD statement pair or OLDACB and IMSACB DD statement pair. Otherwise, specify SDEP=LOGICAL for the UNLOAD or the CHANGE command in HFPSYSIN.

HFPM0005I **SUBSEQUENT MESSAGES ARE SHOWN WITH THE *timetype* TIMESTAMP.**

Explanation

This message notifies that messages that are issued after this message include a timestamp.

System action

Processing continues.

User response

None. This message is informational.

HFPM0101I **MCS CONSOLE COMPLETED. (JOB : *driver_jobname*)**

Explanation

The MCS console task has completed.

System action

Processing continues.

User response

None. This message is informational.

HFPM3001E **UNABLE TO FIND *module_name* IN STEPLIB.**

Explanation

The program could not load the required module. The indicated module was not found in any library that is referenced by STEPLIB.

System action

The job ends with an abend code of U3001.

User response

Specify the library that includes the required load module in the STEPLIB concatenation, and rerun the job.

HFPM3002E **CONTROL CARD IS NOT FOUND IN *xxxxxxx***

Explanation

Control cards were not found in *xxxxxxx*.

System action

The job ends with an abend code of U3002.

User response

Rerun the job with a valid control card in the input stream.

HFPM3003E ***xxxxxxx* TASK ENDED ABNORMALLY.**

Explanation

This message is generated when the program main task detects a critical *xxxxxxx* process error, including *xxxxxxx* subtask abends.

System action

The job ends with an abend code of U3003.

User response

See the other message that was issued by the *xxxxxxx* task. Correct the errors, and rerun the job.

HFPM3004E **HPFP LIBRARY IS NOT APF-AUTHORIZED.**

Explanation

The FPA load module library data set that is specified on the JOBLIB/STEPLIB DD statement is not authorized by APF.

System action

The job ends with an abend code of U3004.

User response

Make the FPA load module library APF authorized for future processing, and rerun the job.

**HFP3005E [HPFP | OSC] DOES NOT SUPPORT
IMS VERSION: nn**

Explanation

The level of the IMS system is not supported by the indicated tool or utility.

System action

The job ends with an abend code of U3005.

User response

Rerun the job under the supported IMS level.

**HFP3006E OSC SUPPORTS IMS MANAGED
ACBS ONLY ON IMS V15.**

Explanation

The Online Structure Change utility supports IMS managed ACBs only on IMS 15 and later. If you are on IMS 14, use ACBLIB-managed ACBs.

System action

The job ends with an abend code of U3006.

User response

Rerun the job on IMS 15, or rerun the job using the ACBs in the ACB library.

**HFP3007E IMAGE COPY OUTPUT FUNCTION
CANNOT BE USED.
- HPIC MODULE: *modulename* IS
NOT FOUND
- HPIC VERSION CHECK MODULE
ENDED WITH ERROR: *nn*.
- THE PTF LEVEL OF THE HPIC
MODULE IS NOT HIGH ENOUGH.
- PRODUCT INFORMATION:
xxxxxxxx.**

Explanation

The image copy output process failed with an error.

System action

The job ends with an abend code of U3007.

User response

Make sure the load module library of IMS HP Image Copy is concatenated to the STEPLIB or JOBLIB and the PTF level is enough. If this situation persists, contact IBM Software Support.

**HFP3008E xxxxxxxx PARM LENGTH IS TOO
LONG.**

Explanation

The length of the xxxxxxxx parameter is too long.

System action

The job ends with an abend code of U3008.

User response

Correct the error by, for example, reducing the length of the parameter, and rerun the job.

**HFP3009I LEVEL IS NOT HIGH ENOUGH TO
SUPPORT OUTPUT REPOSITORY:
*product***

Explanation

The maintenance level of the product does not support the IMS Tools KB Output repository.

System action

Processing continues. The reports of FPA are stored to the IMS Tools KB repository, but the reports of the product are not stored to the IMS Tools KB Output repository.

User response

If you want to store the reports of the product to the IMS Tools KB Output repository, apply the maintenance of the product that supports the IMS Tools KB Output repository.

**HFP3016E SEVERE DEDB DEFINITION
DIFFERENCE DETECTED.**

Explanation

Detected one or more severe DEDB definition differences. SCAN=NO is specified in the GLOBAL command.

System action

The job ends with an abend code of U3016.

User response

Check the DBD Definition report and error messages to determine the cause of the problem. If error messages are issued, follow the user response of those messages. Correct the errors and rerun the job.

HFP3017E *parameter IS NOT VALID FOR THE PARM PARAMETER OF THE EXEC STATEMENT.*

Explanation

The parameter that is specified on the EXEC statement of HFPSTGEN job must be GEN or REPORT.

System action

The job ends with an abend code of U3017.

User response

Correct the parameter of the EXEC statement, and rerun the job.

HFP3061E *UNABLE TO LOAD RANDOMIZER ROUTINE: rmodname
- FROM ddname
- FROM ddname (REASON: NOT APF-AUTHORIZED)
- FROM dsname
- VOLUME: volumename DSN: datasetname
- RC: X'code', RSN: X'xxxxyyyy' FROM CSVAPF*

Explanation

The program cannot load the *rmodname* randomizer routine from *ddname* DD statement. *code* is the return code from CSVAPF macro. *xxxx* is internal information and *yyyy* is the reason code from CSVAPF macro.

System action

The job ends with an abend code of U3061.

User response

Make sure that the specified member *rmodname* exists in the *ddname* library. Look up the CSVAPF macro code in the *MVS Programming: Authorized Assembler Services Reference*. Correct the error, and rerun the job.

HFP3062E *UNABLE TO LOAD USER EXIT ROUTINE: exitname
- FROM ddname
- FROM ddname (REASON: NOT APF-AUTHORIZED)
- VOLUME: volumename DSN: datasetname
- RC: X'code', RSN: X'xxxxyyyy' FROM CSVAPF*

Explanation

The program cannot load the *exitname* user exit routine from *ddname* DD statement. *code* is the return code from CSVAPF macro. *xxxx* is internal information and *yyyy* is the reason code from CSVAPF macro.

System action

The job ends with an abend code of U3062.

User response

Make sure that the specified member *exitname* exists in the *ddname* library. Look up the CSVAPF macro code in the *MVS Programming: Authorized Assembler Services Reference*. Correct the error, and rerun the job.

HFP3070E *DATA SET: dsname IS INCORRECT.
- NO RECORD FOUND.
- INCORRECT RECORD FOUND.*

Explanation

This message consists of multiple parts. These parts are:

- NO RECORD FOUND.

dsname that is specified for the MODSTATx or OLCSTAT is empty.

- INCORRECT RECORD FOUND.

dsname that is specified for the MODSTATx or OLCSTAT is incorrect.

System action

The job ends with an abend code of U3070.

User response

Make sure that the DD statement properly identifies the correct data set for the MODSTATx or OLCSTAT. Correct any errors, and rerun the job.

HFP3360E *UNABLE TO LOAD IMS EDIT/ COMPRESSION ROUTINE: exitname
- FROM DD: ddname*

- FROM DD: *ddname* (REASON: NOT APF-AUTHORIZED)
 - VOLUME: *volumename* DSN: *datasetname*
 - RC: X'*code*', RSN: X'*xxxxyyyy*'
 FROM CSVAPF

Explanation

The program cannot load the *exitname* edit/compression routine from *ddname* DD statement. *code* is the return code from CSVAPF macro. *xxxx* is internal information and *yyyy* is the reason code from CSVAPF macro.

HFPO messages

The following information is about messages and codes that begin with HFPO.

HFPO0001I **DB SENSOR PROCESSING STARTED FOR DATABASE:**
dbdname*, AREA NO: *areanum*, AREANAME: *areaname

Explanation

The DB Sensor process for the indicated area has started.

System action

Processing continues.

User response

None. This message is informational.

HFPO1001I **DB SENSOR STORED SENSOR DATA FOR DATABASE: *dbdname*,**
AREA NO: *areanum*, AREANAME: *areaname*

Explanation

Sensor data for the indicated area was stored into the Sensor Data repository of IMS Tools KB.

System action

Processing continues

User response

None. This message is informational.

HFPO1002W **DB SENSOR DID NOT STORE SENSOR DATA FOR DATABASE: *dbdname*,**
AREA NO: *areanum*, AREANAME: *areaname*

System action

The job ends with an abend code of U3360.

User response

Make sure that the segment edit/compression routine or the user exit routine for the specified segment is in the *ddname* library. Or make sure that the DURDBDFN DD statement properly identifies the correct data set for the area to be reloaded. Look up the CSVAPF macro code in the *MVS Programming: Authorized Assembler Services Reference*. Correct any errors, and rerun the job.

Explanation

Sensor data for the indicated area was not stored into the Sensor Data repository of IMS Tools KB.

System action

Processing continues.

User response

See the preceding IMS Tools KB messages to determine the errors, and correct any errors.

If you want to store the sensor data in the IMS Tools KB Sensor Data repository, rerun the job or run a job that stores statistics data in the Sensor Data repository.

HFPO1003I **DB SENSOR SENT NOTIFICATION TO AD FOR DATABASE: *dbdname*,**
AREA NO: *areanum*, AREANAME: *areaname*

Explanation

Autonomics Director was notified that the sensor data had been stored into the ITKB Sensor Data repository.

System action

Processing continues.

User response

None. This message is informational.

HFPO1004W **DB SENSOR COULD NOT SEND NOTIFICATION TO AD FOR DATABASE: *dbdname*,**
AREA NO: *areanum*, AREANAME: *areaname*

Explanation

Autonomics Director is not notified that the sensor data records had been stored into the ITKB Sensor Data repository.

System action

Processing continues.

User response

See the preceding Autonomics Director messages to determine the errors, and correct any errors.

If you want a sensor data notification to be sent to Autonomics Director to trigger a policy evaluation, rerun the job or run a job that stores statistics data in the Sensor Data repository and that sends a sensor data notification to Autonomics Director.

HFPO1005I DB SENSOR ENDED NORMALLY.

Explanation

Stand-alone DB Sensor ended normally.

System action

Processing ends.

User response

None. This message is informational.

HFPO1006E DB SENSOR ENDED WITH ERRORS.

Explanation

Stand-alone DB Sensor ended with errors.

System action

The return code is set to 8 and processing ends.

User response

See the previous error messages, correct any errors, and rerun the job.

**HFPO1007W UNABLE TO OBTAIN RECON ID.
FUNC=function, RC=return_code,
RSN=reason_code.**

Explanation

DB Sensor failed to obtain a RECON ID.

System action

Processing continues, but IMS HP Fast Path Utilities does not store any sensor data in the Sensor Data repository of IMS Tools KB.

User response

Ensure that an appropriate RECON ID is registered in the IMS Tools KB repository.

If you want to store the sensor data in the IMS Tools KB Sensor Data repository, rerun the job or run a job that stores statistics data in the Sensor Data repository.

**HFPO1008W SENSOR DATA SERVICE FAILED.
FUNC=function, RC=return_code,
RSN=reason_code [,
ELEM=data_element xxxx/yyyy]**

Explanation

The sensor data service function call ended with errors. If the errors are detected in specific data elements, *data_element* shows the name of the data element, and hexadecimal values *xxxx* and *yyyy* show the return and reason codes of the error, respectively.

If this message shows FUNC=INIT, RC=4, and RSN=X'52', the message indicates that the Sensor Data repository is not initialized.

System action

Processing continues, but IMS HP Fast Path Utilities does not store any sensor data in the Sensor Data repository of IMS Tools KB.

User response

See the *Tools Base Policy Services User's Guide* to determine the meaning of the return code and reason code, and then correct any errors.

If FUNC=INIT, RC=4, and RSN=X'52' is shown in the message, initialize the Sensor Data repository by following the instructions in the topic "Initializing the Sensor Data and Input repositories" in the *Tools Base Configuration Guide for IMS*.

If you want to store the sensor data in the IMS Tools KB Sensor Data repository, rerun the job or run a job that stores statistics data in the Sensor Data repository.

**HFPO1009W DATA DICTIONARY SERVICE
FAILED. FUNC=function,
RC=return_code, RSN=reason_code**

Explanation

One or more errors were detected when the indicated function of the Data Dictionary Service was running. Hexadecimal values *return_code* and *reason_code* show the return and reason codes of the error, respectively.

System action

Processing continues, but IMS HP Fast Path Utilities does not store any sensor data in the Sensor Data repository of IMS Tools KB.

User response

Ensure that the correct load module library for Policy Services is specified in JCL. See the *Tools Base Policy Services User's Guide* to determine the meaning of the return code and reason code, and then correct any errors.

If you want to store the sensor data in the IMS Tools KB Sensor Data repository, rerun the job or run a job that stores statistics data in the Sensor Data repository.

If the problem persists, contact IBM Software Support.

HFPO1010W	PHYSICAL ERROR DETECTED IN A CI. DATABASE: <i>dbdname</i>, AREA NO: <i>areanum</i>, AREANAME: <i>areaname</i>
------------------	--

Explanation

DB Sensor detected a physical error in the indicated area.

System action

Processing continues, but IMS HP Fast Path Utilities does not store any sensor data of the indicated area in the Sensor Data repository of IMS Tools KB.

User response

Run OPC or FPA Analyze to identify the errors in the DEDB area and correct any errors.

If you want to store the sensor data in the IMS Tools KB Sensor Data repository, rerun the job or run a job that stores statistics data in the Sensor Data repository.

HFPO1011W	GETMAIN FAILURE OCCURRED (RC=<i>return_code</i>) FOR SENSOR DATA BUFFER. [DATABASE: <i>dbdname</i>, AREA NO: <i>areanum</i>, AREA: <i>areaname</i>]
------------------	--

Explanation

Program GETMAIN failed to obtain space for a sensor data buffer.

System action

Processing continues, but IMS HP Fast Path Utilities does not store any sensor data of the indicated area in the Sensor Data repository of IMS Tools KB.

User response

If the region size that is specified is too small, increase the region size on the JOB statement in the JCL.

If you want to store the sensor data in the IMS Tools KB Sensor Data repository, rerun the job or run a job that stores statistics data in the Sensor Data repository.

HFPO1012W	FAILED TO SEND SENSOR DATA NOTIFICATION TO AD. FUNC=<i>function</i>, RC=<i>return_code</i>, RSN=<i>reason_code</i>, DB: <i>dbdname</i>, AREA: <i>areaname</i>]
------------------	---

Explanation

The Autonomics Director service function call ended with errors.

System action

Processing continues, but IMS HP Fast Path Utilities does not send sensor data notification of the indicated area to Autonomics Director.

User response

See the *Tools Base Autonomics Director User's Guide* to determine the meaning of the return code and reason code, and correct any errors.

If you want to store the sensor data in the IMS Tools KB Sensor Data repository, rerun the job or run a job that stores statistics data in the Sensor Data repository.

HFPO1013W	LOAD FAILED. SC=<i>system_completion_code</i>, RSN=<i>reason_code</i>, MEMBER: <i>member_name</i>
------------------	--

Explanation

The indicated member could not be loaded.

System action

Processing continues, but IMS HP Fast Path Utilities does not store any sensor data in the Sensor Data repository of IMS Tools KB.

User response

Ensure that the correct load module library for IMS Tools Base is specified in JCL. Specify the correct load module library on the STEPLIB DD.

If you want to store the sensor data in the IMS Tools KB Sensor Data repository, rerun the job or run a job that stores statistics data in the Sensor Data repository.

HFPO1014W **DYNAMIC ALLOCATION FAILED FOR DDNAME: *ddname*, RC=*code*, RSN=*eeeeiiii***

Explanation

Dynamic allocation for the indicated ddname and dsname failed. *code* is the return code from SVC99. *eeee* is the S99 ERROR contents. *iiii* is the S99 INFO contents.

System action

Processing continues.

User response

Check the dynamic allocation (SVC99) code in the *MVS Programming: Assembler Services Reference* and correct any errors.

If you want to store the sensor data in the IMS Tools KB Sensor Data repository, rerun the job or run a job that stores statistics data in the Sensor Data repository.

HFPO1015W **OPEN FAILURE OCCURRED FOR DDNAME: *ddname***

Explanation

OPEN processing failed for the file that is associated with the indicated DD statement.

System action

Processing continues.

User response

Ensure that a DD statement is present for the indicated DD and that it is specified correctly. Correct any errors.

If you want to store the sensor data in the IMS Tools KB Sensor Data repository, rerun the job or run a job that stores statistics data in the Sensor Data repository.

HFPO1016W **DB SENSOR ENDED WITH WARNINGS.**

Explanation

Stand-alone DB sensor ended with warnings.

System action

The return code is set to 4 and processing ends.

User response

See the previous warning messages, correct any errors, and rerun the job.

HFPO1017W **FAILED TO SEND SENSOR DATA NOTIFICATION TO AD. MASTER AD ADDRESS SPACE IS NOT ACTIVE. DB: *dbdname*, AREA: *areaname***

Explanation

The Autonomics Director service function call ended with errors because the master AD address space is not active.

System action

Processing continues, but IMS HP Fast Path Utilities does not send sensor data notification of the indicated area to Autonomics Director.

User response

See the *Tools Base Autonomics Director User's Guide* to determine the meaning of the return code and reason code, and correct any errors.

If you want to store the sensor data in the IMS Tools KB Sensor Data repository, rerun the job or run a job that stores statistics data in the Sensor Data repository.

HFPO1018W **FAILED TO SEND SENSOR DATA NOTIFICATION TO AD. DBRC=YES AND RECON1 DD MUST BE SPECIFIED.**

Explanation

DBRC=YES and RECON1 DD must be specified if you want to send sensor data notification to Autonomics Director.

System action

Processing continues, but IMS HP Fast Path Utilities does not send sensor data notification to Autonomics Director.

User response

Specify DBRC=YES and RECON1 DD.

If you want to store the sensor data in the IMS Tools KB Sensor Data repository, rerun the job or run a job that stores statistics data in the Sensor Data repository.

HFPO1019I **OSM SENT NOTIFICATION TO AD FOR DATABASE: *dbname*, AREANAME: *areaname*. RESULT=*result*, STATUS=*status***

Explanation

The Online Space Management utility sent a result notification to Autonomics Director. The notification includes the result and the status of the Online Space Management utility job.

System action

Processing continues.

User response

If RESULT=FAILED is shown in the message, locate the processing messages and identify the error cause. Correct the error, and then rerun the job or run an alternate job that runs the failed process.

HFPO1020W **NOTIFICATION IS NOT SENT TO AD FOR DATABASE: *dbname*, AREANAME: *areaname*, JNAME: *jobname*, JNUM: *jobnumber***

Explanation

Because of a preceding error, the Online Space Management utility did not send the result notification to Autonomics Director.

System action

Processing continues. The Online Space Management utility does not send a result notification to Autonomics Director.

User response

Locate the processing messages and identify the error cause. Then, correct the error and rerun the job or, if necessary, run an alternate job.

HFPO1021W **FAILED TO SEND NOTIFICATION TO AD FOR DATABASE: *dbname*, AREANAME: *areaname*, JNAME: *jobname*, JNUM: *jobnumber***

Explanation

The Online Space Management utility issued a NOTIFY call to Autonomics Director, but Autonomics Director returned a non-zero return code. The result notification did not reach Autonomics Director.

System action

Processing continues.

User response

Locate the processing messages and identify the error cause. Then, correct the error and rerun the job or, if necessary, run an alternate job.

HFPO1022I **FOR AREA *areaname* IN DATABASE *dbname*, OSM PERFORMED *actions***

Explanation

This message shows the actions that the Online Space Management utility performed for the indicated area. The actions were selected based on the parameters specified for the ACTION keyword and the REORGATIOVFEXT keyword.

If multiple actions were selected and an action ends abnormally, subsequent actions are skipped. Skipped actions are not shown in this message.

System action

Processing continues.

User response

This message is informational. If EXTENDIOVF is shown in the message, you must plan IMS online change for each member in the ACB library.

HFPO1024I **INFORMATION ON THE REPOSITORY MEMBER INDEX OF THE STORED UTILITY HISTORY DATA:
- EVENT..... REORG
- PROGRAM_ID... *program_id***

- **PRODUCT_ID...** DF
- **JOB NAME.....** *job_name*
- **RECON_ID.....** *recon_id*

Explanation

This message shows the registration information for the repository member index of the stored utility history data.

System action

Processing continues.

User response

None. This message is informational.

HFPO1025I **UTILITY HISTORY DATA STORED FOR DATABASE: *dbdname*, AREA NO: *nn*, AREANAME: *areaname***

Explanation

Utility history data for the indicated area was stored into the Sensor Data repository of IMS Tools KB.

System action

Processing continues.

User response

None. This message is informational.

HFPO1026I **UTILITY HISTORY DATA NOT STORED FOR DATABASE: *dbdname*, AREA NO: *nn*, AREANAME: *areaname***

Explanation

Utility history data for the indicated area was not stored into the Sensor Data repository of IMS Tools KB.

System action

Processing continues.

User response

See the preceding IMS Tools KB messages to determine the errors, and correct any errors.

HFPO1027I **TIMESTAMP: *timestamp***

Explanation

This message is accompanied with message HFPO1026I. *timestamp* shows the reorganization timestamp that was supposed to be stored into the Sensor Data repository as utility history data.

System action

Processing continues.

User response

None. This message is informational.

HFPO1028E **AN ERROR OCCURRED IN HKTEXTST CALL: FUNCTION=*function* FOR DATABASE: *dbdname*, AREA NO: *nn*, AREANAME: *areaname* HKTxnnnE - LOCID=*aaa* RC=*bb* RSN=*cc* R0=*ddddddd* : *errtext***

Explanation

An error occurred during the internal API call. The utility history member could not be stored. *function* shows the function code.

An HKTxnnnE message accompanies this message. The variables shown in the HKTxnnnE message (*aaa*, *bb*, *cc*, *ddddddd*, and *errtext*) might be required when you contact IBM Software Support.

System action

Processing continues, but the utility history member is not stored.

User response

See the explanation of the HKTxnnnE message in the *Tools Base IMS Tools Knowledge Base User's Guide*.

HFPO1029E **AN ERROR OCCURRED IN HKTEXTMO CALL: FUNCTION=*function*, RC=*rc***

Explanation

An error occurred during the internal API call. Initialization of the module that collects the utility history data failed. *function* shows the function code. *rc* shows the return code from the internal API.

System action

Processing continues, but the utility history member is not stored.

User response

Contact IBM Software Support.

HFPO1030I **UTILITY HISTORY DATA WILL NOT
BE STORED IN THE SENSOR DATA
REPOSITORY.**

Explanation

This message is accompanied with message HFPO1028E or HFPO1029E. Utility history data is not stored in the sensor data repository due to an error.

System action

Processing continues.

User response

See the accompanied HFPO1028E or HFPO1029E message.

HFPO1101I **SENSOR DATA READING STARTED
FOR DATABASE *dbdname*, AREA
*areaname***

Explanation

The DB Sensor Printing utility started to read the sensor data from the Sensor Data repository.

System action

Processing continues.

User response

None. This message is informational.

HFPO1102I **SENSOR DATA READING ENDED
FOR DATABASE *dbdname*, AREA
areaname,
RSI=*record_set_identifier***

Explanation

The DB Sensor Printing utility completed reading the sensor data from the Sensor Data repository. RSI shows the identifier of the sensor data.

System action

Processing continues.

User response

None. This message is informational.

HFPO1103I **SENSOR DATA STATISTICS
REPORT GENERATION STARTED
FOR DATABASE *dbdname*, AREA
*areaname***

Explanation

The DB Sensor Printing utility started to generate the Sensor Data Statistics report.

System action

Processing continues.

User response

None. This message is informational.

HFPO1104I **SENSOR DATA STATISTICS
REPORT GENERATION ENDED FOR
DATABASE *dbdname*, AREA
*areaname***

Explanation

The DB Sensor Printing utility generated the Sensor Data Statistics report.

System action

Processing continues.

User response

None. This message is informational.

HFPO1105I **SENSOR DATA PROCESSED IS
IDENTIFIED BY LOCALE=RECON,
RECON ID: *recon_id*, DSN: *dsname***

Explanation

The DB Sensor Printing utility processes the sensor data that was identified by the RECON locale that has the indicated RECON ID and RECON data set names.

System action

Processing continues.

User response

None. This message is informational.

HFPO1106I **SENSOR DATA PROCESSED IS
IDENTIFIED BY LOCALE=DEFAULT,
RECON ID: NORECON**

Explanation

The DB Sensor Printing utility processes the sensor data that was identified by the default locale that has the indicated RECON ID (NORECON). NORECON is the default locale.

System action

Processing continues.

User response

None. This message is informational.

HFPO1117E *command* **COMMAND MUST NOT BE SPECIFIED MORE THAN ONCE**

Explanation

The indicated command was specified more than once in the HFPSPRIN data set.

System action

The DB Sensor Printing utility ends with return code 8.

User response

Remove the duplicate command and rerun the job.

HFPO1119E **THE GLOBAL COMMAND MUST BE SPECIFIED BEFORE THE *command* COMMAND**

Explanation

The GLOBAL command must be specified before the indicated command.

System action

The DB Sensor Printing utility ends with return code 8.

User response

Change the order of the commands and rerun the job.

HFPO1120E **THE *command* COMMAND MUST BE SPECIFIED**

Explanation

The indicated command is a required command. However, the command is not specified in the control statement.

System action

DB Sensor ends with return code 8.

User response

Specify the correct command in the control statement and rerun the job.

HFPO1121E **STATEMENT CONTAINS A SYNTAX ERROR**

Explanation

Syntax errors were found in the control statements that are specified in the HFPSPRIN data set.

System action

The DB Sensor Printing utility ends with return code 8.

User response

Correct the syntax errors and rerun the job.

HFPO1122E *command* **COMMAND IS NOT CORRECT**

Explanation

The specified command is not a GLOBAL or a DATABASE statement.

System action

The DB Sensor Printing utility ends with return code 8.

User response

Specify the GLOBAL statement or the DATABASE statement, and rerun the job.

HFPO1123E *keyword* **KEYWORD CANNOT BE SPECIFIED IN THE *command* COMMAND**

Explanation

The indicated keyword cannot be specified in the indicated command.

System action

The DB Sensor Printing utility ends with return code 8.

User response

Specify the correct keyword and rerun the job.

HFPO1124E **THE NUMBER OF *keyword* KEYWORDS EXCEEDED THE LIMIT. MAX IS *number*.**

Explanation

The indicated keyword is specified more than the maximum allowable number of times.

System action

The DB Sensor Printing utility ends with return code 8.

User response

Reduce the number of keywords and rerun the job.

HFPO1125E	<i>keyword</i> KEYWORD MUST BE SPECIFIED FOR THE <i>command</i> COMMAND
------------------	--

Explanation

The indicated keyword is a required keyword for the indicated command; however, the keyword was not specified.

System action

The DB Sensor Printing utility ends with return code 8.

User response

Specify the required keyword and rerun the job.

HFPO1126E	THE NUMBER OF PARAMETERS SPECIFIED IN <i>keyword</i> KEYWORD EXCEEDED THE LIMIT. MAX IS <i>number</i>.
------------------	---

Explanation

The number of parameters that are specified on the indicated keyword is more than the maximum allowable number of times.

System action

The DB Sensor Printing utility ends with return code 8.

User response

Correct the parameters and rerun the job.

HFPO1127E	THE <i>n</i>TH PARAMETER ON THE <i>keyword</i> KEYWORD HAS INCORRECT LENGTH
------------------	--

Explanation

The length of the indicated parameter that is specified on the indicated keyword is incorrect.

System action

The DB Sensor Printing utility ends with return code 8.

User response

Correct the length of the parameter and rerun the job.

HFPO1169E	LOAD FAILED FOR <i>modname</i> MODULE. CODE=<i>nnn</i>, RSN=<i>rsn</i>
------------------	---

Explanation

The DB Sensor Printing utility failed to load the indicated module. The hexadecimal value *nnn* shows the system completion code, and the hexadecimal value *rsn* shows the associated reason code that are returned from the LOAD macro.

System action

The DB Sensor Printing utility ends with return code 8.

User response

See the *MVS Programming: Assembler Services Reference* to determine the meaning of the system completion code. Then, correct the error and rerun the job. If the problem persists, contact IBM Software Support.

HFPO1170E	GETMAIN FAILED WITH RC=<i>return_code</i> (SIZE=<i>size</i>) IN <i>modname</i> MODULE, ERROR-ID: <i>error_id</i>
------------------	---

Explanation

The DB Sensor Printing utility failed to obtain storage. *size* shows the size of the storage that could not be obtained. The hexadecimal value *return_code* shows the return code that is returned from the GETMAIN macro. *modname* shows the name of the failed module, and *error_id* shows the error ID that is associated with the module.

System action

The DB Sensor Printing utility ends with a return code of 8.

User response

See the *MVS Programming: Assembler Services Reference* to determine the meaning of the return code. Then, correct the error and rerun the job. If the problem persists, contact IBM Software Support.

HFPO1171E **OPEN FAILED. DD: *ddname*,
RC=*return_code***

Explanation

Open failed for the indicated DD.

System action

The DB Sensor Printing utility ends with return code 8.

User response

Browse the DB Sensor Printing utility JCL and ensure that the indicated DD is specified correctly. Then, correct the error and rerun the job.

HFPO1172E **DEVTYPE FAILED. RC=*return_code*,
RSN=*reason_code*, DD: *ddname***

Explanation

The DB Sensor Printing utility failed to issue the DEVTYPE macro for the indicated DD. The hexadecimal value *return_code* shows the return code and the hexadecimal value *reason_code* shows the associated reason code that are returned from the DEVTYPE macro.

System action

The DB Sensor Printing utility ends with return code 8.

User response

See *DFSMSdfp Advanced Services* to determine the meaning of the return code and the reason code. Correct the error and rerun the job. If the problem persists, contact IBM Software Support.

HFPO1173E **DYNAMIC ALLOCATION FAILED.
RC=*return_code*,
RSN=*reason_code*, DD: *ddname***

Explanation

The DB Sensor Printing utility failed to request the dynamic allocation service for the indicated DD. The hexadecimal value *return_code* shows the return code, and the hexadecimal value *reason_code* shows the associated reason code that are returned from the dynamic allocation service.

System action

The DB Sensor Printing utility ends with return code 8.

User response

See the *z/OS MVS Authorized Assembler Services Guide* to determine the meaning of the return code and the reason code. Correct the error and rerun the job. If the problem persists, contact IBM Software Support.

HFPO1180E **ERRORS WERE DETECTED WHILE
*description***

Explanation

The DB Sensor Printing utility detected errors and stopped processing. *description* shows the processing that was performed when the error was found. *description* can be one of the following strings:

- ANALYZING THE CONTROL STATEMENT
- READING THE SENSOR DATA
- GENERATING THE SENSOR DATA STATISTICS REPORT

System action

The DB Sensor Printing utility ends with return code 8.

User response

Determine the cause of the errors by investigating the HFPSOUT data set, correct the error, and rerun the job.

HFPO1181E **UNABLE TO CONNECT TO THE
ITKB REPOSITORY**

Explanation

The DB Sensor Printing utility could not communicate with the IMS Tools KB server.

System action

The DB Sensor Printing utility ends with return code 8.

User response

The probable cause of this error is that one of the following events occurred:

- The IMS Tools KB server name that was specified is incorrect.
- The IMS Tools KB server ended while the DB Sensor Printing utility was running.

Correct the error and rerun the job.

HFPO1182E **THE SPECIFIED SENSOR DATA
WAS NOT FOUND IN THE
REPOSITORY**

Explanation

The sensor data that is associated with the specified locale was not found in the IMS Tools KB Sensor Data repository.

System action

The DB Sensor Printing utility ends with return code 8.

User response

Ensure that the database name, partition name, and RECON data set name are correctly specified in the HFPSPRIN data set. Correct the error and rerun the job.

HFPO1183E THE SENSOR DATA SERVICE FAILED. RC=return_code, RSN=reason_code, FUNC=function

Explanation

The sensor data service function call ended with errors. *function* shows the function name that is used internally by sensor data service. The hexadecimal value *return_code* shows the return code, and the hexadecimal value *reason_code* shows the reason code from the sensor data service.

System action

The DB Sensor Printing utility ends with return code 8.

User response

See the *Tools Base Policy Services User's Guide* to determine the meaning of the return code and the reason code. If you receive other HFPO messages, check the meaning of those messages and take appropriate actions that are described in the user response section of the messages.

HFPO1184E THE DATA DICTIONARY SERVICE DETECTED ERRORS. FUNC=function, RC=return_code, RSN=reason_code

Explanation

One or more errors were detected when the indicated function of the Data Dictionary Service was running. Hexadecimal values *return_code* and *reason_code* show the return and reason codes from the Data Dictionary Service, respectively. If the errors were detected in specific data elements, HFPO1185E messages follow this message.

System action

When the indicated function is FORM, the DB Sensor Printing utility ends with return code 4. In other cases, the DB Sensor Printing utility ends with return code 8.

User response

Specify the correct load module library for Policy Services in your JCL and rerun the job. If the problem persists, contact IBM Software Support.

HFPO1185E - DATA ELEMENT: data_element, RC=return_code, RSN=reason_code

Explanation

An error was detected while processing the indicated data element. Hexadecimal values *return_code* and *reason_code* show the return and reason codes, respectively. The preceding message, HFPO1184E, indicates the function of Data Dictionary that was in process when this error occurred.

System action

See the system action section for HFPO1184E message.

User response

Specify the correct load module library for Policy Services in your JCL and rerun the job. If the problem persists, contact IBM Software Support.

HFPO1186E THE RECON ID THAT CORRESPONDS TO [RECON1 DD | NORECON] COULD NOT BE OBTAINED

Explanation

The DB Sensor Printing utility failed to retrieve the RECON ID from the IMS Tools KB Input repository.

System action

The DB Sensor Printing utility ends with return code 8.

User response

Ensure that the correct RECON data set name is specified on the RECON1 DD statement in the DB Sensor Printing utility JCL. Also ensure that the RECON data set name is registered in the IMS Tools KB. If the problem persists, contact IBM Software Support.

HFPO1187E AN INCORRECT VALUE IS SPECIFIED FOR THE *keyword* KEYWORD

Explanation

The DB Sensor Printing utility found an incorrect parameter on the indicated keyword.

System action

The DB Sensor Printing utility ends with return code 8.

User response

Specify the correct parameter value for the keyword, and rerun the job.

HFPO1524E "ARDIGIT=" VALUE IS INCORRECT

Explanation

DB Sensor detected an incorrect ARDIGIT= specification while parsing the control statement that you specified in the HFPAPARM DD.

System action

The return code is set to 8 and processing ends.

User response

Correct the control statement, and rerun the job.

HFPO1525E "RGROUP=" VALUE IS INCORRECT

Explanation

DB Sensor detected an incorrect RGROUP= specification while parsing the control statement that you specified in the HFPAPARM DD.

System action

The return code is set to 8 and processing ends.

User response

Correct the control statement, and rerun the job.

HFPO1526I SENSOR DATA IS READ FROM ITKB SERVER *servername*. DATA ELEMENT: *element-name*, VALUE: *value*

Explanation

The value of the indicated data element is read from the IMS Tools KB server.

System action

Processing continues.

User response

None. This message is informational.

HFPO1527I SENSOR DATA IS READ FROM ITKB SERVER *servername*. THE VALUE OF DATA ELEMENT *element-name* IS "N/A".

Explanation

The sensor data for the indicated data element is not stored in the IMS Tools KB server.

