z/OS

SDSF Operation and Customization

Version 2 Release 1

SA23-2274-08

Note
Before using this information and the product it supports, read the information in "Notices" on page 593.

This edition applies to Version 2 Release 1 of z/OS (5650-ZOS) and to all subsequent releases and modifications unless otherwise indicated in new editions.

© Copyright IBM Corporation 1997, 2016.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Figures xi

Tables xiii

About this document xvii

z/OS information xix

How to send your comments to IBM xxi

If you have a technical problem xxi

Summary of changes xxiii

z/OS V2R1 Summary of changes as updated October, 2016 xxiii

z/OS V2R1 Summary of changes as updated June, 2016 xxiii

z/OS V2R1 Summary of changes as updated April, 2016 xxiii

z/OS V2R1 summary of changes as updated March, 2016 xxiv

z/OS V2R1 summary of changes as updated December, 2015 xxv

z/OS V2R1 summary of changes as updated March, 2014 xxv

z/OS Version 2 Release 1 summary of changes xxvi

Chapter 1. Exploiting new functions . . 1

z/OS V2R1 summary as updated October, 2016. . . 1

AS panel 1

DYNX panel 1

z/OS V2R1 summary as updated June, 2016. . . . 2

z/OS V2R1 summary as updated April, 2016 . . . 2

Enqueue panel. 2

System Symbols panel 3

z/OS V2R1 summary as updated December, 2015 . . 3

Authorized program facility panel 3

Link list panel 4

Link pack area panel 4

Page panel 4

PARM panel 5

SRCH panel 5

System panel 6

z/OS V2R1 summary 6

SDSF function in z/OSMF 7

Exploitation tasks. 7

64-bit addressing 8

Exploitation tasks. 8

Enhanced support for browsing with REXX and Java 8

Exploitation tasks. 8

Sort on up to 10 columns 8

Exploitation tasks. 9

Symbols with filtering 9

Exploitation tasks. 9

Display full sort and filter criteria 9

Exploitation tasks. 9

Long JES2 job class names 9

Exploitation tasks 10

Trace SDSF security. 11

Exploitation tasks 11

Print using source attributes, carriage control and other enhancements 11

Exploitation tasks 11

Console name modification 12

Exploitation tasks 12

Display of duplicate data sets 12

Exploitation tasks 12

Custom property for CPU% normalization 13

Exploitation tasks 13

Custom property to control the scope of panels . . 13

Exploitation tasks 13

Column changes. 13

Exploitation tasks 14

Chapter 2. Using ISFPARMS for customization and security 15

ISFPARMS overview 15

ISFPARMS format alternatives 15

Converting ISFPARMS assembler macros to statements. 17

Reverting to the ISFPARMS defined with assembler macros 17

Samples 18

Auditing ISFPARMS 18

Diagnosing security 18

Rules for coding ISFPARMS 18

Statements. 19

Assembler macros 19

Conditional processing 20

WHEN Statement 20

System symbols 22

Global initialization parameters (OPTIONS or ISFPMAC). 23

Example of the OPTIONS statement and ISFPMAC macro. 23

OPTIONS or ISFPMAC reference 23

Server group definition parameters (SERVERGROUP, SERVER, COMM) 30

Example of the SERVERGROUP statement . . . 30

SERVERGROUP statement 31

SERVER statement 31

COMM statement 32

Summary of rules for defining a server group . . 33

Server connection (CONNECT). 34

Example of the CONNECT statement. 34

CONNECT statement 34

Group authorization parameters (GROUP or ISFGRP) 36

Group membership. 36

Group function 37

© Copyright IBM Corp. 1997, 2016

Examples of the GROUP statement and ISFGRP macro . . . 37
Group membership parameters reference . . . 38
Group function parameters reference . . . 40
Action characters and overtypeable fields for each command level . . . 73
Variable field lists (FLD or ISFFLD) . . . 85
Example of the FLD statement and ISFFLD macro . . . 85
FLD and ISFFLD syntax . . . 86
Name tables (NTBL or ISFNTBL) . . . 89
Examples of the NTBL statement and ISFNTBL macro . . . 89
NTBL and ISFNTBL syntax . . . 90
Usage notes . . . 90
Customized properties (PROPLIST) . . . 91
Example of the PROPLIST and associated statements . . . 91
PROPLIST syntax . . . 92
Code page (TRTAB/TRDEF or ISFTR) . . . 101
Examples of the TRTAB statement and ISFTR macro . . . 101
TRTAB and ISFTR syntax . . . 102
Coding a translate table . . . 105

Chapter 3. Using the SDSF server . . 107
Defining the input . . . 107
Starting the server . . . 108
Starting the SDSFAUX server . . . 108
Processing the statements . . . 108
Accessing the server . . . 109
Logging . . . 109
Security . . . 110
Using the server for sysplex data . . . 110
Device panels . . . 110
Sysplex-wide panels . . . 110
SYSLOG panel . . . 111
Servers with server groups and WebSphere MQ 111
Server operator commands . . . 118
Start the SDSF server . . . 118
Change server options . . . 122
Display server options . . . 125
Display information about server communications . . . 126
Refresh ISFPARMS . . . 127
Start communications . . . 128
Stop communications . . . 129
Stop the SDSF server . . . 130

Chapter 4. Columns on the SDSF panels . . . 133
Address Space Memory panel (AS) . . . 133
Authorized Program Facility panel (APF) . . . 135
Display Active Users panel (DA) . . . 135
Dynamic Exits panel (DYNX) . . . 138
Enclaves panel (ENC) . . . 139
Enqueue panel (ENQ) . . . 140
Health Check History panel (CKH) . . . 141
Health Checker panel (CK) . . . 142
Held Output panel (H) . . . 144
Initiator panel (INIT) . . . 146
Input Queue panel (I) . . . 147
JESPLEX panel (JP) . . . 149
Job Class panel (JC) . . . 149
Job Data Set panel (JDS) . . . 151
Job 0 (J0) . . . 154
Lines panel (LI) . . . 155
Link List panel (LNK) . . . 157
Link Pack Area panel (LPA) . . . 158
Multi-Access Spool panel (MAS) and JESPLEX (JP) panel . . . 158
Network Connections (NC) . . . 160
Network Servers (NS) . . . 161
Nodes panel (NO) . . . 162
Output Descriptors panel (OD) . . . 164
Output Queue panel (O) . . . 166
Page panel (PAG) . . . 168
PARMLIB panel (PARM) . . . 169
Printer panel (PR) . . . 169
Processes panel (PS) . . . 174
Punch panel (PUN) . . . 174
Reader panel (RDR) . . . 177
Resource panel (RES) . . . 178
Resource Monitor (RM) panel . . . 179
Scheduling Environment panel (SE) . . . 179
Spool Offload panel (SO) . . . 180
Spool Volumes panel (SP) . . . 182
Search panel (SRCH) . . . 184
System Symbols panel (SYM) . . . 185
System panel (SYS) . . . 186
Status panel (ST) . . . 188
System Requests panel (SR) . . . 189

Chapter 5. Using SAF for security . . 191
Relationship of SAF and ISFPARMS . . . 191
Using ISFPARMS as a backup to SAF . . . 191
Changing authorization dynamically . . . 192
Auditing access attempts . . . 192
Diagnosing security . . . 192
SAF concepts for SDSF resources . . . 193
Protecting SDSF function . . . 193

Chapter 6. SDSF and RACF 205
Security administration . . . 205
Brief summary of RACF commands . . . 205
Delegation of RACF administrative authority . . . 206
SDSF resource group class . . . 206
Establishing SAF security with RACF . . . 207
RACF authorization checking and ISFPARMS security . . . 207
Considerations for broad access . . . 207
Using RACLIST and REFRESH . . . 208
Using RACLIST and REFRESH with the SDSF class . . . 208
Using RACLIST and REFRESH with the OPERCMDS class . . . 208
Using conditional access . . . 209
Sample RACF commands . . . 209
Multilevel Security . . . 209

Chapter 7. Protecting SDSF functions 211

Action characters 211
Protecting the objects of action characters . . . 211
Protecting the generated MVS and JES commands 212
Setting up generic profiles 212
Examples of protecting action characters . . . 213
Tables of action characters 213
Authorized program facility data sets 235
Protecting authorized program facility data sets 235
Example of protecting authorized program facility data sets 235
Authorized SDSF commands 235
Protecting SDSF commands 235
Setting up generic profiles 237
Examples of protecting commands 238
Checks on the CK and CKH panels 238
Protecting checks 238
Example of protecting checks 239
Destination names. 239
Protecting destination names 239
Initializing destinations 240
Example of protecting destination names . . . 240
Destination operator authority. 240
Protecting operator authority by destination . . 241
| Dynamic exit information 242
| Protecting dynamic exits 242
| Example of protecting dynamic exits 242
Enclaves 242
Protecting enclaves 242
Example of protecting enclaves 242
Enqueue information. 243
Protecting Enqueue Information 243
Example of protecting enqueue information . . 243
Initiators 243
Protecting initiators 243
Example of protecting initiators 244
JES2 resources on the RM panel 244
Protecting JES2 resources 244
Example of protecting JES2 resources 245
Job classes 245
Protecting job classes 245
Example of protecting job classes. 245
Jobs, output groups, and SYSIN/SYSOUT data sets 245
Protecting jobs, output groups, and SYSIN/SYSOUT data sets 246
Permitting other users to view your data . . . 247
Providing function comparable to NOTIFY authority 248
Examples of protecting jobs and output groups 248
Lines 250
Protecting lines. 250
Example of protecting lines. 250
Link list data sets 250
Protecting link list data sets 250
Example of protecting link list data sets . . . 251
MAS and JESPLEX members 251
Protecting MAS and JESPLEX members . . . 251
Example of protecting MAS members 251
Membership in groups 252
Controlling membership in groups 252
Example of protecting membership in a group in ISFPARMS 252
MVS and JES commands on the command line . . 252
Protecting the slash command. 252
Slash command and User Log. 253
Example of protecting the slash command. . . 253
Network connections. 253
Protecting network connections 253
Example of protecting network connections . . 254
Network servers 254
Protecting network servers 254
Example of protecting network servers 254
Nodes. 254
Protecting nodes 254
Example of protecting nodes 255
OPERLOG 255
Protecting the log stream 255
Overtypeable fields 255
Protecting the overtypeable field 256
Protecting the objects of overtypeable fields . . 256
Protecting the generated MVS and JES commands 256
Generic profiles 256
Examples of protecting overtypeable fields . . 257
Tables of overtypeable fields 258
Page data sets 294
Protecting page data sets 294
Example of protecting page data sets 294
PARMLIB data sets 295
Protecting PARM data sets 295
Example of protecting PARM data sets 295
Printers 295
Protecting printers. 295
Permitting access only while using SDSF . . . 296
Examples of protecting printers 296
Processes (z/OS UNIX System Services) 296
Protecting processes 296
Example of protecting processes 297
Punches 297
Protecting punches 297
Permitting access only while using SDSF . . . 297
Example of protecting punches 298
Readers 298
Protecting readers 298
Example of protecting readers 298
Resources defined to WLM. 298
Protecting WLM resources 298
Example of protecting resources 299
Scheduling environments 299
Protecting scheduling environments 299
Example of protecting scheduling environments 299
SDSF server 300
Protecting the SDSF server 300
Examples of protecting the SDSF server . . . 300
Spool offloaders 301
Protecting spool offloaders 301
Example of protecting spool offloaders 301
Spool volumes 301
Protecting spool volumes 301
Example of protecting spool volumes 302
SYSLOG 302

Protecting the logical log 302
System Symbol information 303
Protecting system symbol information 303
Example of protecting system symbol information 303
System information 303
Protecting system information 303
Example of protecting system information . . . 304
System requests 304
Protecting system requests 304
Example of protecting system requests 305
User log (ULOG) 305
Protecting the ULOG 305
Examples of protecting ULOG 306
WebSphere MQ. 306
Protecting the queues. 306
Allowing the server to communicate with remote servers 307
Allowing the server to define queues 308
Defining connection security for SDSF 308
Defining context security for SDSF 309
Assigning user IDs 309
Example 309

Chapter 8. Converting ISFPARMS to SAF security 311
Getting started 311
SDSF environment. 311
Migration considerations 311
Examples of RACF security for SDSF groups . . . 312
Providing group authority 313
Group 1 — system programmers 313
Group 2 — operators 315
Group 3 — end users. 318
Summary of profiles for group 1, group 2, and group 3 321
Using the Security Conversion Assist 323
Diagnosing security 328

Chapter 9. Using installation exit routines 329
Installation exit routines 329
Using the ISFUSER module 329
Installation exit points 330
SAF considerations for exit points 330
Initialization exit point 331
Command authority exit point 333
SYSOUT display authority exit point 334
SDSF termination exit point 335
Pre-SAF exit point. 335
Post-SAF exit point 335
SAF indeterminate exit point 336
Table build exit point. 337

Chapter 10. Installation and configuration considerations. 339
JES2 considerations 339
DESTDEF considerations 339
SDSF server group with multiple levels of JES2 339
SDSF with a secondary JES2 subsystem. 339
Installation considerations 340
JES3 considerations 340
Getting started running SDSF in the JES3 environment. 341
SDSF considerations 342
Issuing MVS and JES commands 342
RMF considerations 344
WebSphere MQ considerations 344
z/OS libraries 344
WebSphere MQ configuration 344
Storage estimates 345
Queues 346
ISPF considerations 347
ISFISP entry point 348
Specifying that SDSF should process JES2 . . . 349
SYSLOG 349
Logical Log 349
HASPINDX 350
Reassembling SDSF for JES2 maintenance 351
Reassembling SDSF with ISFASMP 351
Sample SMP/E UCLIN jobs for automatic reassembly 351
z/OSMF considerations 352
Using the SDSF classic interface 362

Chapter 11. Introduction to SDSF application services 363
SDSF panels. 363
SDSF commands 364
Taking action against an object 365
Changing values in columns 365
Filtering and sorting panel information. 365
Complex filters 366
Issuing MVS or JES commands 366
Browsing job output and checks 367
Printing from SDSF Panels 367

Chapter 12. Using SDSF in batch . . . 369
Invoking SDSF in batch 369
Specifying that SDSF should process JES2 . . . 370
Using program name SDSF. 370
SDSF panels and commands 370
Action characters 370
Using program name ISFAFD 370
Commands 371
PF keys 373
Action characters 374
Overtypeable fields 374
Notes on using program name ISFAFD. . . . 375
Security and SDSF in batch. 375
Using SAF 376
Using ISFPARMS 376

Chapter 13. Using SDSF with the REXX programming language 377
Other sources of information 377
Programming practices 378
Quick start 379
Adding the SDSF host command environment with ISFCALLS 379

Result codes 380
Issuing commands with ISFEXEC 380
Examples of using ISFEXEC 381
Return codes for ISFEXEC 381
Messages 382
Issuing panel commands with ISFEXEC . . . 382
Issuing WHO and QUERY commands with ISFEXEC 387
Issuing system commands with ISFEXEC . . . 387
Issuing action characters and modifying columns with ISFACT 388
Modifying related fields 389
Using tokens 390
Examples of using ISFACT 390
Return codes for ISFACT 391
Options for action characters and overtypeable fields 391
Using secondary panels 392
Browsing output 394
Browsing output with ISFBROWSE 394
Browsing jobs with an external utility 397
Browsing checks with the S action character . . 398
Examples of browsing output 398
Printing output 399
Examples of printing 401
Getting all of the values for a single row 401
Return codes for ISFGET 402
Data returned for ISFGET 402
Options for getting all of the values for a row 402
Special variables with ISFGET 403
Browsing the system log with ISFLOG 403
Examples of using ISFLOG 404
Options for the ISFLOG command 404
Special variables for use with the ISFLOG command 404
Return codes for ISFLOG 407
Issuing system commands with ISFSLASH . . . 408
Examples of using ISFSLASH 408
Options for slash (/) commands 409
Special variables for slash (/) commands . . . 409
Return codes for ISFSLASH 410
Using special variables to invoke SDSF function 410
SDSF command 411
Filter commands 411
Options commands 412
Trace commands 413
Dropping special variables with ISFRESET . . 414
SDSF with REXX reference 414
SDSF commands reference 414
Action character reference 418
Special variables reference 420
Examples of REXX execs 429
Access an SDSF panel 429
Cancel a job 430
Cancel a set of jobs 431
List job data sets 432
Modify values in columns 433
Modify a value for a set of jobs 434
Browse job output with EXECIO 435
Browse job output with ISFBROWSE (basic) . . 435
Browse job output with ISFBROWSE 436
Browse a single data set with EXECIO 437
Browse a single data set with ISFBROWSE . . 438
Browse check output from the CK panel . . . 441
Browse check output from the CK panel using ISFBROWSE 441
Browse check output from the CKH panel. . . 443
Print to SYSOUT 444
List action characters 444
Issue system commands using ISFSLASH . . . 445
Work with the last 24 hours of SYSLOG . . . 445
Work with the current day of the system log 446
Find a message in the system log. 446
Work with the last 24 hours of OPERLOG. . . 447
Issue the WHO command 447
System REXX and SDSF 448
Security and REXX 448
Determining which group in ISFPARMS a user is assigned to 448
Diagnosing errors in a REXX exec 449

Chapter 14. Using SDSF with the Java programming language 451
Where to look for information 451
Simplifying systems management with SDSF Java 452
Enabling your application to use SDSF Java . . . 452
Installation verification 453
Writing a Java application 454
Example 454
Working with objects 455
Obtaining column values 455
Actions and overtypes 455
Using runners and request settings 456
Determining which runner to use 457
Samples 458
Running the samples 459
Troubleshooting 459
Tracing 460
Security and Java 460
Determining which group in ISFPARMS a user is assigned to 460
Protecting runners. 461
Protecting methods 461

Chapter 15. SDSF messages and codes. 475
Displaying message help 475
User authorization. 475
SDSF messages 475
Messages with HSF message numbers 503
Messages with ISF message numbers 508
Messages for IBM Health Checker for z/OS . . . 547
SDSF user abend codes 551

Appendix A. SDSF problem management. 555
Observing and identifying a problem 555
Gathering information about a problem 555
Dumps 555
Trace 555
Module information 555

SDSF problem index 556
Problems with the Repeat-Find PF keys (PF5 and PF17) 556
Problems with the LOG, RETRIEVE and TUTOR commands 556
Users are experiencing authorization problems 556
SDSF has abended. 557
Documentation is incorrect 557
An SDSF message is incorrect 557
A message was not found in the tutorial . . . 558
Data on an SDSF panel is garbled or incorrect 558
RMF exit is not installed. 558
Communication problems within a server group 558