System action

Processing continues. The default value is applied to the indicated data element.

User response

If you want to have the value for the indicated data element read from the Sensor Data repository of the IMS Tools KB server, run the integrated DB Sensor function or Stand-alone DB Sensor to store the sensor data.

HFPO1532E [HFPAUOWC | FPXAUOWC] FILE ERRORS. CARD: "xxxxxxxxxxxxxxxxxxxxxxxxxxxx"

Explanation

One or more errors in one of the HFPAUOWC or FPXAUOWC control cards are detected.

System action

The return code is set to 8 and processing ends.

User response

Correct the HFPAUOWC or FPXAUOWC control card, and rerun the job.

HFPO1563E CRITICAL CONTROL CARD ERROR ENCOUNTERED

Explanation

A critical control card error encountered.

System action

The return code is set to 8 and processing ends.

User response

See the other messages generated for further information about the error. Correct the control statement. Rerun the job.

HFPO3400E DB SENSOR SUB TASK ENDED ABNORMALLY.

Explanation

The DB Sensor subtask ended abnormally.

System action

The job ends with an abend code of U3003 or U3006.

User response

Check the preceding message and determine the cause of the DB Sensor subtask error. Correct any errors and rerun the job.

HFPO3401E SORT ENDED ABNORMALLY. (RC: code, TCT: tctname, MSGDDN: msg_ddname)

HFPP messages

The following information is about messages and codes that begin with HFPP.

HFPP0001I [CHECKPOINT | RESTART] PROCESS STARTED.
- THIS PROCESS IS THE [1ST | 2ND | 3RD | nnTH] PROCESS OF RESTART.

Explanation

This informational message indicates whether the job is running in checkpoint mode or restart mode. If the job is running in restart mode, the count for restart processing is also shown.

System action

Processing continues.

User response

None. This message is informational.

HFPP0002I [CHECKPOINT | RESTART] PROCESS COMPLETED.
- THE CHECKPOINT FILE IS IN CHECKPOINT MODE.
- THE CHECKPOINT FILE IS IN RESTART MODE.

Explanation

The SORT program that was internally invoked ended with an error. *code* is the return code from the SORT program.

System action

The job ends with an abend code of U3003 or U3006.

User response

Check the message and return code from the SORT program. Correct the problem and rerun the job. If the SORT program issued message ICE046A, specify the SORTOPT_FILSZ=AUTO option or DD statements for the intermediate storage data sets for the SORT program and rerun the job. For more information about intermediate storage data sets for the SORT program, see the DD statements topic of the function that you are using.

Explanation

This informational message indicates whether the completed process was run in checkpoint mode or restart mode, and whether the checkpoint file is in checkpoint mode or in restart mode.

- THE CHECKPOINT FILE IS IN CHECKPOINT MODE.

If you rerun the job with this checkpoint file, the job is executed in checkpoint mode.

- THE CHECKPOINT FILE IS IN RESTART MODE.

If you rerun the job with this checkpoint file, the job is executed in restart mode.

System action

Processing continues.

User response

None. This message is informational.

HFPP0004E PROCESS FAILED FOR THE DATA SET THAT IS SPECIFIED BY HFPCHKPT DD.

Explanation

Processing failed for the data set that is specified by HFPCHKPT DD because of an I/O error. The restart

processing cannot be executed with this checkpoint file.

System action

The job ends with an abend code of U3003. The contents of the checkpoint file are cleared.

User response

Rerun the job. The job is executed in checkpoint mode.

HFP0005E THE SPECIFICATION IN JCL IS DIFFERENT FROM THE INFORMATION IN THE CHECKPOINT FILE.

- *keyword* **KEYWORD VALUE IS DIFFERENT**
- *keyword* **KEYWORD VALUE IN subcommand SUBCOMMAND (nn) IS DIFFERENT**
- **NUMBER OF AREAS TO BE PROCESSED IS DIFFERENT**
- **UNLOAD MODE IS DIFFERENT**
- **LENGTH OF CONTROL BLOCK IS DIFFERENT**
- **RECORD NUMBER:nn**
- **NUMBER OF subcommand SUBCOMMANDS TO BE PROCESSED IS DIFFERENT**

Explanation

The JCL specification has changed between the time of the checkpoint and the restart. An entry was either deleted, added, or updated. The JCL specification must be consistent to use the checkpoint restart capability.

- *keyword* KEYWORD VALUE IS DIFFERENT

The value for the indicated keyword has been changed since the previous run.

- *keyword* KEYWORD VALUE IN subcommand SUBCOMMAND (nn) IS DIFFERENT

The value for the indicated keyword on the indicated subcommand (nn) has been changed since the previous run.

- NUMBER OF AREAS TO BE PROCESSED IS DIFFERENT

The number of areas to be processed has been changed since the previous run.

- UNLOAD MODE IS DIFFERENT

The mode for unload processing, which is either REORG mode or non-REORG mode, has been changed since the previous run.

- LENGTH OF CONTROL BLOCK IS DIFFERENT

- RECORD NUMBER:nn

The length of the control block that is on the indicated record number has been changed since the previous run.

- NUMBER OF subcommand SUBCOMMANDS TO BE PROCESSED IS DIFFERENT

The number of the indicated subcommand to be processed has been changed since the previous run.

System action

The job ends with an abend code of U3003.

User response

Ensure that the JCL specification remains consistent across checkpoint restart invocations. Correct the JCL specification and rerun the job.

HFP0006E INCORRECT RECORD WAS DETECTED IN THE CHECKPOINT FILE.

- **RECORD NUMBER: nnnnnnnnnn**
- **RECORD WAS INTERRUPTED BY EOF.**

Explanation

An incorrect record was detected in the checkpoint file. The checkpoint file might be corrupted. This checkpoint file cannot be used for restart processing.

System action

The job ends with an abend code of U3003. The contents of the checkpoint file are cleared.

User response

Rerun the same job. The job is executed in checkpoint mode.

HFP0007W DUMMY OR NULLFILE IS SPECIFIED ON THE HFPCHKPT DD STATEMENT. THE CHECKPOINT RESTART FUNCTION IS DISABLED.

Explanation

The JCL specifies DUMMY or NULLFILE for the checkpoint file. HFPCHKPT DD must be specified as a data set when RESTART=YES is specified.

System action

The return code is set to 4 and processing continues.

User response

Correct the specification of HFPCHKPT DD and, if necessary, rerun the job.

HFP0008E THE CHECKPOINT FILE IS NOT DEFINED CORRECTLY.

Explanation

The checkpoint file has an invalid definition. Ensure that the checkpoint file is not defined as a SYSIN, SYSOUT, terminal, or UNIX system file type, and that the file is allocated with the following attributes:

- DISP=MOD is not specified.
- RECFM=VB is specified.
- LRECL is larger than or equal to 22000.

System action

The job ends with an abend code of U3003.

User response

Correct the specification of HFPCHKPT DD and rerun the job.

HFP0009E A NOTIFY.IC COMMAND WAS GENERATED FOR AN AREA WHICH DOES NOT EXIST. AREANAME: *areaname*

Explanation

A NOTIFY.IC command is generated for an area that does not exist or that is not defined in the DBD definition.

System action

The return code is set to 8 and processing continues.

User response

This error is an internal system error. Contact IBM Software Support.

HFPQ messages

The following information is about messages and codes that begin with HFPQ.

HFPQ0001E DL/I #FPU *xxxx* call error, status = *yy*, DATABASE: *databasename* AREANO:#*n* AREA: *areaname*

HFP0010W IN THE NEXT RESTART PROCESSING, THE CHECKPOINT FILE WILL BE CLEARED AND NO RESTART WILL BE PROCESSED. - THE RESTART COUNT HAS REACHED THE LIMIT.

Explanation

The number of restart processing runs has reached the limit that is specified on the RESTART_COUNT keyword. The next time you submit the job, the contents of the checkpoint file are cleared, and the areas are not reprocessed.

System action

Processing continues.

User response

If you want to continue running the job in restart mode, specify a larger number for the RESTART_COUNT keyword.

HFP0011I THE RESTART COUNT HAS EXCEEDED THE LIMIT. RESTART WILL NOT BE PROCESSED ANY MORE.

Explanation

The number of the restart processing runs has exceeded the limit that is specified on the RESTART_COUNT keyword.

System action

The function clears the contents of the checkpoint file.

User response

None. This message is informational.

Explanation

An unexpected IMS status code *yy* was returned during a *xxxx* DL/I call.

System action

The job ends with a return code of 8.

User response

See *IMS Messages and Codes* for the appropriate response to these return codes. Then resubmit the job.

HFPQ0002W **CI DUMP CANNOT BE GENERATED FOR DATABASE: *databasename* AREA NO: *nnnn* AREaname: *areaname***

Explanation

CI dump cannot be generated because of an area connection failure.

System action

The job ends with a return code of 4.

User response

Correct the problem, then rerun the job.

HFPQ0003I **DDNAME: *ddname* FOR TARSDNMASK IS SPECIFIED ON THE JCL DD.**

Explanation

A TARSDNMASK keyword is coded in FPXIN to specify the data set for the tuning aid record data set, but the data set is also coded on the JCL DD statement. The tuning aid record for the area is not generated.

System action

Processing continues.

User response

If you want to generate the tuning aid record, either remove the indicated DD from the JCL stream or remove the TARSDNMASK keyword from FPXIN, and rerun the job.

HFPQ0004I **TARECGEN=YES IS SPECIFIED, BUT *ddname* DD STATEMENT IS NOT FOUND.**

HFPR messages

The following information is about messages and codes that begin with HFPR.

HFPR3042E **CONTROL TABLE OBTAIN FAILURE FOR: *table* TCT: *tctname* MODULE: *modname***

Explanation

The program tried to get storage for *table* but failed.

Explanation

TARECGEN=YES is specified on the ANALYZE or PTRCHK statement. But the DD statement for generating the tuning aid record for the area is not specified in the JCL. The tuning aid record for the area is not generated.

System action

Processing continues.

User response

If you want to generate the tuning aid record, specify the correct TAR*nnnnn* DD statement or specify the TARSDNMASK keyword, and rerun the job.

HFPQ0005I **TARECGEN=YES IS SPECIFIED, BUT TAR*nnnnn* DATA SET IS UNAVAILABLE.**
- DYNALLOC FAILED
- STORAGE GETMAIN FAILED
- FILE OPEN ERROR

Explanation

TARECGEN=YES is specified on the ANALYZE or PTRCHK statement, but the data set that is specified on the TAR*nnnnn* DD statement cannot be used. The tuning aid record for the area is not generated.

System action

Processing continues.

User response

If you want to generate the tuning aid record, specify the correct data set on the TAR*nnnnn* DD statement and rerun the job.

System action

The job ends with an abend code of U3003 or U3006.

User response

Increase the value of the region size parameter on the EXEC statement, and rerun the job.

**HFPR3043E CONTROL TABLE RELEASE
FAILURE FOR: *table* TCT: *tctname*
MODULE: *modname***

Explanation

The program tried to release a control table for *table* but failed.

System action

The job ends with an abend code of U3003 or U3043.

User response

If this situation persists, contact IBM Software Support.

**HFPR3044E BUFFER POOL CREATE FAILURE
FOR: *table* TCT: *tctname* MODULE:
*modname***

Explanation

The program tried to get a buffer pool for *table* but failed.

System action

The job ends with an abend code of U3003.

User response

Increase the value of the region size parameter on the EXEC statement, and rerun the job.

**HFPR3045E BUFFER POOL RELEASE FAILURE
FOR: *table* TCT: *tctname* MODULE:
*modname***

Explanation

The program tried to release a buffer pool but failed.

System action

The job ends with an abend code of U3003.

User response

If this situation persists, contact IBM Software Support.

**HFPR3046E STORAGE OBTAIN FAILURE ON
location FOR: *table* TCT: *tctname*
MODULE: *modname***

Explanation

The program tried to get storage *location* for *table* but failed.

System action

The job ends with an abend code of U3003.

User response

If *location* is data space, check the installation exit IEFUSI. Increase the size of data space. If *location* is address space, increase the value of the region size parameter on the EXEC statement, and rerun the job.

**HFPR3047E STORAGE RELEASE FAILURE ON
location FOR: *table* TCT: *tctname*
MODULE: *modname***

Explanation

The program tried to release storage *location* for *table* but failed.

System action

The job ends with an abend code of U3003.

User response

If this situation persists, contact IBM Software Support.

**HFPR3048E PAGE FIX FAILURE ON *location*
FOR: *table* TCT: *tctname* MODULE:
*modname***

Explanation

The program tried to page fix storage *location* for *table* but failed.

System action

The job ends with an abend code of U3003.

User response

If *location* is data space, increase size of the extended storage, and rerun the job. If *location* is address space, increase the size of the central storage, and rerun the job.

**HFPR3049E PAGE FREE FAILURE ON *location*
FOR: *table* TCT: *tctname* MODULE:
*modname***

Explanation

The program tried to free page storage *location* for *table* but failed.

System action

The job ends with an abend code of U3003.

HFPS messages

The following information is about messages and codes that begin with HFPS.

HFPS0001E DATASET AT DDNAME *ddname* IS NOT FOR AREA *areaname*
- MISMATCH VALIDATING FIELD:
XXXXXXXXXX
- VALUE IN DMB XXXXXXXX:
XXXXXXXXXX
- VALUE IN DMAC: XXXXXXXX

Explanation

The program issued an OPEN macro for the VSAM that is associated with the *ddname* that is specified. A comparison of certain key values that were extracted from the DMB from data sets OLDACB, IMSACB, IMSACBA, or IMSACBB with the contents of the second CI in the VSAM showed that the VSAM was not the database that was described by the DMB.

System action

The program ends with an abend code of 3003. If this error occurs for multiple area data sets (MADS) but not for all the area data sets of the MADS, the program ends with a return code of 8.

User response

Verify that the DD statement that was specified is correct and that the OLDACB DD statement specifies the correct data set. Correct the errors, and rerun the job.

HFPS0002I AREANAME: AREANAME:
areaname*, AREA NO: *nnnnn
- CANNOT BE PROCESSED DUE TO X'80' SET IN EQE LIST.
- DETECT ERROR IN EQE LIST
- NUMBER OF EQE: *nn*

Explanation

This message consists of two parts. The two parts are:

User response

If this situation persists, contact IBM Software Support.

- CANNOT BE PROCESSED DUE TO X'80' SET IN EQE LIST.

Found X'80' in a control interval (CI) that was extracted from the Error Queue Element (EQE) List in the second CI of the area data set *areaname*.

- DETECT ERROR IN EQE LIST - NUMBER OF EQE: *nn*

Found one or more error control intervals (CIs) that were extracted from the Error Queue Element (EQE) List in the second CI of the area data set *areaname*.

System action

Processing continues.

User response

Run the Full Recovery Utility, and rerun the job.

HFPS0003E THE DBD INFORMATION IN THE IMSACB DATA SET IS DIFFERENT FROM THE DBD INFORMATION RECORDS IN THE USR FILE.

Explanation

Inconsistencies were detected between the DBD information in the library that is specified on the IMSACB DD statement and the DBD information in the USR files that were created during the Unload process.

System action

The job ends with a return code of 08.

User response

For more information about the detected inconsistency, see the DBD Definition reports. Ensure that the correct library is specified on the IMSACB DD statement or that correct USR file is specified. Correct any errors, and rerun the job.

HFPS0004I [DATABASE | AREA] LEVEL CHANGE IS REQUIRED.

Explanation

Due to the changes made to the database or the areas, database or area level change is required.

System action

Processing continues.

User response

None. This message is informational.

HFPS0005E NUMBER OF SECONDARY INDEXES HAS BEEN CHANGED.

Explanation

The number of secondary index databases is different between the old DBD description and the new DBD description.

System action

If SCAN=YES is specified for the GLOBAL command, see the system action for the accompanying message HFPM0002E. If SCAN=NO is specified for the GLOBAL command, see the system action for the accompanying message HFPM3016E.

User response

The Online Structure Change utility does not support changes to Fast Path secondary indexes. Remove any changes to the Fast Path secondary indexes from DBD descriptions and rerun the job.

HFPS0006E DETECTED DIFFERENCES IN THE DEFINITIONS. SECONDARY INDEX DATABASE: *database*, RSN: *rsn*

Explanation

Definitions of secondary index databases are different between the old DBD description and the new DBD description.

rsn provides information about the difference detected.

<i>rsn</i>	Explanation
1	The name of secondary index database is different.
2	The name of the user partition exit routine is different.
3	The PSELOPT parameter of the PCB statement is different.

<i>rsn</i>	Explanation
4	The name of maintenance exit routine is different.
5	The NAME parameter of the XDFLD statement is different.
6	The pointer segment name is different.
7	The pointer segment length is different.
8	The source segment name is different.
9	The source segment code is different.
10	The target segment name is different.
11	The target segment code is different.
12	The target segment level is different.
13	The following attributes are different: <ul style="list-style-type: none"> • NULLVAL parameter is specified • SUBSEQ field is specified • DDATA field is specified • the attribute of the segment
14	The NULLVAL parameter of the XDFLD statement is different.
15	The MULTISEG parameter of the LCHILD statement is different.
16	The total length of search fields is different.
17	The pointer segment key length is different.
18	The symbolic pointer length is different.
19	The /CK offset is different.
20	The SEARCH field information is different.
21	The SUBSEQ field information is different.
22	The DDATA field information is different.

System action

If SCAN=YES is specified for the GLOBAL command, see the system action for the accompanying message HFPM0002E. If SCAN=NO is specified for the GLOBAL command, see the system action for the accompanying message HFPM3016E.

User response

The Online Structure Change utility does not support changes to Fast Path secondary indexes. Remove any changes to the Fast Path secondary indexes from DBD descriptions and rerun the job.

HFPS3014E ACB MEMBER *acbmember* IS INCORRECT.

Explanation

An invalid value is set in DMB, or the program could not read the DMB correctly.

System action

The job ends with an abend code of U3003.

User response

Contact IBM Software Support.

HFPS3015E OBTAINING DMB MEMBER: *dbdname* FROM *location* FAILED
- "FIND" MACRO FAILED (RC:xx).
- DMB MEMBER IS NOT FOR DEDB.
- DMB MEMBER IS NOT FOUND.
- IMS LEVEL: *xx* OF DMB MEMBER IS DIFFERENT FROM IMS LEVEL: *xx* OF PROCESSING LIBRARY.
- DATABASE NAME IN DMB IS NOT EQUAL TO DMB MEMBER NAME.
- VALUE OF DATABASE NAME IN DMB: *xxxxxxxx*
- DMB MEMBER HAS NO DATA RECORD.
- "BLDL" MACRO FAILED (RC:xx).

Explanation

The program failed to obtain the indicated DMB member. *location* shows either ACBLIB DDNAME: *acb_ddname* or IMS DIRECTORY (DEFINITION=[CURRENT | PENDING]).

This message consists of multiple parts. These parts are:

- "FIND" MACRO FAILED (RC:xx).

The program issued a FIND macro to get the position to read the member that is associated with *dbdname*. The return code shows that the attempt was unsuccessful.

- DMB MEMBER IS NOT FOR DEDB.

The program obtained the DMB member that is associated with *dbdname* either from the data set that is associated with *acb-ddname* or from the IMS directory. The DMB member was not for DEDB.

- DMB MEMBER IS NOT FOUND.

The program tried to obtain the DMB member that is associated with *dbdname* either from the data set that is associated with *acb-ddname* or from the IMS directory. The DMB member was not found in the *ddname* data set.

- IMS LEVEL: *xx* OF DMB MEMBER IS DIFFERENT FROM IMS LEVEL: *xx* OF PROCESSING LIBRARY.

The program obtained the DMB member that is associated with *dbdname* either from the data set that is associated with *acb-ddname* or from the IMS directory. The DMB member was not of a processable IMS release level.

- DATABASE NAME IN DMB IS NOT EQUAL TO DMB MEMBER NAME.

- VALUE OF DATABASE NAME IN DMB: *xxxxxxxx*

The program obtained the DMB member that is associated with *dbdname* either from the data set that is associated with *acb-ddname* or from the IMS directory. The database name in DMB is not equal to the name of the DMB member.

- "BLDL" MACRO FAILED (RC:xx).

The program issued a BLDL macro to obtain information about the member's directory that is associated with *dbdname*. The return code shows that the attempt was unsuccessful.

- DMB MEMBER HAS NO DATA RECORD.

The program obtained the DMB member that is associated with *dbdname* either from the data set that is associated with *acb-ddname* or from the IMS directory. The DMB member has no data record.

System action

The program ends with an abend code of 3003.

User response

- "FIND" MACRO FAILED (RC:xx).

Correct any errors, and rerun the job. If this situation persists, contact IBM Software Support.

- DMB MEMBER IS NOT FOR DEDB.

Specify the correct DEDB member and rerun the job.

- DMB MEMBER IS NOT FOUND.

Ensure that the correct DBD name is specified. Ensure that the correct data set for the ACB library is specified. If you enabled the IMS management of ACBs, ensure that the correct bootstrap data set for the IMS directory is specified.

Correct any errors, and rerun the job.

- IMS LEVEL: xx OF DMB MEMBER IS DIFFERENT FROM IMS LEVEL: xx OF PROCESSING LIBRARY.

Ensure that the correct data set for the ACB library is specified. If you enabled the IMS management of ACBs, ensure that the correct bootstrap data set for the IMS directory is specified.

Correct any errors, and rerun the job.

- DATABASE NAME IN DMB IS NOT EQUAL TO DMB MEMBER NAME.

- VALUE OF DATABASE NAME IN DMB: xxxxxxxx

Ensure that the correct data set for the ACB library is specified. If you enabled the IMS management of ACBs, ensure that the correct bootstrap data set for the IMS directory is specified.

Correct any errors, and rerun the job.

- "BLDL" MACRO FAILED (RC:xx).

Correct any errors, and rerun the job. If this situation persists, contact IBM Software Support.

- DMB MEMBER HAS NO DATA RECORD.

Ensure that the correct data set for the ACB library is specified. If you enabled the IMS management of ACBs, ensure that the correct bootstrap data set for the IMS directory is specified.

Correct any errors, and rerun the job.

HFPS3027E THE LRECL VALUE OF THE USR FILE IS TOO SMALL.
- ddname LRECL: nnnnn
- LENGTH OF THE FIRST DBD INFORMATION RECORD: nnnnn

Explanation

ACBCHK=YES is specified. However, the specified LRECL for the USR file was smaller than the length of the first DBD information record.

System action

The job ends with an abend code of U3003.

User response

Correct the LRECL for the USR file, and rerun the job.

HFPS3028E INCORRECT DBD INFORMATION RECORD WAS DETECTED IN THE USR FILE (DDNAME: ddname).

- DBD INFORMATION RECORDS ARE NOT FOUND IN THE USR FILE

Explanation

ACBCHK=YES is specified. However, an incorrect DBD information record was detected in the USR file, or the DBD information records were not found in the USR file.

System action

The job ends with an abend code of U3003.

User response

Ensure that the correct USR file is specified. If you want to recreate the USR file, use the Unload function of FPA. Correct any errors, and rerun the job.

HFPS3151E IMS TOOLS CATALOG INTERFACE function FUNCTION (DEFINITION=[CURRENT | PENDING]) FAILED.
- RETURN CODE: rc, REASON CODE: rsn

Explanation

IMS Tools Catalog Interface ended with an error. *function* shows the function code of IMS Tools Catalog Interface. The return code and reason code from IMS Tools Catalog Interface are shown in *rc* and *rsn*, respectively.

System action

The job ends with an abend code of U3151.

User response

If the *function* is OPEN, check if the correct high-level qualifier of the bootstrap data set is specified in the IMSCATHLQ keyword. Otherwise, contact IBM Software Support.

HFPT messages

The following information is about messages and codes that begin with HFPT.

HFPT0001I EXIT ROUTINE: *exit-routine* "END"
CALL FINISHED.
text1
text2

Explanation

The thread task called the user exit routine *exit-routine* with the END call, and the exit routine returned the message.

System action

Processing continues.

User response

None. This message is informational.

HFPT0002W EXIT ROUTINE: *exit-routine*
RETURNED STATUS CODE: *nn* IN
AREA NO: *nn*, AREANAME:
areaname
- SEG-CD: *xx*
- RBA: *xxxxxxxx*
- RECORD NUMBER: *xxxxxxxx*
text1
text2

Explanation

The input thread task or the output thread task received the status code E1 or E2 from the user exit routine *exit-routine* that was specified by the EXITRTN= control statement.

System action

If a status code E1 was returned, the return code is set to 4, and processing continues. If a status code E2 was returned, the job ends with an abend code of U3003.

User response

Correct the problem, if necessary, and rerun the job.

HFPT0003W *xxxx* SEGMENT KEY SEQUENCE
ERROR IN AREA NO: *nn*,
AREANAME: *areaname*
- SEG-CD: *xx*
- RBA: *xxxxxxxx*
- RECORD NUMBER: *xxxxxxxx*

Explanation

A segment key sequence error was found when KEYSEQCHK=BYPASS was specified.

System action

The return code is set to 4 and processing continues.

User response

None.

HFPT0004E A SERIOUS ERROR WAS
DETECTED DURING THE [HASH
CHECK | AREA SENSOR] PROCESS
(AREANO. *nnnn*, AREANAME:
areaname)
- INCORRECT BLK TYPE ID IN CI
AT RBA: *rba*
- DATA VALUE: *xx* OFFSET: *xxxx*
- SPACE MAP AT RBA: *rba* OFFSET:
xxxx HAS CONTROL WORD
DISCREPANCY
- CHECKSUM POINTER (PTF/PCF)
ERRORS DETECTED. (UOW START
RBA: *rba*)
- CHECKSUM POINTER (PCL)
ERRORS DETECTED. (UOW START
RBA: *rba*)
- CI "SPACE USAGE"
DISCREPANCY
- RBA: *rba*
- CI AT RBA: *rba* CONTAINS A
NEGATIVE SEGMENT/FSE LENGTH
- CI AT RBA: *rba* CONTAINS AN
INCORRECT SEGMENT/FSE
LENGTH
- CI AT RBA: *rba* HAS UOW#
DISCREPANCY
- IOVF SPACE MAP: *yyyyyyyy* AND
IOVF CI PREFIX: *zzzzzzzz*
- SEGMENT AT RBA: *rba* (SEGCD:
xx) HAS A PTF OR AN SPTF PTR
THAT POINTS TO ITS OWN
SEGMENT
- TOTAL FSE LENGTH
DISCREPANCY
- RBA: *rba*
- TOTAL NO. OF FSE DISCREPANCY
- RBA: *rba*
- FSE CHAIN POINTS TO A NON
FSE
- RBA: *rba*
- FSE CHAIN OFFSET ERROR
- RBA: *rba*

- **INCORRECT SEGM CODE IN CI AT RBA: *rba***
- **DATA VALUE: *x* OFFSET: *xxxx***
- **SEGMENT AT RBA: *rba* (SEGCD: *xx*) HAS PCF/PCL/SSPTR DISCREPANCY**
- **SPACE MAP AT RBA: *rba* OFFSET: *yyyy* HAS INCORRECT UOW#**
- **CI AT RBA: *rba* CONTAINS AN INCORRECT SCRAP LENGTH**
- **ERROR IN THE FIRST DOVF CI. THE RBA OF THE CURRENT OVERFLOW CI IS NULL. RBA: *rba*, UOW NO: *nnnnn***

- CI AT RBA: *rba* CONTAINS AN INCORRECT SEGMENT/FSE LENGTH

The hash check or the area sensor process encountered a segment or an FSE with an incorrect length during the serial deblocking of the CI at RBA *rba*.

- CI AT RBA: *rba* HAS UOW# DISCREPANCY

- IOVF SPACE MAP: *yyyyyyyyy* AND IOVF CI PREFIX: *zzzzzzzz*

The UOW number *yyyyyyyyy* in the space map that corresponds to the CI at RBA *rba*, and the number *zzzzzzzz* in the IOVF CI prefix at RBA *rba* are different.

- SEGMENT AT RBA: *rba* (SEGCD: *xx*) HAS A PTF OR AN SPTF PTR THAT POINTS TO ITS OWN SEGMENT

The hash check process encountered an IMS pointer chain discrepancy. The segment at RBA *rba* is pointed to by its own PTF pointer.

- TOTAL FSE LENGTH DISCREPANCY

- RBA: *rba*

The hash check process detected that the total free space in the CI at RBA *rba*, as calculated by tracking the FSE chain, did not correspond to the value that was calculated during the serial deblocking of the CI. This condition might have been caused by one of incorrect FSE chain, FSE, or segment length.

- TOTAL NO. OF FSE DISCREPANCY

- RBA: *rba*

The hash check process determined that the number of FSE elements in the CI at RBA *rba*, which was calculated by tracking the FSE chain, did not match the value that was calculated during the serial deblocking of the CI. This condition might have been caused by one of incorrect FSE chain, FSE, or segment length.

- FSE CHAIN POINTS TO A NON FSE

- RBA: *rba*

The hash check process encountered an incorrect FSE while tracking the FSE chain for the CI at RBA *rba*.

- FSE CHAIN OFFSET ERROR

- RBA: *rba*

The FSE pointer value in the CI at RBA *rba* does not point to its own CI.

- INCORRECT SEGM CODE IN CI AT RBA: *rba*

- DATA VALUE: *x* OFFSET: *xxxx*

The hash check or area sensor process encountered an incorrect segment code during the serial deblocking of the CI at RBA *rba*.

- SEGMENT AT RBA: *rba* (SEGCD: *xx*) HAS PCF/PCL/SSPTR DISCREPANCY

The hash check process encountered an error in the segment at RBA *rba* while checking the PCF/PCL/SSPTR pointer inter-dependencies.

Explanation

Each message text means as follows:

- INCORRECT BLK TYPE ID IN CI AT RBA: *rba*

- DATA VALUE: *xx* OFFSET: *xxxx*

The hash check or the area sensor process encountered an incorrect block type (DBLKBTypeID) in the CI at RBA *rba*.

- SPACE MAP AT RBA: *rba* OFFSET: *xxxx* HAS CONTROL WORD DISCREPANCY

The CI that corresponds to the space map at RBA *rba* OFFSET *xxxx* should be the first allocatable CI (that is, 1 byte from OFFSET *xxxx* should be x'80'). However, the space map specifies that the CI is already allocated (that is, 1 byte from OFFSET *xxxx* is x'40').

- CHECKSUM POINTER (PTF/PCF) ERRORS DETECTED. (UOW START RBA: *rba*)

The hash check process encountered checksum pointer (PTF/PCF) errors.

- CHECKSUM POINTER (PCL) ERRORS DETECTED. (UOW START RBA: *rba*)

The hash check process encountered checksum pointer (PCL) errors.

- CI "SPACE USAGE" DISCREPANCY

- RBA: *rba*

The hash check process determined that the sum of the free space element, scrap, and segment length in the CI at RBA *rba* that were encountered during the serial deblocking of that CI, was not equal to the usable space of the CI. This condition might have been caused by one of incorrect FSE chain, FSE, or segment length.

- CI AT RBA: *rba* CONTAINS A NEGATIVE SEGMENT/FSE LENGTH

The hash check or the area sensor process encountered a segment or an FSE with a negative length during the serial deblocking of the CI at RBA *rba*.

- SPACE MAP AT RBA: *rba* OFFSET: *yyyy* HAS INCORRECT UOW#

The space map CI at RBA *rba* OFFSET *yyyy* is larger than the number of UOWs that is defined in the area.

- CI AT RBA: *rba* CONTAINS AN INCORRECT SCRAP LENGTH

The area sensor process encountered a SCRAP that has 4 or more length during the serial deblocking of the CI at RBA *rba*.

- ERROR IN THE FIRST DOVF CI. THE RBA OF THE CURRENT OVERFLOW CI IS NULL. RBA: *rba*, UOW NO: *nnnnn*

The pointer value of the first DOVF CI, which is identified by the indicated RBA and UOW number, is null.

System action

The return code is set to 8 and processing continues.

User response

Correct the errors, and rerun the job.

HFPT0005E	A SEGMENT KEY SEQUENCE ERROR IS FOUND IN AREA NO: <i>nn</i>, AREANAME: <i>areaname</i>
------------------	---

Explanation

A segment key sequence in the segment of the indicated area contains an error.

System action

The return code is set to 8 and processing continues.

User response

Consult your database administration personnel about the procedures for correcting the sequence field in error. Correct the problem, and rerun the unload job.

HFPT0006W	UOW RANKING PROCESS ENDED SUCCESSFULLY, BUT THE NUMBER OF UOWS TO BE REORGANIZED IS ZERO. (AREANO: <i>nnnn</i>, AREANAME: <i>areaname</i>)
------------------	---

Explanation

The UOW Ranking process ended successfully, but the number of UOWs to be reorganized is zero.

System action

The return code is set to 4 and processing continues.

User response

None.

HFPT0007E	A SERIOUS ERROR WAS DETECTED DURING THE UOW RANKING PROCESS IN AREA NO: <i>nnnn</i>, AREANAME: <i>areaname</i> - GETMAIN FAILED IN THE [INITIALIZATION SORT] PROCESS.
------------------	--

Explanation

The GETMAIN macro to obtain space for internal blocks failed.

System action

The return code is set to 8 and processing continues.

User response

If the specified region size is too small, increase the REGION size on the JOB statement in the JCL, and rerun the utility.

HFPT0008E	FAILURE COUNT HAS EXCEEDED THE MAXFAILS NUMBER IN AREA NO: <i>nnnnn</i>, AREANAME: <i>areaname</i>.
------------------	--

Explanation

The total number of failures that occurred in the UOW Reorganization processing has exceeded the value that is specified in the MAXFAILS value.

System action

The return code is set to 8 and processing continues.

User response

Check the reason of the error, and correct the cause of the error.

HFPT0009E	DL/I function CALL FAILED STATUS='xx' - <i>subtext</i>
------------------	---

Explanation

The indicated DL/I function returned the status code that is indicated in this message. This status code was issued while the indicated secondary index database or primary DEDB database was being processed. The hexadecimal key value shows the key of the pointer segment for which the DL/I call failed.

subtext is in either of the following formats.

For a primary DEDB database:

- - PRIMARY DEDB DATABASE: *dedb_dbdname*
- - AREA NAME: *areaname*
- - SEGMENT: *segname*
- - RELATED SECONDARY INDEX DATABASE: *index_dbdname*
- - RELATED POINTER SEGMENT KEY: *key1*
- - (cont) : *key2*
- ...
- - (cont) : *keyn*

For a secondary index database:

- - SECONDARY INDEX DATABASE: *index_dbdname*
- - KEY VALUE: *key1*
- - (cont) : *key2*
- ...
- - (cont) : *keyn*
- - CALL SEQUENCE NUMBER: *nnnn*

System action

If the failed DL/I function is SYNC, FPA issues message HFPT3309E and ends the job with an abend code of U3309. Otherwise, FPA skips processing for the indicated secondary index database, sets the end-of-job return code to 8, and continues processing the next secondary index database.

User response

If RESYNCDUMP=YES is specified, identify the failed segments in the Resync Pointer Segment Dump report. Correct the errors, and rerun the job.

HFPT0010I	[OAE OER OPC SCAN DELETE] PROCESSING WAS INVOKED FOR DATABASE: <i>databasename</i>, AREANAME: <i>areaname</i>
------------------	--

Explanation

The Online Space Management utility started the OAE, OER, OPC, SDEP SCAN, or SDEP DELETE utility to process the indicated area.

System action

Processing continues.

User response

None. This message is informational.

HFPT0011I	[OAE OER OPC SCAN DELETE] PROCESSING COMPLETED SUCCESSFULLY FOR DATABASE: <i>databasename</i>, AREANAME: <i>areaname</i> (Elapsed Time: <i>hh:mm:ss.tt</i>) - FOR THE RESULTS OF OAE, SEE THE FPX0099I MESSAGE. - FOR THE RESULTS OF OER, SEE THE REORGANIZATION PROCESSING SUMMARY REPORT.
------------------	--

Explanation

The OAE, OER, OPC, SDEP SCAN, or SDEP DELETE utility completed successfully for the indicated area.

System action

Processing continues.

User response

None. This message is informational.

HFPT0012W	[OAE OER OPC SCAN DELETE] PROCESSING ENDED WITH WARNINGS FOR DATABASE: <i>databasename</i>, AREANAME: <i>areaname</i> (Elapsed Time: <i>hh:mm:ss.tt</i>) - FOR THE RESULTS OF OAE, SEE THE FPX0099I MESSAGE. - FOR THE RESULTS OF OER, SEE THE REORGANIZATION PROCESSING SUMMARY REPORT. - SEE THE FPX WARNING MESSAGES TO IDENTIFY THE WARNING CONDITIONS.
------------------	--

Explanation

The OAE, OER, OPC, SDEP SCAN, or SDEP DELETE utility ended with warning conditions for the indicated area.

System action

Processing ends with a return code of 4.

User response

Locate the warning messages that are prefixed with FPX (*FPXnnnnW*) and follow the user response for those messages.

HFPT0013E [OAE | OER | OPC | SCAN | DELETE] ENDED WITH RC=*nn* FOR DATABASE: *databasename*, AREANAME: *areaname* (Elapsed Time: *hh:mm:ss.tt*)
- SEE THE FPX ERROR MESSAGES TO IDENTIFY THE ERROR.

Explanation

The OAE, OER, OPC, SDEP SCAN, or SDEP DELETE utility ended with an error for the indicated area.

System action

Processing ends with a return code of 8.

User response

Locate the error messages that are prefixed with FPX (FPX*nnnn*E) and follow the user response for those messages. Correct the error, and then rerun the job.

HFPT0014I AREA *areaname* IS FLAGGED AS EMPTY.

Explanation

The empty area flag DMACEMPT is set in the DMAC.

System action

Processing continues.

User response

None. This message is informational.

HFPT0015E IDCAMS *command* COMMAND ENDED WITH ERROR. SEE RESULT OF ACCESS METHOD SERVICES REPORT FOR DETAILS.
- ERROR DETECTED IN IDCAMS EXAMINE [BEFORE | AFTER] FREESPACE RECLAIM
- SECONDARY INDEX DATABASE DATA SET NAME: *dname*

Explanation

An error occurred while the IDCAMS was issuing the indicated IDCAMS command. The Result of Access Method Services report contains details about the error.

System action

FPA skips processing for the indicated secondary index database, sets the end-of-job return code to 8, and

continues processing the next secondary index database.

User response

See the Result of Access Method Services report and determine the cause of the IDCAMS command failure, correct any errors, and rerun the job.

HFPT0016W IDCAMS *command* COMMAND ENDED WITH WARNING. SEE RESULT OF ACCESS METHOD SERVICES REPORT FOR DETAILS.
- WARNING DETECTED IN IDCAMS EXAMINE [BEFORE | AFTER] FREESPACE RECLAIM
- SECONDARY INDEX DATABASE DATA SET NAME: *dname*

Explanation

The indicated IDCAMS command ended with warning conditions. The Result of Access Method Services report describes the details of the warning conditions.

System action

The return code is set to 4 and processing continues.

User response

See the Result of Access Method Services report.

HFPT0508I OSC INTERNAL JOB STARTED. JOB ID: *job_id*, JOB: *int_jobname*

Explanation

The indicated OSC internal job has started.

System action

Processing continues.

User response

None. This message is informational.

HFPT0509I OSC INTERNAL JOB ENDED NORMALLY. JOB ID: *job_id*, JOB: *int_jobname*

Explanation

The indicated OSC internal job ended normally.

System action

Processing continues.

User response

None. This message is informational.

HFPT0512A **OSC INTERNAL JOB *int_jobname* ENDED WITH ERRORS. (JOB ID: *job_id*) REPLY=>C OR A (CONTINUE OR ABEND).**

Explanation

This message is issued when an OSC internal job fails.

System action

Issues this message to query the user for the action to take.

User response

Correct the failed OSC internal job and rerun the OSC internal job. If the job ends normally, reply to the HFPT0512A message with C so that the OSC driver job continues processing from where it was paused. Otherwise, replay to this message with A to end the OSC driver job.

HFPT3301E **ERROR IN CALL TO RANDOMIZER: *randomizer*, IN AREA NO: *nnnn*, AREANAME: *areaname*, SEGMENT RBA: *xxxxxxxx* - RETURN CODE: *nn* INCORRECT AREA NO: *nn*, RAP NO: *nnnn* WERE RETURNED.**

Explanation

The thread task invoked the randomizer routine that was specified to calculate the area number and the RAP RBA information. The root key value that is used in the call to the randomizer is the value specified by the program that invoked the thread task with a function code of 'CALC'. The return code from the randomizer routine shown in the error message was not zero.

System action

The job ends with an abend code of U3003.

User response

Make sure that the RMODLIB DD statement is identified in the correct data set, and that the randomizer routine has been correctly added, assembled, and link-edited. Correct any errors, and rerun the job.

HFPT3302E **PROCESSING ENDED WITH STATUS CODE: *nn*, IN AREA NO: *nn*, AREANAME: *areaname***

Explanation

The thread task called the user exit routine *exit-routine*, and the exit routine returned E2 status code.

System action

The job ends with an abend code of U3003.

User response

Correct the problem, if necessary, and rerun the job.

HFPT3303E **EXIT ROUTINE: *exit-routine* RETURNED INCORRECT STATUS CODE: *cc(X'xxxx')* IN AREA NO: *nn*, AREANAME: *areaname* - SEG-CD: *xx* - RBA: *xxxxxxxx* - RECORD NUMBER: *xxxxxxxx***

Explanation

The thread task received the incorrect status code indicated by *cc(X'xxxx')* from the user exit routine *exit-routine* that was specified by the EXITRTN= control statement.

System action

The job ends with an abend code of U3003.

User response

Check the logic of the user exit routine. Correct the exit routine, and rerun the job.

HFPT3304E **EXIT ROUTINE: *exit-routine* RETURNED STATUS CODE: *nn(X'xxxx')* IN AREA NO: *nn*, AREANAME: *areaname* BUT PARENT SEGMENT WAS RETURNED 'T2'. - SEG-CD: *xx* - RBA: *xxxxxxxx* - RECORD NUMBER: *xxxxxxxx***

Explanation

The input thread task or the output thread task received the status code E2 from the user exit routine *exit-routine* that was specified by the EXITRTN= control statement.

System action

The job ends with an abend code of U3003.

User response

Investigate the logic of the user exit routine. Correct the exit routine, and rerun the job.

HFPT3305E **SEGMENT KEY SEQUENCE ERROR
IN AREA NO: *nn*, AREANAME:
areaname
- ABEND DUE TO
KEYSEQCHK=ABEND OPTION**

Explanation

A segment key sequence in the segment of the area contains an error. The segment type, ROOT or DDEP, and the segment in which the error was found are shown in the message.

System action

The job ends with an abend code of U3003.

User response

Consult your database administration personnel about the procedures for correcting the sequence field in error. Correct the problem, and rerun the unload job.

HFPT3306E **SEGMENT IN AREA NO: *nn*
(SEGCODE: *nn*)
- AT RBA: X'*xxxxxxxx*' RETURNED
FROM COMPRESSION ROUTINE:
compression_routine_name
- ABEND CODE: *Unnnn* REASON
CODE: X'*xxxxxxxx*'
- AT REC#: *xxxxxxxx* RETURNED
FROM COMPRESSION ROUTINE:
compression_routine_name
- AT RBA: X'*xxxxxxxx*' RETURNED
FROM USER EXIT ROUTINE:
user_exit_routine_name
- AT REC#: *xxxxxxxx* RETURNED
FROM USER EXIT ROUTINE:
user_exit_routine_name
- EXCEEDS MAX DEFINED LENGTH
- TOO SHORT
- KEY FIELD MODIFIED**

Explanation

One of the following error conditions was detected:

- The specified edit/compression routine or user exit routine, which was called by a thread task, returned an error. This message is issued when an error

occurs in the EXPAND or COMPRESS call. If the error occurs in OPEN or CLOSE call, message HFPT3310E is issued instead of this message.

- The length or the key field for the segment is incorrect.

System action

The job ends with an abend code of U3003.

User response

Correct the errors, and rerun the job.

HFPT3307E **RANDOMIZER: *randomizer* USED
LENGTH BEYOND THE MAXIMUM
WORK AREA LENGTH.**

Explanation

The randomizer used area that exceeded the size of the maximum work area.

System action

The job ends with an abend code of U3003.

User response

Check and extend the work area of the randomizer.

HFPT3308E **SORT ENDED ABNORMALLY (RC:
nn, MSGDDN: *msg_ddname*).
- DDNAME: *ddname*, DSNAME:
dsname
- DDNAME: *ddname*, DSNAME:
*dsname(member_name)***

Explanation

The thread task linked the DFSORT program internally to sort a data set that is associated with the SORTIN DD statement. The DFSORT program returned an error code.

System action

The job ends with an abend code of U3003.

User response

Make sure that the DD statements for the SYSOUT and SORTWRK data sets are specified correctly. If there are DFSORT messages on the data set that is associated with the SYSOUT DD statement, check those messages and correct any errors. If the SORT program issued message ICE046A, specify the SORTOPT_FILSZ=AUTO option or DD statements for

the intermediate storage data sets for the SORT program and rerun the job. For more information about intermediate storage data sets for the SORT program, see the DD statements topic of the function that you are using.

HFPT3309E STOPPING THE OUTPUT THREAD TASK DUE TO ERRORS.

Explanation

The output thread task ended abnormally.

System action

The job ends with an abend code of U3309.

User response

Check the preceding message and determine the cause of the thread task error. Correct any errors, and rerun the job.

HFPT3310E COMPRESSION EXIT *exit_name* RETURNED AN ERROR FOR SEGMENT *segment_name* - ABEND CODE: Unnnn REASON CODE: X'xxxxxxxx'

Explanation

The specified edit/compression routine was called, but it returned an error. This message is issued when an error occurs in the OPEN or CLOSE call. If the error occurs in EXPAND or COMPRESS call, message HFPT3306E is issued instead of this message.

System action

The job ends with an abend code of U3003.

User response

Correct the errors, and rerun the job.

HFPU messages

The following information is about messages and codes that begin with HFPU.

HFPU0001I ABEND REQUEST MODULE REGISTER INFORMATION (TCT: *tctname*, MODULE: *modulename*) - register1: *register_value1* register2: *register_value2* register3: *register_value3* register4: *register_value4*

HFPT3712E OSC INTERNAL JOB PROCESS ENDED WITH ERRORS. JOB ID: *job_id*, JOB: *int_jobname*

Explanation

The indicated OSC internal job ended with errors. Because ERRORACT=ABEND is specified, the OSC driver job ended abnormally.

System action

The OSC driver job ends with an abend code of U3712.

User response

Locate other error messages and correct the error condition. See [“Troubleshooting Online Structure Change utility problems” on page 1023](#) to restart the OSC driver job.

HFPT3716E OSC INTERNAL JOB *int_jobname* ABEND REQUEST. (JOB ID: *job_id*)

Explanation

The indicated OSC internal job ended with errors. Because the user reply to message HFPT0512A was A, the OSC driver job ended abnormally.

System action

The OSC driver job ends with an abend code of U3716.

User response

Locate other error messages and correct the error condition. See [“Troubleshooting Online Structure Change utility problems” on page 1023](#) to restart the OSC driver job.

Explanation

This message contains information about the module that caused an abend, including the module name, the TCT name, and associated registers.

System action

The program writes register information about the abend.

User response

Read the user response section of the message that preceded this message. Correct the error, and rerun the job.

HFPU0040I **LIU INTEGRITY CHECKER
INITIALIZED.**

Explanation

The Integrity Checker utility of IMS Library Integrity Utilities has been initialized to create the RDE of the DBD.

System action

Processing continues.

User response

None. This message is informational.

HFPU0041E **LIU INTEGRITY CHECKER
FUNC=INIT ERROR. RC=*rc*,
RSN=*yyyy*, DBD=*dbdname***

Explanation

FPA received an error issued by the Integrity Checker utility of IMS Library Integrity Utilities. *rc* is the return code and *yyyy* is the reason code returned from Integrity Checker. For the detail of the error reason, see the FABLxxxxE message, which is issued by Integrity Checker.

System action

The job ends with a return code of 8.

User response

See the explanation for the FABLxxxxE message in the *IMS Library Integrity Utilities User's Guide*.

HFPU0042W **LIU INTEGRITY CHECKER
FUNC=CREATE ERROR. RC=*rc*,
RSN=*yyyy*, DBD=*dbdname***

Explanation

FPA received an error issued by the Integrity Checker utility of IMS Library Integrity Utilities. This message indicates that FPA process completed successfully but the RDE of the DBD was not created. *rc* is the return code and *yyyy* is the reason code returned from Integrity Checker. For the detail of the error reason, see the FABLxxxxE message, which is issued by Integrity Checker.

System action

The return code is set to 4 and processing continues.

User response

See the explanation for the FABLxxxxE message in the *IMS Library Integrity Utilities User's Guide*.

HFPU0043E **ERROR OCCURRED WHILE
PROCESSING LIU INTEGRITY
CHECKER FUNC=INIT
- GETMAIN FAIL.**

Explanation

GETMAIN macro to obtain space for internal blocks for FUNC=INIT of IMS Library Integrity Utilities Integrity Checker failed.

System action

The job ends with a return code of 8.

User response

If the region size specified is too small, increase the REGION size on the JOB statement in the JCL, and rerun the utility.

HFPU0044W **ERROR OCCURRED WHILE
PROCESSING LIU INTEGRITY
CHECKER FUNC=CREATE
- GETMAIN FAIL.**

Explanation

GETMAIN macro to obtain space for internal blocks for FUNC=CREATE of IMS Library Integrity Utilities Integrity Checker failed. This message indicates that FPA process completed successfully but the RDE of the DBD was not created.

System action

The return code is set to 4 and processing continues.

User response

See the restrictions and considerations about using Integrity Checker in the *IMS Library Integrity Utilities User's Guide*.

HFPU0099E **DATA SET NAME ERROR. * MARKS
THE WRONG PART.
DSNAME: *dsname***

Explanation

This message is generated when the specified DSN: *dsname* is incorrect. The asterisk (*) points to the beginning of the part in error.

System action

If SCAN=YES is specified, the return code is set to 8. If SCAN=NO is specified, an abend occurs with a code of U3003.

User response

Make sure that the data set name is specified correctly. Correct any errors, and rerun the job.

HFPU0110E **OPEN FAILED FOR DDNAME:**
ddname
- **DUMMY OR NULLFILE SPECIFIED.**
- **DD STATEMENT NOT SPECIFIED.**
- **"DEVTYPE" MACRO FAILED. (RC: xx)**
- **"OPEN" MACRO FAILED. (RC: xx)**

Explanation

This message consists of multiple parts. These parts are:

- **DUMMY OR NULLFILE SPECIFIED.**
DUMMY or NULLFILE is specified in *ddname*. You cannot specify DUMMY or NULLFILE for this DDNAME.
- **DD STATEMENT NOT SPECIFIED.**
DD statement *ddname* is not specified in the JCL statement.
- **"DEVTYPE" MACRO FAILED. (RC: xx)**
The DEVTYPE macro failed.
- **"OPEN" MACRO FAILED. (RC: xx)**
The OPEN macro failed.

System action

The return code is set to 8 and processing continues.

User response

- **DUMMY OR NULLFILE SPECIFIED.**
Make sure that the data set name is specified correctly. Correct any errors, and rerun the job.
- **DD STATEMENT NOT SPECIFIED.**
Make sure that DD statement *ddname* is specified correctly. Correct any errors, and rerun the job.

- "DEVTYPE" MACRO FAILED. (RC: xx)

Make sure that a DD statement is present for *ddname* and that it properly identifies the correct data set. Correct any errors, and rerun the job. If this situation persists, contact IBM Software Support.

- "OPEN" MACRO FAILED. (RC: xx)

Make sure that a DD statement is present for *ddname* and that it properly identifies the correct data set. Correct any errors, and rerun the job. If this situation persists, contact IBM Software Support.

HFPU3010E **OPEN FAILED FOR DDNAME:**
ddname
- **DUMMY OR NULLFILE SPECIFIED.**
- **DD STATEMENT NOT SPECIFIED.**
- **"DEVTYPE" MACRO FAILED. (RC: xx)**
- **"OPEN" MACRO FAILED. (RC: xx)**

Explanation

This message consists of multiple parts. These parts are:

- **DUMMY OR NULLFILE SPECIFIED.**
DUMMY or NULLFILE is specified in *ddname*. You cannot specify DUMMY or NULLFILE for this DDNAME.
- **DD STATEMENT NOT SPECIFIED.**
DD statement *ddname* is not specified in the JCL statement.
- **"DEVTYPE" MACRO FAILED. (RC: xx)**
The DEVTYPE macro failed.
- **"OPEN" MACRO FAILED. (RC: xx)**
The OPEN macro failed.

System action

The job ends with an abend code of U3003, U3006, or U3010.

User response

- **DUMMY OR NULLFILE SPECIFIED.**
Make sure that the data set name is specified correctly. Correct any errors, and rerun the job.
- **DD STATEMENT NOT SPECIFIED.**
Make sure that DD statement *ddname* is specified correctly. Correct any errors, and rerun the job.
- **"DEVTYPE" MACRO FAILED. (RC: xx)**
Make sure that a DD statement is present for *ddname* and that it properly identifies the correct data set. Correct any errors, and rerun the job. If

this situation persists, contact IBM Software Support.

- "OPEN" MACRO FAILED. (RC: xx)

Make sure that a DD statement is present for *ddname* and that it properly identifies the correct data set. Correct any errors, and rerun the job. If this situation persists, contact IBM Software Support.

**HFPU3030E UNABLE TO FIND *modulename*
CALLER=*callermodule***

Explanation

The program could not find *modulename*.

System action

The job ends with an abend code of U3003, U3006, or U3030.

User response

This problem may be an internal error. If this situation persists, contact IBM Software Support.

**HFPU3031E I/O ERROR FOR OUTPUT DATA SET
DDNAME: *ddname***

Explanation

The program issued a PUT macro to the specified *ddname*. The PUT macro failed.

System action

The job ends with an abend code of U3003 or U3006.

User response

Correct the errors, and rerun the job. If this situation persists, contact IBM Software Support.

**HFPU3034E DIRECTED LOAD FAILED FOR
MODULE: *modulename* CALLER:
callermodule REASON: *text***

Explanation

The program was unable to load *modulename* for a randomizer, a user exit routine, or an edit compression routine directly.

System action

The job ends with an abend code of U3034.

User response

Correct the errors, and rerun the job. If this situation persists, contact IBM Software Support.

**HFPU3040E INSUFFICIENT STORAGE FOR:
xxxx, TCT: *tctname*, MODULE:
*modulename***

Explanation

The program was unable to obtain sufficient virtual storage.

System action

The job ends with an abend code of U3003, U3006 or U3040.

User response

Correct the errors, and rerun the job. If this situation persists, contact IBM Software Support.

**HFPU3041E STORAGE RELEASE FAILURE FOR:
xxxx, TCT: *tctname*, MODULE:
*modulename***

Explanation

The program could not release virtual storage.

System action

The job ends with an abend code of U3003, U3006, or U3041.

User response

Correct the errors, and rerun the job. If this situation persists, contact IBM Software Support.

**HFPU3042E INSUFFICIENT TABLE STORAGE
FOR: *xxxxxxxx*, TCT: *tctname*,
MODULE: *modulename*, REQUEST
LENGTH: *nnnnnnnnnn*
- OLD TABLE ADDRESS: *xxxxxxxx*,
HEADER: *xx..xx***

Explanation

The program was unable to obtain sufficient virtual storage.

System action

The job ends with an abend code of U3003 or U3006.

User response

Correct the errors, and rerun the job. If this situation persists, contact IBM Software Support.

HFPV0001W **TABLE STORAGE RELEASE FAILURE FOR: xxxxxxxx, TCT: tctname, MODULE: modulename**
- **TABLE ADDRESS: xxxxxxxx,**
HEADER: xx..xx

Explanation

The program was unable to release virtual storage.

System action

The job ends with an abend code of U3003 or U3006.

User response

Correct the errors, and rerun the job. If this situation persists, contact IBM Software Support.

HFPV messages

The following information is about messages and codes that begin with HFPV.

HFPV0001W **OPEN FAILURE FOR AREANAME: areaname, DDNAME: ddname**
- **MEDIA MANAGER CONNECT FAILED. (RC: xx)**
- **MEDIA MANAGER EXTEND FAILED. (RC: xx)**

Explanation

The program issued MMGRSRV to connect the device that is associated with *areaname* and *ddname*. The return code shows that the attempt was unsuccessful.

System action

Processing continues.

User response

Correct any errors, and rerun the job. If this situation persists, contact IBM Software Support.

HFPV0002W **CLOSE FAILED FOR AREANAME: areaname, DDNAME: ddname (RC: xx)**

Explanation

The program issued MMGRSRV to disconnect the device associated with *areaname* and *ddname*. The return code shows that the attempt was unsuccessful.

HFPV0003W **INTERNAL ERROR FOUND TCT: tctname, MODULE: modulename**

Explanation

The program detected an internal error.

System action

The job ends abnormally.

User response

This problem might be an internal error. If this situation persists, contact IBM Software Support.

System action

Processing continues.

User response

Correct any errors, and rerun the job. If this situation persists, contact IBM Software Support.

HFPV0003W **MEDIA MANAGER I/O ERROR - AREA NO: n, AREANAME: areaname**
- **DDNAME: ddname, DSNAME: dsname**
- **REQUESTED RBA: eeeeeeee**
- **MEDIA MANAGER RETURN CODE: cccfffs**

Explanation

When a subtask of the program issued the MMGRCALL to get access to the data set that is associated with the *ddname* indicated, an unexpected Media Manager MMGRCALL error occurred. The variable *ccccfffs* represents the Media Manager error return code that is used for problem determination. Media Manager return codes are described in the *DFSMS: DFSMSdftp Diagnosis Reference*.

System action

Processing continues.

User response

Check the Media Manager return code, correct the error, and rerun the job. If this situation persists, contact IBM Software Support.

HFPV0004I **xxxx PART COULD NOT BE PRELOADED INTO A DATA SPACE FOR PROCESSING IN keyword=YES.**
- **AREA NO: nn, AREANAME: areaname**
- **RC: return-code, RSN: reason-code**

Explanation

The program uses a VSAM area data set or an image copy data set for xxxx (BASE, IOVF, or SDEP) as the work area even if *keyword* (ICACHE or OCACHE)=YES is specified because the work area cannot be acquired in the data space.

System action

If a VSAM area data set is used, processing continues by using the VSAM area data set instead of the data space for an IOVF and/or a SDEP part. If an image copy data set is used, processing ends and the program issues a HFPV error message.

User response

If an image copy data set is used, check the subsequent HFPV message. Otherwise, no action is required.

HFPV0005E **ALL ADS I/O FAILED FOR AREA NO: nnnnn, AREANAME: areaname**

Explanation

An I/O error occurred. If multiple area data sets (MADS) are defined, this message indicates that the I/O error occurred for all area data sets of the MADS.

System action

The return code is set to 8 and processing continues.

User response

See the other messages that are related to I/O error, correct the error, and rerun the job.

HFPV0006E **OPEN FAILED FOR ALL DATA SETS IN AREANAME: areaname**

Explanation

The program tried to open all of the devices that are associated with *areaname*, but no data sets opened.

System action

The return code is set to 8 and processing continues.

User response

See the HFPV0001W messages that are related to the OPEN error. Correct the error, and rerun the job.

HFPV0007I **INPUT DATA SET DDNAME: ddname WAS A CIC IMAGE COPY DATA SET.**

Explanation

The Concurrent image copy was provided as an input data set for the ANALYZE function or the DMACPRT function. Pointer errors are expected with the input data set.

System action

Processing continues.

User response

None. This message is informational.

HFPV0008I **DEFINED AREA DATA SET DOES NOT HAVE THE ENCRYPTION ATTRIBUTE.**
- **AREA NO: nn, AREA NAME: areaname**
- **DDNAME: ddname, DSNAME: dsname**

Explanation

An encryption key label is specified on the KEYLABEL keyword of the ALLOCATE subcommand. However, this key label was not applied to the indicated area data set because the area data set is not in extended format.

System action

Processing continues. The area data set is successfully created as a non-encrypted format data set.

User response

No further action is required. However, if you want the area data set to be an encrypted data set, allocate the

data set by specifying the DATACLAS parameter with the extended format attribute.

**HFPV3050E OPEN FAILED FOR ALL DATA SETS
IN AREANAME: *areaname***

Explanation

The program tried to open all of the devices that are associated with *areaname*, but no data sets opened.

System action

The job ends with an abend code of U3003.

User response

See the HFPV0001W messages that are related to the OPEN error. Correct the error, and rerun the job.

**HFPV3051E CLOSE FAILED FOR ALL DATA SETS
IN AREANAME: *areaname***

Explanation

The program tried to close all of the devices that are associated with *areaname*, but no data sets closed.

System action

The job ends with an abend code of U3003.

User response

See the HFPV0002W messages that are related to the CLOSE error. Correct the error, and rerun the job.

**HFPV3052E UOW BUFFER GET FAILED FOR
AREANAME: *areaname***

Explanation

The program could not get the storage for UOW data accessing.

System action

The job ends with an abend code of U3003.

User response

Increase the region size parameter on the EXEC statement, and rerun the job.

**HFPV3053E RESOURCE SHORTAGE FOR
ACCESS OF IMAGE COPY DATA
SET DDNAME: *ddname*
- RC: *return-code*, RSN: *reason-
code***

**- UNABLE TO GET IC BUFFER.
RESOURCE LIMITED BY THE
IEFUSI EXIT. RC: *return-code*,
RSN: *reason-code***

Explanation

IC buffer exceeded the data space or the storage for the RAA BASE, IOVF, SDEP part, or the required data space could not be created. Either of the subtexts is issued.

System action

The job ends with an abend code of U3003.

User response

Rerun the job with the ICINCOMP=YES keyword. If the problem is caused by the IEFUSI exit, contact your system programmer for local restrictions on creating and using data spaces. For the Analyze function or the Tune function, if this problem persists, rerun the job with the ICACHE=NO keyword.

For information about the keywords, see the following topics:

- [“ICACHE keyword” on page 472](#)
- [“ICINCOMP keyword” on page 474](#)

**HFPV3054E "CSRCSRV" MACRO FAILED.
(FUNCTION: *func*, RC: *xx*)**

Explanation

The program issued a CSRCSRV macro to compress or expand the data. The return code shows that the attempt was unsuccessful.

System action

The job ends with an abend code of U3003.

User response

Correct any errors, and rerun the job. If this situation persists, contact IBM Software Support.

HFPV3055E "PGSER" MACRO FAILED. (RC: *xx*)

Explanation

The program issued a PGSER macro to do page fix or page free. The return code shows that the attempt was unsuccessful.

System action

The job ends with an abend code of U3003.

User response

Correct any errors, and rerun the job. If this situation persists, contact IBM Software Support.

HFPV3056E **ALL ADS I/O FAILED FOR AREA
NO: *nnnnn*, AREaname: *areaname***

Explanation

An I/O error occurred. If multiple area data sets (MADS) are defined, this message indicates that the I/O error occurred for all area data sets of the MADS.

System action

The job ends with an abend code of U3003, or ends with a return code of 12.

User response

See the other messages that are related to I/O error, correct the error, and rerun the job.

HFPV3057E **INPUT DATA SET DDNAME:
ddname WAS NOT A SUPPORTED
IMAGE COPY DATA SET.
- THE IMAGE COPY DATA SET HAS
A FORMAT ERROR.
- UNSUPPORTED IMS RELEASE.
- *command* CANNOT BE
SPECIFIED FOR THE CIC IMAGE
COPY DATA SET.
- IC2 IMAGE COPY DATA SET
CANNOT BE PROCESSED.**

Explanation

The image copy could not be processed as an input data set. The reason is shown by the second text in this message.

System action

The job ends with an abend code of U3003.

User response

Make sure the image copy is correct for the function that is specified in HFPSYSIN. Correct any errors, and rerun the job. If this situation persists, contact IBM Software Support.

HFPV3058E **NOT ENOUGH DATA RECORDS IN
IMAGE COPY DATA SET DDNAME:
*ddname***

Explanation

EOF occurred while the program was reading IC records from the image copy data set.

System action

The job ends with an abend code of U3003.

User response

Ensure that the image copy data set is correct, and rerun the job.

HFPV3060E **NOT EMPTY DATA SET FOR
AREaname: *areaname*, DDNAME:
*ddname***

Explanation

The VSAM data set that was specified in *ddname* for the area *areaname* was not empty.

System action

The job ends with an abend code of U3003.

User response

Delete and redefine the VSAM cluster for the specified area. Do not run the FPA Initialize function or the DEDB Initialization utility (DBFUMINO). Rerun the reload job.

HFPV3063E **CI SIZE OF ADS AND DMB ARE
INCONSISTENT.
- DDNAME: *ddname*, DSNAME:
dsname
- ADS CI SIZE: *nnnnnnnn*
- DMB CI SIZE: *nnnnnnnn***

Explanation

The program detected an inconsistency between the CI size of ADS and DMB.

System action

The job ends with an abend code of U3003.

User response

Check the data set. Correct the error, and rerun the job.

HFPV3065E **FUNCTION *function* FOR
COMPRESSION EXIT ROUTINE
exit-routine-name FAILED WITH
RC=*nn*.**

Explanation

The function for compression exit routine *exit-routine-name* failed with return code *nn*.

The meanings of the functions are:

Function

Meaning

INITIALIZATION

Initialization function call

BLOCK READ

Block read function call

TERMINATION

Termination function call

System action

The job ends with an abend code of U3003.

User response

This error is likely an internal system error. Contact IBM Software Support.

HFPV3066E **THE RECORD LENGTH RETURNED
FROM EXIT ROUTINE: *exit-routine-
name* IS NOT EQUAL TO THE
LENGTH OF THE ORIGINAL CI AT
RBA: *xxxxxxxx***

Explanation

The record length expanded by compression exit routine was not equal to the length of the original CI at RBA *xxxxxxxx*.

System action

The job ends with an abend code of U3003.

User response

This error is likely an internal system error. Contact IBM Software Support.

HFPX messages

The following information is about messages and codes that begin with HFPX.

HFPX0001I **AREANO:*nnnnn*
(AREANAME:*areaname*) IS NOT
REGISTERED IN DBRC**

HFPV3067E **BASE PART COULD NOT BE
PRELOADED.
- IC BUFFER IS OVER THE LIMIT
- AREA NO: *nnnnn*, AREANAME:
*areaname***

Explanation

IC buffer exceeded the data space storage for the RAA BASE, IOVF, SDEP part in the image copy.

System action

The job ends with an abend code of U3003.

User response

If the job is run with ICINCOMP=NO keyword, rerun the job with the ICINCOMP=YES keyword. When ICINCOMP=YES keyword is specified, the image copy cannot be processed as the input data set except for the Analyze function and the Tune function. For the Analyze function or the Tune function, if this problem persists, rerun the job with the ICACHE=NO keyword.

HFPV3068E **THE SIZE OF THE AREA DATA SET
IS MORE THAN 4G BYTES.
- AREA NO: *nn*, AREA NAME:
areaname
- DDNAME: *ddname*, DSNAME:
*dsname***

Explanation

The size of the area data set exceeds 4 GB.

System action

The job ends with an abend code of U3003.

User response

If the indicated ADS contains segment data, run the ODE utility to extract all segments from the ADS, then reload the segments to a new ADS whose size is smaller than 4 GB.

If the indicated ADS is empty, delete and redefine the ADS. Make sure that the size of the ADS does not exceed 4 GB.

Explanation

The specified area is not registered to DBRC.

System action

Processing continues.

User response

None. This message is informational.

HFPX0002E **DBRC REQUEST FAILED.**
FUNCTION: *func*, **RC:** *return-code*
- AREA NO: *nnnn*, **AREANAME:**
areaname

Explanation

The program was unable to successfully complete the DBRC call.

The meanings of the functions are:

Function

Meaning

SIGNON

Sign-on call

SIGNOFF

Sign-off call

AUTH

Area authorization call

UNAUTH

Area unauthorization call

INIT

INIT function call

INITO

INITO function call

INIT1

INIT1 function call

STAT

Data set status function call

EOD

EOD function call

EOJ

EOJ function call

ADSLIST

ADS list request call

The preceding DBRC message explains the reason code.

System action

The job continues.

User response

To rerun the job for the area, check the DBRC message that precedes this message, and follow the response in that message.

HFPX0004I **SECONDARY INDEX DATABASE**
***database* IS NOT CORRECTLY**
REGISTERED TO DBRC.

Explanation

The specified secondary index database is not correctly registered to DBRC.

System action

Processing continues.

User response

None. This message is informational.

HFPX0005I **LIST OF SPECIFIED RECON IMSID.**
- IMS ID: *xxxx*, ... *xxxx*

Explanation

This message shows a list of IMSIDs obtained from the RECON data sets.

System action

Processing continues.

User response

None. This message is informational.

HFPX0006E **IMSID MISMATCH BETWEEN**
DBRC AND IMS TOOLS ONLINE
SYSTEM INTERFACE.
- DIFFERENCE IN THE IMSID
NUMBER.
- NOT FOUND IMSID: *imsid*

Explanation

The IMSIDs obtained from the RECON data sets do not match with the list of IMSIDs that join the XCF group used for the IMS Tools Online System Interface.

System action

The return code is set to 8.

User response

Configure IMS Tools Online System Interface on both the production IMS systems and the IMS DBCTL systems, and rerun the job.

HFPX0101I **CONNECTED TO IMS THROUGH TOOLS ONLINE SYSTEM INTERFACE. XCF GROUP NAME: TOIxxxxx**
- IMS ID: xxxxx

Explanation

The TOSI CONNECT function has completed.

System action

Processing continues.

User response

None. This message is informational.

HFPX0102I **THE [/DBRECOVERY | /DBDUMP | /START] COMMAND FOR SECONDARY INDEX DATABASE *index_name* WAS ISSUED SUCCESSFULLY.**

Explanation

The indicated command was issued successfully.

System action

Processing continues.

User response

None. This message is informational.

HFPX0103E **THE [/DBRECOVERY | /DBDUMP] COMMAND FOR SECONDARY INDEX DATABASE *index_name* FAILED.**
- TOOLS ONLINE SYSTEM INTERFACE API FAILED.
FUNCTION: *function*, RC: *xx*, RSN: *yyyyyyyy*
- **RESPONSE FAILED. IMS ID: *xxxx*, RC: *xxxxxxxx*, RSN: *yyyyyyyy***
- **WAIT TIME LIMIT EXCEEDED.**

Explanation

An attempt to run the /DBRECOVERY or /DBDUMP command for the secondary index database failed.