Appendix B. SAF equivalents for ISFPARMS. 561

ACTION 561
ACTIONBAR 561
APPC 561
AUPDT 561
AUTH=ABEND 561
AUTH=ACTION 561
AUTH=ALL. 561
AUTH=ALLOPER. 562
AUTH=ALLUSER. 562
AUTH=APF. 562
AUTH=AS 562
AUTH=CK 562
AUTH=DA 562
AUTH=DEST 563
AUTH=DYNX 563
AUTH=ENC 563
AUTH=ENQ 563
AUTH=FINDLIM 563
AUTH=H 563
AUTH=I 564
AUTH=INIT. 564
AUTH=INPUT 564
AUTH=JC 564
AUTH=J0 564
AUTH=LI 564
AUTH=LNK 564
AUTH=LOG 565
AUTH=LPA 565
AUTH=MAS 565
AUTH=NC 565
AUTH=NO 565
AUTH=NS 565
AUTH=O. 565
AUTH=PAG. 566
AUTH=PARM 566
AUTH=PR 566
AUTH=PREF 566
AUTH=PS 566
AUTH=PUN 566
AUTH=RDR. 566
AUTH=RES 567
AUTH=RM 567
AUTH=RSYS 567
AUTH=SE 567
AUTH=SO 567
AUTH=SP 567
AUTH=SR 567
AUTH=ST 568
AUTH=SYS 568
AUTH=SYSID 568
AUTH=SYSNAME 568
AUTH=TRACE. 568
AUTH=ULOG 568
BROWSE 568
CMDAUTH=DEST 569
CMDAUTH=DISPLAY 569
CMDAUTH=GROUP. 569
CMDAUTH=INIT 569
CMDAUTH=NOTIFY 569
CMDAUTH=MSG. 569
CMDAUTH=USERID. 570
CMDAUTH=ALL 570
CMDLEV. 570
CONFIRM 571
CPUFMT 571
CTITLE 571
CURSOR 571
DADFLT 571
DATE 571
DATESEP 571
DEST 571
DISPLAY 571
DSPAUTH=ADEST 572
DSPAUTH=ALL 572
DSPAUTH=AMDEST. 572
DSPAUTH=AMSG 572
DSPAUTH=GROUP 573
DSPAUTH=GRPMSG. 573
DSPAUTH=NOTIFY 573
DSPAUTH=USERID 573
EMCSAUTH 573
EMCSREQ 573
GPLEN 573
GPREF 573
ICMD 574
IDEST 574
IDSP 574
IDSPD. 574
ILOGCOL 574
INPUT 574
ISTATUS 575
ISYS 575
LANG. 575
LOG/LOGOPT. 575
OWNER 575
PREFIX 575
RSYS 575
SYSID 575
UPCTAB 575
VALTAB 575
VIO 575
XCMD 575
XDSP 576
XDSPD 576
XSTATUS. 576

Appendix C. SDSF resource names for SAF security 577

Appendix D. Accessibility 589
Accessibility features 589
Consult assistive technologies 589
Keyboard navigation of the user interface 589
Dotted decimal syntax diagrams 589

Notices 593
Policy for unsupported hardware. 594
Minimum supported hardware 595
Programming interface information 595
Trademarks 595

Index 597

Figures

1. Server Group Example 112
2. Server Group Example–Alternate JES2s 113
3. Server Group Example–All Servers 114
4. Server Group Example—Server Group on SY2 115
5. Change Server Options — Syntax 123
6. Display Server Options — Syntax 126
7. Display Information About Server Communications — Syntax 126
8. Refresh ISFPARMS — Syntax 127
9. Start Communications — Syntax 128
10. Stop Communications — Syntax 129
11. STOP the SDSF Server — Syntax 130
12. Example of a Profile Description to Change 326
13. A Sample SDSF Tabular Panel 364
14. Sample ISFPARMS to Restrict Batch 376

© Copyright IBM Corp. 1997, 2016

Tables

1. Exploitation tasks for the AS panel 1
2. Exploitation tasks for the DYNX panel 1
3. Exploitation tasks for the ENQ panel 2
4. Exploitation tasks for the SYM panel 3
5. Exploitation tasks for the APF panel 3
6. Exploitation tasks for the LNK panel 4
7. Exploitation tasks for the LPA panel. 4
8. Exploitation tasks for the PAG panel 5
9. Exploitation tasks for the PARM panel 5
10. Exploitation tasks for the SRCH panel 5
11. Exploitation tasks for the SYS panel 6
12. New Column on the JC Panel 10
13. New Columns on the INIT Panel 10
14. New Columns on the SO Panel 10
15. New SAF Resources. 10
16. Changes to Columns on the I and ST Panels 13
17. New Columns on the SDSF Panels 13
18. SAF Resources for New Overtypeable Columns 14
19. Summary of ISFPARMS Statements and Macros 16
20. Command Level 0 Action Characters 74
21. Command Level 1 Action Characters 75
22. Command Level 2 Action Characters 75
23. Command Level 2 Overtypeable Fields 76
24. Command Level 3 Action Characters 78
25. Command Level 3 Overtypeable Fields 79
26. Command Level 4 Action Characters 84
27. Command Level 4 Overtypeable Fields 84
28. Command Level 5 Overtypeable Fields 84
29. Command Level 6 Overtypeable Fields 85
30. Field List Parameters 87
31. Properties to Specify with the PROPERTY Statement 92
32. WebSphere MQ Queues Used by SDSF 116
33. Queues Used to Define Queues 117
34. Columns on the AS Panel 133
35. Columns on the APF Panel 135
36. Columns on the DA Panel 135
37. Columns on the DYNX Panel 138
38. Columns on the ENC Panel. 139
39. Columns on the ENQ Panel. 141
40. Columns on the CKH Panel. 141
41. Columns on the CK Panel 142
42. Columns on the H Panel. 144
43. Columns on the INIT Panel. 146
44. Columns on the I Panel 147
45. Columns on the JC Panel 149
46. Columns on the JDS Panel 151
47. Columns on the J0 Panel. 154
48. Columns on the LI Panel. 155
49. Columns on the LNK Panel. 157
50. Columns on the LPA Panel 158
51. Columns on the MAS and JP Panel 158
52. Columns on the NC Panel 160
53. Columns on the NS Panel 161
54. Columns on the NO Panel 162
55. Columns on the OD Panel 164
56. Columns on the O Panel. 166
57. Columns on the PAG Panel 168
58. Columns on the PARM Panel 169
59. Columns on the PR Panel 169
60. Columns on the PS Panel 174
61. Columns on the PUN Panel. 174
62. Columns on the RDR Panel 177
63. Columns on the RES Panel 178
64. Columns on the RM Panel 179
65. Columns on the SE Panel 179
66. Columns on the SO Panel 180
67. Columns on the SP Panel 182
68. Columns on the SRCH Panel 184
69. Columns on the System Symbols 185
70. Columns on the SYS Panel 186
71. Columns on the ST Panel 188
72. Columns on the SR Panel 189
73. SDSF Functions and the Classes and Resources Required to Protect Them 194
74. Generic Profiles for Commands Generated by Actions Characters. 212
75. Action Characters 213
76. Action Characters by OPERCMDS Resource Name 228
77. OPERCMDS Resources That Protect Issuing Action Characters for JES2 Resources . . . 234
78. SAF Resources for Authorized Program Facility Data Sets 235
79. SDSF Class Resource Names and SDSF Commands 236
80. Authority Required to Checks for Actions and Overtypes. 238
81. Authority Required to the Log Stream Used for Check History 239
82. Authority Required for Destination Names 239
83. Authority Required for Destination Operator Authority 241
84. SAF Resources for Dynamic Exits 242
85. SAF Resources for Enclaves. 242
86. SAF Resources for Enqueue Information 243
87. Authority Required to Initiator Resource for Actions and Overtypes 243
88. Authority Required to JES2 Resources for Actions and Overtypes 244
89. Authority Required to Job Class Resource for Actions and Overtypes 245
90. Authority Required to JESSPOOL Resources 246
91. Authority Required to Lines Resources for Actions and Overtypes 250
92. SAF Resources for Link List Data Sets 251
93. Authority Required to MAS or JESPLEX Members for Actions and Overtypes 251
94. Authority Required for membership in an ISFPARMS group 252

© Copyright IBM Corp. 1997, 2016

95. Authority Required for the Slash Command 252
96. Authority Required to Network Connection Resources for Actions and Overtypes . . . 253
97. Authority Required to Network Servers Resources for Actions and Overtypes . . . 254
98. Authority Required to Nodes Resources for Actions and Overtypes 254
99. Authority Required for Accessing the Log Stream 255
100. Generic Profiles for Overtypeable Fields 257
101. Overtypeable Fields 259
102. Actions with Overtypes on the PR and PUN Panels in a JES3 Environment 279
103. Overtypeable Fields Sorted by OPERCMDS Resource Name 279
104. OPERCMDS Resources That Protect Overtyping JES2 Resources 294
105. SAF Resources for Page Data Sets. 294
106. SAF Resources for PARM Data Sets 295
107. Authority Required to Printer Resources for Actions and Overtypes 295
108. Authority Required to z/OS UNIX Processes for Actions and Overtypes 296
109. Authority Required to Punch Resources for Actions and Overtypes 297
110. Authority Required to Reader Resources for Actions and Overtypes 298
111. Authority Required to SAF Resources for WLM Resources 299
112. Authority Required to Scheduling Environment Resource for Actions 299
113. Authority Required to Server Functions 300
114. Authority Required to Offloader Resources for Actions and Overtypes 301
115. Authority Required to Spool Volume Resources for Actions and Overtypes . . . 302
116. Authority Required for Accessing the Logical Log 303
117. SAF Resources for System Symbol Information 303
118. SAF Resources for System Information 303
119. Authority Required to System Request Resource for Action Characters. 305
120. Resource that Protects the Extended Console 305
121. Authority Required for the WebSphere MQ Queues 307
122. Authority Required for Communication with Remote Servers 307
123. Authority Required for Server Definition of Queues 308
124. Authority Required for Connection Security 308
125. Authority Required for Context Security 309
126. Sample GROUP statements, Group 1 314
127. Profile for Membership in Group 1 — System Programmers 314
128. Profiles for Function of Group 1 — System Programmers 315
129. Sample GROUP statements, Group 2 316
130. Profiles for Membership in Group 2 — Operators 316
131. Profiles for Function of Group 2 — Operators 316
132. Sample GROUP statements, Group 3 319
133. Profile for Membership in Group 3 — End Users 319
134. Profiles for Function of Group 3 — End Users 319
135. Profiles for Groups 1, 2, and 3 321
136. Data Sets Required by the Security Conversion Assist 325
137. Profile Description Example. 325
138. Profile Descriptions That May Require a Change to the Access List 326
139. GROUP/ISFGRP Parameters Supported 327
140. Summary of Possible WebSphere MQ System Parameters Changes 345
141. Message Size Estimates 345
142. Configuration Options for the HASPINDX Data Set 350
143. Resources in the ZMFAPLA Class. 354
144. z/OSMF Views and Corresponding z/OS SDSF Commands 354
145. Actions on the Active Jobs Table 355
146. Actions on the All Jobs Table 356
147. Actions on the Input Queue Table 357
148. Actions on the Output Queue Table 358
149. Actions on the Held Output Queue Table 359
150. Actions on the Job Data Set Table 359
151. Actions on the Health Check Table 359
152. Actions on the Health Check History Table 360
153. Actions on the System Table 360
154. Actions on the APF Table 361
155. Actions on the PAG Table 361
156. Actions on the LNK Table 361
157. Actions on the PARM Table. 361
158. Summary of SDSF Interfaces 363
159. Commands for Getting Help on Using SDSF 363
160. Summary of Commands for Filtering 366
161. Contents of the Log Stamp 372
162. Using SDSF with REXX 377
163. Controlling the Columns on SDSF Panels 383
164. Special REXX Variables for Printing to SYSOUT 399
165. Special REXX Variables for Printing to a Data Set 400
166. Special Variables for Printing to a File 401
167. SDSF Commands and REXX 415
168. Action Characters Not Supported with REXX 418
169. Special REXX Variables 420
170. SDSF Commands and Runners. 457
171. ISFActive Methods for Action Characters 461
172. ISFApf Methods for Action Characters 462
173. ISFDynx Methods for Action Characters 462
174. ISFEnclave Methods for Action Characters 463
175. ISFENQ Methods for Action Characters 463
176. ISFHealthCheck Methods for Action Characters 463
177. ISFHealthCheckArchive Methods for Action Characters 463
178. ISFHeldOutput Methods for Action Characters 464
179. ISFInitiator Methods for Action Characters 464
180. ISFInput Methods for Action Characters 464
181. ISFJESplex Methods for Action Characters 465

182. ISFJobClass Methods for Action Characters 466
183. ISFJobDataSet Methods for Action Characters 466
184. ISFJob0 Methods for Action Characters 467
185. ISFLine Methods for Action Characters 467
186. ISFLnkLst Methods for Action Characters 467
187. ISFNetworkConnection Methods for Action Characters 467
188. ISFNetworkServer Methods for Action Characters 468
189. ISFNode Methods for Action Characters 468
190. ISFOutput Methods for Action Characters 468
191. ISFPage Methods for Action Characters 469
192. ISFParmlib Methods for Action Characters 469
193. ISFPrinter Methods for Action Characters 469
194. ISFProcess Methods for Action Characters 469
195. ISFPunch Methods for Action Characters 470
196. ISFReader Methods for Action Characters 470
197. ISFRequestSettings Methods for Commands that Require Authorization 470
198. ISFResourceMonitor Methods for Action Characters 471
199. ISFSchedulingEnvironment Methods for Action Characters 471
200. ISFSpool Methods for Action Characters 471
201. ISFSpoolOffload Methods for Action Characters 471
202. ISFStatus Methods for Action Characters 472
203. ISFSystem Methods for Action Characters 473
204. ISFSystemSymbol Methods for Action Characters 474
205. ISFSystemRequest Methods for Action Characters 474
206. ISFWLMResource Methods for Action Characters 474
207. SDSF Abend Codes 551
208. Security Classes, Resource Names, and What They Protect 577
209. SDSF Class Resource Names and Overtypeable Fields 580

About this document

This document is for use with z/OS® System Display and Search Facility (SDSF). It is intended primarily for system programmers and operators, and assumes you are familiar with the z/OS operating system, including JES. This document contains information about migration, customization, security, operation, maintenance and problem determination, including explanations of SDSF messages.

This document also describes how to use SDSF's application services to write REXX execs or Java™ programs that exploit SDSF function. It includes a quick introduction to SDSF function and terminology for people who are not already experienced users of SDSF but want to exploit SDSF's application services.

Complete information about using SDSF, such as commands, action characters and messages, is provided in the online help for z/OS SDSF. In addition, introductory information is available on the Internet at http://www.ibm.com/systems/z/os/zos/features/sdsf/.

© Copyright IBM Corp. 1997, 2016

z/OS information

This information explains how z/OS references information in other documents and on the web.

When possible, this information uses cross document links that go directly to the topic in reference using shortened versions of the document title. For complete titles and order numbers of the documents for all products that are part of z/OS, see *z/OS Information Roadmap*.

To find the complete z/OS library, go to IBM Knowledge Center (http://www.ibm.com/support/knowledgecenter/SSLTBW/welcome).

© Copyright IBM Corp. 1997, 2016

How to send your comments to IBM

We appreciate your input on this publication. Feel free to comment on the clarity, accuracy, and completeness of the information or provide any other feedback that you have.

Use one of the following methods to send your comments:

1. Send an email to mhvrcfs@us.ibm.com.
2. Send an email from the "Contact us" web page for z/OS (http://www.ibm.com/systems/z/os/zos/webqs.html).
3. Mail the comments to the following address:
   IBM Corporation
   Attention: MHVRCFS Reader Comments
   Department H6MA, Building 707
   2455 South Road
   Poughkeepsie, NY 12601-5400
   US
4. Fax the comments to us, as follows:
   From the United States and Canada: 1-845-432-9405
   From all other countries: Your international access code +1 845 432 9405

Include the following information:

- Your name and address.
- Your email address.
- Your telephone or fax number.
- The publication title and order number:
  z/OS V2R1.0 SDSF Operation and Customization
  SA23-2274-08
- The topic and page number that is related to your comment.
- The text of your comment.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute the comments in any way appropriate without incurring any obligation to you.

IBM or any other organizations use the personal information that you supply to contact you only about the issues that you submit.

If you have a technical problem

Do not use the feedback methods that are listed for sending comments. Instead, take one of the following actions:

- Contact your IBM service representative.
- Call IBM technical support.
- Visit the IBM Support Portal at z/OS Support Portal (http://www-947.ibm.com/systems/support/z/zos/).

© Copyright IBM Corp. 1997, 2016

Summary of changes

This information includes terminology, maintenance, and editorial changes. Technical changes or additions to the text and illustrations for the current edition are indicated by a vertical line to the left of the change.

z/OS V2R1 Summary of changes as updated October, 2016

Changes made to z/OS V2R1 as updated October, 2016

New

- "Address Space Memory panel (AS)" on page 133 panel is added. The Address Space Memory (AS) panel shows system storage utilization for all address spaces in the sysplex.
- "Dynamic Exits panel (DYNX)" on page 138 panel is added. The Dynamic Exits (DYNX) panel shows all of the dynamic exits in the sysplex, their status, and the modules that implement the exit.

z/OS V2R1 Summary of changes as updated June, 2016

Changes made to z/OS V2R1 as updated June, 2016

New

- "z/OSMF considerations" on page 352 is updated to show the addition of the APF, LPA, LNK, PAG, PARM, and SYS pages.
- The automatic start of SDSFAUX can be suppressed using special values on the CONNECT statement AUXPROC or AUXNAME keywords, as described in "Server connection (CONNECT)" on page 34.
- The following new messages have been added:
  - "HSF0028W" on page 504
  - "HSF0030W" on page 504
  - "HSF0078W" on page 507

Changed

- "ISF452E" on page 529 is updated with a revised description.
- "ISF453I" on page 530 is updated with return and reason codes.
- Minor editorial updates have been made.

z/OS V2R1 Summary of changes as updated April, 2016

Changes made to z/OS V2R1 as updated April, 2016

New

- SDSF is now enhanced with the following new tabular panels:
  - ENQ (enqueues for each system in the sysplex)
  - SYM (system static and dynamic symbols for each system in the sysplex)

© Copyright IBM Corp. 1997, 2016

You can sort, filter, and arrange columns on the new panels. Authorization is controlled using the existing SDSF security scheme, with new resources added to represent the panels. You can access these panels through SDSF, SDSF REXX, and the SDSF Java API.

- "Display Active Users panel (DA)" on page 135 is updated to show that the N action character invokes the ENQ panel as a secondary display to show all enqueues associated with the ASID for the row.
- New keywords and options have been added to the ISFPARMS GROUP statement function parameters.

  The ISFPRM01 sample in ISF.SISFJCL has been updated accordingly, with every option specified with its default value.
- The following new messages have been added:
  - "ISF450I" on page 529
  - "ISF451I" on page 529
  - "ISF453I" on page 530

Changed

- Table 170 on page 457 is updated to show runners for enqueues and system symbols.
- "Group function parameters reference" on page 40 is updated with additional group function parameters.
- "Command level 0" on page 74 is updated for the ENQ and SYM panels.
- "Command level 1" on page 75 is updated for the ENQ and SYM panels.
- "FLD and ISFFLD syntax" on page 86 is updated for the ENQ and SYM panels.
- "Protecting SDSF commands" on page 235 is updated for the ENQ and SYM panels.
- "Protecting system information" on page 303 is updated with new action characters for the ENQ, SYM, and SYS panels.
- "Tables of action characters" on page 213 is updated with new action characters for the ENQ, SYM, and SYS panels.
- "PROPLIST syntax" on page 92 is updated with new properties for ENQ and SYM.
- "ISF437I" on page 528 is updated with a revised description.
- "DATA NOT AVAIL system-name" on page 480 is updated with a revised description.
- "ISF452E" on page 529 is updated with return and reason codes.
- Minor editorial updates have been made.

z/OS V2R1 summary of changes as updated March, 2016

Changes made to z/OS V2R1 as updated March, 2016

Changed

"Protecting SDSF commands" on page 235 and Table 79 on page 236, are updated to show that some commands (APF, LNK, LPA, PAG, PARM, and SYS) require use of the SDSFAUX address space, and that access to SDSFAUX is controlled through access to the ISF.CONNECT.sysname resource. The user must be permitted to this resource in addition to the resources that protect the individual commands.