System action

The return code is set to 8.

User response

If WAIT TIME LIMIT EXCEEDED is displayed, increase the number that is specified in the TOINUM keyword or in the TOITIME keyword. Otherwise, contact IBM Software Support and provide the function name, the return code, and the reason code that are shown in the message.

HFPX0104W **THE /START COMMAND FOR SECONDARY INDEX DATABASE *index_name* FAILED.**
- **TOOLS ONLINE SYSTEM INTERFACE API FAILED.**
FUNCTION: *function*, RC: *xx*, RSN: *yyyyyyyy*
- **RESPONSE FAILED. IMS ID: *xxxx*, RC: *xxxxxxxx*, RSN: *yyyyyyyy***

Explanation

An attempt to run the /START command for the secondary index database failed.

System action

The secondary index database remains offline, and the return code is set to 4.

User response

Issue the /START command manually for the secondary index database. If the problem persists, contact IBM Software Support and provide the function name, the return code, and the reason code that are shown in the message.

HFPX0105E **TOOLS ONLINE SYSTEM INTERFACE API FAILED.**
FUNCTION: *xxxxxxxx*, RC: *xx*, RSN: *yyyyyyyy*

Explanation

An attempt to use the IMS Tools Online System Interface service failed.

System action

The return code is set to 8.

User response

Contact IBM Software Support and provide the function name, the return code, and the reason code that are shown in the message.

HFPX0107E **TOOLS ONLINE SYSTEM
INTERFACE CANNOT BE USED.
- TOOLS ONLINE SYSTEM
INTERFACE MODULE *modulename*
IS NOT FOUND.**

Explanation

The load module library of IMS Tools Online System Interface is not concatenated to the STEPLIB or JOBLIB.

System action

The return code is set to 8.

User response

Ensure that the load module library of IMS Tools Online System Interface is concatenated to the STEPLIB or JOBLIB.

HFPX3099E **DBRC REQUEST FAILED.
FUNCTION: *function*, RC: *rc*
- AREA NO: *nnnnn*, AREANAME:
areaname
- SECONDARY INDEX DATABASE:
*dbdname***

Explanation

The program was unable to successfully complete the DBRC call.

The meanings of the functions are:

Resolve Pointers report messages

The following information is about messages and codes that are generated in the Resolve Pointers report or the Pointer Checksum report.

0101 **INCORRECT BLK TYPE ID IN CI AT
RBA: *xxxxxxxx*
- CI BYPASSED (DATA VALUE: *yy*
OFFSET: *zzz*)**

Explanation

The Analyze process encountered an incorrect block type (DBLKBTID) in the CI at RBA *xxxxxxxx*.

User response

Correct the errors, and rerun the job.

Function

Meaning

SIGNON

Sign-on call

SIGNOFF

Sign-off call

AUTH

Area authorization call

UNAUTH

Area unauthorization call

INIT

INIT function call

INITO

INITO function call

INIT1

INIT1 function call

STAT

Data set status function call

EOD

EOD function call

EOJ

EOJ function call

ADSLIST

ADS list request call

The preceding DBRC message explains the reason code.

System action

The job ends with an abend code of U3003.

User response

Check the DBRC message that precedes this message, and follow the response in that message.

0102 **INCORRECT SEGM CODE IN CI AT
RBA: *xxxxxxxx*
- CI BYPASSED (DATA VALUE: *yy*
OFFSET: *zzz*)**

Explanation

The Analyze process encountered an incorrect segment code during the serial deblocking of the CI at RBA *xxxxxxxx*.

User response

Correct the errors, and rerun the job.

0105 **CI "SPACE USAGE" DISCREPANCY**
- RBA: xxxxxxxx

Explanation

The Analyze process determined that the sum of the free space element, scrap, and segment length in the CI at RBA xxxxxxxx that were encountered during the serial deblocking of that CI was not equal to the usable space of the CI. This condition might have been caused by one of incorrect FSE chain, FSE, or segment length.

User response

Correct the errors, and rerun the job.

0107 **SEGMENT AT RBA: xxxxxxxx**
(SEGCD: yy) HAS PCF/PCL/SSPTR
DISCREPANCY

Explanation

The Analyze process encountered an error in the segment at RBA xxxxxxxx while checking the PCF/PCL/SSPTR pointer inter-dependencies.

User response

Make sure that the pointer values that are contained in the segment at RBA xxxxxxxx meet the following criteria:

- If the PCF pointer value is zero, the associated PCL pointer value and subset pointer value must also be zero.
- If the PCF pointer value is not zero, the associated PCL pointer value must also be other than zero.

Correct any errors and, if necessary, rerun the job.

0108 **SEGMENT AT RBA: xxxxxxxx**
(SEGCD: yy) HAS AN INCORRECT
SDEP POINTER VALUE

Explanation

The SDEP pointer value that is contained in the segment at RBA xxxxxxxx does not point to the SDEP part of the DEDB area.

User response

Check the address that is pointed to by the SDEP segment. If the address is in the RAA portion or outside of the DEDB area, correct the error and rerun the job.

If you received this message after extending the IOVF portion with the FPO OAE utility, this message might

not indicate a problem. When you use the OAE utility to extend the IOVF portion of a DEDB area, the utility does not update the SDEP pointer value of each segment. Therefore, after the IOVF extension, some SDEP pointers might remain pointing to the address in the extended IOVF portion that was formerly in the SDEP portion. If you run the FPA Analyze function against such a DEDB and the function detects SDEP pointers pointing to the IOVF portion, FPA issues pointer error messages. However, you can ignore such pointer error messages because such pointers do not represent problems. By using the SDEP_PTRIOVF_MSGLVL keyword, you can suppress such pointer error messages or change them to informational messages (messages with the I suffix). For more information about the SDEP_PTRIOVF_MSGLVL keyword, see "SDEP_PTRIOVF_MSGLVL keyword" on page 528.

0109 **CI AT RBA: xxxxxxxx CONTAINS A**
NEGATIVE SEGMENT/FSE LENGTH
- CI BYPASSED

Explanation

The Analyze process encountered a segment or an FSE with a negative length during the serial deblocking of the CI at RBA xxxxxxxx.

User response

Correct any errors, and rerun the job. If this situation persists, report it to system operation personnel.

0110 **CI AT RBA: xxxxxxxx CONTAINS**
AN INCORRECT SEGMENT/FSE
LENGTH
- CI BYPASSED

Explanation

The Analyze process encountered a segment or an FSE with an incorrect length during the serial deblocking of the CI at RBA xxxxxxxx.

User response

Correct any errors, and rerun the job. If this situation persists, report it to system operation personnel.

0111 **SPACE MAP AT RBA: xxxxxxxx**
OFFSET: yyyy HAS CONTROL
WORD DISCREPANCY

Explanation

The CI that corresponds to the space map at RBA xxxxxxxx OFFSET yyyy should be the first allocatable CI (that is, 1 byte from OFFSET yyyy should be x'80').

However, the space map specifies that the CI is already allocated (that is, 1 byte from OFFSET yyyy is x'40').

User response

Correct the errors, and rerun the job.

0112 **CI AT RBA: xxxxxxxx HAS UOW# DISCREPANCY**
- IOVF SPACE MAP: yyyyyyyy and IOVF CI PREFIX: zzzzzzzz

Explanation

The UOW number yyyyyyyy in the space map that corresponds to the CI at RBA xxxxxxxx and the number zzzzzzzz in the IOVF CI prefix at RBA xxxxxxxx are different.

User response

Correct the errors, and rerun the job.

0113 **SEGMENT AT RBA: xxxxxxxx (SEGCD: yy) HAS A PTF OR AN SPTF PTR THAT POINTS TO ITS OWN SEGMENT**

Explanation

The Analyze process encountered an IMS pointer chain discrepancy. The segment at RBA xxxxxxxx is pointed to by its own PTF pointer.

User response

Correct the errors, and rerun the job.

0114 **INCORRECT IMSID IN SDEP CI AT RBA: xxxxxxxx**
- CI BYPASSED

Explanation

The Analyze process encountered an incorrect IMSID in the SDEP CI prefix.

User response

Correct the errors, and rerun the job.

0115 **CI FORMATTED BY CREATE UTILITY IS ENCOUNTERED, CI WAS SKIPPED RBA OF CI: nnnnnnnn**

Explanation

This message is informational. The Analyze process encountered a CI that contains dummy data that was formatted by the DEDB area data create utility. The dummy data CI that is specified by the system ID CREATFMT in the system name that is stored in the SDEP CI prefix was formatted during the copy phase of the DEDB area data set create utility. The Analyze process skipped the dummy data CI.

User response

None. This message is informational.

0116 **TOTAL FSE LENGTH DISCREPANCY**
- RBA: xxxxxxxx

Explanation

The Analyze process detected that the total free space in the CI at RBA xxxxxxxx, as calculated by tracking the FSE chain, did not correspond to the value that was calculated during the serial deblocking of the CI. This condition might have been caused by one of incorrect FSE chain, FSE, or segment length.

User response

Correct the errors, and rerun the job.

0117 **TOTAL NO. OF FSE DISCREPANCY**
- RBA: xxxxxxxx

Explanation

The Analyze process determined that the number of FSE elements in the CI at RBA xxxxxxxx, which was calculated by tracking the FSE chain, did not match the value that was calculated during the serial deblocking of the CI. This condition might have been caused by one of an incorrect FSE chain, FSE, or segment length.

User response

Correct the errors, and rerun the job.

0118 **FSE CHAIN POINTS TO A NON FSE**
- RBA: xxxxxxxx

Explanation

The Analyze process encountered an incorrect FSE while tracking the FSE chain for the CI at RBA xxxxxxxx.

User response

Correct the errors, and rerun the job.

0119 **FSE CHAIN OFFSET ERROR**
- RBA: xxxxxxxx

Explanation

The FSE pointer value in the CI at RBA xxxxxxxx does not point to its own CI.

User response

Correct the errors, and rerun the job.

0120 **INCORRECT RBA VALUE IN CI AT**
RBA: xxxxxxxx (RBA VALUE:
yyyyyyyy OFFSET: zzzz)

Explanation

The Analyze process detected that the indicated RBA value (yyyyyyyy), which is at offset zzzz of a CI at RBA xxxxxxxx, is incorrect.

User response

Run the FPA Unload function and the FPA Reload function for the area to fix the incorrect FPA value in the CI.

0121 **ERROR IN THE FIRST DOVF CI.**
THE RBA OF THE CURRENT
OVERFLOW CI IS NULL. RBA: rba,
UOW#: nnnnn

Explanation

The pointer value of the first DOVF CI, which is identified by the indicated RBA and UOW number, is null.

User response

The pointer must be corrected. Reorganize the area by using the Online Expert Reorganization utility or the High-Speed DEDB Direct Reorganization utility.

0122 **SPACE BYTES NOT ZERO. RBA:**
xxxxxxx OFFSET: yyyy
- CI BYPASSED

Explanation

The free space in the CI at RBA xxxxxxxx offset at yyyy contains a non-zero value.

User response

Correct the errors, and rerun the job.

0123 **THE RBA OF THE CURRENT**
OVERFLOW CI IN THE FIRST DOVF
CI IS NULL. RBA: rba, UOW#:
nnnnn

Explanation

The DOVFNULLPTR_MSGLVL=WARNING option is enabled and the Analyze process detected a null pointer (RBA) value for the current overflow CI in the indicated DOVF CI. The RBA and the UOW number in the message identify the first DOVF CI that contains the null pointer value.

User response

To update the RBA value in the first DOVF CI, use the Online Expert Reorganization utility or the High-Speed DEDB Direct Reorganization utility and reorganize the area.

0124 **SEGMENT DATA EXISTS IN THE**
AREA BUT THE AREA IS FLAGGED
AS EMPTY IN THE DMAC. SEG AT
RBA: xxxxxxxx

Explanation

The empty area flag DMACEMPT is set in the DMAC but segment data exists in the area data set.

User response

Correct the error and then rerun the job.

0125 **SDEP CI FREESPACE OFFSET**
DOES NOT POINT TO AN ADDRESS
AFTER THE LAST SEGMENT IN CI.
RBA: xxxxxxxx - CI BYPASSED
- SDEP CI FREESPACE OFFSET:
yyyy
- OFFSET OF THE BYTE AFTER THE
LAST SEGMENT: zzzz

Explanation

The pointer checking process found an invalid free space offset value in the SDEP CI. The free space offset of the SDEP CI should point to an address after the last valid SDEP segment.

User response

Correct the error and then rerun the job.

0126 **FREE CI IS MARKED AS**
ALLOCATED IN THE IOVF SPACE
MAP CI.

- **RBA OF IOVF SPACE MAP CI:**
xxxxxxx
- **RBA OF TARGET IOVF CI:**
xxxxxxx
- **SPACE MAP ELEMENT OFFSET:**
xxxx
- **SPACE MAP ELEMENT:** vvvvvvvv
- **CI PREFIX:** vvvvvvvvvvvvvvvv

Explanation

The Analyze process found an IOVF CI that is not used by any UOW but is marked as allocated in the IOVF space map CI.

User response

Specify CISNAP=YES to generate the Pointer Chain Reconstruction report and the CI Map/CI Dump report. Correct the error and then rerun the job.

0127 **POSSIBLE LOOP IN THE FREE CHAIN IN THE IOVF SPACE MAP CI.**
 - **RBA OF IOVF SPACE MAP CI:**
xxxxxxx
 - **CALCULATED MAXIMUM ELEMENTS IN CHAIN:** nnn

Explanation

The Analyze process found a possible loop. It found too many elements in a free chain in the IOVF space map CI.

User response

Specify CISNAP=YES to generate the Pointer Chain Reconstruction report and the CI Map/CI Dump report. Correct the error and then rerun the job.

0128 **INFORMATION: SEGMENT AT RBA xxxxxxxx (SEGCD: nn) HAS AN SDEP POINTER THAT POINTS TO THE IOVF PORTION**

Explanation

The Analyze process detected an SPCF or SPTF pointer that points to the IOVF portion in the segment at RBA xxxxxxxx. The Analyze process does not regard this as a pointer error because SDEP_PTRIOVF_MSGLVL=INFORMATIONAL is specified.

User response

None. This message is informational.

0129 **THE NUMBER OF INFORMATIONAL MESSAGES EXCEEDED THE LIMIT**

Explanation

The number of informational messages issued in the Resolve Pointers report exceeded 100. No further informational messages are written to the Resolve Pointers report.

User response

None. This message is informational.

0301 **SEGMENT DATA MISSING FOR UOW#: xxxxxxxx**

Explanation

Pointer records were processed for segments in the specified UOW but no segment information records were found.

User response

Correct the errors, and rerun the job.

0302 **SEG AT RBA: xxxxxxxx (SEGCD: yy) HAS AN INCORRECT PCL/SSPTR THAT REFERS TO SEG AT RBA: zzzzzzzz**

Explanation

The Analyze process determined that the segment at RBA xxxxxxxx contains an incorrect RBA value of zzzzzzzz for a PCL or SSPTR. The pointer value zzzzzzzz refers to a segment that does not exist.

User response

Correct the errors, and rerun the job.

0303 **SEG AT RBA: xxxxxxxx (SEGCD: yy) HAS A PCL/SSPTR THAT MAY POINT TO A WRONG SEG TYPE AT RBA: zzzzzzzz**

Explanation

The Analyze process determined that the PCL or SSPTR that was contained in the segment at RBA xxxxxxxx might be pointing to a wrong segment type at RBA zzzzzzzz.

User response

Correct the errors, and rerun the job.

0304 **SEGMENT AT RBA: xxxxxxxx
(SEGCODE: yy) IS POINTED TO BY
MULTIPLE SEGMENTS
- POINTER IN SEGMENT AT RBA:
xxxxxxx (SEGCODE: yy)**

Explanation

The Analyze process encountered an IMS pointer chain discrepancy. The segment at RBA xxxxxxxx is pointed to by multiple PCF/PTF pointers.

User response

Correct the errors, and rerun the job.

0305 **SEGMENT AT RBA: xxxxxxxx
POINTS TO NON-EXISTENT
SEGMENT (SEGCODE: yy) AT RBA:
zzzzzzzz**

Explanation

The Analyze process encountered an incorrect IMS pointer. No segment is found at RBA yyyyyyyy.

User response

Correct the errors, and rerun the job.

0306 **SEGMENT AT RBA: xxxxxxxx IS AN
ORPHAN**

Explanation

The Analyze process encountered a segment in the database that is not pointed to by another segment or a RAP.

User response

Correct the errors, and rerun the job.

0307 **SEG AT RBA: xxxxxxxx (SEGCD: yy)
HAS A PTF/PCF PTR THAT POINTS
TO A WRONG SEG TYPE AT RBA:
zzzzzzzz**

Explanation

The Analyze process determined that the PTF or the PCF that is contained in the segment at RBA xxxxxxxx is pointing to a wrong segment type at RBA zzzzzzzz.

User response

Correct the errors, and rerun the job.

0308 **SEGMENT AT RBA: xxxxxxxx
(SEGCODE: yy) IS NOT IN RAP
CHAIN
- POINTED TO BY SEGMENT AT
RBA: zzzzzzzz**

Explanation

The Analyze process encountered an IMS pointer chain discrepancy.

User response

Correct the errors, and rerun the job.

0309 **[ROOT|SDEP] SEGMENT AT RBA:
xxxxxxx CONTAINS AN
INCORRECT SDEP POINTER
(CYCLE# REL-BYTE-ADDR)**

Explanation

The SDEP pointer that is contained in the segment at RBA xxxxxxxx points to a nonexistent segment, which is indicated by *CYCLE# REL-BYTE-ADDR*.

User response

Correct the errors, and rerun the job.

0310 **SDEP SEGMENT AT RBA: xxxxxxxx
IS AN ORPHAN**

Explanation

The SDEP segment at RBA xxxxxxxx is not pointed to by another SDEP segment or a ROOT segment.

User response

Correct the errors, and rerun the job.

0311 **SEG AT RBA: xxxxxxxx (SEGCD: yy)
IS LAST IN THE TWIN CHAIN BUT
NOT POINTED TO BY ITS
PARENT'S PCL POINTER**

Explanation

The Analyze process encountered a segment in the database that is last in the twin chain but not pointed to by its parent's PCL pointer.

User response

Correct the errors, and rerun the job.

0312 **SEGMENT AT RBA: xxxxxxxx
(SEGCD: yy) HAS A POINTER THAT
POINTS TO ITS OWN SEGMENT**

Explanation

The Analyze process encountered an IMS pointer chain discrepancy. The segment at RBA xxxxxxxx is pointed to by its own PTF pointer.

User response

Correct the errors, and rerun the job.

0313 **HIGHEST (INCORRECT) SDEP
POINTER VALUE IS
ENCOUNTERED: CYCLE# REL-
BYTE-ADDR**

Explanation

One or more SDEP pointer errors were detected, and the cycle count and the RBA of the highest incorrect SDEP pointer are specified by *CYCLE#* and *REL-BYTE-ADDR*.

User response

Correct the errors, and rerun the job.

0314 **THE NUMBER OF SDEP ORPHAN
MESSAGES EXCEEDED THE LIMIT**

Explanation

The number of SDEP orphan messages has exceeded the limit. Excess messages were not put in the Resolve Pointers report.

User response

Correct the errors, and rerun the job.

0315 **SEQUENCE ERROR FOUND IN A
TWIN CHAIN.
- SOURCE SC nn NAME segname
RBA: xxxxxxxx
- TARGET SC nn NAME segname
RBA: xxxxxxxx
- PREVIOUS TWIN KEY VALUE:
key_value (HEX)
- CURRENT TWIN KEY VALUE:
key_value (HEX)**

Explanation

The pointer checking process found a key sequence error in the indicated segment twin chain.

User response

Specify *CISNAP=YES* to generate the Pointer Chain Reconstruction report and the CI Map/CI Dump report. Correct the error and then rerun the job.

Process Summary of Analysis report messages

The following information is about messages and codes that are generated in the Process Summary of Analysis report and the TUNED Process Summary of Analysis report.

0500I **PROCESSING STARTED FOR AREA
NO: nnnn, AREANAME: areaname**

Explanation

The analysis report process for the area number *nnnn*, area *areaname* has started.

User response

None. This message is informational.

0501I **PROCESSING ENDED NORMALLY
FOR AREA NO: nnnn, AREANAME:
areaname**

Explanation

The analysis report process for the area number *nnnn*, area *areaname* has completed successfully.

User response

None. This message is informational.

0502W **PROCESSING ENDED WITH
WARNINGS FOR AREA NO: nnnn,
AREANAME: areaname**

Explanation

The analysis report process for the area number *nnnn*, area *areaname* has completed, but some warning messages that were issued.

User response

For additional descriptions of the warning conditions, refer to the warning messages that were issued prior to this message.

0503E **PROCESSING ENDED WITH
ERRORS FOR AREA NO: nnnn,
AREANAME: areaname**

Explanation

The analysis report process for the area number *nnnn*, area *areaname* has completed, but some error messages that were issued.

User response

For additional descriptions of the error conditions, refer to the error messages that were issued prior to this message.

0504I	NUMBER OF UOW'S TO BE REORG- ED: <i>xx,xxx</i>
--------------	---

Explanation

This message is issued when the analysis report process has generated and written the REORG Utility control statements to the file that is associated with the HFPARERG DD statement. If *xx,xxx* is zero, no control statements are generated by the analysis report process.

User response

None. This message is informational.

0505E	COLUMN: <i>xx</i>, UNKNOWN KEYWORD
--------------	---

Explanation

The analysis report process detected an incorrect keyword while parsing the control statement that you specified in the HFPAPARM DD.

User response

Correct the control statement, and rerun the job.

0506E	"RBASEFS=" VALUE IS INCORRECT
--------------	--

Explanation

The analysis report process detected an incorrect RBASEFS= specification while parsing the control statement that you specified in the HFPAPARM DD.

User response

Correct the control statement, and rerun the job.

0507E	"RDOVFFS=" VALUE IS INCORRECT
--------------	--

Explanation

The analysis report process detected an incorrect RDOVFFS= specification while parsing the control statement that you specified in the HFPAPARM DD.

User response

Correct the control statement, and rerun the job.

0508E	"UDOVFFS=" VALUE IS INCORRECT
--------------	--

Explanation

The analysis report process detected an incorrect UDOVFFS= specification while parsing the control statement that you specified in the HFPAPARM DD.

User response

Correct the control statement, and rerun the job.

0509E	"SYNTBL=" VALUE IS INCORRECT
--------------	-------------------------------------

Explanation

The analysis report process detected an incorrect SYNTBL= specification while parsing the control statement that you specified in the HFPAPARM DD.

User response

Correct the control statement, and rerun the job.

0510W	THRESHOLD EXCEEDED: <i>descriptive message</i> THRESHOLD: <i>value1</i> ACTUAL: <i>value2</i>
--------------	--

Explanation

The analysis report process detected a situation in which the threshold value that you specified has been exceeded. The *descriptive message* in the text describes the conditions and the associated values. The *descriptive message* can be one of the following indicators:

- FS RELATIONSHIP
(RAA > *xx%* AND DOVF < *yy%*)
- FS RELATIONSHIP
(RAA > *xx%* AND IOVF < *yy%*)
- FS RELATIONSHIP
(DOVF > *xx%* AND IOVF < *yy%*)
- % FS RAA (< *xxx%*)
- % FS DOVF (< *xxx%*)
- % FS IOVF (< *xxx%*)

- RAP CI OVFL USAGE (> xx%)
- UOW'S USING DOVF (> xx%)
- UOW'S USING IOVF (> xx%)
- % USED IOVF CI'S (> xx%)
- AVG. IOVF CI'S USED (> xx.x)
- MAX. IOVF CI'S USED (> xx)
- AVG. SYN LENGTH (> xx.x)
- MAX. SYN LENGTH (> xx)
- AVG. ROOT I/O (> xx.x)
- MAX. ROOT I/O (> xx)
- AVG. RECORD I/O (> xx.x)
- MAX. RECORD I/O (> xx)
- % RECS. USING IOVF (> xx%)
- SYN LENGTH (xx% > LENGTH yy)
- LONG SYNONYM CHAINS (xx% > SYNONYM yy)

User response

Check the threshold exceptions that were reported by this message and other 0510W messages. If you determine that any of those exceptions need to be addressed, follow the guidelines in the topic "Analyzing and tuning IMS data entry databases" in the *IMS Solution Packs IMS Database Space Tuning Guide*.

0512E INTERNAL WORK FILE OPEN ERROR

Explanation

The OPEN processing for the internal work file that is associated with the analysis report process failed.

User response

Contact IBM Software Support.

0513E INPUT DATA ERROR
- EXPECTING RECORD TYPE: "a,b"

Explanation

The analysis report process was expecting a record type *a,b*, but the record that was read was not the correct type.

User response

Contact IBM Software Support.

0514E INSUFFICIENT STORAGE FOR:
aaaa
- INCREASE REGION SIZE

Explanation

The analysis report process issued a GETMAIN macro to get storage for *aaaa*. The return code shows that the attempt was unsuccessful.

User response

If the specified region size is too small, increase the REGION size on the JOB statement in the JCL, and rerun the utility.

0524E "ARDIGIT=" VALUE IS INCORRECT

Explanation

The analysis report process detected an incorrect ARDIGIT= specification while parsing the control statement that you specified in the HFPAPARM DD.

User response

Correct the control statement, and rerun the job.

0525E "RGROUP=" VALUE IS INCORRECT

Explanation

The analysis report process detected an incorrect RGROUP= specification while parsing the control statement that you specified in the HFPAPARM DD.

User response

Correct the control statement, and rerun the job.

0532E [HFPAUOWC | FPXAUOWC] FILE ERRORS.
CARD: "xxxxxxxxxxxxxxxxxxxxxxxxxxxx"

Explanation

One or more errors in one of the HFPAUOWC or FPXAUOWC control cards are detected.

User response

Correct the HFPAUOWC or FPXAUOWC control card, and rerun the job.

0533W [HFPAUOWC | FPXAUOWC] CONTROL CARDS WERE SPECIFIED FOR THE AREAS THAT ARE NOT PROCESSED. THESE CONTROL CARDS WERE IGNORED.

Explanation

One or more HFXAUOWC or FPXAUOWC control cards that specify the areas that are not processed were found. These control cards are ignored.

User response

Ensure that this situation is not an error condition. If it is an error condition, specify the correct file, and rerun the job.

0563E **CRITICAL CONTROL CARD ERROR ENCOUNTERED**

Online Pointer Scan report messages

The following information is about messages and codes that are generated in the Online Pointer Scan report.

For the format of the Online Pointer Scan report, see “Online Pointer Scan report” on page 704.

1001 **Invalid pointer RBA found in the following segment:**
Source SC nn Name segname1 RBA x'rba1'
Target SC mm Name segname2 RBA x'rba2'
Root SC ll Name segname3 RBA x'rba3'
Root key value: root_key_value
Content of the source segment (up to 255 bytes): content

Explanation

The pointer checking process found a segment that contains an invalid RBA value. The RBA value in the source segment is invalid.

User response

Correct the error and then rerun the job.

1002 **Invalid block type found in the RAP CI prefix.**
Source SC nn Name segname1 RBA x'rba1'
Target SC mm Name segname2 RBA x'rba2'

Explanation

The pointer checking process found an invalid block type (DBLKBTID) in the CI that is identified by *rba1*.

User response

Correct the error and then rerun the job.

Explanation

A critical control card error encountered.

User response

See the other messages generated for further information about the error. Correct the control statement. Rerun the job.

1003 **Data at pointer RBA is not the expected segment code.**
Source SC nn Name segname1 RBA x'rba1'
Target SC mm Name segname2 RBA x'rba2'
Root SC ll Name segname3 RBA x'rba3'
Root key value: root_key_value
Content of the source segment (up to 255 bytes): content
Data at target RBA: content

Explanation

The pointer checking process found an incorrect segment code during the serial deblocking of the CI. *rba1* indicates the CI in which the error was found.

User response

Correct the error and then rerun the job.

1004 **Invalid physical child last pointer found.**
Source SC nn Name segname1 RBA x'rba1'
Target SC mm Name segname2 RBA x'rba2'
Root SC ll Name segname3 RBA x'rba3'
Root key value: root_key_value
Content of the source segment (up to 255 bytes): content

Explanation

The pointer checking process found an incorrect physical child last (PCL) pointer.

User response

Correct the error and then rerun the job.

1005 **Root segment contains invalid RBA for a SDEP segment.**
Source SC nn Name segname1 RBA x'rba1'
Target SC mm Name segname2 RBA x'rba2'
Root key value: root_key_value
Content of the source segment (up to 255 bytes): content

Explanation

The pointer checking process found an invalid RBA in the root segment. The RBA does not point to a segment in the SDEP area.

User response

Check the address that is pointed to by the SDEP segment. If the address is in the RAA portion or outside of the DEDB area, correct the error and rerun the job.

If you received this message after extending the IOVF portion with the FPO OAE utility, this message might not indicate a problem. When you use the OAE utility to extend the IOVF portion of a DEDB area, the utility does not update the SDEP pointer value of each segment. Therefore, after the IOVF extension, some SDEP pointers might remain pointing to the address in the extended IOVF portion that was formerly in the SDEP portion. If you run the OPC utility against such a DEDB and the utility detects SDEP pointers pointing to the IOVF portion, OPC issues pointer error messages. However, you can ignore such pointer error messages because such pointers do not represent problems. By using the SDEP_PTRIOVF_MSGLVL keyword, you can suppress such pointer error messages or change them to informational messages (messages with the I suffix). For more information about the SDEP_PTRIOVF_MSGLVL keyword, see "SDEP_PTRIOVF_MSGLVL keyword" on page 915.

1006 **Sequence error found in a dependent segment twin chain.**
Source SC nn Name segname1 RBA x'rba1'
Target SC mm Name segname2 RBA x'rba2'

Previous twin key value:
twin_key_value1
Current twin key value:
twin_key_value2

Explanation

The pointer checking process found a key sequence error in the indicated dependent segment twin chain.

User response

Correct the error and then rerun the job.

1007 **Invalid subset pointer found.**
Source SC nn Name segname1 RBA x'rba1'
Target SC mm Name segname2 RBA x'rba2' <- Invalid pointer value.
Root SC ll Name segname3 RBA x'rba3'
Root key value: root_key_value
Content of the source segment (up to 255 bytes): content
SSP number of the incorrect pointer: ssp_number

Explanation

The pointer checking process found an invalid subset pointer.

User response

Correct the error and then rerun the job.

1008 **IOVF CI is referenced by more than one unit of work.**
Source SC nn Name segname1 RBA x'rba1'
Target SC mm Name segname2 RBA x'rba2'
Root SC ll Name segname3 RBA x'rba3'
Root key value: root_key_value

Explanation

The pointer checking process found an IOVF CI that is referenced by more than one unit of work (UOW). An IOVF CI can be referenced only by one UOW.

User response

Correct the error and then rerun the job.

1009 **SDEP CI has a format earlier than IMS Version 6.**

RBA of CI: *rba*
CI prefix: *ci_prefix*

Root key value: *root_key_value*
Source RAP is at RBA x'rba2' in
UOW number *uow_number*
CI prefix: *ci_prefix*

Explanation

The pointer checking process found an SDEP CI whose format is older than IMS 6.1.

User response

Correct the error and then rerun the job.

1010 **Invalid SDEP CI prefix found.**
RBA of CI: *rba*
CI prefix: *ci_prefix*

Explanation

The pointer checking process found an invalid block type (DBLKBTID) in the SDEP CI.

User response

Correct the error and then rerun the job.

1012 **In-doubt SDEP segment detected.**
Target SC *nn* Name *segname1* RBA
x'rba'
Content of the source segment (up
to 255 bytes): *content*

Explanation

The pointer checking process found an in-doubt SDEP segment.

User response

Correct the error and then rerun the job.

1013 **SDEP CI freespace offset does not**
point to an address after the last
segment in CI.
Target SC *nn* Name *segname1* RBA
x'rba'
CI prefix: *ci_prefix*

Explanation

The pointer checking process found an invalid free space offset value in the SDEP CI.

User response

Correct the error and then rerun the job.

1014 **IOVF CI is not owned by the**
current UOW.
Target SC *nn* Name *segname1* RBA
x'rba'

Explanation

The pointer checking process found an invalid IOVF CI. The IOVF CI is being used by the current UOW, but the UOW number assigned to this IOVF CI is not of the current UOW.

User response

Correct the error and then rerun the job.

1015 **Segment checksum error in UOW**
starting at RBA *rba*
Segment code: *nn* **Segment name:**
***segname1* Checksum value:** *value*

Explanation

The pointer checking process found a segment checksum error in the UOW.

User response

Correct the error and then rerun the job.

1016 **Invalid CIDF/RDF found in the CI**
suffix.
RBA of CI: *rba*
Hex offset in CI: *offset*

Explanation

The pointer checking process found an invalid CIDF or RDF value in the CI suffix.

User response

Correct the error and then rerun the job.

1018 **RBA in the CI suffix does not**
match the CI RBA.
RBA of CI: *rba*
Hex offset in CI: *offset*

Explanation

The pointer checking process found an invalid RBA value in the CI suffix.

User response

Correct the error and then rerun the job.

1019 **CI type mismatch. CI type is not**
RAA or DOVF.

RBA of CI: *rba*
Hex offset in CI: *offset*

Explanation

The pointer checking process found an invalid CI. The expected CI type is either a RAA CI or a DOVF CI, but the actual CI type is different.

User response

Correct the error and then rerun the job.

1020	CI type mismatch. CI type is not IOVF. RBA of CI: <i>rba</i> Hex offset in CI: <i>offset</i>
-------------	---

Explanation

The pointer checking process found an invalid CI. The expected CI type is IOVF, but the actual CI type is different.

User response

Correct the error and then rerun the job.

1021	IOVF CI ownership error. RBA of IOVF space map CI: <i>rba1</i> RBA of target IOVF CI: <i>rba2</i> Space map element offset: <i>offset</i> Space map element: <i>element</i> CI prefix: <i>ci_prefix</i>
-------------	--

Explanation

The pointer checking process found an inconsistency between the UOW number in the IOVF space map CI and the UOW number in the IOVF CI.

User response

Correct the error and then rerun the job.

1022	CI type is invalid or the current RBA is not in the SDEP range. RBA of CI: <i>rba</i>
-------------	--

Explanation

The pointer checking process found an SDEP CI that is not in the SDEP range.

User response

Correct the error and then rerun the job.

1023	Suspected loop in FSEs. Too many FSEs found in the CI.
-------------	---

RBA of CI: *rba*

Explanation

The pointer checking process found more than the allowable number of FSEs in the CI.

User response

Correct the error and then rerun the job.

1024	Invalid FSE found. Flag not set or a length error. RBA of CI: <i>rba</i> Hex offset in CI: <i>offset</i> The remainder of the CI is skipped
-------------	--

Explanation

The pointer checking process found an FSE whose flag is not set, or an FSE whose length is invalid.

User response

Correct the error and then rerun the job.

1025	Bad scrap bytes detected in the CI. RBA of CI: <i>rba</i> Hex offset in CI: <i>offset</i>
-------------	--

Explanation

The pointer checking process found bad scrap bytes in the CI.

User response

Correct the error and then rerun the job.

1026	Offset points to an area beyond the CI end. RBA of CI: <i>rba</i> Hex offset in CI: <i>offset</i> The remainder of the CI is skipped
-------------	---

Explanation

The pointer checking process found an FSE that has an offset value pointing to an area beyond the end of the CI.

User response

Correct the error and then rerun the job.

1027	Space bytes not zero. RBA of CI: <i>rba</i> Hex offset in CI: <i>offset</i>
-------------	--

Explanation

The pointer checking process found an FSE whose space bytes are not filled with zeros.

User response

Correct the error and then rerun the job.

1028	Invalid O2/SDEP segment found. RBA of CI: <i>rba</i> Hex offset in CI: <i>offset</i> The remainder of the CI is skipped
-------------	--

Explanation

The pointer checking process found an SDEP segment. SDEP segments are not allowed in this database because they are not defined in the DBD.

User response

Correct the error and then rerun the job.

1029	Invalid segment length. RBA of CI: <i>rba</i> Hex offset in CI: <i>offset</i> The remainder of the CI is skipped
-------------	---

Explanation

The pointer checking process found a segment with invalid length in the CI.

User response

Correct the error and then rerun the job.

1030	Invalid FSEs in CI, checksum error. RBA of CI: <i>rba</i> Hex offset in CI: <i>offset</i>
-------------	--

Explanation

The pointer checking process found an FSE checksum error in the CI.

User response

Correct the error and then rerun the job.

1031	CI size does not match the total length of CI fields. RBA of CI: <i>rba</i>
-------------	--

Explanation

The pointer checking process found a CI whose size is not equal to the total length of CI prefix, CI suffix, FSEs, scrap bytes, and segments in the CI.

User response

Correct the error and then rerun the job.

1032	CI type identifier of the IOVF space map CI is incorrect. RBA of IOVF space map CI: <i>rba</i> CI prefix: <i>ci_prefix</i>
-------------	---

Explanation

The pointer checking process found incorrect CI type identifier in the IOVF space map CI. The expected CI type is IOVF space map CI, but the actual CI type is different.

User response

Correct the error and then rerun the job.

1033	CUSN bit is not set in the IOVF space map CI. RBA of IOVF space map CI: <i>rba</i> CI prefix: <i>ci_prefix</i>
-------------	---

Explanation

The pointer checking process found an IOVF space map CI whose CI Update Sequence Number (CUSN) bit is not set in the CI prefix.

User response

Correct the error and then rerun the job.

1034	Invalid free chain pointer in the IOVF space map CI. RBA of IOVF space map CI: <i>rba</i> Space map element offset: <i>offset</i> Space map element: <i>element</i>
-------------	--

Explanation

The pointer checking process found an invalid free chain pointer in the IOVF space map CI.

User response

Correct the error and then rerun the job.

1035	Invalid element in the free chain pointer in the IOVF space map CI. RBA of IOVF space map CI: <i>rba</i> Space map element offset: <i>offset</i>
-------------	---

Space map element: *element*

Explanation

The pointer checking process found a free chain pointer that points to an IOVF CI used by a UOW.

User response

Correct the error and then rerun the job.

1036 Possible loop in the free chain in the IOVF space map CI.
RBA of IOVF space map CI: *rba*
Calculated maximum elements in chain: *nnn*

Explanation

The pointer checking process found a possible loop. It found too many elements in a free chain in the IOVF space map CI.

User response

Correct the error and then rerun the job.

1038 Free CI is marked as allocated in the IOVF space map CI.
RBA of IOVF space map CI: *rba1*
RBA of target IOVF CI: *rba2*
Space map element offset: *offset*
Space map element: *element*
CI prefix: *ci_prefix*

Explanation

The pointer checking process found an IOVF CI that is not used by any UOW but is marked as allocated in the IOVF space map CI.

User response

Correct the error and then rerun the job.

1039 Invalid element in the IOVF space map CI.
RBA of IOVF space map CI: *rba*
Space map element offset: *offset*
Space map element: *element*

Explanation

The pointer checking process found an invalid element in the IOVF space map CI. The element is neither a free element nor a used element.

User response

Correct the error and then rerun the job.

1040 Orphan free element in the IOVF space map CI.
RBA of IOVF space map CI: *rba1*
RBA of target IOVF CI: *rba2*
Space map element offset: *offset*
Space map element: *element*

Explanation

The pointer checking process found an orphan element in the IOVF space map CI.

User response

Correct the error and then rerun the job.

1041 Segment code is zero or invalid.
RBA of CI: *rba*
Hex offset in CI: *offset*

Explanation

The pointer checking process found an invalid segment code in the CI.

User response

Correct the error and then rerun the job.

1042 Segment has invalid [SPCF | SPTF] pointer value.
Source SC *nn* Name *segname1* RBA *x'rba'*
Pointer value: *value*

Explanation

The pointer checking process found an invalid SPCF or SPTF pointer value.

User response

Check the address that is pointed to by the SDEP segment. If the address is in the RAA portion or outside of the DEDB area, correct the error and rerun the job.

If you received this message after extending the IOVF portion with the FPO OAE utility, this message might not indicate a problem. When you use the OAE utility to extend the IOVF portion of a DEDB area, the utility does not update the SDEP pointer value of each segment. Therefore, after the IOVF extension, some SDEP pointers might remain pointing to the address in the extended IOVF portion that was formerly in the SDEP portion. If you run the OPC utility against such a

DEDB and the utility detects SDEP pointers pointing to the IOVF portion, OPC issues pointer error messages. However, you can ignore such pointer error messages because such pointers do not represent problems. By using the SDEP_PTRIOVF_MSGLVL keyword, you can suppress such pointer error messages or change them to informational messages (messages with the I suffix). For more information about the SDEP_PTRIOVF_MSGLVL keyword, see “SDEP_PTRIOVF_MSGLVL keyword” on page 915.

1043 **Logically deleted SDEP segment is flagged as in-doubt.**
Target SC nn Name segname1 RBA x'rbal' TIMESTAMP x'time'
Content of the source segment (up to 255 bytes): content

Explanation

The pointer checking process found a logically deleted SDEP segment that is flagged as an in-doubt SDEP segment.

User response

Correct the error and then rerun the job.

1045 **Segment data exists in the area but the area is flagged as empty in the DMAC.**
RBA of CI: xxxxxxxx
Hex offset in CI: xxxxxxxx
The remainder of the CI is skipped

Explanation

The empty area flag DMACEMPT is set in the DMAC, but segment data exists in the area data set.

User response

Correct the error and then rerun the job.

Gathering diagnostic information

Before you report a problem with IMS HP Fast Path Utilities to IBM Software Support, you need to gather the appropriate diagnostic information.

Provide the following information for all IMS HP Fast Path Utilities problems:

- A clear description of the problem and the steps that are required to re-create the problem
- The version of IMS that you are using and the version of the operating system that you are using
- A complete log of the job
- Snap dump generated in the HFPABEND data set

The HFPABEND data set is generated only when the FPA process ends abnormally. If the HFPABEND DD is not specified in the JCL, FPA dynamically allocates the data set by using SYSOUT=*

1046

Information: The root segment contains an SDEP segment RBA that points to the IOVF portion.
Source SC nn Name segname1 RBA x'rbal'
Target SC mm Name segname2 RBA x'rbal2'

Explanation

The pointer checking process detected an SPCF pointer that points to the IOVF portion in the root segment. The pointer checking process does not regard this as a pointer error because SDEP_PTRIOVF_MSGLVL=INFORMATIONAL is specified.

User response

None. This message is informational.

1047

Information: A segment has an SPTF pointer value that points to the IOVF portion.
Source SC nn Name segname1 RBA x'rbal'
Pointer value: value

Explanation

The pointer checking process detected an SPTF pointer that points to the IOVF portion in the SDEP segment. The pointer checking process does not regard this as a pointer error because SDEP_PTRIOVF_MSGLVL=INFORMATIONAL is specified.

User response

None. This message is informational.

- A Load Module/Macro APAR Status report

For information about creating a Load Module/Macro APAR Status report, see [Chapter 51, “Diagnostics aid,”](#) on page 1433.

Chapter 51. Diagnostics aid

If you have a problem that you think is not a user error, you should run the diagnostics aid program to obtain the Load Module/Macro APAR Status report.

After you obtain the Load Module/Macro APAR Status report, attach it to the other diagnostic documents (such as job dump list or I/O of the utility), and contact IBM Software Support. The diagnostics aid generates a Load Module/Macro APAR Status report. This report shows the latest APAR fixes that were applied to each module and macro.

The diagnostics aid program is HFPUDIAG, which is the alias name of FABADIAG. You can invoke the diagnostics aid program by using either HFPUDIAG or FABADIAG.

Topics:

- [“How to run diagnostics aid with JCL” on page 1433](#)
- [“APAR status reports” on page 1434](#)

How to run diagnostics aid with JCL

To run the diagnostics aid program, supply an EXEC statement and a DD statement that defines the output data set.

Procedure

1. Specify the EXEC statement. It must be in the following form:

```
//stepname EXEC PGM=HFPUDIAG
```

2. Specify a DD statement.

DD statement	Description
STEPLIB	This statement defines the library that contains the HFPUDIAG program (usually <i>prefix.SHFPLMD0</i>).
SHPSLMD	This statement defines the library that contains the load modules (usually <i>prefix.SHFPLMD0</i>) for which you have a problem. The Load Module APAR Status report is not generated if this DD statement is not provided or if DD DUMMY is specified. It is recommended that you specify this DD statement.
SHPSMAC	This statement defines the library that contains the macros (usually <i>prefix.SHFPLMD0</i>) for which you have a problem. The Macro APAR Status report is not generated if this DD statement is not provided or if DD DUMMY is specified.
SYSPRINT	This output data set contains the Load Module/Macro APAR Status report. The data set contains 133-byte, fixed-length records. It can reside on a tape, direct-access device, or printer; or it can be routed through the output stream. If BLKSIZE is coded in the DD statement, it must be a multiple of 133. However, it is recommended that you use:

```
//SYSPRINT DD SYSOUT=A
```

3. Run the JCL.

APAR status reports

The diagnostics aid generates two reports for maintenance by IBM.

The generated reports are:

- Load Module APAR Status report
- Macro APAR Status report

Load Module APAR Status report

The Load Module APAR Status report contains information about the modules and their applied APARs.

This report contains the following information:

MODULE LIBRARY

This includes the data set names that were specified in the SHPSLMD DD statement. If more than 30 data sets are concatenated, only the first 30 data sets are listed.

MODULE NAME

This is the name of the load module member or the alias.

ALIAS-OF

This is the name of the original member of the alias. If the module name is not an alias, this field is left blank.

CSECT NAME

This is the name of the included CSECT in the module. The CSECT names are reported in the included order in the module.

APAR NUMBER

This is the latest APAR number that was applied to the module that is represented by the CSECT name. If no APAR is applied, NONE is shown.

APAR FIX-DATE

This is the date when the modification was prepared for the module that is represented by the CSECT name. If no APAR is applied, N/A is shown.

Notes:

1. If the CSECT name does not start with *FAB*, *HPS*, *HFP*, *FPX* or the program structure of the CSECT does not conform to the IMS HP Fast Path Utilities module standard to identify the APAR number and the APAR fixed date, the fields APAR NUMBER and APAR FIX-DATE are filled with asterisks (*).
2. If the load module is a member of the PDSE library, the following statement is shown on the report line, and the job completes with a return code of 4.

```
** IT CAN NOT BE ANALYZED DUE TO PDSE LIBRARY MEMBER **
```

3. If the load macro fails for a utility member, the following statement is shown on the report line, and the job completes with a return code of 8.

```
** IT CAN NOT BE ANALYZED DUE TO LOAD FAILED MEMBER **
```

Macro APAR Status report

The Macro APAR Status report contains information about macros and their applied APARs.

This report contains the following information:

MACRO LIBRARY

This includes the data set names that were specified in the SHPSMAC DD statement. If more than 30 data sets are concatenated, only the first 30 data sets are listed.

MACRO NAME

This is the name of the macro member or the alias.

ALIAS-OF

This is the name of the original member of the alias. If the macro name is not an alias, this field is left blank.

APAR NUMBER

This is the latest APAR number that was applied to the macro. If no APAR is applied, NONE is shown.

APAR FIX-DATE

This is the date when the modification was prepared for the macro. If no APAR is applied, N/A is shown.

Note: If the macro source statement structure does not conform to the IMS HP Fast Path Utilities macro standard to identify the APAR number and the APAR fixed date, the fields APAR NUMBER and APAR FIX-DATE are filled with asterisks (*).

Notices

This information was developed for products and services offered in the U.S.A.

This material may be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Director of Licensing
IBM Corporation
North Castle Drive

Armonk, NY 10504-1785
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Programming interface information

This information is intended to help the customer use the functional capabilities of IMS High Performance Fast Path Utilities of IBM IMS Fast Path Solution Pack for z/OS.

However, this information also documents Product-Sensitive programming interface information and associated guidance information.

Product-Sensitive programming interfaces are provided to allow the customer installation to perform tasks such as tailoring, monitoring, modification, or diagnosis of this IBM product. Use of such interfaces creates dependencies on the detailed design or implementation of the IBM product. Product-Sensitive interfaces should be used only for these specialized purposes. Because of their dependencies on detailed design and implementation, it is to be expected that programs written to such interfaces may need to be changed in order to run with new product releases or versions, or as a result of service.

Product-Sensitive programming interface information is explicitly identified where it occurs, as an introductory statement to a chapter or section that is entirely Product-Sensitive programming interface information.

Product-Sensitive Programming Interface and Associated Guidance Information is identified where it occurs, either as an introductory statement to a chapter or section or by the following marking:



Product-sensitive programming interface and associated guidance information...



Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com)® are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at <http://www.ibm.com/legal/copytrade.shtml>.

Other company, product, and service names may be trademarks or service marks of others.

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions:

Applicability: These terms and conditions are in addition to any terms of use for the IBM website.

Personal use: You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use: You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights: Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

Privacy policy considerations

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM's Privacy Policy at <http://www.ibm.com/privacy> and the section titled "Cookies, Web Beacons, and Other Technologies" in IBM's Online Privacy Statement at <http://www.ibm.com/privacy/details>. Also, see the "IBM Software Products and Software-as-a-Service Privacy Statement" at <http://www.ibm.com/software/info/product-privacy>.

Index

Special Characters

/STA AREA, ODE [731](#)

A

abend codes

FPA [1212](#)

FPO [1212](#)

OSM [1218](#)

ACB 4

ACBCHK keyword, FPA [438](#)

ACBLIB

OAE [762](#)

ACBLIB keyword, FPA [439](#)

ACBLIB keyword, OSM [987](#)

accessibility

overview [30](#)

ACTION keyword, OSM [984](#)

activating new member in ACBLIB, OAE [757](#)

active autonomics

Online Space Management utility [963](#)

ADS, allocating for OAE [749](#), [754](#)

ADXCGRP keyword, FPA [439](#)

ADXCGRP keyword, FPO [861](#)

ADXCGRP keyword, OSM [980](#)

AFTERTIME keyword, FPO [862](#)

ALLOCATE subcommand

keywords for CHANGE command

AMSOPT [153](#)

ATTRIBUTE [153](#)

AVGREC [153](#)

DATACLAS [153](#)

MGMTCLAS [153](#)

OAREA [153](#)

SPACE [153](#)

STORCLAS [153](#)

VOLSER [153](#)

keywords for INITIALIZE command

AMSOPT [376](#)

ATTRIBUTE [376](#)

AVGREC [376](#)

DATACLAS [376](#)

MGMTCLAS [376](#)

OAREA [376](#)

SPACE [376](#)

STORCLAS [376](#)

VOLSER [376](#)

keywords for RELOAD command

AMSOPT [112](#)

ATTRIBUTE [112](#)

AVGREC [112](#)

DATACLAS [112](#)

MGMTCLAS [112](#)

OAREA [112](#)

SPACE [112](#)

STORCLAS [112](#)

ALLOCATE subcommand (*continued*)

keywords for RELOAD command (*continued*)

VOLSER [112](#)

keywords, OSC

ATTRIBUTE [1048](#)

OAREA [1048](#)

allocating ADS for OAE [749](#), [754](#)

ALLOCONLY keyword, FPA [440](#)

AMSOPT keyword, FPA [440](#)

ANALYZE command

keywords

ATASKCTL [219](#)

CISNAP [219](#)

DBD [219](#)

DOVFNULPTR_MSGLVL [219](#)

HISTORYDD [220](#)

IAREA [220](#)

ICACHE [220](#)

ICHECK [220](#)

ICINCOMP [220](#)

ICNOTIFY [220](#)

ICOUT [220](#)

IDSNMASK [220](#)

IDXDUMP [220](#)

IDXMAXERROR [220](#)

IDXTASKCTL [220](#)

INDEXDBD [220](#)

KEYSEQCHK [220](#)

MAXERROR [220](#)

PTRCHKLVL [220](#)

SDEP_ORPHAN_MSGLVL [220](#)

SDEP_PTIOVF_MSGLVL [220](#)

SDEPCHKLVL [220](#)

SENSOR [220](#)

SENSOR_DBREC [220](#)

SENSOR_PTRERR [220](#)

SENSOR_UOW [220](#)

SENSOR_UOW_REPORT [220](#)

SORTOPT_FILSZ [221](#)

ANALYZE keyword, OSC [1040](#)

ANALYZE statement

keywords

ARDIGIT [863](#)

AREA [864](#)

DOVFNULPTR_MSGLVL [869](#)

ERRORACT [870](#)

EXC_RBASEFS [872](#)

EXC_RDOVFFS [873](#)

EXC_UDOVFFS [873](#)

FULLSTEP [882](#)

HISTORYDD [882](#)

IC [883](#)

IC1OUTDD [886](#)

IC2OUTDD [886](#)

ICNOTIFY [885](#)

IOVFPOOL [888](#)

MAXERROR [894](#)

ANALYZE statement (*continued*)

keywords (*continued*)

MAXTWINCTR [895](#)
MFILE [896](#)
PCHKONLY [901](#)
PROC_AFT_ERROR [902](#)
RCEXLOAD [904](#)
RFILE [908](#)
RGROUP [910](#)
SDEP [912](#)
SDEP_ORPHAN_MSGLVL [914](#)
SDEP_PTRIOVF_MSGLVL [915](#)
STAT [928](#)
SYNTBL [932](#)
TARDSNMASK [932](#)
TARECGEN [933](#)
TYPRUN [935](#)

ANDFIELD keyword, [FPO 862](#)

ARDIGIT keyword, [FPO 863](#)

AREA keyword, [FPO 764, 864](#)

Area Tuning Summary report, [FPA 649](#)

AREA_INFORMATION_RECORD keyword, [FPA 443](#)

area-level control [730](#)

AREADEF subcommand

keywords for TUNE command

CISIZE [250](#)
OAREA [250](#)
ROOT [250](#)
UOW [250](#)

AREADSN keyword, [FPA 441](#)

AREASCAN keyword, [FPA 442](#)

ATASKCTL keyword, [FPA 443](#)

ATTRIBUTE keyword, [FPA 444](#)

ATTRIBUTE keyword, [OSC 1048](#)

Audit report, [FPA 564](#)

Audit report, [OSM 1001](#)

automating database maintenance tasks [9](#)

Autonomics Director [16](#)

AVGREC keyword, [FPA 445](#)

B

Basic Tools package [673](#)

Batch DEDB Pointer Checker [673](#)

benefits

[FPA 35](#)

[FPO](#)

[OAE 670](#)

[ODE 670](#)

[ODM 671](#)

[OER 671](#)

[OPC 669](#)

[OSC 1007, 1023](#)

[OSM 956](#)

BLDMFILECTL subcommand

keywords for INDEXBLD command

AVGREC [312](#)
DATACLAS [312](#)
DISP [312](#)
DSNAME [312](#)
EXPDT [312](#)
INDEXDBD [312](#)
LIKE [312](#)
MGMTCLAS [312](#)

BLDMFILECTL subcommand (*continued*)

keywords for INDEXBLD command (*continued*)

RETPD [312](#)

SPACE [312](#)

STORCLAS [312](#)

UNIT [312](#)

VOLCNT [312](#)

VOLSER [312](#)

BREAK keyword, [FPA 446](#)

BREAK keyword, [FPO 865](#)

build pointer segment records [296](#)

BUILDDSNMASK keyword, [FPA 446](#)

BUILDMODE keyword, [FPA 447](#)

C

CASE keyword, [FPO 866](#)

CHANGE command

keywords

ATASKCTL [151](#)

CISNAP [151](#)

DBD [151](#)

DBDREC_DISCARD_MSGLVL [151](#)

EDITCOMP_SKIP [151](#)

HISTORYDD [151](#)

IAREA [151](#)

ICACHE [151](#)

ICHECK [151](#)

ICINCOMP [151](#)

ICNOTIFY [151](#)

ICOUT [152](#)

IDSNMASK [152](#)

IEXITRTN [152](#)

ICINCOMP [151](#)

IMSCATACB_OUTPUT [152](#)

ITASKCTL [152](#)

KEYCHGCHK [152](#)

KEYSEQCHK [152](#)

MAXERROR [152](#)

OAREA [152](#)

OCACHE [152](#)

ODSNMASK [152](#)

OEXITRTN [152](#)

OTASKCTL [152](#)

PAD [152](#)

PTRCHK [152](#)

PTRCHKLVL [152](#)

RELOCATE_SDEP [152](#)

SDEP [152](#)

SDEP_BYPASS_MSGLVL [152](#)

SDEP_ORPHAN_MSGLVL [152](#)

SDEPCHKLVL [152](#)

SENSOR [152](#)

SENSOR_DBREC [153](#)

SENSOR_PTRERR [153](#)

SENSOR_UOW [153](#)

SENSOR_UOW_REPORT [153](#)

SORTOPT_FILSZ [153](#)

SSP [153](#)

checking pointers, [OPC 674](#)

checkpoint restart

changing keywords [49](#)

checkpoint restart option

[FPA 41, 49](#)

CHKUOW keyword, [FPO 866](#)

CI Map/CI Dump report, FPA [588](#)
 CI SNAP Dump report, FPO [719](#)
 CIPRT keyword, FPO [867](#)
 CISIZE keyword, FPA [448](#)
 CISNAP keyword, FPA [448](#)
 commands, FPA
 ANALYZE [382](#)
 CHANGE [386](#)
 DMACPRT [390](#)
 END [391](#)
 EXTRACT [391](#)
 GLOBAL [394](#)
 INDEXBLD [397](#), [400](#), [403](#)
 INITIALIZE [404](#)
 RELOAD [405](#)
 REORG [409](#)
 TUNE [413](#)
 UNLOAD [415](#)
 commands, OSC
 GLOBAL [1036](#)
 RESTRUCTURE [1039](#)
 commands, OSM
 GLOBAL [980](#)
 MANAGE [984](#)
 comments, FPA [381](#)
 COMPRESS keyword, FPA [449](#)
 COMPRESS keyword, OSC [1052](#)
 compressing/expanding segment data
 DEDB Reload [1201](#), [1203](#)
 DEDB Unload [1201](#), [1203](#)
 CONFIGLIB_DSN keyword, OSC [1052](#)
 considerations
 OSC [1014](#)
 Consolidated Largest Database Records report, FPA [604](#)
 continuation characters, FPA [381](#)
 cookie policy [1437](#)
 Create Utility [754](#), [757](#)
 CSC area, OAE [751](#)

D

DAR120, OPC [680](#), [681](#), [685](#)
 DAR13SO, OPC [680](#), [681](#), [685](#)
 DAR3CSO, OPC [680](#), [681](#), [685](#)
 data set mask

DSNAME
 &AREA [457](#)
 &AREA# [457](#)
 &AREAn# [457](#)
 &DATE [457](#)
 &DBD [457](#)
 &IMSID [457](#)
 &JOBNAME [457](#)
 &PROCSN [457](#)
 &STEPN [457](#)
 &TIME [457](#)
 IDSNMASK
 &AREA [477](#)
 &AREA# [477](#)
 &AREAn# [477](#)
 &DBD [477](#)
 LIKE
 &AREA [492](#)
 &AREA# [492](#)

data set mask (*continued*)

LIKE (*continued*)
 &AREAn# [492](#)
 &DATE [493](#)
 &DBD [492](#)
 &JOBNAME [493](#)
 &PROCSN [493](#)
 &STEPN [493](#)
 &TIME [493](#)
 ODSNMASK
 &AREA [501](#)
 &AREA# [501](#)
 &AREAn# [501](#)
 &DBD [501](#)
 TARDNSMASK
 &AREA [549](#), [933](#)
 &AREA# [549](#), [933](#)
 &AREAn# [549](#), [933](#)
 &DBD [549](#), [933](#)

data set size for OPC [681](#)

DATACLAS keyword, FPA [449](#)
 DATACLAS keyword, FPO [867](#)
 DATACLAS keyword, OSM [988](#)
 DATACOMP_SUFFIX keyword, OSC [1052](#)
 DB Record Length Distribution report, FPA [647](#)
 DB Record Placement Analysis report, FPA [599](#)
 DB Record Profile Analysis report, FPA [596](#)
 DBD [4](#)
 DBD Definition report, FPA [569](#)
 DBD information [111](#)
 DBD information records [438](#)
 DBD keyword, FPA [450](#)
 DBD keyword, OSC [1040](#)
 DBD keyword, OSM [985](#)
 DBDLIB_DSN keyword, OSC [1053](#)
 DBFUMRIO utility [754](#), [757](#)
 DBR_NOFEOV keyword, OSC [1054](#)
 DBR_NOFEOV_APPLY keyword, OSC [1054](#)
 DBRC

 and OAE [750](#), [755](#)

DBRC keyword, FPA [451](#)
 DBRC keyword, FPO [868](#)
 DBRC SCI registration exit routine [682](#)
 DBRCGRP keyword, OSC [1053](#)
 DBRCGRP_APPLY keyword, OSC [1054](#)
 DBREC_DISCARD_MSGLVL keyword, FPA [452](#)
 DBRLDIST keyword, FPA [452](#)
 DBRLDIST keyword, FPO [868](#)
 DBT [4](#)
 DCLASS_TEMP keyword, OSC [1055](#)
 DD statements
 Analyze command input
 addname/areaname DD [203](#)
 HFPAARM DD [203](#)
 HFPAUOWC DD [203](#)
 HFPCSITE DD [203](#)
 HFPSYSIN DD [203](#)
 ICEIN DD [203](#)
 IMSACB DD [203](#)
 IMSACBA DD [203](#)
 IMSACBB DD [203](#)
 IMSDALIB DD [203](#)
 index DD [203](#)
 indexovrflw DD [203](#)

DD statements (*continued*)

Analyze command input (*continued*)

JOBLIB/STEPLIB DD [203](#)
MODSTAT DD [203](#)
MODSTAT2 DD [203](#)
OLCSTAT DD [203](#)

Analyze command input/output

HFPInnnn DD [203](#)
HFPSnnnn DD [203](#)
itttWKnn DD [203](#)
RECON1/RECON2/RECON3 DD [203](#)

Analyze command output

dataout1 DD [203](#)
dataout2 DD [203](#)
DFSPRINT DD [203](#)
HFPABEND DD [203](#)
HFPAHST DD [203](#)
HFPAHST2 DD [203](#)
HFPARERG DD [203](#)
HFPASNAP DD [203](#)
HFPPRINT DD [203](#)
HFPRPTS DD [203](#)
HFPSRPT DD [203](#)
ICEPRINT DD [203](#)
ICERPRT DD [203](#)
itttMSG DD [203](#)

Change command input

EXITLIB DD [134](#)
HFPAPARM DD [134](#)
HFPAUOWC DD [134](#)
HFPCSITE DD [134](#)
HFPSYSIN DD [134](#)
IAREAnnn/IARnnnnn DD [134](#)
ICEIN DD [134](#)
IMSACB DD [134](#)
IMSACBA DD [134](#)
IMSACBB DD [134](#)
IMSDALIB DD [134](#)
IMSRESLB DD [134](#)
JOBLIB/STEPLIB DD [134](#)
MODSTAT DD [134](#)
MODSTAT2 DD [134](#)
OLCSTAT DD [134](#)
OLDACB DD [134](#)
OLDRESLB DD [134](#)

Change command input/output

ctttWKnn DD [134](#)
itttWKnn DD [134](#)
RECON1/RECON2/RECON3 DD [134](#)

Change command output

addnname/areaname DD [134](#)
ctttMSG DD [134](#)
dataout1 DD [134](#)
dataout2 DD [134](#)
DFSPRINT DD [134](#)
HFPABEND DD [134](#)
HFPAHST DD [134](#)
HFPAHST2 DD [134](#)
HFPARERG DD [134](#)
HFPASNAP DD [134](#)
HFPPRINT DD [134](#)
HFPRPTS DD [134](#)
HFPSEQER DD [134](#)
HFPSRPT DD [134](#)

DD statements (*continued*)

Change command output (*continued*)

HFPTOTAL DD [134](#)
ICEPRINT DD [134](#)
ICERPRT DD [134](#)
itttMSG DD [134](#)

DMACPRT command input

addnname/areaname DD [286](#)
HFPSYSIN DD [286](#)
IMSACB DD [286](#)
IMSACBA DD [286](#)
IMSACBB DD [286](#)
IMSDALIB DD [286](#)
JOBLIB/STEPLIB DD [286](#)
MODSTAT DD [286](#)
MODSTAT2 DD [286](#)
OLCSTAT DD [286](#)

DMACPRT command input/output

RECON1/RECON2/RECON3 DD [286](#)

DMACPRT command output

HFPABEND DD [286](#)
HFPPRINT DD [286](#)
HFPRPTS DD [286](#)

Extract command input

addnname/areaname DD [265](#)
HFPCSITE DD [265](#)
HFPSYSIN DD [265](#)
IMSACB DD [265](#)
IMSACBA DD [265](#)
IMSACBB DD [265](#)
IMSDALIB DD [265](#)
JOBLIB/STEPLIB DD [265](#)
MODSTAT DD [265](#)
MODSTAT2 DD [265](#)
OLCSTAT DD [265](#)

Extract command input/output

ctttWKnn DD [265](#)
RECON1/RECON2/RECON3 DD [265](#)

Extract command output

ctttMSG DD [265](#)
HFPABEND DD [265](#)
HFPPRINT DD [265](#)
HFPRPTS DD [265](#)
HFPTOTAL DD [265](#)
OAREAnnn/OARnnnnn DD [265](#)

INDEXBLD command input

HFPSYSIN DD [301](#)
IMS DD [301](#)
IMSACB DD [301](#)
IMSACBA DD [301](#)
IMSACBB DD [301](#)
IMSDALIB DD [301](#)
IMSRESLB DD [301](#)
JOBLIB/STEPLIB DD [301](#)
MODSTAT DD [301](#)
MODSTAT2 DD [301](#)
OLCSTAT DD [301](#)

INDEXBLD command input (Freespace Reclaim)

HFPSYSIN DD [358](#)
IMSACB DD [358](#)
IMSACBA DD [358](#)
IMSACBB DD [358](#)
IMSDALIB DD [358](#)

DD statements (*continued*)

INDEXBLD command input (Freespace Reclaim) (*continued*)
 JOBLIB/STEPLIB DD [358](#)
 MODSTAT DD [358](#)
 MODSTAT2 DD [358](#)
 OLCSTAT DD [358](#)

INDEXBLD command input (Resync)
 HFPSYSIN DD [333](#)
 IMSACB DD [333](#)
 IMSACBA DD [333](#)
 IMSACBB DD [333](#)
 IMSDALIB DD [333](#)
 IMSRESLB DD [333](#)
 JOBLIB/STEPLIB DD [333](#)
 MODSTAT DD [333](#)
 MODSTAT2 DD [333](#)
 OLCSTAT DD [333](#)

INDEXBLD command input/output
 RECON1/RECON2/RECON3 DD [301](#)
 S0nnnn0 DD [301](#)
 UtttWKnn DD [301](#)

INDEXBLD command input/output
 (Freespace Reclaim)
 index DD [358](#)
 indexovrflw DD [358](#)
 RECON1/RECON2/RECON3 DD [358](#)

INDEXBLD command input/output
 (Resync)
 index DD [333](#)
 indexovrflw DD [333](#)
 RECON1/RECON2/RECON3 DD [333](#)
 S0nnnn0 DD [333](#)
 S0nnnnP DD [333](#)
 S0nnnnR DD [333](#)
 UtttWKnn DD [333](#)

INDEXBLD command output
 HFPABEND DD [301](#)
 HFPPRINT DD [301](#)
 HFPRPTS DD [301](#)
 index DD [301](#)
 indexovrflw DD [301](#)
 S0nnnnR DD [301](#)

INDEXBLD command output
 (Freespace Reclaim)
 HFPABEND DD [358](#)
 HFPPRINT DD [358](#)
 HFPRPTS DD [358](#)
 SYSPRINT DD [358](#)

INDEXBLD command output (Resync)
 HFPABEND DD [333](#)
 HFPPRINT DD [333](#)
 HFPRPTS DD [333](#)
 SYSPRINT DD [333](#)

INITIALIZE command input
 HFPSYSIN DD [369](#)
 IMSACB DD [369](#)
 IMSACBA DD [369](#)
 IMSACBB DD [369](#)
 IMSDALIB DD [369](#)
 IMSRESLB DD [369](#)
 JOBLIB/STEPLIB DD [369](#)
 MODSTAT DD [369](#)
 MODSTAT2 DD [369](#)
 OLCSTAT DD [369](#)

DD statements (*continued*)

INITIALIZE command input/output
 RECON1/RECON2/RECON3 DD [369](#)

INITIALIZE command output
 addnname/areaname DD [369](#)
 HFPABEND DD [369](#)
 HFPPRINT DD [369](#)
 HFPRPTS DD [369](#)

OSC
 input [1027](#)
 output [1028](#)

OSM
 input [972](#)
 input and output [976](#)
 output [973](#)

Reload command input
 EXITLIB DD [94](#)
 HFPAPARM DD [94](#)
 HFPAUOWC DD [94](#)
 HFPCHKPT DD [94](#)
 HFPCSITE DD [94](#)
 HFPSYSIN DD [94](#)
 IAREAnnn/IARnnnnn DD [94](#)
 ICEIN DD [94](#)
 IMSACB DD [94](#)
 IMSACBA DD [94](#)
 IMSACBB DD [94](#)
 IMSDALIB DD [94](#)
 IMSRESLB DD [94](#)
 JOBLIB/STEPLIB DD [94](#)
 MODSTAT DD [94](#)
 MODSTAT2 DD [94](#)
 OLCSTAT DD [94](#)

Reload command input/output
 ctttWKnn DD [94](#)
 itttWKnn DD [94](#)
 RECON1/RECON2/RECON3 DD [94](#)

Reload command output
 addnname/areaname DD [94](#)
 ctttMSG DD [94](#)
 dataout1 DD [94](#)
 dataout2 DD [94](#)
 DFSPRINT DD [94](#)
 HFPABEND DD [94](#)
 HFPAHST DD [94](#)
 HFPAHST2 DD [94](#)
 HFPARERG DD [94](#)
 HFPASNAP DD [94](#)
 HFPPRINT DD [94](#)
 HFPRPTS DD [94](#)
 HFPSRPT DD [94](#)
 HFPTOTAL DD [94](#)
 ICEPRINT DD [94](#)
 ICERPRT DD [94](#)
 itttMSG DD [94](#)