"Summary of SAF resources for SDSF function" on page 194 is updated to reflect the connection to SDSFAUX, and the APF, LNK, LPA, PAG, PARM, and SYS resources.

Minor editorial updates have been made.

z/OS V2R1 summary of changes as updated December, 2015

Changes made to z/OS V2R1 as updated December, 2015

New

- SDSF is now enhanced with a new address space called SDSFAUX that provides information about the systems in the sysplex and their system data sets. The information is shown on the following new tabular panels:
  - LNK (link list data sets)
  - LPA (link pack area data set list)
  - APF (authorized program facility data set list)
  - PAGE (page data sets)
  - PARM (parmlib data sets)
  - SYS (system information)

  You can sort, filter, and arrange columns on the new panels. Authorization is controlled using the existing SDSF security scheme, with new resources added to represent the panels. You can access these panels through SDSF, SDSF REXX, and the SDSF Java API.
- A new SRCH (search) command allows you to locate members in the LNK, LPA, APF, and PARM data sets.

  Note: The SRCH command provides a different capability from the SEARCH command. SRCH implements a member search using a data set list, whereas SEARCH searches the SDSF help and tutorial.
- New keywords and options have been added to the ISFPARMS GROUP statement function parameters.

  The ISFPRM01 sample in ISF.SISFJCL has been updated accordingly, with every option specified with its default value.
- The HSF component prefix is now used for parts and messages. Both ISF and HSF are registered to the SDSF product. The SMP/E packaging contains new elements that reference the HSF prefix.
- When SDSF trace is active, SDSFAUX tracing is also active. The trace records are written to the HSFTRACE data set allocated by the SDSFAUX address space.

z/OS V2R1 summary of changes as updated March, 2014

The following changes are made for z/OS Version 2 Release 1 (V2R1) as updated March, 2014.

Changed

SDSF provides support for function in z/OSMF. For more information refer to "SDSF function in z/OSMF" on page 7.

z/OS Version 2 Release 1 summary of changes

See the following publications for all enhancements to z/OS Version 2 Release 1 (V2R1):

- *z/OS Migration*
- *z/OS Planning for Installation*
- *z/OS Summary of Message and Interface Changes*
- *z/OS Introduction and Release Guide*

Chapter 1. Exploiting new functions

Migration information is in *z/OS Migration*. This topic contains information about exploiting new functions in this release. It describes changes to the security and customization of SDSF and is intended for system programmers. Information about using the new functions can be found under "What's New" in SDSF's online help.

z/OS V2R1 summary as updated October, 2016

The updates that were introduced to SDSF in z/OS V2R1 as updated October, 2016 are as follows:

- "AS panel"
- "DYNX panel"

AS panel

The Address Space Memory (AS) panel shows system storage utilization for all address spaces in the sysplex. You display this panel with the AS action character.

Exploitation tasks

Table 1. Exploitation tasks for the AS panel

Task	Reference Information
Ensure that the SDSF server and SDSFAUX address space are started.	"Starting the SDSFAUX server" on page 108
Control use of the AS command with the ISFCMD.ODSP.*xxx* resources	"Protecting SDSF commands" on page 235
Control use of the action characters using SAF or the CMDLEV parameter in ISFPARMS	"Action characters and overtypeable fields for each command level" on page 73 or "Group function" on page 37
Optionally, customize columns on the panel using the ASFLDS and ASFLD2 parameters and FLD statements in ISFPARMS	"FLD and ISFFLD syntax" on page 86

DYNX panel

The Dynamic Exits (DYNX) panel shows all of the dynamic exits in the sysplex, their status, and the modules that implement the exit. You display this panel with the DYNX action character.

Exploitation tasks

Table 2. Exploitation tasks for the DYNX panel

Task	Reference Information
Ensure that the SDSF server and SDSFAUX address space are started.	"Starting the SDSFAUX server" on page 108
Control use of the DYNX command with the ISFCMD.ODSP.*xxx* resources	"Protecting SDSF commands" on page 235
Control use of the action characters using SAF or the CMDLEV parameter in ISFPARMS	"Action characters and overtypeable fields for each command level" on page 73 or "Group function" on page 37

© Copyright IBM Corp. 1997, 2016

Table 2. Exploitation tasks for the DYNX panel (continued)

Task	Reference Information
Optionally, customize columns on the panel using the DYNXFLDS and DYNXFLD2 parameters and FLD statements in ISFPARMS	"FLD and ISFFLD syntax" on page 86

z/OS V2R1 summary as updated June, 2016

The updates that were introduced to SDSF in z/OS V2R1 as updated June 2016 are as follows PTF UI90050, APAR PI60412:

- "z/OSMF considerations" on page 352 is updated to show the addition of the APF, LPA, LNK, PAG, PARM, and SYS pages. If z/OSMF was never installed, you should install it. If it has already been installed, review the security configuration to ensure that users have access to the new panels.
- The automatic start of SDSFAUX can be suppressed using special values on the CONNECT statement AUXPROC or AUXNAME keywords. See "Server connection (CONNECT)" on page 34 for information about exploiting this feature.

z/OS V2R1 summary as updated April, 2016

The updates that were introduced to SDSF in z/OS V2R1 as updated April 2016 are as follows:

- "Enqueue panel"
- "System Symbols panel" on page 3

Enqueue panel

The Enqueue (ENQ) panel allows authorized users to display active system enqueues. The panel shows the major and minor names for the enqueuer, as well as the job name waiting for or holding the enqueue. You display this panel with the ENQ action character. The ENQC command provides a convenient means of showing all enqueues with contention. That is, ENQC shows currently held enqueues that are required by another job.

Exploitation tasks

Table 3. Exploitation tasks for the ENQ panel

Task	Reference Information
Ensure that the SDSF server and SDSFAUX address space are started.	"Starting the SDSFAUX server" on page 108
Control use of the ENQ command with the ISFCMD.ODSP.*xxx* resources	"Protecting SDSF commands" on page 235
Control use of the action characters using SAF or the CMDLEV parameter in ISFPARMS	"Action characters and overtypeable fields for each command level" on page 73 or "Group function" on page 37
Optionally, customize columns on the panel using the ENQFLDS and ENQFLD2 parameters and FLD statements in ISFPARMS	"FLD and ISFFLD syntax" on page 86

System Symbols panel

The System Symbols panel (SYM) allows authorized users to display the system dynamic and static symbols defined for each system in the sysplex. System symbols are elements that allow systems to share parmlib definitions while retaining unique values in those definitions. System symbols act like variables in a program; they can take on different values, based on the input to the program. You display this panel with the SYM action character.

Exploitation tasks

Table 4. Exploitation tasks for the SYM panel

Task	Reference Information
Ensure that the SDSF server and SDSFAUX address space are started.	"Starting the SDSFAUX server" on page 108
Control use of the SYM command with the ISFCMD.DSP.*xxx* resources	"Protecting SDSF commands" on page 235
Control use of the action characters using SAF or the CMDLEV parameter in ISFPARMS	"Action characters and overtypeable fields for each command level" on page 73 or "Group function" on page 37
Optionally, customize columns on the panel using the SYMFLDS and SYMFLD2 parameters and FLD statements in ISFPARMS	"FLD and ISFFLD syntax" on page 86

z/OS V2R1 summary as updated December, 2015

The updates that were introduced to SDSF in z/OS V2R1 as updated December 2015 are as follows:

- "Authorized program facility panel"
- "Link list panel" on page 4
- "Link pack area panel" on page 4
- "Page panel" on page 4
- "PARM panel" on page 5
- "SRCH panel" on page 5
- "System panel" on page 6

Authorized program facility panel

The APF List (APF) panel allows authorized users to display the data sets in the APF list. You display this panel with the APF action character.

Exploitation tasks

Table 5. Exploitation tasks for the APF panel

Task	Reference Information
Ensure that the SDSF server and SDSFAUX address space are started.	"Starting the SDSFAUX server" on page 108
Control use of the APF command with the ISFCMD.ODSP.*xxx* resources	"Protecting SDSF commands" on page 235
Control use of the action characters using SAF or the CMDLEV parameter in ISFPARMS	"Action characters and overtypeable fields for each command level" on page 73 or "Group function" on page 37

Table 5. Exploitation tasks for the APF panel (continued)

Task	Reference Information
Optionally, customize columns on the panel using the APFFLDS and APFFLD2 parameters and FLD statements in ISFPARMS	"FLD and ISFFLD syntax" on page 86

Link list panel

The link list (LNK) panel allows authorized users to display the data sets in the link list. You display this panel with the LNK action character.

Exploitation tasks

Table 6. Exploitation tasks for the LNK panel

Task	Reference Information
Ensure that the SDSF server and SDSFAUX address space are started.	"Starting the SDSFAUX server" on page 108
Control use of the LNK command with the ISFCMD.ODSP.*xxx* resources	"Protecting SDSF commands" on page 235
Control use of the action characters using SAF or the CMDLEV parameter in ISFPARMS	"Action characters and overtypeable fields for each command level" on page 73 or "Group function" on page 37
Optionally, customize columns on the panel using the LNKFLDS and LNKFLD2 parameters and FLD statements in ISFPARMS	"FLD and ISFFLD syntax" on page 86

Link pack area panel

The link pack area (LPA) panel allows authorized users to display the link pack area data sets. You display this panel with the LPA action character.

Exploitation tasks

Table 7. Exploitation tasks for the LPA panel

Task	Reference Information
Ensure that the SDSF server and SDSFAUX address space are started.	"Starting the SDSFAUX server" on page 108
Control use of the LPA command with the ISFCMD.ODSP.*xxx* resources	"Protecting SDSF commands" on page 235
Control use of the action characters using SAF or the CMDLEV parameter in ISFPARMS	"Action characters and overtypeable fields for each command level" on page 73 or "Group function" on page 37
Optionally, customize columns on the panel using the LPAFLDS and LPAFLD2 parameters and FLD statements in ISFPARMS	"FLD and ISFFLD syntax" on page 86

Page panel

The page (PAG) panel allows authorized users to display the page data sets. You display this panel with the PAG action character.

Exploitation tasks

Table 8. Exploitation tasks for the PAG panel

Task	Reference Information
Ensure that the SDSF server and SDSFAUX address space are started.	"Starting the SDSFAUX server" on page 108
Control use of the PAG command with the ISFCMD.ODSP.*xxx* resources	"Protecting SDSF commands" on page 235
Control use of the action characters using SAF or the CMDLEV parameter in ISFPARMS	"Action characters and overtypeable fields for each command level" on page 73 or "Group function" on page 37
Optionally, customize columns on the panel using the PAGFLDS and PAGFLD2 parameters and FLD statements in ISFPARMS	"FLD and ISFFLD syntax" on page 86

PARM panel

The PARMLIB (PARM) panel allows authorized users to display the data sets in the PARM concatenation. You display this panel with the PARM action character.

Exploitation tasks

Table 9. Exploitation tasks for the PARM panel

Task	Reference Information
Ensure that the SDSF server and SDSFAUX address space are started.	"Starting the SDSFAUX server" on page 108
Control use of the PARM command with the ISFCMD.ODSP.*xxx* resources	"Protecting SDSF commands" on page 235
Control use of the action characters using SAF or the CMDLEV parameter in ISFPARMS	"Action characters and overtypeable fields for each command level" on page 73 or "Group function" on page 37
Optionally, customize columns on the panel using the PARMFLDS and PARMFLD2 parameters and FLD statements in ISFPARMS	"FLD and ISFFLD syntax" on page 86

SRCH panel

The SRCH panel shows the results of a member search from a data set list. You display this panel with the SRCH action character from the LNK, LPA, APF, or PARM panels.

Note: The SRCH command provides a different capability from the SEARCH command. SRCH implements a member search using a data set list, whereas SEARCH searches the SDSF help and tutorial.

Exploitation tasks

Table 10. Exploitation tasks for the SRCH panel

Task	Reference Information
Ensure that the SDSF server and SDSFAUX address space are started.	"Starting the SDSFAUX server" on page 108

Table 10. Exploitation tasks for the SRCH panel (continued)

Task	Reference Information
Control use of the action characters using SAF or the CMDLEV parameter in ISFPARMS	"Action characters and overtypeable fields for each command level" on page 73 or "Group function" on page 37
Optionally, customize columns on the panel using the SRCHFLDS and SRCHFLD2 parameters and FLD statements in ISFPARMS	"FLD and ISFFLD syntax" on page 86

System panel

The System Panel (SYS) allows authorized users to display information about systems in the sysplex, such as CPU busy, storage utilization, and IPL information. You display this panel with the SYS action character.

Exploitation tasks

Table 11. Exploitation tasks for the SYS panel

Task	Reference Information
Ensure that the SDSF server and SDSFAUX address space are started.	"Starting the SDSFAUX server" on page 108
Control use of the SYS command with the ISFCMD.ODSP.*xxx* resources	"Protecting SDSF commands" on page 235
Control use of the action characters using SAF or the CMDLEV parameter in ISFPARMS	"Action characters and overtypeable fields for each command level" on page 73 or "Group function" on page 37
Optionally, customize columns on the panel using the SYSFLDS and SYSFLD2 parameters and FLD statements in ISFPARMS	"FLD and ISFFLD syntax" on page 86

z/OS V2R1 summary

The updates that were introduced to SDSF in z/OS V2R1 are as follows:

- "SDSF function in z/OSMF" on page 7
- "64-bit addressing" on page 8
- "Enhanced support for browsing with REXX and Java" on page 8
- "Sort on up to 10 columns" on page 8
- "Symbols with filtering" on page 9
- "Long JES2 job class names" on page 9
- "Trace SDSF security" on page 11
- "Print using source attributes, carriage control and other enhancements" on page 11
- "Console name modification" on page 12
- "Display of duplicate data sets" on page 12
- "Custom property for CPU% normalization" on page 13
- "Custom property to control the scope of panels" on page 13
- "Column changes" on page 13

SDSF function in z/OSMF

IBM z/OS Management Facility (z/OSMF) provides a framework for managing z/OS through a Web browser interface. You can access SDSF function through z/OSMF.

The SDSF task of z/OSMF lets you see key summary information about your sysplex in graphical form, work with jobs and checks for IBM z/OS Health Checker, and issue system commands. It includes function that is analogous to these functions of z/OS SDSF:

- DA, H, I, O, ST, Job Data Set and Output Data Set (browse) panels, for jobs and job output
- CK and Health Check History panels, for health checks
- APF, LNK, LPA, PAG, PARM, and SYS panels
- ULOG panel, for command and message responses issued during the current session
- Editing JCL
- Action characters for controlling jobs and checks
- Overtypeable fields, for modifying the attributes of jobs and checks
- Slash (/) command, for issuing system commands
- PREFIX, DEST, OWNER, SYSNAME, FILTER and SORT commands, for filtering and sorting tabular data
- ARRANGE command, for customizing the order and widths of columns

Exploitation tasks

The SDSF task requires:

- z/OSMF APAR PM98630 and SDSF APAR PM86303. The SDSF APAR is functional only when the z/OSMF APAR is also installed.
- A TSO logon proc and related settings, which you specify with the **SDSF Settings** task in the z/OSMF Settings category. The TSO logon proc is used to launch a TSO address space that is created on behalf of the user. For details on the settings, refer to "Defining required settings for the SDSF task" on page 353.
- For remote systems to be displayed on the overview page of the SDSF task, SDSF communications must be active (through either WebSphere MQ or XCF) with a minimum level of z/OS SDSF V2R1. For more information, refer to "Using the server for sysplex data" on page 110.

Adding the SDSF task to z/OSMF

To add the SDSF task to z/OSMF, you import a properties file through the Import Manager task of z/OSMF, which is in the z/OSMF Administration category. This process is described in the z/OSMF online help.

The properties file for SDSF is /usr/lpp/sdsf/zosmf/sdsf.properties. Specify this file name in the Import dialog.

The import is generally a one-time process. The SDSF plugin only needs to be imported the first time you are installing SDSF or after you have deleted the plugin and want to restore it.

Configuring and securing the SDSF task

For more information about settings for the SDSF task, and about providing security for the SDSF task, refer to "z/OSMF considerations" on page 352.

For more information about z/OSMF, refer to *IBM z/OS Management Facility Configuration Guide* and http://www.ibm.com/systems/z/os/zos/zosmf/.

64-bit addressing

SDSF supports 64-bit addressing, which allows SDSF panels to handle very large numbers of rows.

Exploitation tasks

Use of 64-bit storage requires that the MEMLIMIT for the user be non-zero. If MEMLIMIT is zero, SDSF uses 31-bit storage. For more information about MEMLIMIT, refer to *z/OS MVS Programming: Extended Addressability Guide*.

Enhanced support for browsing with REXX and Java

A new ISFBROWSE host command and new special variables provide support for browsing job output, check output and the system log using a REXX exec. This provides an alternative to the previously available support, which allowed you to allocate the data sets and then use EXECIO or a similar utility to browse them. New special variables provide function that corresponds to finding a string in the data and scrolling the data.

New special variables support a FIND function to search for character strings when browsing data.

A new WTOR option on the ISFLOG command and ISFWTOR special variable provide support for WTORs when browsing the system log.

New special variables provide support for the color and highlighting attributes, and description codes, of lines on the OPERLOG.

New methods provide the same support for browsing job output, check output and the system log using a Java program.

Exploitation tasks

For more information about the new support for browsing with REXX execs, refer to "Browsing output with ISFBROWSE" on page 394.

For more information about using SDSF with Java, refer to Chapter 14, "Using SDSF with the Java programming language," on page 451.

Sort on up to 10 columns

Tabular panels now allow sorting on up to 10 columns, and you can now display the full set of sort and filter criteria.

- With the SORT command, you can use plus (+) or minus (-) parameters to add or remove sort criteria.
- The SET DISPLAY command, which shows the values for SORT, FILTER and other user settings just above the scrollable data, now accepts a LONG

parameter, which shows the full set of SORT and FILTER criteria, rather than the simple counts shown with SET DISPLAY ON. The associated REXX special variable, ISFDISPLAYMODE, accepts ON and LONG.

Exploitation tasks

None required. For more information about the commands, refer to the online help. You might use the SEARCH command, for example, `SEARCH SORT COMMAND`. For more information about the REXX special variables ISFSORT, ISFSORT2 and ISFDISPLAYMODE, refer to Chapter 13, "Using SDSF with the REXX programming language," on page 377.

Symbols with filtering

You can use system symbols (both static and dynamic) with filtering. The Filter pop-up lets you display a list of the current symbols and their values.

To disable symbol support, use the Command.FILTER.SymbolsDisabled custom property in ISFPARMS.