Reorg command input
 HFPAPARM DD [170](#)
 HFPAUOWC DD [170](#)
 HFPCSITE DD [170](#)
 HFPSYSIN DD [170](#)
 ICEIN DD [170](#)
 IMSACB DD [170](#)
 IMSACBA DD [170](#)
 IMSACBB DD [170](#)

DD statements (*continued*)

Reorg command input (*continued*)

IMSDALIB DD [170](#)
JOB LIB/STEPLIB DD [170](#)
MODSTAT DD [170](#)
MODSTAT2 DD [170](#)
OLCSTAT DD [170](#)

Reorg command input/output

addname/areaname DD [170](#)
itttWKnn DD [170](#)
RECON1/RECON2/RECON3 DD [170](#)

Reorg command output

dataout1 DD [170](#)
dataout2 DD [170](#)
DFSPRINT DD [170](#)
HFPABEND DD [170](#)
HFPAHST DD [170](#)
HFPAHST2 DD [170](#)
HFPARERG DD [170](#)
HFPASNAP DD [170](#)
HFPPRINT DD [170](#)
HFPRPTS DD [170](#)
HFPSRPT DD [170](#)
ICEPRINT DD [170](#)
ICERPRT DD [170](#)
itttMSG DD [170](#)

Tune command input

addname/areaname DD [238](#)
HFPAPARM DD [238](#)
HFPCSITE DD [238](#)
HFPSYSIN DD [238](#)
IMSACB DD [238](#)
IMSACBA DD [238](#)
IMSACBB DD [238](#)
IMSDALIB DD [238](#)
JOB LIB/STEPLIB DD [238](#)
MODSTAT DD [238](#)
MODSTAT2 DD [238](#)
OLCSTAT DD [238](#)

Tune command input/output

itttWKnn DD [238](#)
RECON1/RECON2/RECON3 DD [238](#)
TARnnnnn DD [238](#)

Tune command output

HFPABEND DD [238](#)
HFPASNAP DD [238](#)
HFPPRINT DD [238](#)
HFPRPTS DD [238](#)
itttMSG DD [238](#)

Unload command input

addname/areaname DD [56](#)
EXITLIB DD [56](#)
HFPCHKPT DD [56](#)
HFPCSITE DD [56](#)
HFPSYSIN DD [56](#)
ICEIN DD [56](#)
IMSACB DD [58](#)
IMSACBA DD [58](#)
IMSACBB DD [56](#)
IMSDALIB DD [56](#)
IMSRESLB DD [56](#)
JOB LIB/STEPLIB DD [58](#)
MODSTAT DD [56](#)
MODSTAT2 DD [56](#)

DD statements (*continued*)

Unload command input (*continued*)

NEWACB DD [56](#)
NEWRESLB DD [56](#)
OLCSTAT DD [56](#)

Unload command input/output

ctttWKnn DD [56](#)
RECON1/RECON2/RECON3 DD [56](#)

Unload command output

ctttMSG DD [56](#)
dataout1 DD [56](#)
dataout2 DD [56](#)
DFSPRINT DD [56](#)
HFPABEND DD [56](#)
HFPASNAP DD [56](#)
HFPPRINT DD [56](#)
HFPRPTS DD [56](#)
HFPSEQER DD [56](#)
HFPTOTAL DD [56](#)
ICEPRINT DD [56](#)
ICERPRT DD [56](#)
itttMSG DD [56](#)
OAREAnnn/OARnnnnn DD [56](#)

DDNAME keyword, FPA [453](#)

DDNAME keyword, FPO [764, 869](#)

DDNAME keyword, OSM [988](#)

DDNPREFIX_NEW keyword, OSC [1055](#)

DDNPREFIX_SHADOW keyword, OSC [1056](#)

DDNPREFIX_UIC keyword, OSC [1056](#)

DEDB Area Analysis reports, FPA [591](#)

DEDB encryption [19](#)

DEDB Pointer Checker

History file record layout [1173](#)

History2 file record layout [1175](#)

DEDB Pointer Checker and Online Pointer Checker [673](#)

DEDB Reload (FABCUR3)

compressing/expanding segment data [1201, 1203](#)

database definition record [1202](#)

edit/compression routine interface [1202](#)

exit routine [1197](#)

exit routine interface [1197](#)

unloaded segment record [1203](#)

DEDB Reload Segment Data Set Create utility (FABCUR6)

exit routine [1197](#)

exit routine interface [1197](#)

DEDB Unload (FABCUR1)

compressing/expanding segment data [1201, 1203](#)

database definition record [1202](#)

edit/compression routine interface [1202](#)

exit routine [1197](#)

exit routine interface [1197](#)

unloaded segment record [1203](#)

DFSORT [4](#)

diagnostic messages [1220](#)

diagnostics aid

messages and codes [1220](#)

DISP keyword, FPA [453](#)

DMACPRT command

keywords

DBD [292](#)

IAREA [292](#)

IDSNMASK [292](#)

DMACPRT report, FPA [612](#)

DMACPRT report, FPO [826](#)

DMACPRT statement

keywords

AREA [864](#)
CIPRT [867](#)
MFILE [896](#)
PCHKONLY [901](#)
PROC_AFT_ERROR [902](#)
RFILE [908](#)

DMB 4

DMCB 4

documentation

accessing [28](#)

sending feedback [28](#)

documentation changes 3

DOVFFS_PCT keyword, FPA [455](#)

DOVFFS_PCT keyword, FPO [869](#)

DOVFNULPTR_MSGLVL keyword, FPA [455](#)

DOVFNULPTR_MSGLVL keyword, FPO [869](#)

DSNAME keyword, FPA [456](#)

DSNAME keyword, FPO [870](#)

DSNAME keyword, OSM [989](#)

DSNMASK_NEW keyword, OSC [1056](#)

DSNMASK_SHADOW keyword, OSC [1057](#)

DSNMASK_UIC keyword, OSC [1057](#)

DUPKEYACT keyword, FPA [457](#)

DUPKEYDSNMASK keyword, FPA [458](#)

DUPKEYMAX keyword, FPA [458](#)

DURDBDFN

record layout [1185](#)

E

EDITCOMP_SKIP keyword, FPA [459](#)

EFORMAT keyword, FPA [460](#)

END command [391](#)

ENQDSN keyword, OSC [1058](#)

ERRORACT keyword, FPO [870](#)

ERRORACT keyword, OSC [1041](#)

ERRORACT keyword, OSM [993](#)

ERRORSEG

record layout [1185](#)

EVERY keyword, FPA [461](#)

EVERY keyword, FPO [871](#)

examples

FPA

JCL statement for controlling the FPA process [40](#)

monitoring and tuning DEDBs [1129](#)

FPA, Analyze

checksum validation of pointers in an area [223](#)

checksum validation of pointers in an image copy [225](#)

cross-reference validation of pointers in an area [224](#)

cross-reference validation, secondary index database [225](#), [226](#)

generating the analysis report of an area in the HPPRPTS data set [224](#)

generating the analysis reports of an area and storing the sensor data in the IMS Tools KB repository [227](#)

monitoring and tuning DEDBs [1129](#)

FPA, Build Index

building a secondary index database [316](#)

building all secondary index databases [315](#)

examples (continued)

FPA, Build Index (continued)

building specific secondary index databases [315](#)

generate records and concurrently build secondary indexes [317](#)

FPA, Change

monitoring and tuning DEDBs [1141](#)

reorganizing an area registered with DBRC [157](#)

reorganizing an area that is registered with DBRC and creating its image copy [159](#)

reorganizing and analyzing the reorganized area [160](#)

reorganizing multiple areas registered with DBRC while renaming the input ADS [158](#)

restructuring all areas registered with DBRC [157](#)

storing the sensor data in the IMS Tools KB repository [160](#)

FPA, DMAC Print

printing a DMAC block in an area [293](#)

printing DMAC blocks in all areas of a DEDB [293](#)

FPA, Extract

changing the style of the segment records [277](#)

creating a test database [278](#)

extracting in user-defined format [279](#)

extracting segments into designated output data sets [280](#)

extracting segments with the selection criteria [277](#)

FPA, Freespace Reclaim

multiple data sets [365](#)

reports [365](#)

single data set [365](#)

FPA, Initialize

initializing an area that is registered with DBRC [377](#)

initializing multiple areas in an SMS environment [378](#)

FPA, Reload

all areas of a DEDB registered with DBRC [117](#)

an area of which some of the parameters for the DEDB area definition are changed [119](#)

an area registered with DBRC [116](#)

an area registered with DBRC as MADS area into a specified ADS [118](#)

monitoring and tuning DEDBs [1135](#)

multiple areas of a DEDB from one input unloaded segment record data set [117](#)

reloading all areas by using the concatenated unloaded segment records data sets [121](#)

reloading an area from a user-specified format data set [124](#)

reloading an area registered with DBRC by using the LOADCTL subcommand [120](#)

reloading an area that is registered with DBRC and creating its image copy [123](#)

reloading and analyzing an area [124](#)

reloading and analyzing an area registered with DBRC [120](#)

reloading areas by using the concatenated unloaded segment records data sets [122](#)

reloading areas in an SMS environment [125](#)

reloading areas without DBRC [126](#)

reloading when IMS management of ACBs is enabled [127](#)

storing the sensor data in the IMS Tools KB repository [124](#)

FPA, Reorg

examples (continued)

FPA, Reorg (continued)

- monitoring and tuning DEDBs [1148](#)
- reorganizing all UOWs in an area [189](#)
- reorganizing and analyzing the reorganized area [191](#)
- reorganizing some UOWs that satisfy one of RBASEFS or RDOVFFS [190](#)
- reorganizing UOWs that use IOVF CIs and generating an image copy [189](#)
- storing the sensor data in the IMS Tools KB repository [191](#)

FPA, Resync

- offline synchronization [352](#)
- two-step method [347](#), [349](#)
- verify update method [346](#)

FPA, Tune

- evaluating a randomizer [255](#)
- evaluating area reorganization [254](#)
- rerunning with different parameters [256](#)
- simulating changes in area definitions [254](#)
- simulating changes in area definitions of online areas [257](#)

FPA, Unload

- a registered area with DBRC [74](#)
- all areas of a DEDB and changing some or all of the DEDB area definitions (IAREA=ALL, OAREA=ALL) [77](#)
- all areas of a DEDB and changing some or all of the DEDB area definitions (IAREA=ALL, OAREA=specific) [78](#)
- all areas of a DEDB and changing some or all of the DEDB area definitions (IAREA=specific, OAREA=ALL) [79](#)
- all areas of a DEDB and changing some or all of the DEDB area definitions (IAREA=specific, OAREA=specific) [80](#)
- all areas of a DEDB registered with DBRC [74](#)
- all areas of a DEDB that are registered with DBRC and creating their image copies [81](#)
- an area from the specified ADS of MADS area [76](#)
- an area to change the DEDB area definition [77](#)
- multiple areas of a DEDB into one output unloaded segment record data set [75](#)
- unloading areas and creating image copies with checkpoint restart option [84](#)
- unloading areas in user-specified format [82](#)
- unloading multiple areas in an SMS environment [84](#)
- unloading when IMS management of ACBs is enabled [86](#)
- unloading without DBRC [85](#)

FPB, DEDB Pointer Checker

- monitoring and tuning DEDBs [1154](#)

FPO, OAE

- original expansion [772](#)
- original expansion, extending both IOVF and SDEP portions in an SMS environment [773](#)
- original expansion, extending IOVF portion in an SMS environment [772](#)
- original expansion, extending the IOVF portion with EXTEND_TYPE=RENAME [774](#)
- original expansion, extending the IOVF portion, SDEP segments [775](#)
- original expansion, extending the portions with ACBLIB under non-SMS environment [773](#)

examples (continued)

FPO, OAE (continued)

- original expansion, extending the SDEP and image copy data sets [776](#)
- original expansion, verifying the parameter to extend the SDEP portion in an SMS environment [774](#)
- shadow expansion [776](#)
- shadow expansion, extending IOVF portion [776](#)
- shadow expansion, extending SDEP and image copy data sets [777](#)

FPO, ODE

- running ODE with area-level control [745](#)
- running ODE with run-level control [744](#)

FPO, ODM

- reporting only DMAC information in IMS storage [832](#)
- reporting two kinds of DMAC information [833](#)

FPO, OER

- monitoring and tuning DEDBs [1162](#)
- running OER with OPC [818](#)
- running OER with OPC and storing sensor data [819](#)
- running OER with OPC for multiple areas and taking image copies [819](#)
- running OER with the segment shunting function [820](#)
- running OER without OPC [817](#)

FPO, OPC

- collecting segment information records for database tuning [723](#)
- cross-reference validation of pointers [722](#)
- cross-reference validation of pointers and generating analysis reports [722](#)
- cross-reference validation of pointers and storing the sensor data [727](#)
- fast scan of database integrity [720](#)
- fast scan of database integrity and collecting pointer verification data [721](#)
- monitoring and tuning DEDBs [1157](#)
- obtaining SDEP record information [728](#)
- processing multiple areas and taking image copies [727](#)

IMS HP Image Copy

- monitoring and tuning DEDBs [1167](#)

OSM

- JCL statement for stand-alone mode [960](#)

examples OSC [1085](#)

EXC_RBASEFS keyword, FPO [872](#)

EXC_RDOVFFS keyword, FPO [873](#)

EXC_UDOVFFS keyword, FPO [873](#)

EXEC statement

- DBRCGRP [53](#), [90](#), [131](#), [167](#), [198](#), [233](#), [262](#), [284](#), [297](#), [328](#), [330](#), [331](#), [356](#), [367](#)

- GSGNAME [53](#), [90](#), [131](#), [167](#), [198](#), [233](#), [262](#), [284](#), [297](#), [328](#), [330](#), [331](#), [356](#), [367](#)

- IMSPLEX [53](#), [90](#), [131](#), [167](#), [198](#), [233](#), [262](#), [284](#), [297](#), [328](#), [330](#), [331](#), [356](#), [367](#)

- OAE [756](#)

- ODE [732](#)

- ODM [822](#)

- OER [784](#)

- OPC [684](#)

EXIT keyword, FPO [765](#)

exit routine

exit routine (*continued*)

DEDB Reload (FABCUR3)

interface [1197](#)

DEDB Reload Segment Data Set Create utility
(FABCUR6)

interface [1197](#)

DEDB Unload (FABCUR1)

interface [1197](#)

FABCUR7 (DEDB Unloaded Segment Data Set Retrieve
utility)

interface [1197](#)

EXITRTN keyword, FPA [461](#)

EXPAND keyword, FPO [874](#)

EXPDT keyword, FPA [462](#)

EXTEND keyword, FPO [765](#)

EXTEND statement

keywords

AREA [864](#)

DATACLAS [867](#)

DDNAME [869](#)

DSNAME [870](#)

EXTEND_TYPE [875](#)

ICERR_CONT [884](#)

ICEXTEND [884](#)

IOVF_ROOT [889](#)

IOVF_UOW [889](#)

MAXVOLS [896](#)

MGMTCLAS [897](#)

RUNTYPE [911](#)

SDEP_CI [913](#)

STORCLAS [930](#)

VOLUME [939](#)

VOLUME_EXT [939](#)

EXTEND_TYPE keyword, FPO [875](#)

EXTEND_TYPE keyword, OSM [989](#)

extending IOVF with OAE [749](#), [754](#)

extending SDEP with OAE [749](#), [754](#)

EXTRACT command

keywords

COMPRESS [273](#)

DBD [273](#)

EFORMAT [273](#)

IAREA [273](#)

ICACHE [273](#)

ICHECK [273](#)

ICINCOMP [273](#)

IDSNMASK [273](#)

ITASKCTL [274](#)

KEYSEQCHK [274](#)

OCHECK [274](#)

ODSNMASK [274](#)

PTRCHK [274](#)

SEGREC_PREFIX [274](#)

SEGREC_SUFFIX [274](#)

SORT [274](#)

SORTOPT_FILSZ [274](#)

EXTRACT statement

examples [844](#)

keywords

AREA [864](#)

PROC_AFT_ERROR [902](#)

RCEXLOAD [904](#)

Extracted report, FPA [609](#)

F

FABADIAG [1205](#), [1433](#)

FABCUR7 (DEDB Unloaded Segment Data Set Retrieve
utility)

exit routine [1197](#)

exit routine interface [1197](#)

FIELD keyword, FPO [875](#)

FIELDS keyword, FPA [463](#)

FIELDS keyword, FPO [876](#), [877](#)

FILECTL subcommand

keywords for EXTRACT command

AVGREC [274](#)

DATACLAS [274](#)

DDNAME [274](#)

DISP [274](#)

DSNAME [274](#)

EXPDT [274](#)

LIKE [274](#)

MGMTCLAS [274](#)

OAREA [274](#)

RETPD [274](#)

SPACE [274](#)

STORCLAS [274](#)

UNIT [274](#)

VOLCNT [274](#)

VOLSER [274](#)

keywords for INDEXBLD command

AVGREC [313](#), [345](#)

DATACLAS [313](#), [345](#)

DISP [313](#), [345](#)

DSNAME [313](#), [345](#)

EXPDT [313](#), [345](#)

INDEXDBD [313](#), [345](#)

LIKE [313](#), [345](#)

MGMTCLAS [313](#), [345](#)

RETPD [313](#), [345](#)

SPACE [313](#), [345](#)

STORCLAS [313](#), [345](#)

UNIT [313](#), [345](#)

VOLCNT [313](#), [345](#)

VOLSER [313](#), [345](#)

keywords for TUNE command

AVGREC [250](#)

DATACLAS [250](#)

DDNAME [250](#)

DISP [250](#)

DSNAME [250](#)

EXPDT [250](#)

IAREA [250](#)

LIKE [250](#)

MGMTCLAS [250](#)

RETPD [250](#)

SPACE [250](#)

STORCLAS [250](#)

UNIT [250](#)

VOLCNT [250](#)

VOLSER [250](#)

keywords for UNLOAD command

AVGREC [71](#)

DATACLAS [71](#)

DDNAME [71](#)

DISP [71](#)

DSNAME [71](#)

FILECTL subcommand (*continued*)

keywords for UNLOAD command (*continued*)

[EXPDT 71](#)
[LIKE 71](#)
[MGMTCLAS 71](#)
[OAREA 71](#)
[RETPD 71](#)
[SPACE 71](#)
[STORCLAS 71](#)
[UNIT 71](#)
[VOLCNT 71](#)
[VOLSER 71](#)

FORMAT keyword, [FPA 465](#)

FPA 4

FPB 4

FPBP64U keyword, [FPO 879](#)

FPBP64U keyword, [OSM 994](#)

FPLIB_DSN keyword, [OSC 1058](#)

FPO

security [951](#)

FPUTIL

[OAE 756](#)

[OPC 684](#)

[FPX0151E 759](#)

[FPX0188E 759](#)

[FPX0191E 759](#)

FPXBINTP DD

[OAE 763](#)

FPXCTL statement

examples [847](#)

keywords

[CASE 866](#)

[EXPAND 874](#)

[INDOUBT 887](#)

[IOVFPOOL 888](#)

[MAXTWINCTR 895](#)

[MFILE 896](#)

[OFILE 898](#)

[OLRECL 899](#)

[OUTPUT 899](#)

[PCHKONLY 901](#)

[SDEP_BYPASS_MSGLVL 912](#)

[SDEPHIER 913](#)

[SEGREC_PREFIX 919](#)

[SEGREC_SUFFIX 920](#)

[SORT 927](#)

[STAT 928](#)

FPXIN DD

[OAE 762](#)

[ODE 734](#)

[ODM 823](#)

[OER 787](#)

[OPC 687](#)

FPXOFIL, [ODE 731](#)

FPXRCEX0 [947](#)

FPXRPTS statement

[ODM 824](#)

[OER 788](#)

Freespace Analysis report, [FPA 592](#)

FS_DOVFIOVF keyword, [FPA 466](#)

FS_DOVFIOVF keyword, [FPO 880](#)

FS_RAADOVF keyword, [FPA 467](#)

FS_RAADOVF keyword, [FPO 880](#)

FS_RAADOVFIOVF keyword, [FPA 467](#)

FS_RAADOVFIOVF keyword, [FPO 881](#)

FS_RAADIOVF keyword, [FPA 468](#)

FS_RAADIOVF keyword, [FPO 881](#)

FS_RECLAIM_REPORT keyword, [FPA 468](#)

FULLSTEP keyword, [FPO 882](#)

functions

FPA

[Analyze 194](#)

[Build Index 295](#)

[Change 129](#)

[DMAC Print 283](#)

[Extract 259](#)

[Freespace Reclaim 355](#)

[Initialize 367](#)

[Reload 89](#)

[Reorg 163](#)

[Resync 321](#)

[Tune 229](#)

[Unload 51](#)

FPO

[DEDB analysis, OPC 676](#)

[DEDB integrity verification, OPC 674](#)

[OAE 747](#)

[ODE 729](#)

[ODM 821](#)

[OER 779](#)

[OPC 674](#)

Online Space Management utility (OSM) [955](#)

Online Structure Change utility (OSC) [1007](#)

G

GLOBAL command

keywords

[ADXCFGRP 109, 150, 183, 218, 861](#)

[AREADSN 218, 248](#)

[CISNAP 109, 150, 184, 218, 248](#)

[DBRC 68, 109, 150, 184, 218, 248, 273, 292, 311, 343, 363, 376](#)

[HISTORYDD 109, 150, 184, 218](#)

[ICINCOMP 68, 109, 150, 184, 219, 273, 292, 311](#)

[ICNOTIFY 68, 109, 150, 184, 219](#)

[ICOUT 68, 109, 150, 184, 219](#)

[IMSCATACB_INPUT 68, 109, 150, 184, 219, 248, 273, 292, 311, 343, 363, 376](#)

[IMSCATHLQ 69, 109, 150, 184, 219, 248, 273, 292, 311, 343, 363, 376](#)

[IMSID 343, 363](#)

[ITKBDATA 110, 150, 184](#)

[ITKBLOAD 69, 110, 150, 184, 219, 292, 311, 343, 363, 376](#)

[ITKBSRVR 69, 110, 150, 184, 219, 292, 311, 344, 363, 376](#)

[LIUGOPT 69, 110, 150, 184](#)

[MSTIMESTAMP 69, 110, 151, 184, 219, 248, 273, 292, 311, 344, 363, 376](#)

[PTRCHKLVL 110, 151, 184, 219](#)

[RESTART 69, 110](#)

[RESTART_COUNT 69, 110](#)

[SCAN 69, 110, 151, 184, 219, 248, 273, 292, 311, 344, 364, 376](#)

[SDEP_ORPHAN_MSGLVL 110, 151, 184, 219](#)

[SDEPCHKLVL 110, 151, 184, 219](#)

GLOBAL command (*continued*)

keywords (*continued*)

SITE_DSNAME [69](#), [110](#), [151](#), [184](#), [219](#), [248](#), [273](#),
[292](#), [311](#), [344](#), [364](#), [376](#)
SITE_MEMBER [69](#), [110](#), [151](#), [184](#), [219](#), [248](#), [273](#),
[292](#), [311](#), [344](#), [364](#), [376](#)
SYNC [344](#)
TOINUM [344](#)
TOITIME [344](#)
TOIXCFGRP [344](#)

keywords, OSC

IMSCATHLQ [1036](#)
IMSID [1037](#)
ITKBSRVR [1037](#)
MSTIMESTAMP [1037](#)
SCAN [1038](#)
TOIXCFGRP [1038](#)

keywords, OSM

ADXCFGRP [980](#)
ICNOTIFY [981](#)
ICOUT [981](#)
IMSID [982](#)
ITKBDATA [982](#)
ITKBSRVR [983](#)
SCAN [983](#)

GLOBAL statement

keywords

DBRC [868](#)
ITKBDATA [890](#)
ITKBLOAD [890](#)
ITKBSRVR [891](#)
KEYWORD_DUPCHK [892](#)

GO keyword, FPO [765](#)

H

HFPABEND statement, OPC [685](#)

HFPCSITE module [665](#)

HFPPRINT, OPC [685](#)

HFPSYSIN DD data set

Analyze [218](#)
Change [149](#)
DMACPRT [291](#)
Extract [272](#)
INDEXBLD [310](#), [343](#), [363](#)
INITIALIZE [375](#)
Online Space Management utility [977](#)
Online Structure Change utility [1031](#)
Reload [108](#)
Reorg [183](#)
Tune [247](#)
Unload [68](#)

HFPUDIAG [1205](#), [1433](#)

HFPZTOT [1183](#)

HFPZUSR [1177](#)

History file record layout [1173](#)

History2 file record layout [1175](#)

HISTORYDD keyword, FPA [469](#)

HISTORYDD keyword, FPO [882](#)

HPIC [687](#)

I

i001SMSG statement

OPC [685](#)

i001WKnn statement

OPC [685](#)

IAREA keyword, FPA [469](#)

IAREA keyword, OSC [1041](#)

IAREA keyword, OSM [985](#)

IC keyword, FPO [883](#)

IC1OUTDD keyword, FPO [886](#)

IC2OUTDD keyword, FPO [886](#)

ICACHE keyword, FPA [472](#)

ICERR_CONT keyword, FPO [884](#)

ICEXTEND keyword, FPO [884](#)

ICHECK keyword, FPA [473](#)

ICINCOMP keyword, FPA [474](#)

ICNOTIFY keyword, FPA [475](#)

ICNOTIFY keyword, FPO [885](#)

ICNOTIFY keyword, OSM [981](#)

ICOUT keyword, FPA [476](#)

ICOUT keyword, OSM [981](#)

IDSNMASK keyword, FPA [476](#)

IDXDUMP keyword, FPA [477](#)

IDXMAXERROR keyword, FPA [478](#)

IDXPROC keyword, FPA [478](#)

IDXTASKCTL keyword, FPA [479](#)

IEXITRTN keyword, FPA [479](#)

IFP utility [671](#), [754](#)

Image Copy Output report, FPA [608](#)

image copy parameters, OPC [886](#)

improving performance

FPA [1091](#)

FPA Analyze [1098](#)

FPA Build Index [1102](#)

FPA Change [1094](#)

FPA DMAC Print [1101](#)

FPA Extract [1100](#)

FPA Freespace Reclaim [1105](#)

FPA Initialize [1105](#)

FPA Reload [1093](#)

FPA Reorg [1096](#)

FPA Resync [1103](#)

FPA Tune [1099](#)

FPA Unload [1091](#)

FPO [1106](#)

FPO OAE [1107](#)

FPO ODE [1107](#)

FPO ODM [1109](#)

FPO OER [1108](#)

FPO OPC [1106](#)

OSC [1110](#)

OSM [1109](#)

IMS Fast Path Advanced Tool [11](#), [33](#)

IMS Fast Path Basic Tools [12](#)

IMS Fast Path Online Tools [12](#), [669](#), [673](#)

IMS High-Speed DEDB Direct Reorganization (HSR)

MAXBUFFS keyword [894](#)

IMS HP Fast Path Utilities [4](#)

IMS Library Integrity Utilities

change [132](#)

reload [92](#)

reorg [169](#)

unload [54](#)

IMS managed ACBs [18](#)
 IMS Tools Base [16](#)
 IMS Tools Catalog Interface [16](#)
 IMS Tools Knowledge Base [16](#)
 IMS Tools Online System Interface [16](#)
 IMSCATACB_INPUT keyword, FPA [480](#)
 IMSCATACB_OUTPUT keyword, FPA [481](#)
 IMSCATHLQ keyword, FPA [481](#)
 IMSCATHLQ keyword, OSC [1036](#)
 IMSID keyword, FPA [482](#)
 IMSID keyword, OSC [1037](#)
 IMSID keyword, OSM [982](#)
 IMSID_APPLY keyword, OSC [1059](#)
 IMSLIB_DSN keyword, OSC [1059](#)
 IMSLIB_DSN_APPLY keyword, OSC [1059](#)
 IMSPLEX keyword, OSC [1060](#)
 IMSPLEX_APPLY keyword, OSC [1060](#)
 IMSRESLB_EDC_DSN keyword, OSC [1060](#)
 IMSRESLB_RND_DSN keyword, OSC [1061](#)
 increasing size of IOVF, OAE [749](#), [754](#)
 increasing size of SDEP, OAE [749](#), [754](#)
 INDEXBLD command
 keywords
 AREASCAN [344](#)
 BUILDDSNMASK [312](#)
 BUILDMODE [312](#)
 DBD [312](#), [344](#), [364](#)
 DUPKEYACT [312](#)
 DUPKEYDSNMASK [312](#)
 DUPKEYMAX [312](#), [344](#)
 FS_RECLAIM_REPORT [364](#)
 IAREA [312](#), [344](#)
 ICACHE [312](#), [344](#)
 ICHECK [312](#), [345](#)
 ICINCOMP [312](#)
 IDSNMASK [312](#), [345](#)
 IDXPROC [312](#), [345](#), [364](#)
 INDEXDBD [312](#), [345](#), [364](#)
 ITASKCTL [312](#), [345](#)
 OTASKCTL [312](#), [345](#)
 PROC_AFT_DUPKEY [312](#), [345](#)
 RESYNCDSNMASK [345](#)
 RESYNCDUMP [345](#)
 RESYNCDUMPMAX [345](#)
 RESYNCMODE [345](#)
 SORT [312](#)
 TOICTL [345](#)
 INDEXDBD keyword, FPA [482](#)
 INDOUBT keyword, FPO [887](#)
 Initialization Processing Summary report, FPA [632](#)
 INITIALIZE command
 keywords
 ALLOCONLY [376](#)
 DBD [376](#)
 OAREA [376](#)
 OCHECK [376](#)
 ODSNMASK [376](#)
 OTASKCTL [376](#)
 input and output
 FPA
 Analyze [196](#)
 Build Index [295](#)
 Change [130](#)
 DMAC Print [283](#)

input and output (*continued*)
 FPA (*continued*)
 Extract [262](#)
 Freespace Reclaim [355](#)
 Initialize [367](#)
 Reload [90](#)
 Reorg [166](#)
 Resync [321](#)
 Tune [231](#)
 Unload [52](#)
 FPO
 OAE [748](#)
 ODE [731](#)
 ODM [821](#)
 OER [782](#)
 OPC [677](#)
 OSM [956](#)
 input data sets
 OPC [696](#)
 Input File report, FPA [583](#)
 INSERT_LIMIT_COUNT keyword, FPA [484](#)
 INSERT_LIMIT_COUNT keyword, OSC [1072](#)
 Integrity Checker utility
 change [132](#)
 reload [92](#)
 reorg [169](#)
 unload [54](#)
 integrity verification, OPC [674](#), [684](#)
 INTRDR keyword, OSC [1061](#)
 introduction, OPC [674](#)
 invoking OAE [752](#)
 IOVF
 buffer pool, OPC [888](#)
 IOVF extension, OAE [749](#), [754](#)
 processing, OPC [676](#)
 IOVF keyword, FPO [765](#)
 IOVF_ROOT keyword, FPO [889](#)
 IOVF_UOW keyword, FPO [889](#)
 IOVF_UOW keyword, OSM [990](#)
 IOVFCI_RBA_REPORT keyword, FPA [484](#)
 IOVFCI_RBA_REPORT keyword, FPO [887](#)
 IOVFFS_PCT keyword, FPA [485](#)
 IOVFFS_PCT keyword, FPO [887](#)
 IOVFPPOOL keyword, FPO [888](#)
 IOVFUSED_PCT keyword, FPA [485](#)
 IOVFUSED_PCT keyword, FPO [888](#)
 IRENAME subcommand
 keywords for CHANGE command
 DSNAME [153](#)
 IAREA [153](#)
 ITASKCTL keyword, FPA [486](#)
 ITASKCTL keyword, OSC [1042](#)
 ITKBDATA keyword, FPA [487](#)
 ITKBDATA keyword, FPO [890](#)
 ITKBDATA keyword, OSM [982](#)
 ITKBLOAD keyword, FPA [487](#)
 ITKBLOAD keyword, FPO [890](#)
 ITKBSRVR keyword, FPA [488](#)
 ITKBSRVR keyword, FPO [891](#)
 ITKBSRVR keyword, OSC [1037](#)
 ITKBSRVR keyword, OSM [983](#)

J

JCLDSN keyword, OSC [1061](#)
JCLPARM subcommand
 keywords, OSC
 COMPRESS [1052](#)
 CONFIGLIB_DSN [1052](#)
 DATACOMP_SUFFIX [1052](#)
 DBDLIB_DSN [1053](#)
 DBR_NOFEOV [1054](#)
 DBR_NOFEOV_APPLY [1054](#)
 DBRCGRP [1053](#)
 DBRCGRP_APPLY [1054](#)
 DCLASS_TEMP [1055](#)
 DDNPREFIX_NEW [1055](#)
 DDNPREFIX_SHADOW [1056](#)
 DDNPREFIX_UIC [1056](#)
 DSNMASK_NEW [1056](#)
 DSNMASK_SHADOW [1057](#)
 DSNMASK_UIC [1057](#)
 ENQDSN [1058](#)
 FPLIB_DSN [1058](#)
 IMSID_APPLY [1059](#)
 IMSLIB_DSN [1059](#)
 IMSLIB_DSN_APPLY [1059](#)
 IMSPLEX [1060](#)
 IMSPLEX_APPLY [1060](#)
 IMSRESLB_EDC_DSN [1060](#)
 IMSRESLB_RND_DSN [1061](#)
 INTRDR [1061](#)
 JCLDSN [1061](#)
 JOB CARDx [1062](#)
 JOBN_PREFIX [1062](#)
 LOGDSN [1063](#)
 MCLASS_TEMP [1063](#)
 NBA_APPLY [1064](#)
 NEWDBDLIB_DSN [1064](#)
 NEWRESLB_EDC_DSN [1064](#)
 NEWRESLB_RND_DSN [1065](#)
 OBA_APPLY [1065](#)
 PROCLIB_DSN [1066](#)
 PROCLIB_DSN_APPLY [1066](#)
 PSBNAME_APPLY [1066](#)
 RECONx_DSN [1067](#)
 RECONx_DSN_APPLY [1067](#)
 SCLASS_TEMP [1067](#)
 SGLXLOAD [1068](#)
 SPACE [1068](#)
 SPOC_WAITTIME [1069](#)
 SYSAFF [1069](#)
 SYSAFF_APPLY [1070](#)
 TOIXCFGRP_APPLY [1070](#)
 TOSILOAD [1070](#)
 UNIT_TEMP [1071](#)
 USRHLQ [1071](#)
 job control language (JCL)
 Analyze [198](#)
 Change [131](#)
 DMACPRt [284](#)
 Extract [262](#)
 INDEXBLD [297](#), [356](#)
 INDEXBLD (Resync) [328](#), [330](#), [331](#)
 INITIALIZE [367](#)
 OAE [756](#), [762](#)

job control language (JCL) (continued)

 ODE [732](#), [735](#)
 ODM [822](#)
 OER [784](#)
 Reload [90](#)
 Reorg [167](#)
 Tune [233](#)
 Unload [53](#)

job steps

 OPC [681](#)

JOB CARDx keyword, OSC [1062](#)

JOBN_PREFIX keyword, OSC [1062](#)

K

keyboard shortcuts [30](#)

KEYCHGCHK keyword, FPA [489](#)

KEYLABEL keyword, FPA [489](#)

KEYSEQCHK keyword, FPA [490](#)

keyword abbreviation

 FPA [381](#)

KEYWORD_DUPCHK keyword, FPO [892](#)

keywords

 FPA

 ACBCHK [438](#)

 ACBLIB [439](#)