Exploitation tasks

Symbols use these special characters: &():. Setting the search characters to any of those characters with the SCHARS parameter of ISFPARMS or the SET SCHARS command will cause symbols to behave incorrectly. For information, refer to the description of SCHARS in "Global initialization parameters (OPTIONS or ISFPMAC)" on page 23.

For more information about the custom property, refer to "Customized properties (PROPLIST)" on page 91.

Display full sort and filter criteria

You can display full sort and filter criteria with the SET DISPLAY LONG command. SET DISPLAY ON continues to show a count for the number of filters, and shows the first two sort columns with a count of any additional columns. Both forms of SET DISPLAY also show the values for PREFIX, DEST, OWNER and SYSNAME.

Exploitation tasks

For more information, see the description of SET DISPLAY in the online help.

Long JES2 job class names

SDSF supports the JES2 enhancement for job class names of up to 8 characters. There are several changes to SDSF panels.

The default width of columns that display job class now expands to 8 when there are long (greater than 1-character) class names in the MAS. This affects:

- C on the DA, I, ST and RDR panels
- JC on the H and O panels.

On the JC panel, the Group column is now displayed and overtypeable for JES2, and a new column is added, as described in Table 12 on page 10.

Table 12. New Column on the JC Panel

Column Name	Title (Displayed)	Width	Description
CLACTIVE	Active	6	Indicates if the class is currently active (JES2 only)

On the INIT panel, the Classes column now shows periods for multi-character classes or groups in a JES2 environment. A new set of columns is added to show multi-character classes and groups. The first one is overtypeable. The new set of columns is described in Table 13.

Table 13. New Columns on the INIT Panel

Column Name	Title (Displayed)	Width	Description
ICLASS1-8	Class1-8	8	JES2 initiator classes 1-8, including multi-character classes and groups (JES2 only)

On the SO panel, the SClass column now shows periods for multi-character classes or groups in a JES2 environment. A new set of columns is added to show multi-character classes and groups. The first one is overtypeable. The new set of columns is described in Table 14.

Table 14. New Columns on the SO Panel

Column Name	Title (Displayed)	Width	Description
SCLASS1-8	SClass1-8	8	Selection classes 1-8, including multi-character classes and groups (job transmitters and receivers)

Exploitation tasks

If you have customized field lists for the panels that are affected by this enhancement, you may want to update your field lists to reflect the changes to columns. For more information, refer to "Variable field lists (FLD or ISFFLD)" on page 85.

Update security for the new overtypeable columns. If you are using ISFPARMS for security, the new Class1 column in the INIT panel, the Active and Group columns on the JC panel, and the SClass1 column on the SO panel, require command level 3. For the SAF resources, refer to Table 15.

Table 15. New SAF Resources

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
ACTIVE	JC	ISFATTR.JOBCL.ACTIVE	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
CLASS1-8	INIT	ISFATTR.SELECT.JOBCLASS	$T	*jesx*.MODIFY.INITIATOR	CONTROL
GROUP	JC	ISFATTR.JOBCL.GROUP	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
SCLASS1-8	SO	ISFATTR.SELECT.CLASS	$T	*jesx*.MODIFY.OFF	CONTROL

Trace SDSF security

SDSF's security trace function helps you understand and diagnose SDSF security (SAF or ISFPARMS). In response to the actions that you take, such as issuing commands or overtyping columns, it issues messages that describe the associated SAF resources or ISFPARMS statements. You control security trace with commands, REXX variable or Java methods.

- With the SET SECTRACE command, you turn security tracing on and specify how the associated messages are handled.
  - SET SECTRACE ON causes the trace messages to be sent to the ULOG.
  - SET SECTRACE WTP causes the messages to be issued as write-to-programmer messages. Use this if security prevents you from accessing SDSF or the user log.
- With the SECTRACE option on the SDSF command, you can turn security tracing on as soon as you access SDSF.
- When SDSF SECTRACE is active, SDSFAUX SECTRACE is also activated. SDSFAUX uses SECTRACE to record the results of security calls for diagnosis.
- With the ISFSECTRACE REXX special variable, you can control security tracing from a REXX exec.
- With ISFRequestSettings methods addISFSecTrace and removeISFSecTrace, you can control security tracing from a Java program.

Exploitation tasks

For more information about the commands, refer to the online help. You could use the SEARCH command, for example, `SEARCH SET SECTRACE`. For more information about the REXX special variable, refer to Chapter 13, “Using SDSF with the REXX programming language,” on page 377. For more information about Java, refer to Chapter 14, “Using SDSF with the Java programming language,” on page 451.

Print using source attributes, carriage control and other enhancements

When printing to SYSOUT, you can request that the SYSOUT use the attributes of the source. You can also specify additional attributes, including writer name, record length and record format.

You can request that SDSF use the record format of the output data set to determine how to handle carriage control for all printing, as follows:

- If the record format includes A, then the print function uses ASA (ANSI) carriage control.
- If the record format includes M, then the print function uses machine carriage control.
- Otherwise, SDSF removes carriage control characters if they are present in the source.

By default, SDSF always prints with ASA carriage control.

Exploitation tasks

To request that a print to SYSOUT use the attributes of the source, use a new field on the Open Print panel or the REXX special variable ISFPRTSOURCEATTS. To specify the new attributes when printing to SYSOUT, use fields on the Print panel or REXX special variables.

To request that SDSF's print function handle carriage control based on the record format, use the SET PRTCCASA command, REXX special variable ISFPRTCCASA or a custom property in ISFPARMS, Print.CCTL.AlwaysUseASA. For more information, see the online help and "Customized properties (PROPLIST)" on page 91. For more information about printing, refer to the online help.

Console name modification

When SDSF needs to activate an extended console and the default console name is in use (for example, when you invoke SDSF from a REXX exec while also using SDSF interactively) SDSF attempts to activate a new console with a different name, which is derived by modifying the default console name. To modify the name, SDSF appends a single-character suffix. SDSF can try up to 32 different characters until a unique console name is obtained. The original console name must be fewer than 8 characters long.

You can control console name modification with:

- The SET CONMOD (ON | OFF) command, which turns console name modification on and off.
- In ISFPARMS, the custom property Console.EMCS.ConModChars, which specifies the characters to be used as the suffix. By default, the characters are $#@12345.
- In ISFPARMS, the custom property Console.EMCS.NoConMod, which turns console name modification off.
- In a REXX exec, with the ISFCONMOD special variable.
- In a Java program, with ISFRequestSettings.

By default, console name modification is enabled.

Exploitation tasks

None required. For more information about the command, refer to the online help. For more information about console names, refer to "Extended console name" on page 343. For more information about the custom properties, refer to "Customized properties (PROPLIST)" on page 91. For information about the REXX special variable, refer to Chapter 13, "Using SDSF with the REXX programming language," on page 377. For information about ISFRequestSettings, refer to the Chapter 14, "Using SDSF with the Java programming language," on page 451.

Display of duplicate data sets

You can now control whether SDSF displays duplicate SYSOUT data sets when you browse or print job output. Duplicate data sets result from DD statements referencing more than one OUTPUT JCL control card. Control the display of duplicate data sets with:

- SET DUPDS command
- ISFDUPDS special variable in REXX
- Browse.Suppress.DupDS custom property in ISFPARMS.

Exploitation tasks

For more information about the command, refer to the online help. For more information about the custom properties, refer to "Customized properties (PROPLIST)" on page 91. For information about the REXX special variable, refer to Chapter 13, "Using SDSF with the REXX programming language," on page 377.

Custom property for CPU% normalization

A custom property, Panel.DA.CPUPctBasedLPAR, is added to control which CPU-busy value (MVS or LPAR) is used to normalize the CPU% column on the DA panel. The default is to use the MVS value.

Exploitation tasks

None required. For more information about the custom properties, refer to "Customized properties (PROPLIST)" on page 91.

Custom property to control the scope of panels

You can use custom properties in ISFPARMS to set the scope of the CK, DA, ENC and PS panels to the JESPLEX rather than the sysplex. The property names are in the format Panel.*panel-name*.JESPlexScope, where *panel-name* is CK, DA, ENC, PS or All.

Exploitation tasks

None required. For more information about the custom properties, refer to "Customized properties (PROPLIST)" on page 91.

Column changes

On the I and ST panels, the TGNum and TGPct columns, which show information about track groups, are now available in the JES3 environment. They are described in Table 16.

Table 16. Changes to Columns on the I and ST Panels

Column Name	Title (Displayed)	Width	Description
TGNUM	TGNum	5	Track groups used by a job
TGPCT	TGPct	6	Percentage of total track group usage

Columns are added to the DA, ENC, I, INIT, JC, JDS, ST and SO panels. They are described in Table 17.

Table 17. New Columns on the SDSF Panels

Panel	Column Name	Title (Displayed)	Width	Description
DA, H, I, O, ST	JOBCORR	JobCorrelator	32	User portion of the job correlator (JES2 only)
DA, ENC	IOPRIOGRP	IOPrioGrp	9	WLM I/O priority group
ENC	ARRTIME	Arrival-Time	19	Date and time the enclave was created
ENCdev	ARRINTV	Arrival-Int	11	Interval since the enclave was created (*hh:mm:ss*)
I, ST	DELAYRSN	DelayRsn	32	Reason for the job delay (JES2 only). The width can be expanded to 127.
JC	DSENQSHR	DSEnqShr	8	Indicates if JES should change data set enqueues to shared access when exclusive access is not required (JES2 only)

Table 17. New Columns on the SDSF Panels (continued)

Panel	Column Name	Title (Displayed)	Width	Description
JC	SYSSYM	SysSym	8	Indicates if system symbols are allowed in batch jobs (JES2 only)
JDS	LRECL	LRecL	5	Logical record length

Exploitation tasks

If you have customized field lists for the panels that are affected by this enhancement, you may want to update your field lists to reflect the changes to columns. For more information, refer to "Variable field lists (FLD or ISFFLD)" on page 85.

Update security for the new overtypeable columns. If you are using ISFPARMS for security, the new SysSym and DSEnqShr columns on the JC panel require command level 3. For the SAF resources, refer to Table 18.

Table 18. SAF Resources for New Overtypeable Columns

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 / Command, JES3	OPERCMDS Resource, JES2 / OPERCMDS Resource, JES3	Required Access
DSENQSHR	JC	ISFATTR.JOBCL.DSENQSHR	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
SYSSYM	JC	ISFATTR.JOBCL.SYSSYM	$T	*jesx*.MODIFY.JOBCLASS	CONTROL

Chapter 2. Using ISFPARMS for customization and security

This topic describes SDSF's internal parameters, ISFPARMS, and explains how to use ISFPARMS to customize SDSF and provide security for SDSF.

ISFPARMS overview

ISFPARMS defines global and group options and the format of the panels. The options include things like the name of the JES subsystem to process, what generic and wildcard characters to allow in SDSF commands, and whether to display the action bar on SDSF panels. The format of the panels includes the order and titles of the columns.

In a JES2 environment, ISFPARMS can also be used to provide security for SDSF functions, though SAF, which provides better granularity and auditability, is the recommended method. When you use SAF for user authorization, you need ISFPARMS only to define your global and group options. You might also use ISFPARMS as a backup to SAF. Using SAF for security is described in Chapter 5, "Using SAF for security," on page 191.

In a JES3 environment, SDSF function can be protected only with SAF.

You can further customize authorization with an installation exit routine, as described in Chapter 9, "Using installation exit routines," on page 329.

ISFPARMS format alternatives

SDSF provides two alternatives for ISFPARMS:

- Assembler macros that you define, assemble, and then link into the SDSF load library. This is the original format for defining ISFPARMS, and it continues to be supported for compatibility in the JES2 environment. However, some functions, such as conditional processing and defining a server group for sysplex support, are not available using this format, and it is not supported in the JES3 environment.
- The ISFPRM*xx* member of PARMLIB. This is the **recommended format**. The statements in ISFPRM*xx* are easier to define and are more dynamic than the assembler macros: they can be updated without reassembling or link-editing. For a JES3 environment, you must use the ISFPRM*xx* member.

  The statements in ISFPRM*xx* are processed by an SDSF server, which is controlled by MVS operator commands. The server and associated commands are described in detail in Chapter 3, "Using the SDSF server," on page 107.

  SDSF provides a utility for converting ISFPARMS assembler macros to ISFPRM*xx* statements. See "Converting ISFPARMS assembler macros to statements" on page 17.

  In some cases, SDSF may revert from processing ISFPRM*xx* to the ISFPARMS defined with assembler macros. This is described in "Reverting to the ISFPARMS defined with assembler macros" on page 17.

For simplicity, this information refers to both the assembler macro ISFPARMS and PARMLIB member ISFPRM*xx* as ISFPARMS.

© Copyright IBM Corp. 1997, 2016

To assist you in defining your ISFPARMS, SDSF provides sample ISFPRM*xx* members as well as a sample ISFPARMS defined with assembler macros. You can modify the appropriate sample to meet the needs of your installation.

The statements and corresponding assembler macros that make up ISFPARMS are summarized in Table 19.

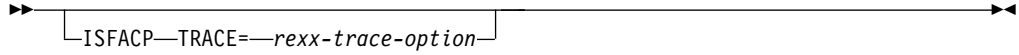
Table 19. Summary of ISFPARMS Statements and Macros

Statement	Assembler Macro	Required	Description	Refer to
OPTIONS	ISFPMAC	Assembler only	Specifies global SDSF initialization parameters.	"Global initialization parameters (OPTIONS or ISFPMAC)" on page 23
SERVERGROUP + SERVER + COMM	Not available	For sysplex data only	Defines the group of servers that provide sysplex data, and the communication between servers, when using WebSphere MQ. Used only in a JES2 environment.	"Server group definition parameters (SERVERGROUP, SERVER, COMM)" on page 30
CONNECT	Not available	No	Defines server connection properties, SDSFAUX options, and the XCF application server name.	"CONNECT statement" on page 34
GROUP	ISFGRP	No	Defines a group of users and the SDSF functions that will be available to a member of the group. Also includes initialization parameters. You can use SAF along with your group definitions to control membership and authorization.	"Group authorization parameters (GROUP or ISFGRP)" on page 36
FLD + FLDENT	ISFFLD	No	Customizes the fields shown on an SDSF primary or alternate panel for members of a group. Associated with an ISFGRP macro or GROUP statement.	"Variable field lists (FLD or ISFFLD)" on page 85
NTBL + NTBLENT	ISFNTBL	No	Specifies such things as user IDs, job names, and destination names to further qualify group membership and authority. Associated with an ISFGRP macro or GROUP statement.	"Name tables (NTBL or ISFNTBL)" on page 89
PROPLIST + PROPERTY	Not available	No	Specifies a property to customize. Provides an alternative to a user exit routine. Associated with a GROUP statement.	"Customized properties (PROPLIST)" on page 91
TRTAB + TRDEF	ISFTR	Assembler only	Specifies the code page that SDSF uses for members of the group. Associated with an ISFGRP macro or GROUP statement.	"Code page (TRTAB/TRDEF or ISFTR)" on page 101
WHEN	Not available	No	Provides conditional processing of statements	"Conditional processing" on page 20

Converting ISFPARMS assembler macros to statements

If you already have defined ISFPARMS with the assembler macros, you can use a conversion utility provided by SDSF to convert them to the statement format used in ISFPRM*xx*. The utility displays a panel on which you type the names of the input data set (ISFPARMS assembler macros) and output data set (statements), as well as the assembler macro library you use when assembling SDSF. You invoke the utility by typing the ISFACP command on the ISPF command line.

The syntax of the ISFACP command is shown below:



Because the macros in ISFPARMS can be coded in different ways, the conversion may not be exact. You may need to make corrections to the generated statements.

The conversion utility processes only SDSF macros. For the utility to resolve macro labels to names, the labels must be on the same line as the macros. The conversion utility will attempt to add the labels if it recognizes common coding conventions, such as a label defined with a `DS 0H`, `DS 0F`, or `EQU *` immediately preceding an SDSF macro.

Reverting to the ISFPARMS defined with assembler macros

SDSF is connected to the SDSF server during SDSF initialization, and uses the server to process the ISFPARMS defined with statements. However, if the server is not active or if no statements are in effect, SDSF reverts to the ISFPARMS defined with the assembler macros.

If you have migrated to an ISFPARMS defined with statements, you may want to restrict or prevent SDSF from reverting to the ISFPARMS defined with assembler macros. You can do this by:

- Defining SAF profiles for the SERVER.NOPARM resource in the SDSF class. This profile is checked when the required server is not active or when no ISFPARMS statements are in effect. If the user has READ access to the profile (or if SAF returns an indeterminate response) SDSF reverts to the ISFPARMS in assembler macro format.

  If the user does not have access to the profile, SDSF does not revert to the assembler macros and the user is not authorized to SDSF.
- Coding only one ISFGRP macro, for system programmers, so that only those users can access SDSF when the server is not available.
- Coding no ISFGRP macros. In that case, users cannot be assigned to a group, and so they cannot access SDSF.

For information about defining the ISFGRP macro, see “Group authorization parameters (GROUP or ISFGRP)” on page 36.

Note that if you code the SERVER parameter in the assembler ISFPARMS, SDSF will use it to define the default SDSF server. All other parameters will be taken from the ISFPARMS statements processed by the server.

Samples

SDSF supplies the following samples in the AISFJCL library:

- ISFPRM00, which is an ISFPARMS in statement format that matches SDSF's sample ISFPARMS in assembler macro format
- ISFPRM01, which is the same as ISFPRM00 with the addition of field lists for the tabular displays.

A sample ISFPARMS module in assembler macro format for the JES2 environment is provided in member ISFPARMS of the data set ISF.SISFSRC1, with the JES2 feature. The ISFPARMS sample provides security with ISFPARMS parameters only. Sample ISFPARMS parameters in conjunction with SAF profiles are shown in "Examples of RACF security for SDSF groups" on page 312.

Auditing ISFPARMS

When you use the statement format of ISFPARMS, SDSF provides an audit trail of all statements that have been processed. The statements and any associated error messages are written to a log file that you allocate in the server JCL.

Diagnosing security

SDSF's security trace function helps you understand and diagnose SDSF security (SAF or ISFPARMS). In response to the actions that you take, such as issuing commands or overtyping columns, it issues messages that describe the associated SAF resources or ISFPARMS statements. You control security trace with commands, REXX variable or Java methods.

- With the SET SECTRACE command, you turn security tracing on and specify how the associated messages are handled.
  - SET SECTRACE ON causes the trace messages to be sent to the ULOG.
  - SET SECTRACE WTP causes the messages to be issued as write-to-programmer messages. Use this if security prevents you from accessing SDSF or the user log.
- With the SECTRACE option on the SDSF command, you can turn security tracing on as soon as you access SDSF.
- When SDSF SECTRACE is active, SDSFAUX SECTRACE is also activated. SDSFAUX uses SECTRACE to record the results of security calls for diagnosis.
- With the ISFSECTRACE REXX special variable, you can control security tracing from a REXX exec.
- With ISFRequestSettings methods addISFSecTrace and removeISFSecTrace, you can control security tracing from a Java program.

For more information about the commands, refer to the online help. You could use the SEARCH command, for example, `SEARCH SET SECTRACE`. For more information about the REXX special variable, refer to Chapter 13, "Using SDSF with the REXX programming language," on page 377. For more information about Java, refer to Chapter 14, "Using SDSF with the Java programming language," on page 451.

Rules for coding ISFPARMS

This section describes the rules for syntax and implementation of ISFPARMS.

Statements

Enter statements as card images in a data set that you create with any editor. The data set is identified to the SDSF server through the server startup JCL.

The ISFPARMS statements use a *keyword*(*value*) format. For example, a GROUP statement might look like this:

```
GROUP TSOAUTH(JCL,OPER,ACCT),
  AUTH(LOG,I,O,H,DA,INIT,PR,LI,NO,DEST)
```

The complete set of rules for specifying ISFPARMS statements follows.

General rules for coding statements

- A statement is 80 characters long. Use columns 1 through 72 for the statement; columns 73 through 80 are ignored.
- A statement can span any number of lines. To indicate that the statement continues on the next line, use a trailing comma.
- Enclose comments in a `/**/` pair, for example, `/* comment */`. You can include comments anywhere in a record that a blank is valid. A comment cannot span lines; it must be closed on the line on which it begins.
- When you use a trailing comma to continue a statement, the only thing that can follow the comma on that line is a comment.
- Completely blank lines (in columns 1 through 72) are ignored; you can intersperse them freely with statements.

Rules for statement types, keywords, and values

The exact syntax of each of the statements is defined in the remainder of this topic. However, the following general rules apply to the statements and their keywords:

- Parameters must be separated from one another by a comma or a blank. Any number of blanks may appear between keywords, values, and commas, and parentheses.
- Each statement must have at least one keyword on the same line.
- Values are translated to uppercase. If the value contains embedded blanks or is case-sensitive, enclose it in single quotes.
- Parameters can be in any order in a statement.
- Statements can appear in any order; however, FLDENT statements must appear after an FLD statement and NTBLENT statements must appear after an NTBL statement.
- To specify a value of blanks, enclose one or more blanks in single quotation marks, for example, `' '`.

Duplicate statements:
In general, when SDSF encounters a duplicate statement, it uses the values from the last statement. However, duplicate FLDENT and NTBLENT statements are processed multiple times. For example, a duplicate field appears twice in the list.

Assembler macros

Code the ISFPARMS module according to standard MVS assembler language rules. The macros use a *keyword=value* format. In addition,

- The ISFPMAC macro must be the first macro in ISFPARMS, and only one ISFPMAC macro may be coded.
- The ISFGRP macros must be coded second, after the ISFPMAC macro and before any ISFNTBL, ISFFLD, and ISFTR macros.

- At least one ISFTR macro must be included.

After coding the ISFPARMS module, assemble and link-edit it. ISFPARMS must be reentrant. You can use the SMP/E procedure described in Chapter 10, "Installation and configuration considerations," on page 339.

Conditional processing

To facilitate using a common ISFPARMS for multiple systems, SDSF provides support for:

- A WHEN statement that allows you to identify statements that apply to a particular system
- System symbols in the ISFPARMS statements.

Conditional processing is available only with the statement format of ISFPARMS. It is not available with the assembler format of ISFPARMS.

Note that, even with conditional processing, if you want to use a common ISFPARMS with different levels of SDSF, you must ensure that the ISFPARMS does not include support (such as new keywords or values) that was introduced in the higher level of SDSF.

WHEN Statement

The WHEN statement can be used to conditionally process an entire ISFPARMS statement (OPTIONS, GROUP, and so on). The WHEN statement specifies one or more conditions which are compared to the current environment. All of the conditions must be true for the statements that follow to be processed.

In processing a WHEN statement, SDSF checks each of the values against the current system. If all values match the current system, the statements that follow the WHEN statement are processed until the next WHEN is encountered, or until the end of the file is reached. If any of the values do not match the current system, the statements that follow the WHEN statement are checked for syntax but not processed, until the next WHEN is encountered.

The WHEN statement cannot be used to conditionally process a single parameter within a statement. For example, use WHEN to conditionally process an entire OPTIONS statement with all of its parameters, not to conditionally process just the TIMEOUT parameter of OPTIONS. This means that if even a few parameters in a statement vary between systems, multiple versions of the statement may be required. (System symbols, described in "System symbols" on page 22, can be used to replace the value for a single parameter.)

Messages logged by the server indicate which initialization statements are being processed.

WHEN and all of its parameters are optional. WHEN with no parameters causes the statements that follow (until the next WHEN) to be selected; this can be used to end a preceding WHEN.

The parameters are in the format *keyword*(*value*). The value for *value* can be any text string, including standard pattern matching characters:

- *, which represents any string of characters
- %, which represents any single character.

The SYMBOL keyword lets you specify an expression for the value.

WHEN parameters

The parameters that describe the processing conditions are described below.

Parameter	Description
LPARNAME(*lpar-name*)	Name of the LPAR
SYSNAME (*system-name*)	Name of the system
SYSPLEX (*sysplex-name*)	Name of the sysplex
HWNAME (*processor-name*)	Name of the CPC
VMUSERID (*vm-userid*)	User ID of a VM system under which MVS is running
SERVER (*sdsf-server-name*)	Name of the SDSF server
SYMBOL(*expression*)	Evaluate an expression using one or more symbols

LPARNAME (*lpar-name*)

Names a logical partition that is defined to a processor, which is one of the following: the partition name specified on the 'add partition' panel in HCD, or the partition name specified on the resource or chpid statement that is input to the I/O configuration program (IOCP). Specify a value of ' ' (one or more blanks enclosed by single quotation marks) to indicate a processor that is not initialized in lpar mode.

SYSNAME (*system-name*)

Specifies the name assigned to an MVS system.

SYSPLEX (*sysplex-name*)

Names the sysplex this MVS system is in.

HWNAME (*processor-name*)

Names the central processor complex (CPC) as defined to HCD. Note: specify a value of ' ' (one or more blanks enclosed by single quotation marks) to indicate a processor with no name.

VMUSERID (*vm-userid*)

Specifies the user ID of a VM system under which MVS is running as a guest. Specify a value of ' ' (one or more blanks enclosed by single quotation marks) to indicate a system not running as a guest under VM.

SERVER (*sdsf-server-name*)

Names the SDSF server processing the statements.

SYMBOL (*expression*)

Checks for a value for any system static symbol. These are defined in the IEASYMxx parmlib member.

The format is WHEN SYMBOL(*x*= | ^=*y*,...) where the operands *x* and *y* can be either strings or symbols. The comparison is either equal or not equal. A symbol is expressed as &*name*. The operands can be specified in either order (for example, &SYSNAME=SYS1 or SYS1=&SYSNAME). If an operand does not evaluate to a symbol, the string is checked as is.

For the "equal" condition, the strings must match in length and content. Strings are case sensitive. To specify a "not equal" condition, use ^=, /= or \=.

You can specify any number of conditions, separated by a comma; all must be true for the statement to be accepted.

You can combine the SYMBOL keyword with any other WHEN keyword; all keywords must evaluate to true to be accepted.

If more than one SYMBOL keyword is present, the last one replaces any prior ones regardless of the previous conditions that were processed (that is, conditions cannot be replaced individually).

Examples of the WHEN statement:

1. WHEN SYMBOL(&SYSNAME ^=SY1)

   This is accepted when the value of symbol SYSNAME is not equal to SY1. Note that this will also be accepted if SYSNAME is not a defined symbol, as the character string &SYSNAME is not equal to the string SY1.
2. WHEN SYMBOL(&SYSNAME=SY1, &SYSPLEX=PLEX1)

   This is accepted when the value of symbol SYSNAME is equal to SY1, and the value of symbol SYSPLEX is equal to PLEX1.
3. WHEN SYMBOL(&SYSPLEX=PLEX1) SYSNAME(SY1)

   This example shows a WHEN with two conditions, one of which uses a symbol. This WHEN is accepted when the value of the symbol SYSPLEX is PLEX1 and the sysname is SY1.

System symbols

Statements can include system symbols for keyword values. Symbols in ISFPARMS are identified by an initial ampersand (&). They also have an ending period, though the period is required only if omitting it would cause ambiguity. It is required if the character that follows is a period.

System symbols are not supported in the assembler macro format of ISFPARMS.

For example, the HASPINDX data set name may vary by system. A system symbol can be used to substitute the data set name when the parms are processed, as follows:

- &SYSNAME is a symbol for the system name. It might be defined in the IEASYMxx or IEASYSxx parmlib members; if not, its default value is the processor identifier.
- You update your ISFPARMS to use the symbol in defining the name of the HASPINDX data set:

  ```
  INDEX(&SYSNAME..HASPINDX)
  ```
- When the ISFPARMS are processed, the system name is substituted for &SYSNAME, resulting in a HASPINDX data set name that is correct for the system. Note that in this example, the ending period for &SYSNAME. is required, so that the period used to separate data set qualifiers is preserved.
- The server initialization log will show the actual value used when the statement was processed.

Global initialization parameters (OPTIONS or ISFPMAC)

The OPTIONS statement or ISFPMAC macro specifies the global initialization parameters for SDSF.

In ISFPARMS assembler macros, ISFPMAC must be the first macro, and there can be only one ISFPMAC macro.

Example of the OPTIONS statement and ISFPMAC macro

OPTIONS Statement	ISFPMAC Macro
1 OPTIONS NIDBUF(5), 2 SYSOUT(A),IDBLKS(4096),LINECNT(55), 3 MENUS(ISF.SISFPLIB), 4 INDEX(ISF.HASPINDX), 5 FINDLIM(100000),SCRSIZE(3440), 6 SCHARS('*%'),DCHAR('?'),TRACE(C000)	1 ISFPMAC NIDBUF=5, 2 SYSOUT=A,IDBLKS=4096,LINECNT=55, 3 MENUS='ISF.SISFPLIB', 4 INDEX='ISF.HASPINDX', 5 FINDLIM=100000,SCRSIZE=3440, 6 SCHARS=*%,DCHAR=?,TRACE=C000

On line **1** in the example, the NIDBUF parameter indicates that five SYSLOG index buffers are to be allocated to each user.

On line **2**, the SYSOUT parameter specifies the default SYSOUT class for the SDSF PRINT command. The IDBLKS parameter specifies the block size for the HASPINDX data set, which determines the maximum number of SYSLOG data sets that SDSF will support. The more SYSLOG data sets there are, the larger the block size needs to be; the more data in the data sets, the bigger the HASPINDX data set must be. The LINECNT parameter specifies 55 lines per page of printed output when using the PRINT command to print portions of the system log or output data sets.

On line **3**, the MENUS parameter specifies that the name of the SDSF help panel data set is ISF.SISFPLIB.

Note: This parameter applies only when SDSF is running under TSO or as a TMP. It is not used when SDSF is running as an ISPF dialog.

On line **4**, the INDEX parameter specifies that the name of the HASPINDX data set is ISF.HASPINDX.

On line **5**, the FINDLIM parameter specifies that the FIND command will search up to 100,000 lines on a single pass before displaying the number of lines searched. In the above example, the SCRSIZE parameter specifies that the maximum screen size on which SDSF will be used is 3440 characters.

On line **6**, the SCHARS parameter specifies the search character used for PREFIX and OWNER pattern matching. The DCHAR parameter specifies the display query character. The TRACE parameter specifies the events you wish to trace with the TRACE facility.

OPTIONS or ISFPMAC reference

The parameters that can be coded in the OPTIONS statement or ISFPMAC macro are show below. Defaults are underlined.

OPTIONS	ISFPMAC	Description
ADMSYMBL (*symbol-sets-dsn*)	ADMSYMBL=*symbol-sets-dsn*	GDDM symbols
ATHOPEN (YES) | (NO)	ATHOPEN=YES | NO	Obsolete and ignored
DCHAR ('?') | ('*query-char*')	DCHAR=? | *query-char*	Query character
DSI (NO) | (YES)	DSI=NO | YES	Data set integrity enqueue
FINDLIM (5000) | (*lines-searched*)	FINDLIM=5000 | *lines-searched*	Lines searched by FIND
IDBLKS (4096) | (*block-size*)	IDBLKS=4096 | *block-size*	HASPINDX block size
INDEX (ISF.HASPINDX) | (*ds-name*)	INDEX=ISF.HASPINDX | *ds-name*	HASPINDX data set name
INDXVOL (*volume-serial*)	INDXVOL=*volume-serial*	HASPINDX volser
JESNAME (*user-JES2-name*) | (*JES2-name*)	JESNAME=*user-JES2-name* | *JES2-name*	Name of the JES2 subsystem that is processed
JES3NAME (*user-JES3-name*) | (*JES3-name*)	Not available	Name of the JES3 subsystem that is processed
LINECNT (55) | (*lines*)	LINECNT=55 | *lines-per-page*	Lines per page
LOGLIM (0) | (*hours-searched*)	LOGLIM=0 | *hours-searched*	Hours of OPERLOG data filtered
MENUS (ISF.SISFPLIB) | (*ds-name*)	MENUS=ISF.SISFPLIB | *ds-name*	SDSF panels data set
MENUVOL (*volume-serial*)	MENUVOL=*volume-serial*	Panels data set volser
NIDBUF (5) | (*number-of-buffers*)	NIDBUF=5 | *number-of-buffers*	Buffers used for HASPINDX
SCHARS ('*%') | ('*search-characters*')	SCHARS=*% | *search-characters*	Pattern matching characters
SCRSIZE (1920) | (*screen-size*)	SCRSIZE=1920 | *screen-size*	Screen size
N/A	SERVER=SDSF | *SDSF-server-name*	Default SDSF server name
SYSOUT (A) | (*class*)	SYSOUT=A | *class*	Default print class
TIMEOUT (5) | (*seconds*)	TIMEOUT=5 | *seconds*	Default timeout interval (JES2 only)
TRACE (C000) | (*trace-masks*)	TRACE=C000 | *trace-masks*	Default trace masks
TRCLASS (A) | (*class*)	TRCLASS=A | *class*	Default trace SYSOUT class
UNALLOC (NO) | (YES)	UNALLOC=NO | YES	Free files at termination

OPTIONS	ISFPMAC
ADMSYMBL (*symbol-sets-data-set-name*)	ADMSYMBL=*symbol-sets-data-set-name*

Defines a default GDDM symbol sets data set to be used when displaying page-mode data with the V action character.

symbol-sets-data-set-name is the name of a cataloged data set for the GDDM symbol sets. This data set will be dynamically allocated by SDSF only if the ADMSYMBL ddname is not already allocated.

There is no default for ADMSYMBL. If you don't specify this keyword, SDSF will not allocate a symbol sets data set.

OPTIONS	ISFPMAC
DCHAR ('?') | ('*query-char*')	DCHAR= ? | *query-char*

Defines the query character for use with commands, to display their current values. The character you specify must be different from the SCHARS value. Also, be sure to tell your users what the new query character is. The default is ?. When using statements, enclose the query character in quotation marks.

OPTIONS	ISFPMAC
DSI (NO) | (YES)	DSI=NO | YES

YES
specifies that dynamically allocated data sets are to be enqueued upon by SDSF for the user when they are allocated.

NO is the default and specifies that dynamically allocated data sets are not to be enqueued upon (for data set reservation) by SDSF for the user when they are allocated.

OPTIONS	ISFPMAC
FINDLIM (5000) | (*lines-searched*)	FINDLIM=5000 | *lines-searched*

Specifies the maximum number of lines the FIND command will search on a single pass before displaying the number of lines searched. When running under ISPF, the FINDLIM value is saved and restored across sessions if the user is authorized to issue the command. See the online help for a description of the FIND command.

OPTIONS	ISFPMAC
IDBLKS (4096) | (*block-size*)	IDBLKS=4096 | *block-size*

Specifies the block size for the HASPINDX data set, which SDSF uses for indexing information in the SYSLOG spool data sets for JES2 prior to z/OS V1R11. The block size determines the maximum number of SYSLOG data sets in the spool data set that SDSF will support. For most installations, the default block size of 4096 will be adequate.

OPTIONS	ISFPMAC
INDEX (ISF.HASPINDX) | (*data-set-name*)	INDEX=ISF.HASPINDX | *data-set-name*

Specifies the name of the HASPINDX data set. The name can be up to 44 characters long. The INDEX and INDXVOL parameters are used only if the data set is to be dynamically allocated. If a HASPINDX DD statement was included in the TSO logon procedure, that data set is used.

This dataset requires the user to have UPDATE access. If the user does not have UPDATE access to HASPINDX, the open fails and the SYSLOG function using HASPINDX is inoperative for the session.

Note: In some cases, SDSF falls back to HASPINDX when the SYSID command is used to specify a member in the MAS that has been defined to JES but has never been IPLed. In this case, the level of SYSLOG is not known. SDSF uses HASPINDX but the SYSLOG dataset is not found. You can correct this by specifying a SYSID for a member that exists.

Note that if you specify the same HASPINDX data set in ISFPARMS for more than one system, those systems share the HASPINDX data set. Sharing a HASPINDX

data set reduces the number of data sets required, but may result in degraded performance due to contention. For that reason, IBM® does not recommend sharing the HASPINDX data set. For more information, see "SYSLOG" on page 349. The HASPINDX data set is used only for JES2 systems prior to z/OS V1R11.

OPTIONS	ISFPMAC
INDXVOL (*volser*)	INDXVOL=*volume-serial*

Specifies the volume serial number of the HASPINDX data set. The maximum length is 6 characters. If this parameter is omitted, the index data set is assumed to be cataloged correctly. The INDEX and INDXVOL parameters are used only if the data set is to be dynamically allocated, and only for JES2 systems prior to z/OS V1R11. If a HASPINDX DD statement was included in the TSO logon procedure, that data set is used.

OPTIONS	ISFPMAC
JESNAME (*user-JES-name*) | (*JES-name*)	JESNAME=*user-JES-name* | *JES-name*

Indicates the name of the JES2 subsystem. The name can be 1 to 4 characters. The default is the JES system the user is currently running under.

For information on specifying this parameter when SDSF is installed to run with a secondary JES2 subsystem, see "SDSF with a secondary JES2 subsystem" on page 339. This applies to JES2 only; for JES3, use the JES3NAME parameter.

OPTIONS	ISFPMAC
JES3NAME (*) | (*JES-name*)	not available

Indicates the name of the JES3 subsystem. The name can be 1 to 8 characters. The default is *, which requests the JES system the user is currently running under.

OPTIONS	ISFPMAC
LINECNT (55) | (*lines*)	LINECNT=55 | *lines-per-page*

Specifies the number of lines per page of printed output when using the PRINT command to print portions of the SYSLOG or OPERLOG.

OPTIONS	ISFPMAC
LOGLIM (0) | (*hours-searched*)	LOGLIM=0 | (*hours-searched*)

Specifies the maximum amount of OPERLOG data, in hours, that SDSF will search on a single pass for OPERLOG records that meet filter criteria. If LOGLIM is omitted, the value is set to 0, which indicates no maximum.

Valid values are 0-999.

SDSF searches the OPERLOG data until it finds enough records to fill the screen, or until it reaches the limit, whichever comes first.

Users can override *hours* with the LOGLIM command. Under ISPF, the LOGLIM value is saved across sessions.

OPTIONS	ISFPMAC
MENUS (ISF.SISFPLIB) | (*data-set-name*)	MENUS=ISF.SISFPLIB | *data-set-name*

Specifies the name of the SDSF panel data set. This dataset requires READ access when SDSF is running as a TSO command. The MENUS and MENUVOL parameters are used only for dynamic allocation of the panels when running under TSO. If the SDSFMENU DD statement was included in the TSO logon procedure, that data set is used.