 ADXCFCGRP [439](#)

 ALLOCONLY [440](#)

 AMSOPT [440](#)

 AREA_INFORMATION_RECORD [443](#)

 AREADSN [441](#)

 AREASCAN [442](#)

 ATASKCTL [443](#)

 ATTRIBUTE [444](#)

 AVGREC [445](#)

 BREAK [446](#)

 BUILDDSNMASK [446](#)

 BUILDMODE [447](#)

 CISIZE [448](#)

 CISNAP [448](#)

 COMPRESS [449](#)

 DATACLAS [449](#)

 DBD [450](#)

 DBRC [451](#)

 DBREC_DISCARD_MSGLVL [452](#)

 DBRLDIST [452](#)

 DDNAME [453](#)

 DISP [453](#)

 DOVFFS_PCT [455](#)

 DOVFNULLPTR_MSGLVL [455](#)

 DSNAME [456](#)

 DUPKEYACT [457](#)

 DUPKEYDSNMASK [458](#)

 DUPKEYMAX [458](#)

 EDITCOMP_SKIP [459](#)

 EFORMAT [460](#)

 EVERY [461](#)

 EXITRTN [461](#)

 EXPDT [462](#)

 FIELDS [463](#)

 FORMAT [465](#)

 FS_DOVFIOVF [466](#)

 FS_RAADOVF [467](#)

 FS_RAADOVFIOVF [467](#)

keywords (continued)

FPA (continued)

FS_RAAIOVF [468](#)
 FS_RECLAIM_REPORT [468](#)
 HISTORYDD [469](#)
 IAREA [469](#)
 ICACHE [472](#)
 ICHECK [473](#)
 ICINCOMP [474](#)
 ICNOTIFY [475](#)
 ICOUT [476](#)
 IDSNMASK [476](#)
 IDX DUMP [477](#)
 IDXMAXERROR [478](#)
 IDXPROC [478](#)
 IDXTASKCTL [479](#)
 IEXITRTN [479](#)
 IMSCATACB_INPUT [480](#)
 IMSCATACB_OUTPUT [481](#)
 IMSCATHLQ [481](#)
 IMSID [482](#)
 INDEXDBD [482](#)
 INSERT_LIMIT_COUNT [484](#)
 IOVFCI_RBA_REPORT [484](#)
 IOVFFS_PCT [485](#)
 IOVFUSED_PCT [485](#)
 ITASKCTL [486](#)
 ITKBDATA [487](#)
 ITKBLOAD [487](#)
 ITKBSRVR [488](#)
 KEYCHGCHK [489](#)
 KEYLABEL [489](#)
 KEYSEQCHK [490](#)
 LARGEREC_COUNT [491](#)
 LARGEREC_REPORT [491](#)
 LIKE [492](#)
 LIUGOPT [493](#)
 LOCATION [494](#)
 LRECL [494](#)
 MAXERROR [495](#)
 MAXFAILS [496](#)
 MGMTCLAS [496](#)
 MSTIMESTAMP [497](#)
 OAREA [497](#)
 OCACHE [499](#)
 OCHECK [500](#)
 ODSNMASK [500](#)
 OEXITRTN [502](#)
 OTASKCTL [503](#)
 PAD [504](#)
 PROC_AFT_DUPKEY [504](#)
 PTRCHK [505](#)
 PTRCHKLVL [506](#)
 RAAFS_PCT [507](#)
 RAP_LONGSYN_PCT [507](#)
 RAPOVFL_PCT [508](#)
 RBASEFS [509](#)
 RDOVFFS [509](#)
 RECIO_AVG [510](#)
 RECIO_MAX [511](#)
 RECIOVF_PCT [510](#)
 RECOVOPT [511](#)
 REORGLIM [513](#)
 REPORTDD [513](#)

keywords (continued)

FPA (continued)

RESTART [514](#)
 RESTART_COUNT [515](#)
 RESYNCDSNMASK [515](#)
 RESYNCDUMP [516](#)
 RESYNCDUMPMAX [517](#)
 RESYNCMODE [517](#)
 RETPD [519](#)
 RFSSCAN [519](#)
 RFSSTAT [520](#)
 RMOD [521](#)
 RMODLIB [521](#)
 ROOT [522](#)
 ROOTIO_AVG [522](#)
 ROOTIO_MAX [523](#)
 SCAN [523](#)
 SDEP [524](#)
 SDEP_BYPASS_MSGLVL [525](#)
 SDEP_ORPHAN_MSGLVL [527](#)
 SDEP_PTRIOVF_MSGLVL [528](#)
 SDEPCHKLVL [526](#)
 SEGLDIST [529](#)
 SEGLSTAT [529](#)
 SEGMENT [530](#)
 SEGREC_PREFIX [531](#)
 SEGREC_SUFFIX [533](#)
 SENSOR [535](#)
 SENSOR_DBREC [536](#)
 SENSOR_PTRERR [537](#)
 SENSOR_UOW [537](#)
 SENSOR_UOW_REPORT [538](#)
 SITE_DSNAME [539](#)
 SITE_MEMBER [540](#)
 SKIP [540](#)
 SORT [541](#)
 SORTOPT_FILSZ [542](#)
 SPACE [543](#)
 SSP [544](#)
 STARTUOW [545](#)
 STOPAFT [545](#)
 STOPUOW [546](#)
 STORCLAS [546](#)
 SYN_LEN_AVG [547](#)
 SYN_MAX [548](#)
 SYN_PCT_LEN [548](#)
 SYNC [547](#)
 TARDSNMASK [549](#)
 TOICTL [550](#)
 TOINUM [551](#)
 TOITIME [551](#)
 TOIXCFGRP [552](#)
 UNIT [552](#)
 UOW [553](#)
 UOW_REPORT [555](#)
 UOWDOVF_PCT [553](#)
 UOWIOVF_AVG [554](#)
 UOWIOVF_MAX [554](#)
 UOWIOVF_PCT [555](#)
 VOLCNT [556](#)
 VOLSER [556](#)
 WHERE [557](#)
 FPO
 ADXCFGRP [861](#)

keywords (continued)

FPO (continued)

[AFTERTIME 862](#)
[ANDFIELD 862](#)
[ARDIGIT 863](#)
[AREA 864](#)
[BREAK 865](#)
[CASE 866](#)
[CHKUOW 866](#)
[CIPRT 867](#)
[DATACLAS 867](#)
[DBRC 868](#)
[DBRLDIST 868](#)
[DDNAME 869](#)
[DOVFFS_PCT 869](#)
[DOVFNULLPTR_MSGLVL 869](#)
[DSNAME 870](#)
[ERRORACT 870](#)
[EVERY 871](#)
[EXC_RBASEFS 872](#)
[EXC_RDOVFFS 873](#)
[EXC_UDOVFFS 873](#)
[EXPAND 874](#)
[EXTEND_TYPE 875](#)
[FIELD 875](#)
[FIELDS 876, 877](#)
[FPBP64U 879](#)
[FS_DOVFIOVF 880](#)
[FS_RAADOVF 880](#)
[FS_RAADOVFIOVF 881](#)
[FS_RAAIOVF 881](#)
[FULLSTEP 882](#)
[HISTORYDD 882](#)
[IC 883](#)
[IC1OUTDD 886](#)
[IC2OUTDD 886](#)
[ICERR_CONT 884](#)
[ICEXTEND 884](#)
[ICNOTIFY 885](#)
[INDOUBT 887](#)
[IOVF_ROOT 889](#)
[IOVF_UOW 889](#)
[IOVFCI_RBA_REPORT 887](#)
[IOVFFS_PCT 887](#)
[IOVFPPOOL 888](#)
[IOVFUSED_PCT 888](#)
[ITKBDATA 890](#)
[ITKBLOAD 890](#)
[ITKBSRVR 891](#)
[KEYWORD_DUPCHK 892](#)
[LARGEREC_COUNT 893](#)
[LARGEREC_REPORT 893](#)
[MAXBUFFS 894](#)
[MAXERROR 894](#)
[MAXFAILS 895](#)
[MAXTWINCTR 895](#)
[MAXVOLS 896](#)
[MFILE 896](#)
[MGMTCLAS 897](#)
[MSG_LIM 897](#)
[MSG_NUM 898](#)
[OFILE 898](#)
[OLRECL 899](#)
[OUTPUT 899](#)

keywords (continued)

FPO (continued)

[PCHKONLY 901](#)
[PROC_AFT_ERROR 902](#)
[PTRCHK 902](#)
[RAAFS_PCT 903](#)
[RAPOVFL_PCT 903](#)
[RBASEFS 904](#)
[RCEXLOAD 904](#)
[RDOVFFS 905](#)
[RECIO_AVG 905](#)
[RECIO_MAX 906](#)
[RECIOVF_PCT 906](#)
[REORGLIM 906](#)
[REORGMSG 907](#)
[REPORTDD 907](#)
[RFILE 908](#)
[RFSSCAN 909](#)
[RFSSTAT 909](#)
[RGROUP 910](#)
[ROOTIO_AVG 910](#)
[ROOTIO_MAX 911](#)
[RUNTYPE 911](#)
[SDEP 912](#)
[SDEP_BYPASS_MSGLVL 912](#)
[SDEP_CI 913](#)
[SDEP_ORPHAN_MSGLVL 914](#)
[SDEP_PTRIOVF_MSGLVL 915](#)
[SDEPHIER 913](#)
[SEG 916](#)
[SEGLDIST 918](#)
[SEGLSTAT 918](#)
[SEGREC_PREFIX 919](#)
[SEGREC_SUFFIX 920](#)
[SKIP 926](#)
[SORT 927](#)
[STARTUOW 928](#)
[STAT 928](#)
[STOPAFT 929](#)
[STOPUOW 930](#)
[STORCLAS 930](#)
[SYN_LEN_AVG 930](#)
[SYN_MAX 931](#)
[SYN_PCT_LEN 931](#)
[SYNTBL 932](#)
[TARDSNMASK 932](#)
[TARECGEN 933](#)
[TIMECHECK 934](#)
[TOITIME 935](#)
[TOIXCFGR 935](#)
[TYPRUN 935](#)
[UOW_REPORT 939](#)
[UOWDOVF_PCT 937](#)
[UOWIOVF_AVG 937](#)
[UOWIOVF_MAX 938](#)
[UOWIOVF_PCT 938](#)
[VOLUME 939](#)
[VOLUME_EXT 939](#)

OSC

[ANALYZE 1040](#)
[ATTRIBUTE 1048](#)
[COMPRESS 1052](#)
[CONFIGLIB_DSN 1052](#)
[DATACOMP_SUFFIX 1052](#)

keywords (continued)

OSC (continued)

DBD [1040](#)
DBDLIB_DSN [1053](#)
DBR_NOFEOV [1054](#)
DBR_NOFEOV_APPLY [1054](#)
DBRCGRP [1053](#)
DBRCGRP_APPLY [1054](#)
DCLASS_TEMP [1055](#)
DDNPREFIX_NEW [1055](#)
DDNPREFIX_SHADOW [1056](#)
DDNPREFIX_UIC [1056](#)
DSNMASK_NEW [1056](#)
DSNMASK_SHADOW [1057](#)
DSNMASK_UIC [1057](#)
ENQDSN [1058](#)
ERRORACT [1041](#)
FPLIB_DSN [1058](#)
IAREA [1041](#)
IMSCATHLQ [1036](#)
IMSID [1037](#)
IMSID_APPLY [1059](#)
IMSLIB_DSN [1059](#)
IMSLIB_DSN_APPLY [1059](#)
IMSPLEX [1060](#)
IMSPLEX_APPLY [1060](#)
IMSRESLB_EDC_DSN [1060](#)
IMSRESLB_RND_DSN [1061](#)
INSERT_LIMIT_COUNT [1072](#)
INTRDR [1061](#)
ITASKCTL [1042](#)
ITKBSRVR [1037](#)
JCLDSN [1061](#)
JOB CARDx [1062](#)
JOB_PREFIX [1062](#)
LOCATION [1072](#)
LOGDSN [1063](#)
MCLASS_TEMP [1063](#)
MSTIMESTAMP [1037](#)
NBA_APPLY [1064](#)
NEWDBDLIB_DSN [1064](#)
NEWRESLB_EDC_DSN [1064](#)
NEWRESLB_RND_DSN [1065](#)
OAREA [1042](#), [1048](#)
OBA_APPLY [1065](#)
ORIGADS_PROC [1043](#)
OTASKCTL [1044](#)
PAD [1044](#)
PROCLIB_DSN [1066](#)
PROCLIB_DSN_APPLY [1066](#)
PSBNAME_APPLY [1066](#)
RECONx_DSN [1067](#)
RECONx_DSN_APPLY [1067](#)
REPLRAND_AREALVL [1045](#)
RUNTYPE [1045](#)
SCAN [1038](#)
SCLASS_TEMP [1067](#)
SEGMENT [1073](#)
SGLXLOAD [1068](#)
SKIP_OLC [1046](#)
SPACE [1068](#)
SPOC_WAITTIME [1069](#)
SYSAFF [1069](#)
SYSAFF_APPLY [1070](#)

keywords (continued)

OSC (continued)

TAKEOVER_TIME [1046](#)
TOIXCFGRP [1038](#)
TOIXCFGRP_APPLY [1070](#)
TOSILOAD [1070](#)
UNIT_TEMP [1071](#)
UPDATE_NUMBER [1046](#)
USR_MAX [1047](#)
USRHLQ [1071](#)

OSM

ACBLIB [987](#)
ACTION [984](#)
ADXCFCGRP [980](#)
DATACLAS [988](#)
DBD [985](#)
DDNAME [988](#)
DSNAME [989](#)
ERRORACT [993](#)
EXTEND_TYPE [989](#)
FPBP64U [994](#)
IAREA [985](#)
ICNOTIFY [981](#)
ICOUT [981](#)
IMSID [982](#)
IOVF_UOW [990](#)
ITKBDATA [982](#)
ITKBSRVR [983](#)
MAXBUFFS [994](#)
MAXFAILS [995](#)
MAXVOLS [990](#)
MGMTCLAS [990](#)
RBASEFS [995](#)
RDOVFFS [996](#)
REORGATIOVFEXT [986](#)
REORGLIM [997](#)
REORGMSG [997](#)
RFSSCAN [998](#)
RFSSTAT [998](#)
SCAN [983](#)
SCANDELT [986](#)
SDEP_CI [991](#)
STORCLAS [991](#)
VOLUME [992](#)
VOLUME_EXT [992](#)

L

LARGEREC_COUNT keyword, FPA [491](#)
LARGEREC_COUNT keyword, FPO [893](#)
LARGEREC_REPORT keyword, FPA [491](#)
LARGEREC_REPORT keyword, FPO [893](#)
largest database record layout [1176](#)
Largest Database Records report, FPA [603](#)
legal notices
 cookie policy [1437](#)
 notices [1437](#)
 programming interface information [1437](#)
 trademarks [1437](#)
LIKE keyword, FPA [492](#)
LIUGOPT keyword, FPA [493](#)
LOADCTL subcommand
 keywords for CHANGE command
 INSERT_LIMIT_COUNT [154](#)

LOADCTL subcommand (*continued*)

keywords for CHANGE command (*continued*)

LOCATION [154](#)

SEGMENT [154](#)

keywords for RELOAD command

INSERT_LIMIT_COUNT [113](#)

LOCATION [113](#)

SEGMENT [113](#)

keywords for REORG command

INSERT_LIMIT_COUNT [186](#)

LOCATION [186](#)

SEGMENT [186](#)

keywords, OSC

INSERT_LIMIT_COUNT [1072](#)

LOCATION [1072](#)

SEGMENT [1073](#)

LOCATION keyword, FPA [494](#)

LOCATION keyword, OSC [1072](#)

LOGDSN keyword, OSC [1063](#)

LRECL and BLKSIZE, OPC [681](#)

LRECL keyword, FPA [494](#)

M

MADS [4](#)

MANAGE command

keywords, OSM

ACTION [984](#)

DBD [985](#)

IAREA [985](#)

REORGATIOVFEXT [986](#)

SCANDELTD [986](#)

Management Console [16](#)

manual recovery procedure [759](#)

MAXBUFFS keyword, FPO [894](#)

MAXBUFFS keyword, OSM [994](#)

MAXERROR keyword, FPA [495](#)

MAXERROR keyword, FPO [894](#)

MAXFAILS keyword, FPA [496](#)

MAXFAILS keyword, FPO [895](#)

MAXFAILS keyword, OSM [995](#)

MAXTWINCTR keyword, FPO [895](#)

MAXVOLS keyword, FPO [896](#)

MAXVOLS keyword, OSM [990](#)

MCLASS_TEMP keyword, OSC [1063](#)

message and return code table [948](#)

message file, OPC [896](#)

messages

FPA

Analyze process [1301](#)

area I/O process [1409](#)

Build Index process and Analyze (index verify)

process [1308](#)

checkpoint restart process [1389](#)

command process [1315](#)

common routine process [1405](#)

data set management process [1340](#)

DB Sensor process [1377](#)

DBRC interface process [1413](#)

DMAC Print process [1355](#)

DMB process [1394](#)

Extract process [1357](#)

function task process [1343](#)

image copy process [1355](#)

messages (*continued*)

FPA (*continued*)

IMS Tools KB support [1352](#)

main process [1373](#)

Online Pointer Scan report messages [1425](#)

OSC driver job [1287](#)

OSC internal apply job [1299](#)

OSC internal job generator [1294](#)

Policy evaluation by using Policy Services [1296](#)

Process Summary of Analysis report messages [1422](#)

Reload process [1366](#)

report process [1372](#)

Resolve Pointers report messages [1416](#)

resource control process [1392](#)

thread task process [1398](#)

Unload process [1357](#)

FPB

Diagnostics Aid [1220](#)

FPO

online pointer check process [1391](#)

OSM [1218](#)

MFILE keyword, FPO [896](#)

MGMTCLAS keyword, FPA [496](#)

MGMTCLAS keyword, FPO [897](#)

MGMTCLAS keyword, OSM [990](#)

MSG_LIM keyword, FPO [897](#)

MSG_NUM keyword, FPO [898](#)

MSTIMESTAMP keyword, FPA [497](#)

MSTIMESTAMP keyword, OSC [1037](#)

N

NBA_APPLY keyword, OSC [1064](#)

NEWDBDLIB_DSN keyword, OSC [1064](#)

NEWRESLB_EDC_DSN keyword, OSC [1064](#)

NEWRESLB_RND_DSN keyword, OSC [1065](#)

notices [1437](#)

number2 parm on ROOT= parameter [751](#)

O

OAE Rename program

JCL requirements

EXEC statement [771](#)

FPXBINTP DD [771](#)

FPXIN DD [771](#)

STEPLIB DD [771](#)

SYSPRINT DD [771](#)

TIMECHECK keyword [934](#)

TOITIME keyword [935](#)

TOIXCFGR keyword [935](#)

utility control statements [771](#)

OAREA keyword, FPA [497](#)

OAREA keyword, OSC [1042](#), [1048](#)

OBA_APPLY keyword, OSC [1065](#)

OCACHE keyword, FPA [499](#)

OCHECK keyword, FPA [500](#)

ODE

exit routine interface [947](#)

exit routine overview [947](#)

ODESORT statement, ODE [736](#)

ODSNMASK keyword, FPA [500](#)

OEXITRTN keyword, FPA [502](#)
 OFILE keyword, FPO [731](#), [898](#)
 OLCUTL utility [757](#)
 OLRECL keyword, FPO [899](#)
 Online Area Extender (OAE)
 ACBGEN and DBDGEN [751](#)
 ACBLIB DD [762](#)
 activating new member in ACBLIB [757](#)
 allocating a larger ADS [749](#), [754](#)
 AREA keyword [764](#), [864](#)
 benefits [670](#)
 considerations [749](#)
 control statements [764](#), [765](#)
 DATACLAS keyword [867](#)
 DBRC keyword [868](#)
 DDNAME keyword [764](#), [869](#)
 deleting SDEP segments [750](#)
 DSNAME keyword [870](#)
 EXIT keyword
 EXTEND keyword [765](#)
 IOVF keyword [765](#)
 RESTART keyword [765](#)
 SDEP keyword [765](#)
 SDEPWARN keyword [765](#)
 TEST keyword [765](#)
 EXTEND statement [766](#)
 EXTEND_TYPE keyword [875](#)
 extending the same area twice [752](#)
 GLOBAL statement [765](#), [847](#)
 GO keyword [765](#)
 ICEXTEND keyword [884](#)
 IOVF_ROOT keyword [889](#)
 IOVF_UOW keyword [889](#)
 ITKBLOAD keyword [890](#)
 ITKBSRVR keyword [891](#)
 JCL
 EXEC statement [756](#)
 MAXVOLS keyword [896](#)
 MGMTCLAS keyword [897](#)
 MSG_LIM keyword [897](#)
 MSG_NUM keyword [898](#)
 OVERRIDE statement [850](#)
 performance [754](#)
 recovery [757](#)
 registering new ADS with DBRC [754](#)
 report [768](#), [770](#)
 restarting [757](#)
 restrictions [748](#), [751](#)
 RSR [748](#)
 RUNTYPE keyword [911](#)
 SDEP_CI keyword [913](#)
 STORCLAS keyword [930](#)
 SYSIN file [762](#)
 SYSOUT file [762](#)
 SYSPRINT file [762](#)
 TYPE keyword [764](#)
 VOLUME keyword [939](#)
 VOLUME_EXT keyword [939](#)
 VSO
 reloading area [757](#)
 removing area [756](#)
 restriction [748](#)
 Online Area Extender Statistics report, FPO [768](#), [770](#)
 Online Data Extract (ODE)

Online Data Extract (ODE) (*continued*)
 AFTERTIME keyword [862](#)
 and SDEP segments [730](#), [732](#)
 and Sort program [730](#)
 ANDFIELD keyword [862](#)
 AREA keyword [864](#)
 area-level control [730](#)
 benefits [670](#)
 BREAK keyword [865](#)
 CASE keyword [866](#)
 EVERY keyword [871](#)
 EXPAND keyword [874](#)
 EXTRACT statement [737](#), [738](#), [843](#)
 FIELD keyword [875](#)
 FIELDS keyword [876](#), [877](#)
 FPUTIL procedure [732](#)
 FPXCTL statement [738](#), [844](#), [858](#)
 FPXOFILE [731](#)
 GLOBAL statement [737](#), [847](#)
 INDOUBT keyword [887](#)
 IOVFPOOL keyword [888](#)
 JCL
 control statement [735](#)
 EXEC statement [732](#)
 ODESORT statement [736](#)
 STEPLIB statement [734](#)
 SYSIN statement [734](#)
 SYSOUT statement [736](#)
 TOOLSNP1 statement [736](#)
 TOOLSNP2 statement [736](#)
 KEYWORD_DUPCHK keyword [892](#)
 MAXBUFFS keyword [894](#)
 MAXTWINCTR keyword [895](#)
 MFILE keyword [896](#)
 MSG_LIM keyword [897](#)
 MSG_NUM keyword [898](#)
 OFILE keyword [731](#), [898](#)
 OLRECL keyword [899](#)
 output file format [730](#)
 OUTPUT keyword [899](#)
 OUTPUT statement [739](#), [848](#)
 OVERRIDE statement [850](#)
 PCHKONLY keyword [901](#)
 performance [732](#)
 PROC_AFT_ERROR keyword [902](#)
 RCEXLOAD keyword [904](#)
 recovery and restart [732](#)
 Restrictions
 /STA AREA [731](#)
 EPST [731](#)
 EXPAND [731](#)
 FPXCTL [731](#)
 IFP separate job step [731](#)
 PRELOAD attribute [731](#)
 run-level control [730](#)
 SDEP_BYPASS_MSGLVL keyword [912](#)
 SDEPHIER keyword [913](#)
 SEG keyword [916](#)
 SEGREC_PREFIX keyword [919](#)
 SEGREC_SUFFIX keyword [920](#)
 SELECT statement [739](#), [857](#)
 SKIP keyword [926](#)
 SORT keyword [927](#)
 STAT keyword [928](#)

Online Data Extract (ODE) *(continued)*

STOPAFT keyword [929](#)
USER_RECORD statement [740](#), [861](#)
utility control statements [737](#)

Online Data Extract Statistics report, FPO [742](#)

Online DMAC Print (ODM)

AREA keyword [864](#)
benefits [671](#)
CIPRT keyword [867](#)
DBRC keyword [868](#)
DMACPRT statement [825](#), [841](#)
GLOBAL statement [825](#), [847](#)
ITKBLOAD keyword [890](#)
ITKBSRVR keyword [891](#)
JCL

EXEC statement [822](#)
FPXIN [823](#)
FPXRPTS [824](#)
STEPLIB [823](#)
SYSPRINT [824](#)

KEYWORD_DUPCHK keyword [892](#)
MFILE keyword [896](#)
MSG_LIM keyword [897](#)
MSG_NUM keyword [898](#)
OVERRIDE statement [850](#)
PCHKONLY keyword [901](#)
PROC_AFT_ERROR keyword [902](#)
RFILE keyword [908](#)
utility control statements [824](#)

Online Expert Reorganization (OER)

AREA keyword [864](#)
benefits [671](#)
CHKUOW keyword [866](#)
DBRC keyword [868](#)
DOVFFS_PCT keyword [869](#)
DOVFNULLPTR_MSGLVL keyword [869](#)
ERRORACT keyword [870](#)
EXC_RBASEFS keyword [872](#)
EXC_RDOVFFS keyword [873](#)
EXC_UDOVFFS keyword [873](#)
FPBP64U keyword [879](#)
FS_DOVFIOVF keyword [880](#)
FS_RAADOVF keyword [880](#)
FS_RAADOVFIOVF keyword [881](#)
FS_RAAIOVF keyword [881](#)
FULLSTEP keyword [882](#)
GLOBAL statement [796](#)
HISTORYDD keyword [882](#)
IC keyword [883](#)
IC1OUTDD keyword [886](#)
IC2OUTDD keyword [886](#)
ICNOTIFY keyword [885](#)
IOVFCI_RBA_REPORT keyword [887](#)
IOVFFS_PCT keyword [887](#)
IOVFPPOOL keyword [888](#)
IOVFUSED_PCT keyword [888](#)
ITKBDATA keyword [890](#)
ITKBLOAD keyword [890](#)
ITKBSRVR keyword [891](#)
JCL

EXEC statement [784](#)
FPXIN [787](#)
FPXRPTS [788](#)
SYSPRINT [789](#)

Online Expert Reorganization (OER) *(continued)*

KEYWORD_DUPCHK keyword [892](#)
LARGEREC_COUNT keyword [893](#)
LARGEREC_REPORT keyword [893](#)
MAXERROR keyword [894](#)
MAXFAILS keyword [895](#)
MAXTWINCTR keyword [895](#)
MFILE keyword [896](#)
MSG_LIM keyword [897](#)
MSG_NUM keyword [898](#)
OVERRIDE statement [850](#)
PCHKONLY keyword [901](#)
PROC_AFT_ERROR keyword [902](#)
PTRCHK keyword [902](#)
PTRCHK statement [798](#), [850](#)
RAAFS_PCT keyword [903](#)
RAPOVFL_PCT [903](#)
RBASEFS [904](#)
RCEXLOAD keyword [904](#)
RDOVFFS keyword [905](#)
RECIO_AVG keyword [905](#), [906](#)
RECIOVF_PCT keyword [906](#)
REORG statement [797](#), [854](#)
REORGLIM keyword [906](#)
REORGMMSG keyword [907](#)
REPORT statement [800](#), [856](#)
REPORTDD keyword [907](#)
RFILE keyword [908](#)
RFSSCAN keyword [909](#)
RFSSTAT keyword [909](#)
ROOTIO_AVG keyword [910](#)
ROOTIO_MAX keyword [911](#)
SDEP keyword [912](#)
SDEP_ORPHAN_MSGLVL keyword [914](#)
SDEP_PTRIOVF_MSGLVL keyword [915](#)
SENSOR keyword [922](#)
SENSOR_DBREC keyword [923](#)
SENSOR_PTRERR keyword [924](#)
SENSOR_UOW keyword [925](#)
SENSOR_UOW_REPORT keyword [925](#)
STARTUOW keyword [928](#)
STAT keyword [928](#)
STOPUOW keyword [930](#)
SYN_LEN_AVG keyword [930](#)
SYN_MAX keyword [931](#)
SYN_PCT_LEN keyword [931](#)
SYNTBL keyword [932](#)
TARECGEN keyword [933](#)
THRESHOLD statement [801](#), [859](#)
TYPRUN keyword [935](#)
UOW_REPORT keyword [939](#)
UOWDOVF_PCT keyword [937](#)
UOWIOVF_AVG keyword [937](#)
UOWIOVF_MAX keyword [938](#)
UOWIOVF_PCT keyword [938](#)
utility control statements [796](#)

Online Expert Reorganization (ORE)

GLOBAL statement [847](#)

Online Pointer Checker (OPC)

ADXCFGRP keyword [861](#)
ANALYZE statement [697](#), [837](#)
ARDIGIT keyword [863](#)
AREA keyword [864](#)
benefits [669](#)

Online Pointer Checker (OPC) (continued)

CI SNAP Dump [719](#)
DAR120 [680](#)
DAR13SO [680](#)
DAR3CSO [680](#)
DBRC keyword [868](#)
DOVFFS_PCT keyword [869](#)
DOVNULLPTR_MSGLVL keyword [869](#)
ERRORACT keyword [870](#)
EXC_RBASEFS keyword [872](#)
EXC_RDOVFFS keyword [873](#)
EXC_UDOVFFS keyword [873](#)
EXEC FPUTIL statement [684](#)
EXTEND statement [842](#)
FS_DOVFIOVF keyword [880](#)
FS_RAADOVF keyword [880](#)
FS_RAADOVFIOVF keyword [881](#)
FS_RAIOVF keyword [881](#)
FULLSTEP keyword [882](#)
GLOBAL statement [696](#), [847](#)
HISTORYDD keyword [882](#)
IC keyword [883](#)
IC1OUTDD keyword [886](#)
IC2OUTDD keyword [886](#)
ICERR_CONT keyword [884](#)
ICNOTIFY keyword [885](#)
image copy parameters [870](#), [886](#)
IOVF buffer pool [888](#)
IOVFCI_RBA_REPORT keyword [887](#)
IOVFFS_PCT keyword [887](#)
IOVFPOOL keyword [888](#)
IOVFUSED_PCT keyword [888](#)
ITKBLOAD keyword [890](#)
ITKBSRVR keyword [891](#)
JCL
 DAR120 statement [685](#)
 DAR13SO statement [685](#)
 DAR3CSO statement [685](#)
 DASYSIN statement [688](#)
 EXEC statement [684](#)
 HFPABEND statement [685](#)
 HFPPRINT statement [685](#)
 i001SMSG statement [685](#)
 i001WKnn statement [685](#)
 SNAPPIT statement [685](#)
 SORT12 statement [685](#)
 SORT3CS statement [685](#)
 SYSIN statement [688](#)
 SYSOUT statement [685](#)
 TARnnnnn statement [685](#)
 TOOLSNP1 statement [692](#)
 TOOLSNP2 statement [692](#)
KEYWORD_DUPCHK keyword [892](#)
LARGEREC_COUNT keyword [893](#)
LARGEREC_REPORT keyword [893](#)
LRECL and BLKSIZE [681](#)
MAXERROR keyword [894](#)
MAXTWINCTR keyword [895](#)
message file [896](#)
MFILE keyword [896](#)
operating mode [935](#)
output data sets [676](#), [680](#)
OVERRIDE statement [850](#)
PCHKONLY keyword [901](#)

Online Pointer Checker (OPC) (continued)

PROC_AFT_ERROR keyword [902](#)
RAAFS_PCT keyword [903](#)
RAPOVFL_PCT [903](#)
RCEXLOAD keyword [904](#)
RECIO_AVG keyword [905](#), [906](#)
RECIOVF_PCT keyword [906](#)
REPORT statement [699](#), [856](#)
REPORTDD keyword [907](#)
RFILE keyword [908](#)
RGROUP keyword [910](#)
ROOTIO_AVG keyword [910](#)
ROOTIO_MAX keyword [911](#)
SDEP keyword [912](#)
SDEP_ORPHAN_MSGLVL keyword [914](#)
SDEP_PTRIOVF_MSGLVL keyword [915](#)
SENSOR keyword [922](#)
SENSOR_DBREC keyword [923](#)
SENSOR_PTRERR keyword [924](#)
SENSOR_UOW keyword [925](#)
SENSOR_UOW_REPORT keyword [925](#)
SNAPPIT [680](#), [718](#)
SORT12 [680](#)
SORT3CS [680](#)
space requirements for output data sets [681](#)
STARTUOW keyword [928](#)
STAT keyword [718](#), [928](#)
STOPUOW keyword [930](#)
SYN_LEN_AVG keyword [930](#)
SYN_MAX keyword [931](#)
SYN_PCT_LEN keyword [931](#)
SYNTBL keyword [932](#)
SYSPRINT [680](#)
SYSPRINT data set [718](#)
TARECGEN keyword [933](#)
THRESHOLD statement [700](#), [859](#)
TYPRUN keyword [935](#)
UOW_REPORT keyword [939](#)
UOWDOVF_PCT keyword [937](#)
UOWIOVF_AVG keyword [937](#)
UOWIOVF_MAX keyword [938](#)
UOWIOVF_PCT keyword [938](#)
Online Pointer Checker and DEDB Pointer Checker [673](#)
Online Pointer Checker Statistics report, FPO [704](#)
Online Reorganization Expert (OER)
 ADXCFGRP keyword [861](#)
 ARDIGIT keyword [863](#)
 RGROUP keyword [910](#)
Online Space Management utility
 active autonomics [963](#)
 DD statements [971](#)
 JCL requirements [971](#)
 policies [964](#)
 region size [959](#)
 reports [1001](#)
 scenarios [960](#)
 stand-alone mode [959](#)
Online Structure Change utility
 JCL requirements [1027](#)
 reports [1075](#)
Online Tools package [673](#)
OPC
 exit routine interface [947](#)
 exit routine overview [947](#)

- operating mode, OPC [935](#)
- ORIGADS_PROC keyword, OSC [1043](#)
- original expansion [671](#), [765](#)
- OSC [4](#)
- OSC internal jobs [1010](#)
- OSM [4](#)
- OTASKCTL keyword, FPA [503](#)
- OTASKCTL keyword, OSC [1044](#)
- Output File report, FPA [580](#)
- output files
 - OPC [676](#), [680](#), [702](#), [719](#)
 - SNAPPIT [719](#)
- OUTPUT keyword, FPO [899](#)
- OUTPUT statement
 - examples [849](#)
 - keywords
 - AFTERTIME [862](#)
 - EVERY [871](#)
 - FIELDS [876](#)
 - SEG [916](#)
 - SKIP [926](#)
 - STOPAFT [929](#)
- OUTPUT subcommand
 - keywords for EXTRACT command
 - FIELDS [275](#)
 - SEGMENT [275](#)
 - WHERE [275](#)
- overflow2 parm on ROOT= parameter [751](#)
- OVERRIDE statement
 - keywords
 - MSG_LIM [897](#)
 - MSG_NUM [898](#)
- overview
 - IMS Fast Path Advanced Tool [33](#)
 - IMS Fast Path Online Tools [669](#)
 - Online Space Management utility [955](#)
 - Online Structure Change utility [1007](#)