OPTIONS	ISFPMAC
MENUVOL (*volume-serial*)	MENUVOL=*volume-serial*

Specifies the volume serial number of the SDSF panel data set. It can be 1 to 6 characters long. If this parameter is omitted, the data set is assumed to be cataloged correctly. The MENUS and MENUVOL parameters are used only for dynamic allocation. If the SDSFMENU DD statement was included in the TSO logon procedure, that data set is used.

OPTIONS	ISFPMAC
NIDBUF (5) | (*number-of-buffers*)	NIDBUF=5 | *number-of-buffers*

Specifies the number of HASPINDX data set buffers to be allocated to each user. It can be a decimal number from 3 to 9. The default is 5. The HASPINDX data set is used only for JES2 systems prior to z/OS V1R11.

OPTIONS	ISFPMAC
SCHARS ('*%') | ('*search-characters*')	SCHARS=*% | *search-characters*

Specifies the generic and placeholder characters. These characters are used wherever pattern matching is supported.

The values for *search-characters* are of the form *ab*, where *a* is the generic character and *b* is the placeholder character. The values cannot be alphabetic, numeric, or national characters; they cannot be @, #, $, &,; the ISPF end-of-line character, the current query character, blank, or equal to each other. In addition, using :, (or) may interfere with using system symbols with filtering. The defaults are * and %.

When you use statements, enclose the characters in quotation marks.

OPTIONS	ISFPMAC
SCRSIZE (1920) | (*screen-size*)	SCRSIZE=1920 | *screen-size*

Specifies the maximum size, in characters, of the largest terminal screen on which SDSF will be used.

OPTIONS	ISFPMAC
not available	SERVER=SDSF | *SDSF-server-name*

Not valid in statements. Specifies the default SDSF server job name to be used in processing ISFPARMS statements. It is 1 to 8 characters. If this parameter is omitted, the default server is SDSF. Users can override the server name with the SERVER keyword on the SDSF command. See Chapter 3, "Using the SDSF server," on page 107.

Specifying a value with the SERVER parameter allows you to restrict use of a particular server. This may be desirable in a test environment, or when SDSF maintenance is applied.

As an alternative to this parameter, you can define a default server with the CONNECT statement. Using CONNECT may eliminate your need for any assembler ISFPARMS. See "CONNECT statement" on page 34.

OPTIONS	ISFPMAC
SYSOUT (A) | (*class*)	SYSOUT=A | *class*

Specifies the default SYSOUT class for the SDSF PRINT command.

OPTIONS	ISFPMAC
TIMEOUT (5) | (*seconds*)	TIMEOUT=5 | *seconds*

Specifies the default timeout interval, in seconds, for awaiting sysplex data on the JES2 device and resource panels, and on the SYSLOG panel, when sysplex data is provided with WebSphere® MQ or XCF. A value of 0 means that SDSF should not wait, that is, sysplex data is not available on those panels. This parameter is allowed in the assembler ISFPMAC macro, but the sysplex support requires the statement format of ISFPARMS.

If this parameter is omitted, 5 seconds is used.

This is ignored in a JES3 environment.

For more information, see "Using the server for sysplex data" on page 110.

OPTIONS	ISFPMAC
TRACE (C000) | (*trace-masks*)	TRACE=C000 | *trace-masks*

Specifies the default event mask to be used by the trace facility. You can trace several events at one time by combining the mask values (in hexadecimal). The *mask* is a hexadecimal number that is 2, 4, 6, or 8 characters long. Each bit in the number represents a specific SDSF event to be traced. Leading zeros are not required, but the resulting mask must have an even number of digits.

The trace masks are:

Mask	Description
FFFFFFFF	Unconditional trace
00800000	Message service
00400000	Communications events
00200000	ISFPARMS statements

Mask	Description
00100000	Filter
00080000	Log processing
00040000	Internal interfaces
00020000	ISPF services
00010000	RMF processing
00008000	SDSF initialization
00004000	SDSF JES2 initialization
00002000	Call
00001000	Return
00000800	TSO data stream, ISPF buffers, batch input and output
00000400	Device and node processing
00000200	GDDM processing
00000100	SJF processing
00000080	SAF processing
00000040	Spool I/O and SRB processing
00000020	SSI processing, MVS/JES commands and job classes
00000010	Data set processing
00000008	External interfaces, WLM scheduling environments and WLM resources
00000004	User exit call, return, and parameter list
00000002	ULOG functions
00000001	Reserved
00000000	No trace

SDSF trace is intended to be used under the direction of IBM service.

OPTIONS	ISFPMAC
TRCLASS (A) | (*class*)	TRCLASS=A | *class*

Specifies the default sysout class used by SDSF when dynamically allocating a trace file.

OPTIONS	ISFPMAC
UNALLOC (NO) | (YES)	UNALLOC=NO | YES

YES
indicates that when an SDSF session is terminated, all dynamically allocated data sets are to be freed.

NO is the default and indicates that SDSF will not free dynamically allocated data sets. They will be available if the user should begin another SDSF session before logging off.

Server group definition parameters (SERVERGROUP, SERVER, COMM)

A server group is a group of SDSF servers that communicate to provide sysplex-wide data using WebSphere MQ. It is not needed if all systems in the sysplex are at the z/OS V1R13 level or higher.

When one or more systems in the sysplex is at the z/OS V1R12 level or lower, you may need a server group to display sysplex-wide data on device and resource panels. For more information on the requirements for sysplex-wide data, refer to "Using the server for sysplex data" on page 110.

You define a server group for each SDSF server, using these statements:

- SERVERGROUP. This marks the beginning of the server group definition.
- SERVER. Each SERVER statement provide details about an SDSF server in the group.
- COMM. This statement provides information about the method of communication between servers.

You can define a server group only with statements; there are no equivalent assembler macros.

In addition to defining a server group, you may want to define a default server, using the CONNECT statement. See "CONNECT statement" on page 34. You can also define a default server with the SERVER parameter of the ISFGRP macro in the assembler verison of ISFPARMS.

Example of the SERVERGROUP statement

SERVERGROUP Statement

```
1 SERVERGROUP
2       SERVER NAME(SDSF),SYSNAME(SY1),JESNAME(JES2),MEMBER(AQFT),
                          COMM(SDSFSY1)
3       SERVER NAME(SDSF),SYSNAME(SY2),JESNAME(JES2),MEMBER(AFTS)
4       COMM NAME(SDSFSY1),TYPE(MQS),QMGR(MQ1),QPREFIX(ISF)
```

On line **1** in the example, the SERVERGROUP statement begins the server group definition.

On line **2**, the first server in the group is defined. The name of the server is SDSF, and it resides on system SY1. This server will process requests for JES2 subsystem JES2, member name AQFT. The COMM parameter refers to a COMM statement named SDSFSY1. The COMM statement, which follows, defines values that are used in communicating with the server.

On line **3**, a second server, also named SDSF but residing on system SY2, and processing JES2 subsystem JES2, member AQTS, is defined. No COMM parameter is present, so the communication will use defaults.

On line **4**, a COMM statement defines characteristics for communicating with the server. The value for the TYPE parameter, MQS, identifies this communication as using WebSphere MQ. The QMGR and QPREFIX parameters define the WebSphere MQ queue manager name and queue prefix values.

As a result of this server group definition, data from server SDSF on SY1 will be merged with the data from server SDSF on SY2.

For more examples and further discussion of server groups, see Chapter 3, "Using the SDSF server," on page 107.

SERVERGROUP statement

The SERVERGROUP statement has no parameters. It marks the beginning of a server group definition. You should code only one SERVERGROUP statement; if a second one is encountered, it replaces the previous one.

Note that the presence of a SERVERGROUP statement causes SDSF to use the server to gather sysplex-wide data. If you do not require the sysplex support provided by the server group, it is recommended that you not define a SERVERGROUP statement or its associated SERVER and COMM statements.

SERVER statement

A server group must include at least two SERVER statements, including one for the *local* server, up to a maximum of 32 servers. The local server is the one specified on the SERVER parameter of the ISFPMAC macro in ISFPARMS. From the user's perspective, it is the server the user is connected to.

The following table shows the parameters that you can code on a SERVER statement.

Parameter	Description
NAME (*server-name*)	Name of the SDSF server
SYSNAME (*system-name*)	Name of the system on which the server resides
JESNAME (*JES-subsystem-name*)	Name of the JES subsystem for which this server will obtain data
MEMBER (*jes-member-name*)	Name of the JES member for which this server will obtain data
COMM (*COMM-statement-name*)	Characteristics of the communication method
STOP (YES | NO)	Initial state of the server

The parameters are described in detail below.

NAME (*server-name***)**
Names the SDSF server to be included in the server group. The server name is a 1-8 character job name.

The server name must be unique on a system. If you code more than one SERVER statement with the same values for NAME and SYSNAME, the last statement is used.

SYSNAME (*system-name***)**
Names the system on which the SDSF server runs. The system must be in the same sysplex as systems for the other servers in the group.

JESNAME (*JES-subsystem-name***)**
Names the JES subsystem for which data is to be gathered.

MEMBER (*JES-member-name***)**
Names the member of the MAS for which data is to be gathered. The member must be in the same MAS as the members for the other servers in the group.

COMM (*COMM-statement-name*)
Names the COMM statement that describes the communication for this server group. If this is omitted, the communication uses default values.

STOP (YES | NO)
Specifies the initial state of the server when the server group is activated. STOP(YES) indicates that the server will initially be in the stopped state. You might want to use STOP(YES) when you:

- Include servers in the server group definition before the servers are actually available
- Include servers for JES2s that you don't ordinarily use. An initial state of stopped avoids the need to enter the operator STOP command as soon as the server comes up.

To start the server, you must issue the START command. See "Server operator commands" on page 118 for more information.

COMM statement

The COMM statement is optional. If it is omitted, default values for all parameters are used.

The following table shows the parameters that you code on a COMM statement.

Parameter	Description
COMM (*name*)	Name of the COMM statement, referenced by the COMM parameter of a SERVER statement
TYPE (MQS)	Communication type, WebSphere MQ
QMGR (*MQ-queue-manager-name*)	Name of the WebSphere MQ queue manager
QPREFIX (ISF) | (*MQ-prefix*)	WebSphere MQ queue prefix
CLUSTER(*cluster-name*) | CLUSNL(*namelist*)	WebSphere MQ clustering options
QREPLACE(YES|NO)	Setting for whether the SDSF server should replace existing definitions for the queues it creates
QDELETE(YES|NO)	Setting for whether queues created by the SDSF server will be deleted when the server is shut down
QDEFINE(YES | NO)	Setting for whether the SDSF server should issue WebSphere MQ DEFINE commands to define required WebSphere MQ objects during SDSF initialization

The parameters are described in detail below.

COMM (*name*)
Names the COMM statement, for reference by the COMM parameter of a SERVER statement. The name can be 1 to 8 alphabetic, numeric, or national characters (@, #, $) and must begin with an alphabetic character.

TYPE (MQS)
Names the type of communication between SDSF servers. If you specify this parameter, the value must be MQS, for WebSphere MQ. MQS is the default.

QMGR (*MQ-queue-manager*)
Names the WebSphere MQ queue manager on which the server request queue and client queue are defined. If you want the default queue manager for the system, either omit QMGR or specify QMGR with a value of at least one

blank, for example, `QMGR (' ')`. You can define a default queue manager as part of WebSphere MQ customization, using program CSQBDEFV.

For more information on the queues, see "WebSphere MQ" on page 306.

QPREFIX (ISF) | (*queue-prefix*)
WebSphere MQ queue prefix. The prefix is the high-level qualifier of the queue names that are used by SDSF. Because WebSphere MQ uses the queue names for security, changing the prefix may affect your security profiles. The default, if you omit the QPREFIX parameter, is ISF.

For more information on protecting the queues, see "WebSphere MQ" on page 306.

CLUSTER(*cluster-name*)|CLUSNL(*namelist*)
Names the WebSphere MQ cluster or cluster list. Specify either CLUSTER or CLUSNL if you have implemented clusters.

CLUSTER(*cluster-name*)
specifies the name of the WebSphere MQ cluster.

CLUSNL(*namelist*)
specifies the name of the list of WebSphere MQ clusters.

QREPLACE(YES|NO)
Defines whether the SDSF server will replace any existing WebSphere MQ objects that it creates, that is, the client request queue and the model queue. The default is NO, meaning that the server will use the existing definitions as is. YES specifies that the server will replace the WebSphere MQ objects. Use QREPLACE to alter a queue definition if it was created without WebSphere MQ clustering, and clustering has since been implemented.

Note that QREPLACE(NO) does not prevent the SDSF server from issuing WebSphere MQ DEFINE commands; to do that, use QDEFINE(NO).

QDEFINE(YES|NO)
Defines whether the SDSF server should issue WebSphere MQ DEFINE commands during SDSF initialization to define the required WebSphere MQ objects, that is, the client request queue and the model queue. The default is YES. If you specify QDEFINE(NO), you must have first defined the queues, either by issuing the DEFINE commands, or by previously starting the server with QDEFINE(YES). The DEFINE commands used to define the queues are logged in the server log. See "Logging" on page 109 for details. If QDEFINE(NO) is specified, and the SDSF server later requires an WebSphere MQ object that is not present, SDSF initialization will fail.

QDELETE(YES|NO)
Defines whether WebSphere MQ objects created by the SDSF server in this instance will be deleted when communications is ended. The default is NO. The affected WebSphere MQ objects are the client request queue and the model queue.

Summary of rules for defining a server group

1. A server may have only one server group. If a second SERVERGROUP statement is encountered, it replaces the first statement.
2. A server group must have at least two servers, one of which must be the local server.
3. A server group may include no more than 32 servers.
4. All servers in a server group must be in the sysplex.

5. All JES2s in the server group must be in the MAS.

Server connection (CONNECT)

The CONNECT statement defines the server connection, including whether the server is the default SDSF server and the XCF application server name. It can also request that XCF not be used to provide sysplex data. For more information, refer to "Using the server for sysplex data" on page 110.

CONNECT can be placed anywhere in the ISFPARMS statements.

You can use a CONNECT AUXPROC or AUXNAME of NONE to prevent the automatic start of SDSFAUX.

Example of the CONNECT statement

CONNECT Statement
`CONNECT DEFAULT(YES),` `  XCFSRVNM(SAME)`

This statement indicates that the server processing ISFPRM*xx* is the default server, replacing any other default server that might have been defined previously, and that the XCF application server name is derived based on the SDSF server name.

CONNECT statement

The following table shows the parameters that you code on a CONNECT statement.

Parameter	Description
AUXPROC(*SDSFAUX-procedure-name* | *NONE*)	Specifies the SDSFAUX procedure name or NONE to suppress starting of SDSFAUX.
AUXNAME(*SDSFAUX-jobname* | *NONE*)	Specifies the SDSFAUX job name or NONE to suppress starting of SDSFAUX.
AUXSAF(FAILRC4 | NOFAILRC4)	Specifies the action to be taken by SDSFAUX when a SAF authentication request results in a return code 04 (indeterminate response).
DEFAULT(YES|NO|COND)	Indicates whether the server is identified as the default server. Users are connected to the default server unless another server is specified, either with the SERVER keyword in the assembler ISFPARMS, or with the SERVER parameter on the SDSF command used to access SDSF.
XCFSRVNM(*server-name*|SAME|NONE)	Defines the XCF application server name, or requests that XCF should not be used to provide sysplex data

The parameters are described in detail below.

AUXPROC(*SDSFAUX-procedure-name*)|NONE

SDSFAUX-procedure-name|**NONE**
indicates the procedure name for starting SDSFAUX. The default is SDSFAUX. You can use an AUXPROC of NONE to prevent the automatic start of SDSFAUX.

If you were to specify AUXPROC(HSFSRJCL) and AUXNAME(SDSFAUX), then the start command would be S HSFSRJCL.SDSFAUX.

AUXNAME(*SDSFAUX-job-name*)|NONE

SDSFAUX-job-name|**NONE**
indicates the job name to use when starting the SDSFAUX address space. The default is SDSFAUX. You can use an AUXNAME of NONE to prevent the automatic start of SDSFAUX.

If you were to specify AUXPROC(HSFSRJCL) and AUXNAME(SDSFAUX), then the start command would be S HSFSRJCL.SDSFAUX.

AUXSAF(FAILRC4|NOFAILRC4)

FAILRC4
indicates that the requests should fail (not authorized). This is the default.

NOFAILRC4
indicates that the request should not fail (authorized).

DEFAULT(YES|NO|COND)

YES
indicates this server is to be made the default server unconditionally, replacing any other default server if necessary.

NO indicates the server is not to be made the default. If the server had previously been made the default, it will remain the default until it terminates. This is the case if the DEFAULT parameter is not specified.

COND
indicates that the server will be made the default unless another default server is already defined.

If you define a default server, you do not need to code the SERVER keyword in the assembler ISFPARMS.

There can be only one default server for a system at any given time.

XCFSRVNM(SAME|*server-name*|NONE)

SAME
indicates that the XCF application server name is derived from the SDSF server name. This is the default, and so is the case if the XCFSRVNM parameter is omitted.

When you use SAME, all SDSF servers that are to participate in sysplex requests must have the same name. (The server name is either the job name or the started task ID.)

server-name
specifies the customizable portion of the XCF application server name, ISFSRVR.*server-name*. *server-name* can be up to 8 characters, and can consist of alphabetic characters, numeric characters and the national characters @, #, or $.

When you use *server-name*, the names of the SDSF servers that are to participate in sysplex requests do not need to be the same.

NONE
indicates that the server should not identify itself to XCF and so will not respond to sysplex requests through XCF. A value of NONE for the local system (the system the user is logged on to) causes SDSF to attempt to

revert to using server groups and WebSphere MQ to provide sysplex-wide data. A value of NONE for a remote system requests that this remote system not be included in the sysplex-wide data.

Group authorization parameters (GROUP or ISFGRP)

A GROUP statement or ISFGRP macro defines:

- The members of a group of users
- Which functions the members of the group may perform
- Customization values, such as columns on SDSF panels, and date format

Group membership

You can define membership in the groups in ISFPARMS with:

- SAF. This is required for the JES3 environment. For the JES2 environment, it is optional, but recommended, as it is dynamic and allows you to assign users to the same group regardless of the environment from which they invoke SDSF (interactive, batch, REXX or Java).
- Parameters on the GROUP statements or ISFGRP macros. You define who belongs to the group on the basis of procedure name, terminal name, user ID, and TSO authority. This applies only to the JES2 environment. SDSF does not use ISFPARMS statements for group membership in the JES3 environment.

  SDSF scans ISFPARMS from the beginning and assigns users to the first group for which they are qualified. This means that the order of the group definitions is important: Arrange them from most selective to least selective.

  Users can display the name of the group to which they belong with the WHO command.

A user must be assigned to a group in order to use SDSF. When a user tries to access SDSF but is not assigned to any group, SDSF issues message ISF024I.

Using SAF to control group membership

When using SAF to define who belongs to an ISFPARMS group, you:

1. Assign a name to each group, as follows:
   - With a GROUP statement, using the NAME parameter.
   - With an ISFGRP macro, using the macro label. The label must start in column 1 and be 1-8 characters. It must conform to standard assembler language programming conventions and be unique within ISFPARMS.
2. Define SAF profiles GROUP.*group-name.server-name*, in the SDSF class, and permit users to them as appropriate. For more information, see “Membership in groups” on page 252.

SDSF works through the groups in ISFPARMS, checking for READ access to the SAF resource GROUP.*group-name.server-name* in the SDSF class. (If the SDSF client is not connected to the SDSF server, *server-name* is blank.) If the user is authorized to the group through the SAF profile, then the user is assigned to the group, regardless of whether he may be authorized to groups that occur later in ISFPARMS. If the user is not authorized to the group through the SAF profile, SDSF goes on to the next group.

In a JES2 environment, if SAF cannot make a decision because the SDSF class is inactive or the profile is not defined, SDSF reverts to ISFPARMS to determine membership in the group. In a JES3 environment, SAF fails the request.

If you do not assign a name to a group, SDSF generates one: ISF plus the index value of the group, in the format ISF*nnnnn*. However, because this name will change when you add or subtract groups from ISFPARMS, it is not suitable for use with SAF. To avoid conflicts with the SDSF-generated names, you should *not* assign names in the format ISF*nnnnn*.

The ISFPARMS and statements shipped with SDSF use the following names:
- ISFSPROG for group 1
- ISFOPER for group 2
- ISFUSER for group 3

If you do not want SAF checking to occur, you can write a user exit using the pre-SAF exit point. See Chapter 9, “Using installation exit routines,” on page 329.

Group function

The group function parameters can be used to determine which functions the members of a group can perform.

Some of these parameters have equivalents in SAF. For more information, see Chapter 5, “Using SAF for security,” on page 191 and Appendix B, “SAF equivalents for ISFPARMS,” on page 561.

Examples of the GROUP statement and ISFGRP macro

GROUP Statement	ISFGRP Macro
1 GROUP IUID(LOGAUTH), 2 PREFIX(USERID), 3 AUTH(ALLUSER) 4 NTBL NAME(LOGAUTH) NTBLENT STRING(ASR),OFFSET(1) NTBLENT STRING(RND),OFFSET(1)	1 ISFGRP IUID=LOGAUTH, 2 PREFIX=USERID, 3 AUTH=(ALLUSER) 4 LOGAUTH ISFNTBL ASR,1,RND,1

On line 1 in the example, the IUID parameter works with an ISFNTBL macro or NTBL statement labeled LOGAUTH, on the line marked by 4. The result is to include in the group any user whose ID contains the character string *ASR* beginning in the first position or the character string *RND* beginning in the first position.

On line 2, the PREFIX parameter limits the jobs displayed on the DA, I, O, H, and ST panels to those jobs whose job names begin with the group member's user ID.

On line 3, the AUTH parameter identifies the SDSF panels that members of this user group are allowed to display, and the SDSF commands that they are allowed to issue. In this case, they are authorized to all “end user” commands, that is, I, O, H, DA, ST and SE.

GROUP Statement	ISFGRP Macro
1 GROUP TSOAUTH(JCL,OPER,ACCT), 2 XUID(XLIST), 3 AUTH(LOG,I,O,H,DA,INIT,PR,NO,DEST) 4 GROUP TSOAUTH(JCL), 5 IFIELDS(DFLD) 6 NTBL NAME(XLIST) NTBLENT STRING($S),OFFSET(1) NTBLENT STRING(OPER),OFFSET(3) 7 FLD NAME(DFLD) TYPE(IN) FLDENT COLUMN(JNUM),TITLE('JOB NUM'),WIDTH(7) FLDENT COLUMN(JPRIO),TITLE(PRTY),WIDTH(4)	1 ISFGRP TSOAUTH=(JCL,OPER,ACCT), 2 XUID=XLIST, 3 AUTH=(LOG,I,O,H,DA,INIT,PR,NO,DEST)) 4 ISFGRP TSOAUTH=(JCL), 5 IFIELDS=DFLD 6 XLIST ISFNTBL $S,1,OPER,3 7 DFLD ISFFLD JNUM,'JOB NUM',7, JPRIO,PRTY,4,TYPE=IN

On line 1 in the example, TSOAUTH defines a group of users with a TSO authority of JCL, OPER, and ACCT.

On line 2, the XUID parameter is also a group membership parameter. It works with an ISFNTBL macro or NTBL statement on the line marked by 6 to exclude from the group of users defined by the TSOAUTH parameter any user whose ID contains the character string *$S* beginning in the first position or the character string *OPER* beginning in the third position.

On line 3, the AUTH parameter identifies the SDSF commands this user group is allowed to issue.

On line 4, the second group definition begins. TSOAUTH defines a group of users based on TSO authority.

On line 5, the IFIELDS parameter works with an ISFFLD macro or FLD statement beginning on the line marked by 7 to define a variable field list on the Input Queue panel for the group. The list contains the column JNUM, with the title *JOB NUM,* with a width of seven characters, and the column JPRIO, with the title *PRTY,* with a width of four characters. TYPE=IN indicates that the field list is for the Input Queue panel.

GROUP Statement	ISFGRP Macro
1 GROUP NAME(ISFUSER) AUTH(DA,I,O,H,ST,DEST,PREF), 2 PREFIX(USERID)	1 ISFUSER ISFGRP AUTH=(DA,I,O,H,ST,DEST,PREF), 2 PREFIX=USERID

On line 1 in the example, the group is given a name, ISFUSER. The name is assigned through the label on the ISFGRP macro and through the NAME parameter on the GROUP statement. All members of this group are authorized to a SAF resource in the format GROUP.*group-name.server-name*. In this case, the SAF resource is GROUP.ISFUSER.*server-name*. With RACF®, to authorize users to this group, for a server named SDSF, you place these users in the access list for the profile GROUP.ISFUSER.SDSF.

On line 2, the PREFIX parameter specifies that the users will only see jobs whose names begin with their user IDs.

For more examples, see samples ISFPRM00 and ISFPRM01 in ISF.SISFJCL.

Group membership parameters reference

The group membership parameters are not used in the JES3 environment. For JES3, see "Using SAF to control group membership" on page 36.

GROUP	ISFGRP	Description
NAME (*group-name*)	*macro label*	Group name, used in SAF resource
ILPROC (*NTBL-name*)	ILPROC=*ISFNTBL-label*	Includes users by logon procedure. See note below.
XLPROC (*NTBL-name*)	XLPROC=*ISFNTBL-label*	Excludes users by logon procedure. See note below.
ITNAME (*NTBL-name*)	ITNAME=*ISFNTBL-label*	Includes users by terminal name. See note below.
XTNAME (*NTBL-name*)	XTNAME=*ISFNTBL-label*	Excludes users by terminal name. See note below.
IUID (*NTBL-name*)	IUID=*ISFNTBL-label*	Includes users by user ID. See note below.
XUID (*NTBL-name*)	XUID=*ISFNTBL-label*	Excludes users by user ID. See note below.
TSOAUTH (*attributes*)	TSOAUTH=*attributes*	Includes users by TSO authority

Note:

The ILPROC, ITNAME, and IUID parameters *include* members. If you use more than one of these to define a group, a user must meet the requirements of all of them in order to qualify for inclusion in the group. For instance, if you specify IUID=Y and ILPROC=Z, a user must have both of these attributes (Y and Z) to be included in the group. If none of the parameters is specified on an ISFGRP macro, all users will qualify for that group.

The XLPROC, XTNAME, and XUID parameters *exclude* members, and they override the parameters that include members. For instance, if a user qualifies for a group based on the IUID parameter, but is excluded from the group based on the XLPROC parameter, the user is excluded.

The values for logon proc, user ID and terminal name vary with how users invoke SDSF (interactively, batch, REXX or Java). For more information, see "Security and SDSF in batch" on page 375, "Security and REXX" on page 448 and "Security and Java" on page 460.

GROUP	ISFGRP
NAME (*group-name*)	*macro label*

Names the group. Assign a name to a group when you want to use SAF to control membership in the group. SDSF checks authorization to the resource GROUP.*group-name.server-name*.

You might also name groups to make them easier to find and identify.

The *group-name* must be 1-8 alphanumeric characters, beginning with an alphabetic character.

GROUP	ISFGRP
ILPROC (*NTBL-name*)	ILPROC=*ISFNTBL-macro-label*

Specifies that a user whose **logon procedure name** is in the list created by the specified ISFNTBL macro or NTBL statement is included in this group. If this parameter is omitted, logon procedure names are not used to determine inclusion in this group.

GROUP	ISFGRP
XLPROC (*NTBL-name*)	XLPROC=*ISFNTBL-macro-label*

Specifies that a user whose logon procedure name is in the list created by the specified NTBL macro or NTBL statement is excluded from this group. If this parameter is omitted, logon procedure names are not used to determine exclusion from this group.

GROUP	ISFGRP
ITNAME (*NTBL-name*)	ITNAME=*ISFNTBL-macro-label*

Specifies that a user whose **terminal name** is in the list created by the specified ISFNTBL macro or NTBL statement is included in this group. If this parameter is omitted, terminal names are not used to determine inclusion in this group.

GROUP	ISFGRP
XTNAME (*NTBL-name*)	XTNAME=*ISFNTBL-macro-label*

Specifies that a user whose terminal name is in the list created by the specified SFNTBL macro or NTBL statement is excluded from this group. If this parameter is omitted, terminal names are not used to determine exclusion from this group.

GROUP	ISFGRP
IUID (*NTBL-name*)	IUID=*ISFNTBL-macro-label*

Indicates that a user whose **user ID** is in the list created by the specified ISFNTBL macro or NTBL statement is included in this group. If this parameter is omitted, user IDs are not used to determine inclusion in this group.

GROUP	ISFGRP
XUID (*NTBL-name*)	XUID=*ISFNTBL-macro-label*

Indicates that a user whose user ID is in the list created by the specified ISFNTBL macro or NTBL statement is excluded from this group. If this parameter is omitted, user IDs are not used to determine exclusion from this group.

GROUP	ISFGRP
TSOAUTH (*TSO-authority-list*)	TSOAUTH=(*TSO-authority-list*)

Indicates that a user with *all* of the **TSO authorities** listed (ACCT, OPER, JCL, MOUNT) is included in this group. If the list contains more than one TSO authority, separate them with a comma. The TSO authorities work together in a logical "AND" process. That is, if you specify more than one TSO authority, a user must have all those specified to become eligible for inclusion in the group. If this parameter is omitted, TSO authorities are not used to determine inclusion in this group.

Group function parameters reference

All parameters apply in the JES2 environment; those parameters that apply in the JES3 environment are indicated in the table. Parameters that do not apply in the JES3 environment are primarily parameters that:

- Provide security. You must use SAF for security in the JES3 environment.

- Relate to SDSF's sysplex support that uses the SDSF server and WebSphere MQ. This does not apply in the JES3 environment.
- Define field lists for panels that are not available in the JES3 environment.

In a JES3 environment, parameters that do not apply are ignored.

GROUP	ISFGRP	Description
ACTION (NONE) | (ALL) | (*routing-code-list*)	ACTION=NONE | ALL | (*routing-code-list*)	Display of outstanding WTORs in LOG
ACTIONBAR (YES) | (NO)	ACTIONBAR=YES | NO	Display of the action bar
APFFLDS (*FLD-name*)	APFFLDS=*ISFFLD-label*	Primary field list for APF
APFFLD2 (*FLD-name*)	APFFLD2=*ISFFLD-label*	Alternate field list for APF
APPC (ON) | (OFF)	APPC=ON | OFF	Display of APPC transaction sysout (JES2 only)
ASFLDS (*FLD-name*)	ASFLDS=*ISFFLD-label*	Primary field list for AS
ASFLD2 (*FLD-name*)	ASFLD2=*ISFFLD-label*	Alternate field list for AS
AUPDT (2) | (*interval*)	AUPDT=2 | *interval*	Minimum auto update interval
AUTH (*command-list*)	AUTH=(*command-list*)	SDSF commands (JES2 only)
BROWSE (S | SB | SE | NONE)	BROWSE=S | SB | SE | NONE	Default browse action character
CKFLDS (*FLD-name*)	CKFLDS=*ISFFLD-label*	Primary field list for CK
CKFLD2 (*FLD-name*)	CKFLD2=*ISFFLD-label*	Alternate field list for CK
CKHFLDS (*FLD-name*)	CKHFLDS=*ISFFLD-label*	Primary field list for CKH
CKHFLD2 (*FLD-name*)	CKHFLD2=*ISFFLD-label*	Alternate field list for CKH
CMDAUTH (*auth-list*)	CMDAUTH=(*auth-list*)	Action characters, overtypes, / commands
CMDLEV (0) | (*level*)	CMDLEV=0 | *level*	Command authorization level (JES2 only)
CONFIRM (ON) | (OFF) | (ALWAYS)	CONFIRM=ON | OFF | ALWAYS	Confirmation of action characters
CPUFMT(LONG) | (SHORT)	CPUFMT=LONG | SHORT	Format of CPU on DA title line
CTITLE (ASIS) | (UPPER)	CTITLE=ASIS | UPPER	Case of text, such as column titles
CURSOR (ON) | (OFF) | TOP	CURSOR=ON | OFF | TOP	Cursor placement
CUSTOM(*proplist-name*)	Not supported	Customization of properties
DADFLT (*types-and-pos*)	DADFLT=(*types-and-pos*)	Types of jobs on DA
DATE (MMDDYYYY) | (DDMMYYYY) | (YYYYMMDD)	DATE=MMDDYYYY | DDMMYYYY | YYYYMMDD	Date format
DATESEP (/) | (-) | (.)	DATESEP=/ | - | .	Date separator
DEST (*NTBL-name*)	DEST=*ISFNTBL-label*	Destinations
DFIELDS (*FLD-name*)	DFIELDS=*ISFFLD-label*	Primary field list for DA
DFIELD2 (*FLD-name*)	DFIELD2=*ISFFLD-label*	Alternate field list for DA
DISPLAY (OFF) | (ON)	DISPLAY=OFF | ON	Display of current values
DSPAUTH (*auth-list*)	DSPAUTH=(*auth-list*)	Types of jobs the group can browse
DYNXFLDS (*FLD-name*)	DYNXFLDS=*ISFFLD-label*	Primary field list for DYNX
DYNXFLD2 (*FLD-name*)	DYNXFLD2=*ISFFLD-label*	Alternate field list for DYNX
EMCSAUTH (MASTER | ALL)	EMCSAUTH=MASTER | ALL	Authority used with the EMCS console

GROUP	ISFGRP	Description
EMCSREQ (YES | NO)	EMCSREQ=YES | NO	EMCS required for system commands
ENCFLDS (*FLD-name*)	ENCFLDS=*ISFFLD-label*	Primary field list for ENC
ENCFLD2 (*FLD-name*)	ENCFLD2=*ISFFLD-label*	Alternate field list for ENC
ENQFLDS (*FLD-name*)	ENQFLDS=*ISFFLD-label*	Primary field list for ENQ
ENQFLD2 (*FLD-name*)	ENQFLD2=*ISFFLD-label*	Alternate field list for ENQ
GPLEN (*prefix-length*)	GPLEN=*prefix-length*	Length of the group prefix
GPREF (*group-prefix*)	GPREF=*group-prefix*	Group prefix string
HFIELDS (*FLD-name*)	HFIELDS=*ISFFLD-label*	Primary field list for H
HFIELD2 (*FLD-name*)	HFIELD2=*ISFFLD-label*	Alternate field list for H
ICMD (*NTBL-name*)	ICMD=*ISFNTBL-label*	Jobs to be included with CMDAUTH
IDEST (*NTBL-name*)	IDEST=*ISFNTBL-label*	Initial list of destinations
IDSP (*NTBL-name*)	IDSP=*ISFNTBL-label*	Jobs to be included with DSPAUTH
IDSPD (*NTBL-name*)	IDSPD=*ISFNTBL-statement*	Jobs for which messages can be displayed
IFIELDS (*FLD-name*)	IFIELDS=*ISFFLD-label*	Primary field list for I
IFIELD2 (*FLD-name*)	IFIELD2=*ISFFLD-label*	Alternate field list for I
ILOGCOL (1) | (*position*)	ILOGCOL=1 | *position*	Starting column for LOG
INPUT (OFF) | (ON)	INPUT=OFF | ON	SYSIN data sets shown with browse
INTFLDS (*FLD-name*)	INTFLDS=*ISFFLD-label*	Primary field list for INIT
INTFLD2 (*FLD-name*)	INTFLD2=*ISFFLD-label*	Alternate field list for INIT
ISTATUS (*NTBL-name*)	ISTATUS=*ISFNTBL-name*	Jobs included on DA, H, I, O, PS and ST
ISYS (LOCAL) | (NONE)	ISYS=LOCAL | NONE	Systems shown on sysplex panels
JCFLDS (*FLD-name*)	JCFLDS=*ISFFLD-label*	Primary field list for JC
JCFLD2 (*FLD-name*)	JCFLD2=*ISFFLD-label*	Alternate field list for JC
JDSFLDS (*FLD-name*)	JDSFLDS=*ISFFLD-label*	Primary field list for JDS
JDSFLD2 (*FLD-name*)	JDSFLD2=*ISFFLD-label*	Alternate field list for JDS
J0FLDS (*FLD-name*)	J0FLDS=*ISFFLD-label*	Primary field list for J0 (JES3 only)
J0FLD2 (*FLD-name*)	J0FLD2=*ISFFLD-label*	Alternate field list for J0 (JES3 only)
LANG (ENGLISH) | (ENG) | (JAPANESE) | (JPN)	LANG=ENGLISH | ENG | JAPANESE | JPN	Default language
LINEFLDS (*FLD-name*)	LINEFLDS=*ISFFLD-name*	Primary field list for LI
LINEFLD2 (*FLD-name*)	LINEFLD2=*ISFFLD-name*	Alternate field list for LI
LNKFLDS (*FLD-name*)	LNKFLDS=*ISFFLD-name*	Primary field list for LNK
LNKFLD2 (*FLD-name*)	LNKFLD2=*ISFFLD-name*	Alternate field list for LNK
LOG (OPERACT) | (OPERLOG) | (SYSLOG)	LOG=OPERACT | OPERLOG | SYSLOG	Default Log panel
LPAFLDS (*FLD-name*)	LPAFLDS=*ISFFLD-name*	Primary field list for LPA
LPAFLD2 (*FLD-name*)	LPAFLD2=*ISFFLD-name*	Alternate field list for LPA
MASFLDS (*FLD-name*)	MASFLDS=*ISFFLD-name*	Primary field list for MAS and JP
MASFLD2 (*FLD-name*)	MASFLD2=*ISFFLD-name*	Alternate field list for MAS and JP
NCFLDS(*FLD-name*)	NCFLDS=*ISFFLD-name*	Primary field list for NC