P

- PAD keyword, FPA [504](#)
- PAD keyword, OSC [1044](#)
- parameters, FPA [381](#)
- PCHKONLY keyword, FPO [901](#)
- physical child first pointer, OPC [674](#)
- physical child last pointer, OPC [674](#)
- physical twin forward pointer, OPC [674](#)
- Pointer Chain Reconstruction report, FPA [587](#)
- pointer checker products [673](#)
- pointer checking [674](#)
- Pointer Checksum report, FPA [586](#)
- Pointer Segment Dump report, FPA [633](#)
- pointer types verified [674](#)
- pointers
 - to root segments, OAE [751](#)
 - to SDEP segments, OAE [751](#)
- PRELOAD attribute, ODE [731](#)
- prerequisite knowledge [28](#)
- prerequisite publications [28](#)
- PROC_AFT_DUPKEY keyword, FPA [504](#)
- PROC_AFT_ERROR keyword, FPO [902](#)
- Process Summary of Analysis report, FPA [590](#)
- Process Summary report, FPA [636](#)
- Processing report, FPA [565](#)

- Processing report, OSM [1001](#)
- PROCLIB_DSN keyword, OSC [1066](#)
- PROCLIB_DSN_APPLY keyword, OSC [1066](#)
- programming interface information [1437](#)
- PSBNAME_APPLY keyword, OSC [1066](#)
- PTRCHK keyword, FPA [505](#)
- PTRCHK keyword, FPO [902](#)
- PTRCHK statement
 - IC keyword [883](#)
 - keywords
 - ARDIGIT [863](#)
 - CHKUOW [866](#)
 - DOVFNULPTR_MSGLVL [869](#)
 - EXC_RBASEFS [872](#)
 - EXC_RDOVFFS [873](#)
 - EXC_UDOVFFS [873](#)
 - FULLSTEP [882](#)
 - HISTORYDD [882](#)
 - IC1OUTDD [886](#)
 - IC2OUTDD [886](#)
 - ICNOTIFY [885](#)
 - IOVFPOOL [888](#)
 - MAXERROR [894](#)
 - RGROUP [910](#)
 - SDEP [912](#)
 - SDEP_ORPHAN_MSGLVL [914](#)
 - SDEP_PTRIOVF_MSGLVL [915](#)
 - STAT [928](#)
 - SYNTBL [932](#)
 - TARDSNMASK [932](#)
 - TARECGEN [933](#)
 - TYPRUN [935](#)
- PTRCHKLVL keyword, FPA [506](#)

R

- RAAFS_PCT keyword, FPA [507](#)
- RAAFS_PCT keyword, FPO [903](#)
- RACF considerations, FPO [951](#)
- RAP_LONGSYN_PCT keyword, FPA [507](#)
- RAP, OAE [751](#)
- RAPOVFL_PCT keyword, FPA [508](#)
- RAPOVFL_PCT keyword, FPO [903](#)
- RBA of Used IOVF CI report, FPA [656](#)
- RBASEFS keyword, FPA [509](#)
- RBASEFS keyword, FPO [904](#)
- RBASEFS keyword, OSM [995](#)
- RCEXLOAD keyword, FPO [904](#)
- RDOVFFS keyword, FPA [509](#)
- RDOVFFS keyword, FPO [905](#)
- RDOVFFS keyword, OSM [996](#)
- reader comment form [28](#)
- RECIO_AVG keyword, FPA [510](#)
- RECIO_AVG keyword, FPO [905](#)
- RECIO_MAX keyword, FPA [511](#)
- RECIO_MAX keyword, FPO [906](#)
- RECIOVF_PCT keyword, FPA [510](#)
- RECIOVF_PCT keyword, FPO [906](#)
- RECON [4](#)
- RECONx_DSN keyword, OSC [1067](#)
- RECONx_DSN_APPLY keyword, OSC [1067](#)
- record layout
 - build pointer segment records layout [1193](#)
 - duplicate pointer segment records layout [1193](#)

record layout (*continued*)

DURDBDFN record layout [1185](#)
ERRORSEG record layout [1185](#)
extracted record layout [1186](#)
History file record layout [1173](#)
History2 file record layout [1175](#)
largest database record layout [1176](#)
OUTFILE record layout [1190](#)
resync pointer segment records layout [1193](#)
sequence error segment record layout [1183](#)
total segment record layout [1183](#)
unloaded segment record layout [1177](#)
unverified resync pointer segment records layout [1193](#)
UOW group definition record layout [1191](#)

RECOVPT keyword, FPA [511](#)

RELOAD command

keywords

ACBCHK [111](#)
ATASKCTL [111](#)
CISNAP [111](#)
DBD [111](#)
EXITRTN [111](#)
FORMAT [111](#)
HISTORYDD [111](#)
IAREA [111](#)
ICHECK [111](#)
ICNOTIFY [111](#)
ICOUT [111](#)
IDSNMASK [111](#)
ITASKCTL [111](#)
KEYCHGCHK [111](#)
KEYSEQCHK [111](#)
MAXERROR [111](#)
OAREA [111](#)
OCACHE [111](#)
OCHECK [111](#)
ODSNMASK [111](#)
PTRCHKLVL [111](#)
SDEP_ORPHAN_MSGLVL [111](#)
SDEPCHKLVL [111](#)
SEGREC_PREFIX [112](#)
SEGREC_SUFFIX [112](#)
SENSOR [112](#)
SENSOR_DBREC [112](#)
SENSOR_PTRERR [112](#)
SENSOR_UOW [112](#)
SENSOR_UOW_REPORT [112](#)
SORT [112](#)
SORTOPT_FILSZ [112](#)
SSP [112](#)

Reloaded report, FPA [578](#)

RELOCATE_SDEP [71](#), [512](#)

REORG command

keywords

ATASKCTL [185](#)
CISNAP [185](#)
DBD [185](#)
HISTORYDD [185](#)
IAREA [185](#)
ICACHE [185](#)
ICNOTIFY [185](#)
ICOUT [185](#)
IDSNMASK [185](#)
ITASKCTL [185](#)

REORG command (*continued*)

keywords (*continued*)

MAXFAILS [185](#)
PTRCHKLVL [185](#)
RBASEFS [185](#)
RDOVFFS [185](#)
RECOVPT [185](#)
REORGLIM [185](#)
RFSSCAN [185](#)
RFSSTAT [185](#)
SDEP_ORPHAN_MSGLVL [185](#)
SDEP_PTRIOVF_MSGLVL [185](#)
SDEPCHKLVL [185](#)
SENSOR [185](#)
SENSOR_DBREC [185](#)
SENSOR_PTRERR [186](#)
SENSOR_UOW [186](#)
SENSOR_UOW_REPORT [186](#)
SORTOPT_FILSZ [186](#)

REORG statement

keywords

AREA [864](#)
ERRORACT [870](#)
FPBP64U [879](#)
MAXBUFFS [894](#)
MAXFAILS [895](#)
MFILE [896](#)
PCHKONLY [901](#)
PROC_AFT_ERROR [902](#)
PTRCHK [902](#)
RBASEFS [904](#)
RCEXLOAD [904](#)
RDOVFFS [905](#)
REORGLIM [906](#)
REORGMSG [907](#)
RFILE [908](#)
RFSSCAN [909](#)
RFSSTAT [909](#)

reorganization free space score (RFS score) [779](#)

Reorganization Processing Summary report, FPA [615](#)

Reorganization Processing Summary report, FPO [803](#)

REORGATIOVFEXT keyword, OSM [986](#)

REORGLIM keyword, FPA [513](#)

REORGLIM keyword, FPO [906](#)

REORGLIM keyword, OSM [997](#)

REORGMSG keyword, FPO [907](#)

REORGMSG keyword, OSM [997](#)

REPLRAND_AREALVL keyword, OSC [1045](#)

REPORT statement

keywords

DBRLDIST [868](#)
IOVFCI_RBA_REPORT [887](#)
LARGEREC_COUNT [893](#)
LARGEREC_REPORT [893](#)
REPORTDD [907](#)
SEGLDIST [918](#)
SEGLSTAT [918](#)
STARTUOW [928](#)
STOPUOW [930](#)
UOW_REPORT [939](#)

REPORT subcommand

keywords for ANALYZE command

DBRLDIST [221](#)
IOVFCI_RBA_REPORT [221](#)

REPORT subcommand *(continued)*

keywords for ANALYZE command *(continued)*

LARGEREC_COUNT [221](#)
LARGEREC_REPORT [221](#)
REPORTDD [221](#)
SEGLDIST [221](#)
SEGLSTAT [221](#)
STARTUOW [221](#)
STOPUOW [221](#)
UOW_REPORT [221](#)

keywords for CHANGE command

DBRLDIST [154](#)
IOVFCI_RBA_REPORT [154](#)
LARGEREC_COUNT [154](#)
LARGEREC_REPORT [154](#)
REPORTDD [154](#)
SEGLDIST [154](#)
SEGLSTAT [154](#)
STARTUOW [154](#)
STOPUOW [154](#)
UOW_REPORT [154](#)

keywords for RELOAD command

DBRLDIST [113](#)
IOVFCI_RBA_REPORT [113](#)
LARGEREC_COUNT [113](#)
LARGEREC_REPORT [113](#)
REPORTDD [113](#)
SEGLDIST [113](#)
SEGLSTAT [113](#)
STARTUOW [113](#)
STOPUOW [113](#)
UOW_REPORT [113](#)

keywords for REORG command

DBRLDIST [186](#)
IOVFCI_RBA_REPORT [186](#)
LARGEREC_COUNT [186](#)
LARGEREC_REPORT [186](#)
REPORTDD [186](#)
SEGLDIST [186](#)
SEGLSTAT [186](#)
STARTUOW [186](#)
STOPUOW [186](#)
UOW_REPORT [186](#)

keywords for TUNE command

DBRLDIST [251](#)
LARGEREC_COUNT [251](#)
LARGEREC_REPORT [251](#)
REPORTDD [251](#)
SEGLDIST [251](#)
SEGLSTAT [251](#)
STARTUOW [251](#)
STOPUOW [251](#)
UOW_REPORT [251](#)

REPORTDD keyword, FPA [513](#)

REPORTDD keyword, FPO [907](#)

reports

FPA

Area Tuning Summary report [649](#)
Audit report [564](#)
CI Map/CI Dump report [588](#)
Consolidated Largest Database Records report [604](#)
DB Record Length Distribution report [647](#)
DB Record Placement Analysis report [599](#)
DB Record Profile Analysis report [596](#)

reports *(continued)*

FPA *(continued)*

DBD Definition report [569](#)
DEDB Area Analysis reports [591](#)
DMACPRT report [612](#)
Extracted report [609](#)
Freespace Analysis report [592](#)
Image Copy Output report [608](#)
Initialization Processing Summary report [632](#)
Input File report [583](#)
Largest Database Records report [603](#)
Output File report [580](#)
Pointer Chain Reconstruction report [587](#)
Pointer Checksum report [586](#)
Pointer Segment Dump report [633](#)
Process Summary of Analysis report [590](#)
Process Summary report [636](#)
Processing report [565](#)
RBA of Used IOVF CI report [656](#)
Reloaded report [578](#)
Reorganization Processing Summary report [615](#)
Resolve Pointers report [585](#)
Result of Access Method Services report [631](#)
Resync Pointer Segment Dump report [629](#)
Secondary Index Analysis report [626](#)
Secondary Index Definition report [617](#)
Secondary Index Duplicate Key report [627](#)
Secondary Index Freespace Reclaim report [630](#)
Secondary Index Processing report [620](#)
Secondary Index Resync report [623](#)
Segment I/O Analysis report [600](#)
Segment Length Distribution report [645](#)
Segment Placement Analysis report [600](#)
Sensor Data Statistics report [637](#)
Site and System Default report [634](#)
Synonym Chain Analysis report [601](#)
TUNED reports [653](#)
Tuning Aid Record Data Set report [652](#)
Unloaded report [575](#)
UOW Exception report [607](#)
UOW report [606](#)
UOW Space Statistics report [616](#)

FPO

CI MAP/CI Dump report [719](#)
DMACPRT report [826](#)
Online Area Extender Statistics report [768](#), [770](#)
Online Data Extract Statistics report [742](#)
Online Pointer Checker Statistics report [704](#)
Reorganization Processing Summary report [803](#)
UOW Space Statistics report [804](#)

OSC [1075](#)

OSM

Audit report [1001](#)
Processing report [1001](#)
Sensor Data Statistics report [1004](#)

Resolve Pointers report, FPA [585](#)

RESTART keyword, FPA [514](#)

RESTART keyword, FPO [765](#)

RESTART_COUNT keyword, FPA [515](#)

restrictions

FPA [36](#)

FPO

OAE [748](#), [751](#)

ODE [731](#)

restrictions (*continued*)
 OSC [1013](#)
 RESTRUCTURE command
 keywords, OSC
 ANALYZE [1040](#)
 DBD [1040](#)
 ERRORACT [1041](#)
 IAREA [1041](#)
 ITASKCTL [1042](#)
 OAREA [1042](#)
 ORIGADS_PROC [1043](#)
 OTASKCTL [1044](#)
 PAD [1044](#)
 REPLRAND_AREALVL [1045](#)
 RUNTYPE [1045](#)
 SKIP_OLC [1046](#)
 TAKEOVER_TIME [1046](#)
 UPDATE_NUMBER [1046](#)
 USR_MAX [1047](#)
 Result of Access Method Services report [FPA 631](#)
 Resync Pointer Segment Dump report [FPA 629](#)
 resync pointer segment records [324](#)
 RESYNCDNSMASK keyword, [FPA 515](#)
 RESYNCDUMP keyword, [FPA 516](#)
 RESYNCDUMPMAX keyword, [FPA 517](#)
 RESYNCMODE keyword, [FPA 517](#)
 RETPD keyword, [FPA 519](#)
 return code
 exit routine [947](#)
 return code user exit routine [947](#)
 return codes
 diagnostics aid [1212](#)
 FPA [1211](#)
 FPO [1211](#)
 Online Space Management utility [1212](#)
 OSM [1212](#)
 RFILE keyword, [FPO 908](#)
 RFSSCAN keyword, [FPA 519](#)
 RFSSCAN keyword, [FPO 909](#)
 RFSSCAN keyword, [OSM 998](#)
 RFSSTAT keyword, [FPA 520](#)
 RFSSTAT keyword, [FPO 909](#)
 RFSSTAT keyword, [OSM 998](#)
 RGROUP keyword, [FPO 910](#)
 RMOD keyword, [FPA 521](#)
 RMODLIB keyword, [FPA 521](#)
 root addressable processing, [OPC 676](#)
 root anchor point, [OPC 674](#)
 ROOT keyword, [FPA 522](#)
 ROOT= parameter, [OAE 751](#)
 ROOTIO_AVG keyword, [FPA 522](#)
 ROOTIO_AVG keyword, [FPO 910](#)
 ROOTIO_MAX keyword, [FPA 523](#)
 ROOTIO_MAX keyword, [FPO 911](#)
 RSR [748](#)
 run-level control, [ODE 730](#)
 RUNTYPE keyword, [FPO 911](#)
 RUNTYPE keyword, [OSC 1045](#)
 RUOW, [OAE 751](#)

S

SCAN keyword, [FPA 523](#)
 SCAN keyword, [OSC 1038](#)
 SCAN keyword, [OSM 983](#)
 SCANDELT keyword, [OSM 986](#)
 scenario
 Online Space Management utility [960](#)
 restructuring DEDB areas [72](#)
 SCLASS_TEMP keyword, [OSC 1067](#)
 SDEP extension for OAE [749, 754](#)
 SDEP keyword, [FPA 524](#)
 SDEP keyword, [FPO](#)
 OPC [674](#)
 SDEP segments, deleting [750](#)
 SDEP_BYPASS_MSGLVL keyword, [FPA 525](#)
 SDEP_BYPASS_MSGLVL keyword, [FPO 912](#)
 SDEP_CI keyword, [FPO 913](#)
 SDEP_CI keyword, [OSM 991](#)
 SDEP_ORPHAN_MSGLVL keyword, [FPA 527](#)
 SDEP_ORPHAN_MSGLVL keyword, [FPO 914](#)
 SDEP_PTRIOVF_MSGLVL keyword, [FPA 528](#)
 SDEP_PTRIOVF_MSGLVL keyword, [FPO 915](#)
 SDEPCHKLVL keyword, [FPA 526](#)
 SDEPHIER keyword, [FPO 913](#)
 SDEPWARN keyword, [FPO 765](#)
 search hierarchy, [ODE 858](#)
 Secondary Index Analysis report, [FPA 626](#)
 secondary index database [196](#)
 Secondary Index Definition report, [FPA 617](#)
 Secondary Index Duplicate Key report, [FPA 627](#)
 Secondary Index Freespace Reclaim report, [FPA 630](#)
 Secondary Index Processing report, [FPA 620](#)
 Secondary Index Resync report, [FPA 623](#)
 security
 FPO [951](#)
 SEG keyword, [FPO 916](#)
 SEGLDIST keyword, [FPA 529](#)
 SEGLDIST keyword, [FPO 918](#)
 SEGLSTAT keyword, [FPA 529](#)
 SEGLSTAT keyword, [FPO 918](#)
 Segment I/O Analysis report, [FPA 600](#)
 SEGMENT keyword, [FPA 530](#)
 SEGMENT keyword, [OSC 1073](#)
 Segment Length Distribution report, [FPA 645](#)
 Segment Placement Analysis report, [FPA 600](#)
 segment record, unloaded [1203](#)
 SEGREC_PREFIX keyword, [FPA 531](#)
 SEGREC_PREFIX keyword, [FPO 919](#)
 SEGREC_SUFFIX keyword, [FPA 533](#)
 SEGREC_SUFFIX keyword, [FPO 920](#)
 SELECT statement
 examples [858](#)
 keywords
 ANDFIELD [862](#)
 FIELD [875](#)
 SEG [916](#)
 SELECT subcommand
 keywords for EXTRACT command
 EVERY [275](#)
 SEGMENT [275](#)
 SKIP [275](#)
 STOPAFT [275](#)
 WHERE [275](#)
 Sensor Data Statistics report, [FPA 637](#)
 Sensor Data Statistics report, [OSM 1004](#)
 SENSOR keyword [922](#)
 SENSOR keyword, [FPA 535](#)

SENSOR_DBREC keyword [923](#)
 SENSOR_DBREC keyword, FPA [536](#)
 SENSOR_PTRERR keyword [924](#)
 SENSOR_PTRERR keyword, FPA [537](#)
 SENSOR_UOW keyword [925](#)
 SENSOR_UOW keyword, FPA [537](#)
 SENSOR_UOW_REPORT keyword [925](#)
 SENSOR_UOW_REPORT keyword, FPA [538](#)
 separator characters, FPA [381](#)
 sequence error segment records layout [1183](#)
 service information [19](#)
 SETEXTENDPARM subcommand
 keywords, OSM
 ACBLIB [987](#)
 DATACLAS [988](#)
 DDNAME [988](#)
 DSNAME [989](#)
 EXTEND_TYPE [989](#)
 IOVF_UOW [990](#)
 MAXVOLS [990](#)
 MGMTCLAS [990](#)
 SDEP_CI [991](#)
 STORCLAS [991](#)
 VOLUME [992](#)
 VOLUME_EXT [992](#)
 SETREORGPARM subcommand
 keywords, OSM
 ERRORACT [993](#)
 FPBP64U [994](#)
 MAXBUFFS [994](#)
 MAXFAILS [995](#)
 RBASEFS [995](#)
 RDOVFFS [996](#)
 REORGLIM [997](#)
 REORGMMSG [997](#)
 RFSSCAN [998](#)
 RFSSTAT [998](#)
 setting up
 Online Space Management utility
 active autonomics [963](#)
 stand alone [959](#)
 SGLXLOAD keyword, OSC [1068](#)
 shadow expansion [671](#), [764](#)
 simplifying database maintenance tasks [9](#)
 SIMULATE subcommand
 keywords for TUNE command
 ACBLIB [249](#)
 IMSCATACB_OUTPUT [249](#)
 OAREA [249](#)
 REPORTDD [249](#)
 RMOD [249](#)
 RMODLIB [249](#)
 Site and System Default report, FPA [634](#)
 site default module [658](#)
 site default option
 FPA [657](#)
 site default table [661](#)
 SITE_DSNAME keyword, FPA [539](#)
 SITE_MEMBER keyword, FPA [540](#)
 size of OPC output data sets [681](#)
 SKIP keyword, FPA [540](#)
 SKIP keyword, FPO [926](#)
 SKIP_OLC keyword, OSC [1046](#)
 SNAPPIT
 SNAPPIT (*continued*)
 data set [680](#)
 statement [685](#)
 SNAPPIT data set [718](#), [719](#)
 SORT keyword, FPA [541](#)
 SORT keyword, FPO [927](#)
 SORT12, OPC [680](#), [681](#), [685](#)
 SORT3CS, OPC [680](#), [681](#), [685](#)
 SORTOPT_FILSZ keyword, FPA [542](#)
 SPACE keyword, FPA [543](#)
 SPACE keyword, OSC [1068](#)
 space requirements for OPC output data sets [681](#)
 SPOC_WAITTIME keyword, OSC [1069](#)
 SSP keyword, FPA [544](#)
 SSP, OPC [674](#)
 STARTUOW keyword, FPA [545](#)
 STARTUOW keyword, FPO [928](#)
 STAT keyword, FPO [928](#)
 STAT keyword, OPC [718](#)
 statistics gathering [779](#)
 STEPLIB statement
 OAE [761](#)
 ODE [734](#)
 ODM [823](#)
 STOPAFT keyword, FPA [545](#)
 STOPAFT keyword, FPO [929](#)
 STOPUOW keyword, FPA [546](#)
 STOPUOW keyword, FPO [930](#)
 STORCLAS keyword, FPA [546](#)
 STORCLAS keyword, FPO [930](#)
 STORCLAS keyword, OSM [991](#)
 subcommands, FPA
 ALLOCATE [419](#)
 AREADEF [421](#)
 BLDMFILECTL [422](#)
 FILECTL [425](#)
 IRENAME [428](#)
 LOADCTL [429](#)
 OUTPUT [431](#)
 REPORT [432](#)
 SELECT [433](#)
 SIMULATE [434](#)
 THRESHOLD [435](#)
 USER_RECORD [437](#)
 subcommands, OSC
 ALLOCATE [1047](#)
 JCLPARM [1049](#)
 LOADCTL [1071](#)
 subcommands, OSM
 SETEXTENDPARM [987](#)
 SETREORGPARM [992](#)
 summary of changes [3](#)
 support information [19](#)
 SYN_LEN_AVG keyword, FPA [547](#)
 SYN_LEN_AVG keyword, FPO [930](#)
 SYN_MAX keyword, FPA [548](#)
 SYN_MAX keyword, FPO [931](#)
 SYN_PCT_LEN keyword, FPA [548](#)
 SYN_PCT_LEN keyword, FPO [931](#)
 SYNC keyword, FPA [547](#)
 Synonym Chain Analysis report, FPA [601](#)
 syntax diagrams
 how to read [1207](#)
 SYNTBL keyword, FPO [932](#)

SYSAFF keyword, OSC [1069](#)
 SYSAFF_APPLY keyword, OSC [1070](#)
 SYSIN statement
 OAE [762](#)
 ODE [734](#)
 OPC [688](#)
 SYSOUT statement
 OAE [762](#)
 ODE [736](#)
 OPC [685](#)
 SYSPRINT statement
 OAE [762](#)
 ODM [824](#)
 OER [789](#)
 OPC [680](#), [718](#)

T

TAKEOVER_TIME keyword, OSC [1046](#)
 TARDSNMASK keyword, FPA [549](#)
 TARDSNMASK keyword, FPO [932](#)
 TARECGEN keyword, FPO [933](#)
 TARnnnnn, OPC [685](#)
 technotes [28](#)
 terminology [4](#)
 TEST keyword, FPO [765](#)
 THRESHOLD statement
 keywords
 DOVFFS_PCT [869](#)
 FS_DOVFIOVF [880](#)
 FS_RAADOVF [880](#)
 FS_RAADOVFIOVF [881](#)
 FS_RAAIOVF [881](#)
 IOVFFS_PCT [887](#)
 IOVFUSED_PCT [888](#)
 RAAFS_PCT [903](#)
 RAPOVFL_PCT [903](#)
 RECIO_AVG [905](#)
 RECIO_MAX [906](#)
 RECIOVF_PCT [906](#)
 ROOTIO_AVG [910](#)
 ROOTIO_MAX [911](#)
 SYN_LEN_AVG [930](#)
 SYN_MAX [931](#)
 SYN_PCT_LEN [931](#)
 UOWDOVF_PCT [937](#)
 UOWIOVF_AVG [937](#)
 UOWIOVF_MAX [938](#)
 UOWIOVF_PCT [938](#)

THRESHOLD subcommand
 keywords for ANALYZE command
 DOVFFS_PCT [221](#)
 FS_DOVFIOVF [221](#)
 FS_RAADOVF [221](#)
 FS_RAADOVFIOVF [221](#)
 FS_RAAIOVF [221](#)
 IOVFFS_PCT [221](#)
 IOVFUSED_PCT [221](#)
 RAAFS_PCT [221](#)
 RAPOVFL_PCT [221](#)
 RECIO_AVG [221](#)
 RECIO_MAX [221](#)
 RECIOVF_PCT [221](#)
 ROOTIO_AVG [221](#)

THRESHOLD subcommand (*continued*)
 keywords for ANALYZE command (*continued*)
 ROOTIO_MAX [221](#)
 SYN_LEN_AVG [221](#)
 SYN_MAX [221](#)
 SYN_PCT_LEN [221](#)
 UOWDOVF_PCT [221](#)
 UOWIOVF_AVG [221](#)
 UOWIOVF_MAX [221](#)
 UOWIOVF_PCT [221](#)
 keywords for CHANGE command
 DOVFFS_PCT [155](#)
 FS_DOVFIOVF [155](#)
 FS_RAADOVF [155](#)
 FS_RAADOVFIOVF [155](#)
 FS_RAAIOVF [155](#)
 IOVFFS_PCT [155](#)
 IOVFUSED_PCT [155](#)
 RAAFS_PCT [155](#)
 RAPOVFL_PCT [155](#)
 RECIO_AVG [155](#)
 RECIO_MAX [155](#)
 RECIOVF_PCT [155](#)
 ROOTIO_AVG [155](#)
 ROOTIO_MAX [155](#)
 SYN_LEN_AVG [155](#)
 SYN_MAX [155](#)
 SYN_PCT_LEN [155](#)
 UOWDOVF_PCT [155](#)
 UOWIOVF_AVG [155](#)
 UOWIOVF_MAX [155](#)
 UOWIOVF_PCT [155](#)
 keywords for RELOAD command
 DOVFFS_PCT [114](#)
 FS_DOVFIOVF [114](#)
 FS_RAADOVF [114](#)
 FS_RAADOVFIOVF [114](#)
 FS_RAAIOVF [114](#)
 IOVFFS_PCT [114](#)
 IOVFUSED_PCT [114](#)
 RAAFS_PCT [114](#)
 RAPOVFL_PCT [114](#)
 RECIO_AVG [114](#)
 RECIO_MAX [114](#)
 RECIOVF_PCT [114](#)
 ROOTIO_AVG [114](#)
 ROOTIO_MAX [114](#)
 SYN_LEN_AVG [114](#)
 SYN_MAX [114](#)
 SYN_PCT_LEN [114](#)
 UOWDOVF_PCT [114](#)
 UOWIOVF_AVG [114](#)
 UOWIOVF_MAX [114](#)
 UOWIOVF_PCT [114](#)
 keywords for REORG command
 DOVFFS_PCT [187](#)
 FS_DOVFIOVF [187](#)
 FS_RAADOVF [187](#)
 FS_RAADOVFIOVF [187](#)
 FS_RAAIOVF [187](#)
 IOVFFS_PCT [187](#)
 IOVFUSED_PCT [187](#)
 RAAFS_PCT [187](#)
 RAPOVFL_PCT [187](#)

THRESHOLD subcommand (*continued*)

keywords for REORG command (*continued*)

RECIO_AVG [187](#)
RECIO_MAX [187](#)
RECIOVF_PCT [187](#)
ROOTIO_AVG [187](#)
ROOTIO_MAX [187](#)
SYN_LEN_AVG [187](#)
SYN_MAX [187](#)
SYN_PCT_LEN [187](#)
UOWDOVF_PCT [187](#)
UOWIOVF_AVG [187](#)
UOWIOVF_MAX [187](#)
UOWIOVF_PCT [187](#)

keywords for TUNE command

DOVFFS_PCT [252](#)
FS_DOVFIOVF [252](#)
FS_RAADOVF [252](#)
FS_RAADOVFIOVF [252](#)
FS_RAAIOVF [252](#)
IOVFFS_PCT [252](#)
IOVFUSED_PCT [252](#)
RAAFS_PCT [252](#)
RAP_LONGSYN_PCT [252](#)
RAPOVFL_PCT [252](#)
RECIO_AVG [252](#)
RECIO_MAX [252](#)
RECIOVF_PCT [252](#)
ROOTIO_AVG [252](#)
ROOTIO_MAX [252](#)
SYN_LEN_AVG [252](#)
SYN_MAX [252](#)
SYN_PCT_LEN [252](#)
UOWDOVF_PCT [252](#)
UOWIOVF_AVG [252](#)
UOWIOVF_MAX [252](#)
UOWIOVF_PCT [252](#)

TIMECHECK keyword, FPO [934](#)

tips

FPA [1091](#)
FPA Analyze [1098](#)
FPA Build Index [1102](#)
FPA Change [1094](#)
FPA DMAC Print [1101](#)
FPA Extract [1100](#)
FPA Initialize [1105](#)
FPA Reload [1093](#)
FPA Reorg [1096](#)
FPA Resync [1103](#), [1105](#)
FPA Tune [1099](#)
FPA Unload [1091](#)
FPO [1106](#)
FPO OAE [1107](#)
FPO ODE [1107](#)
FPO ODM [1109](#)
FPO OER [1108](#)
FPO OPC [1106](#)
OSC [1110](#)
OSM [1109](#)

TOICTL keyword, FPA [550](#)
TOINUM keyword, FPA [551](#)
TOITIME keyword, FPA [551](#)
TOITIME keyword, FPO [935](#)
TOIXCFGR keyword, FPO [935](#)

TOIXCFGRP keyword, FPA [552](#)
TOIXCFGRP keyword, OSC [1038](#)
TOIXCFGRP_APPLY keyword, OSC [1070](#)
TOOLSNP1 statement, ODE [736](#)
TOOLSNP1 statement, OPC [692](#)
TOOLSNP2 statement, ODE [736](#)
TOOLSNP2 statement, OPC [692](#)
TOSILOAD keyword, OSC [1070](#)
total segments records file format
total segments records layout [1183](#)
trademarks [1437](#)
TUNE command

keywords

AREASCAN [249](#)
CISNAP [249](#)
DBD [249](#)
IAREA [249](#)
ICACHE [249](#)
ICHECK [249](#)
ICINCOMP [249](#)
IDSNMASK [249](#)
ITASKCTL [249](#)
MAXERROR [249](#)
OTASKCTL [249](#)
SORTOPT_FILSZ [249](#)
TARDSNMASK [249](#)

TUNED reports, FPA [653](#)

Tuning Aid Record Data Set report, FPA [652](#)

tuning aid records [231](#)

TYPE keyword, FPO [764](#)

TYPRUN keyword, FPA [679](#)

TYPRUN keyword, FPO [935](#)

U

UNIT keyword, FPA [552](#)

UNIT_TEMP keyword, OSC [1071](#)

UNLOAD command

keywords

ACBCHK [69](#)
AREA_INFORMATION_RECORD [69](#)
CISNAP [69](#)
COMPRESS [69](#)
DBD [70](#)
DBDREC_DISCARD_MSGLVL [70](#)
DOVFNULLPTR_MSGLVL [70](#)
EDITCOMP_SKIP [70](#)
EXITRTN [70](#)
FORMAT [70](#)
IAREA [70](#)
ICACHE [70](#)
ICHECK [70](#)
ICINCOMP [70](#)
ICNOTIFY [70](#)
ICOUT [70](#)
IDSNMASK [70](#)
IMSCATACB_OUTPUT [70](#)
ITASKCTL [70](#)
KEYCHGCHK [70](#)
KEYSEQCHK [70](#)
LRECL [70](#)
OAREA [70](#)
OCHECK [70](#)
ODSNMASK [70](#)

UNLOAD command (*continued*)

keywords (*continued*)

PAD [70](#)
PTRCHK [71](#)
RELOCATE_SDEP [71](#)
SDEP [71](#)
SDEP_BYPASS_MSGLVL [71](#)
SEGREC_PREFIX [71](#)
SEGREC_SUFFIX [71](#)
SORT [71](#)
SORTOPT_FILSZ [71](#)
SSP [71](#)

Unloaded report, FPA [575](#)

unloaded segment records file format

area information record layout [1179](#)

unloaded segment records layout [1177](#)

unverified resync pointer segment records [324](#)

UOW Evaluation

considerations [781](#)

reports and statistics [803](#)

RFSSCAN keyword [909](#)

UOW Exception report, FPA [607](#)

UOW keyword, FPA [553](#)

UOW Ranking

considerations [781](#)

RFSSCAN keyword [909](#)

UOW report, FPA [606](#)

UOW Space Statistics report, FPA [616](#)

UOW Space Statistics report, FPO [804](#)

UOW_REPORT keyword, FPA [555](#)

UOW_REPORT keyword, FPO [939](#)

UOWDOVF_PCT keyword, FPA [553](#)

UOWDOVF_PCT keyword, FPO [937](#)

UOWIOVF_AVG keyword, FPA [554](#)

UOWIOVF_AVG keyword, FPO [937](#)

UOWIOVF_MAX keyword, FPA [554](#)

UOWIOVF_MAX keyword, FPO [938](#)

UOWIOVF_PCT keyword, FPA [555](#)

UOWIOVF_PCT keyword, FPO [938](#)

UPDATE_NUMBER keyword, OSC [1046](#)

user exit, DEDB Reload (FABCUR3)

interface [1197](#)

user exit, DEDB Reload Segment Data Set Create utility
(FABCUR6)

interface [1197](#)

user exit, DEDB Unload (FABCUR1)

interface [1197](#)

user exit, FABCUR7 (DEDB Unloaded Segment Data Set
Retrieve utility)

interface [1197](#)

USER_RECORD statement

keywords

BREAK [865](#), [877](#)

USER_RECORD subcommand

keywords

BREAK [72](#)

FIELDS [72](#)

keywords for EXTRACT command

BREAK [276](#)

FIELDS [276](#)

USR_MAX keyword, OSC [1047](#)

USRHLQ keyword, OSC [1071](#)

utility control statements

OAE [764](#), [765](#)

utility control statements (*continued*)

ODE [735](#)

ODM [824](#)

OER [796](#)

OPC [696](#)

OSC [1031](#)

OSM [977](#)

utility message file, OPC [896](#)

V

VOLCNT keyword, FPA [556](#)

VOLSER keyword, FPA [556](#)

VOLUME keyword, FPO [939](#)

VOLUME keyword, OSM [992](#)

VOLUME_EXT keyword, FPO [939](#)

VOLUME_EXT keyword, OSM [992](#)

VSO [748](#), [756](#), [757](#)

W

what's new [3](#)

WHERE keyword, FPA [557](#)



Product Number: 5698-FPP

SC27-9536-02