GROUP	ISFGRP	Description
NCFLD2(*FLD-name*)	NCFLD2S=*ISFFLD-name*	Alternate field list for NC
NODEFLDS (*FLD-name*)	NODEFLDS=*ISFFLD-name*	Primary field list for NO
NSFLDS (*FLD-name*)	NSFLDS=*ISFFLD-name*	Primary field list for NS
NSFLD2 (*FLD-name*)	NSFLD2=*ISFFLD-name*	Alternate field list for NS
NODEFLD2 (*FLD-name*)	NODEFLD2=*ISFFLD-name*	Alternate field list for NO
ODFLDS (*FLD-name*)	ODFLDS=*ISFFLD-name*	Primary field list for OD
ODFLD2 (*FLD-name*)	ODFLD2=*ISFFLD-name*	Alternate field list for OD
OFIELDS (*FLD-name*)	OFIELDS=*ISFFLD-name*	Primary field list for O
OFIELD2 (*FLD-name*)	OFIELD2=*ISFFLD-name*	Alternate field list for O
OWNER (NONE) | (USERID)	OWNER=NONE | USERID	Default for OWNER
PAGFLDS (*FLD-name*)	PAGFLDS=*ISFFLD-name*	Primary field list for PAG
PAGFLD2 (*FLD-name*)	PAGFLD2=*ISFFLD-name*	Alternate field list for PAG
PARMFLDS (*FLD-name*)	PARMFLDS=*ISFFLD-name*	Primary field list for PARM
PARMFLD2 (*FLD-name*)	PARMFLD2=*ISFFLD-name*	Alternate field list for PARM
PREFIX (NONE) | (USERID) | (GROUP)	PREFIX=NONE | USERID | GROUP	Default for PREFIX
PRTFLDS (*FLD-name*)	PRTFLDS=*ISFFLD-label*	Primary field list for PR
PRTFLD2 (*FLD-name*)	PRTFLD2=*ISFFLD-label*	Alternate field list for PR
PSFLDS (*FLD-name*)	PSFLDS=*ISFFLD-label*	Primary field list for PS
PSFLD2 (*FLD-name*)	PSFLD2=*ISFFLD-label*	Alternate field list for PS
PUNFLDS (*FLD-name*)	PUNFLDS=*ISFFLD-label*	Primary field list for PUN
PUNFLD2 (*FLD-name*)	PUNFLD2=*ISFFLD-label*	Alternate field list for PUN
RDRFLDS (*FLD-name*)	RDRFLDS=*ISFFLD-label*	Primary field list for RDR
RDRFLD2 (*FLD-name*)	RDRFLD2=*ISFFLD-label*	Alternate field list for RDR
RESFLDS (*FLD-name*)	RESFLDS=*ISFFLD-label*	Primary field list for RES
RESFLD2 (*FLD-name*)	RESFLD2=*ISFFLD-label*	Alternate field list for RES
RMFLDS (*FLD-name*)	RMFLDS=*ISFFLD-label*	Primary field list for RM (JES2 only)
RMFLD2 (*FLD-name*)	RMFLD2=*ISFFLD-label*	Alternate field list for RM (JES2 only)
RSYS (LOCAL | NONE)	RSYS=LOCAL | NONE	WTORs shown on Log
SEFLDS (*FLD-name*)	SEFLDS=*ISFFLD-label*	Primary field list for SE
SEFLD2 (*FLD-name*)	SEFLD2=*ISFFLD-label*	Alternate field list for SE
SOFLDS (*FLD-name*)	SOFLDS=*ISFFLD-label*	Primary field list for SO (JES2 only)
SOFLD2 (*FLD-name*)	SOFLD2=*ISFFLD-label*	Alternate field list for SO (JES2 only)
SPFLDS (*FLD-name*)	SPFLDS=*ISFFLD-label*	Primary field list for SP
SPFLD2 (*FLD-name*)	SPFLD2=*ISFFLD-label*	Alternate field list for SP
SRCHFLDS (*FLD-name*)	SRCHFLDS=*ISFFLD-name*	Primary field list for SRCH
SRCHFLD2 (*FLD-name*)	SRCHFLD2=*ISFFLD-name*	Alternate field list for SRCH
SRFLDS (*FLD-name*)	SRFLDS=*ISFFLD-label*	Primary field list for SR
SRFLD2 (*FLD-name*)	SRFLD2=*ISFFLD-label*	Alternate field list for SR
STFLDS (*FLD-name*)	STFLDS=*ISFFLD-label*	Primary field list for ST
STFLD2 (*FLD-name*)	STFLD2=*ISFFLD-label*	Alternate field list for ST

GROUP	ISFGRP	Description
SYMFLDS (*FLD-name*)	SYMFLDS=*ISFFLD-label*	Primary field list for SYM
SYMFLD2 (*FLD-name*)	SYMFLD2=*ISFFLD-label*	Alternate field list for SYM
SYSFLDS (*FLD-name*)	SYSFLDS=*ISFFLD-name*	Primary field list for SYS
SYSFLD2 (*FLD-name*)	SYSFLD2=*ISFFLD-name*	Alternate field list for SYS
SYSID (*system-id*)	SYSID=*system-id*	System ID for LOG in a JES2 environment (JES2 only)
SYSID3 (*system-id*)	Not supported	System ID for LOG in a JES3 environment
UPCTAB (TRTAB2) | (*TRTAB-name*)	UPCTAB=TRTAB2 | *TRTAB-name*	Upper case translation table
VALTAB (TRTAB) | (*TRTAB-name*)	VALTAB=TRTAB | *TRTAB-name*	Valid character translation table
VIO (SYSALLDA) | (*unit-name*)	VIO=SYSALLDA | *unit-name*	VIO unit name for viewing page-mode output
XCMD (*NTBL-name*)	XCMD=*ISFNTBL-label*	Jobs to be excluded when processing CMDAUTH
XDSP (*NTBL-name*)	XDSP=*ISFNTBL-label*	Jobs to be excluded when processing DSPAUTH
XDSPD (*NTBL-name*)	XDSPD=*ISFNTBL-label*	Jobs to be excluded for which messages can be displayed
XSTATUS (*NTBL-name*)	XSTATUS=*ISFNTBL-label*	Jobs excluded from DA, H, I, O, PS and ST

GROUP	ISFGRP
ACTION (NONE) | (ALL) | (*routing-code-list*)	ACTION=NONE | ALL | (*routing-code-list*)

Specifies routing codes that determine which write-to-operator-with-reply (WTOR) messages should be displayed at the bottom of the SYSLOG panel for members of this group.

ALL
specifies that WTOR messages for MCS routing codes 1 through 28 are to be displayed.

NONE
specifies that no WTOR messages are to be displayed. This is the default.

(*routing-code-list***)**
specifies that WTOR messages for specific routing codes are to be displayed. If you specify more than one option in your routing code list, enclose the list in parentheses and separate each option with a comma. The list can be made up of one or more of the following options:

- One or more decimal routing codes. The possible routing codes are 1 through 28.
- **MVS**, which enables the 12 routing codes used by MVS-JES. The routing codes used by MVS-JES are 1 through 12.
- **USER**, which enables the routing codes reserved for customer use. The routing codes reserved for customer use are 13 through 28.
- **ALL** or **NONE**, if you are using statements. ALL and NONE are described above. If included in the list, they are added to other items in the list.

The setting of the ACTION parameter can be changed by an authorized user through the use of the ACTION command. (See the AUTH parameter.)

GROUP	ISFGRP
ACTIONBAR (YES) | (NO)	ACTIONBAR=YES | NO

Sets an initial value for the display of the action bar.

YES
indicates that the action bar is displayed.

NO indicates that the action bar is not displayed.

If the ACTIONBAR parameter is omitted, the initial setting is to display the action bar.

Users can override the ACTIONBAR setting with the Set Screen Characteristics pop-up.

GROUP	ISFGRP
APFFLDS (*FLD-statement-name*)	APFFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the APF panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
APFFLD2 (*FLD-statement-name*)	APFFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **Alternate** variable field list for the APF panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
APPC (ON) | (OFF)	APPC=ON | OFF

Controls whether a group member will see APPC transactions on the H and O panels. (Applies to JES2 only.)

ON indicates that APPC transactions are displayed.

OFF
indicates that APPC transactions are not displayed.

If the APPC parameter is omitted, APPC transactions are displayed. Users can override the APPC setting with the APPC command or pull-down choice.

GROUP	ISFGRP
ASFLDS (*FLD-statement-name*)	ASFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the AS panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
ASFLD2 (*FLD-statement-name*)	ASFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **Alternate** variable field list for the AS panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
AUPDT (2) | (*interval*)	AUPDT=2 | *interval*

Specifies the minimum automatic update interval, in seconds, that can be specified by members of this group. *interval* is a number from 0 to 255. The default is 2. A value of 0 indicates that the members of this group are not allowed to use the automatic update facility.

GROUP	ISFGRP
AUTH (*authorized-command-list*)	AUTH=(*authorized-command-list*)

Indicates which SDSF commands a member of the group is authorized to use. (Applies to JES2 only.) The values that can be included in *authorized-command-list* are:

- **ALL**, for all SDSF commands.
- **ALLOPER**, for all "operator" commands. The list of operator commands is the same as that for **ALL**, except for the omission of **ABEND**, **INPUT** and **TRACE**.
- **ALLUSER**, for all "end user" commands. The end user commands are **DA**, **H**, **I**, **O**, **ST**, **SYM**, and **SE**.
- Any SDSF command that requires authorization, which is: **ABEND**, **ACTION**, **APF**, **AS**, **CK**, **DA**, **DEST**, **DYNX**, **ENC**, **ENQ**, **FINDLIM**, **H**, **I**, **INIT**, **INPUT**, **JC**, **LI**, **LNK**, **LOG**, **LPA**, **MAS**, **NC**, **NO**, **NS**, **O**, **PAG**, **PARM**, **PR**, **PREF**, **PS**, **PUN**, **RDR**, **RES**, **RM**, **RSYS**, **SE**, **SO**, **SP**, **SR**, **ST**, **SYM**, **SYS**, **SYSID**, **SYSNAME**, **TRACE**, and **ULOG**

You can combine any value with any other value or values. If the list contains more than one item, separate the items with a comma. Using ALL, ALLOPER or ALLUSER rather than a list of individual commands can eliminate the need to update the AUTH parameter when new authorized commands are added to SDSF.

For information about further controlling the use of a panel accessed by one of the commands listed above, see the descriptions of the PREFIX, DEST, and OWNER commands, and the description of the XSTATUS parameter.

GROUP	ISFGRP
BROWSE (S | SB | SE | NONE)	BROWSE=S | SB | SE | NONE

Specifies the default browse action character, which is invoked when a user selects a row on a panel by placing the cursor in the NP column and pressing Enter. This applies to all panels that support browse.

S is SDSF browse.

SB is ISPF browse.

SE is ISPF edit.

NONE
specifies that there should be no default browse action character. This is also the case if this parameter is omitted.

GROUP	ISFGRP
CKFLDS (*FLD-statement-name*)	CKFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the CK panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
CKFLD2 (*FLD-statement-name*)	CKFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the CK panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
CKHFLDS (*FLD-statement-name*)	CKHFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the CKH panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
CKHFLD2 (*FLD-statement-name*)	CKHFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the CKH panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
CMDAUTH (*authorization-list*)	CMDAUTH=(*authorization-list*)

For JES2, indicates the jobs, TSO user IDs, started tasks, initiators, printers and punches for which members of this group can:
- Issue action characters and overtype fields
- Issue MVS and JES system commands from the command line with the SDSF / command

For either JES2 or JES3, you can request that a WTO be issued for attempts to issue unauthorized system commands. (See the MSG option.)

To use the / command, users must have authority to issue commands for all jobs. To give this authority, use either the ALL value described below or a SAF equivalent. Users also need authority to the console that SDSF uses to issue the command. See “Issuing MVS and JES commands” on page 342 for more information. To see command responses on their terminals, users must also be

authorized to the ULOG command, which is controlled with the AUTH parameter or the ISFCMD.ODSP.ULOG.*jesx* SAF resource in the SDSF class.

To allow members of a group to control printers from the Printer panel, or punches from the Punch panel, when relying on ISFPARMS for user authorization, you must include CMDAUTH values of ALL or DEST in the group's *command-authorization-list* and must give sufficient command level authority with the CMDLEV parameter. You can authorize more flexible control of printers or punches by using a SAF security scheme.

The CMDAUTH parameter works with the CMDLEV, ICMD, and XCMD parameters. To specify CMDAUTH using the SAF security scheme, see Appendix B, "SAF equivalents for ISFPARMS," on page 561.

(*authorization-list*) specifies CMDAUTH values. If the list contains more than one value, the values must be separated by a comma.

ALL (JES2 only)
authorizes a group member to issue commands for all the jobs, TSO user IDs, started tasks, initiators, and printers authorized by the CMDAUTH values of DEST, DISPLAY, GROUP, INIT, NOTIFY, and USERID. The ALL value does not control write-to-operator (WTO) messages; you must specify MSG to have WTO messages issued.

Note: Specifying ALL for CMDAUTH is the only way through ISFPARMS to authorize a user to issue MVS and JES2 commands by use of the / command. However, you can give this authority with a SAF equivalent.

When you specify ALL for CMDAUTH, you can use the CMDLEV parameter to limit the MVS and JES2 commands that a user is authorized to issue with the / command, as shown below.

CMDLEV	Authorized Commands when CMDAUTH is ALL	MVS/JES2 Commands Authorized for Use with the / Command
0	This is the default.	None authorized
1	Displays information using display and list commands.	$D, $L, D
2	Cancels or purges a job, started task, or TSO user. Releases held output.	$C, $O, $P, C
3	Releases or holds a job.	$A, $H, E
4	JES2 SET command. Restarts a job, or the MVS RESET command.	$E, $T, $TO
5	Routes job output.	$R
6	JES2 printer control commands or any JES2 command.	$B, $F, $I, $S, $Z
7	Any MVS command, the command to stop or withdraw JES2 from the system ($PJES2), or JES2 commands that send commands to other MAS members or nodes.	All MVS/JES2 commands

You should control use of the / command as you would the master console.

DEST (JES2 only)
allows a group member to issue action characters and use overtypeable fields for any job, printer or punch whose destination matches the value specified in the ISFNTBL macro or NTBL statement of the DEST parameter.

When destination names are not protected by a SAF security scheme, SDSF displays only those jobs, printers and punches having destination names specified with the DEST and IDEST parameters.

For more information, see "Destination names" on page 239.

Note:

1. The DEST function does not affect the DA panel.
2. For jobs, the destination is the user-defined name for the JES2 route code and is defined on the *DESTID* statement in the JES2 parameters. It can also be the route code in the form of PRT*nnnn*, PUN*nn*, RMT*nnnn*, U*nnnn*, R*nnnn*, and LOCAL. For printers and punches, *destid* is the internal route code name in the form U*nnnn* and R*nnnn*. U is used for a local printer defined by PRT*nnnn*, or a local punch defined by PUN*nn*. R is used for a remote printer or punch defined by RMT*nnnn* statements in the JES2 parameters.

   The destination name coded on the ISFNTBL macro or NTBL statement for the DEST parameter must match the DEST field on the panel, for all panels except the JDS panel. For the JDS panel, the DEST value that SDSF uses for authority checking is the DEST value shown on the panel from which the JDS panel was invoked, such as ST, O, or I.
3. Commands affecting printers and punches have further restrictions on them. One type of command authority is based on the destination name of the printer or punch. If you specify DEST for the CMDAUTH parameter and define a list of destination names with the DEST parameter, the user can only issue commands to the printers or punches that are in the list pointed to by the DEST parameter. For example, if the list contains PRT22 as a valid destination name, then any command of form $xPRT22,yyy would be allowed. If the list contains RMT22 as a valid destination name, then any command of the form $xR22,yyy would be allowed (where x is any command and yyy are any operands).

DISPLAY (JES2 only)
allows a member to issue the D (Display) and L (List) action characters for any job, regardless of CMDLEV. These action characters cause the $D (Display) and $L (List) commands to be generated.

GROUP (JES2 only)
allows a member to issue commands for any job whose job name begins with the group's prefix. (See the GPREF and GPLEN parameters below.) If NOTIFY is also specified, a member can issue commands for jobs whose NOTIFY matches the group prefix.

INIT (JES2 only)
authorizes the user to control the system initiators from the Initiator panel.

NOTIFY (JES2 only)
allows a member to issue commands for any job for which the NOTIFY parameter of the job card contains the member's user ID. If GROUP is also specified, a member can issue commands for jobs whose NOTIFY matches the group prefix.

MSG
issues a security WTO message whenever a member of this group issues a command (the WTO message is always issued when an SDSF user attempts to issue a system command for which the user is not authorized). The WTO message is also issued for all SSI requests.

USERID (JES2 only)
allows a member to issue commands for any job whose job name begins with the member's user ID.

GROUP	ISFGRP
CMDLEV (0) | (*level*)	CMDLEV=0 | *level*

Specifies command level to which a group member is authorized. Use a value of 0 through 7 for *level*. (Applies to JES2 only.)

The command level determines the action characters that a group member can issue, the fields that a group member can overtype, and the MVS and JES2 commands that a group member can issue from the command line. The CMDLEV parameter works with the CMDAUTH, ICMD, and XCMD parameters.

Each command level is inclusive of all those with a lower number. For example, a user with a command level of 3 can perform the functions requiring a command level of 3, 2, 1, and 0.

For a complete list of the action characters and overtypeable fields for each command level, see "Action characters and overtypeable fields for each command level" on page 73.

To authorize use of the / command so that MVS and JES2 commands can be issued from the command line, you must specify ALL for CMDAUTH. For a summary of the authorized MVS and JES2 commands for each command level when CMDAUTH is set to ALL, see the discussion of CMDAUTH.

A member can issue the D and L action characters for any job, on the panels to which he is authorized, when CMDAUTH is set to DISPLAY, regardless of CMDLEV.

To allow members of a group to control printers from the Printer panel, or punches from the PUN panel, you must give sufficient command level authority with the CMDLEV parameter and must include CMDAUTH values of ALL or DEST in the group's *command-authorization-list*. You can authorize more flexible control of printers or punches by using a SAF security scheme.

To specify CMDLEV using the SAF security scheme, see Appendix B, "SAF equivalents for ISFPARMS," on page 561.

GROUP	ISFGRP
CONFIRM (ON) | (OFF) | (ALWAYS)	CONFIRM=ON | OFF | ALWAYS

Specifies whether SDSF requests confirmation of destructive action characters (such as cancel or purge).

ON indicates that the action characters will require confirmation.

If CONFIRM is omitted, the value is ON.

OFF
indicates that the action characters will not require confirmation.

ALWAYS
indicates that the action characters will not require confirmation.

GROUP	ISFGRP
CPUFMT (LONG) | (SHORT)	CPUFMT=LONG | SHORT

Specifies whether SDSF displays the MVS, LPAR and zAAP views of CPU busy on the title line of the DA panel, or only the MVS view. The LPAR and zAAP views require RMF.

LONG
indicates that all values are displayed. The LPAR view is shown only when in LPAR mode. The zAAP view is shown only when a zAAP is defined and the system is in LPAR-mode.

SHORT
indicates that only the MVS view is shown.

The MVS view (the first value on the title line) is a better indicator of a CPU bottleneck. The LPAR view (the second value, if present) takes into account several states related to PR/SM™. The zAAP view (the third value, if present) shows usage of the System z® Application Assist Processor.

GROUP	ISFGRP
CTITLE (ASIS) | (UPPER)	CTITLE= ASIS | UPPER

Specifies how the case of text is displayed, specifically:

- Column titles on SDSF panels
- Text on the primary option menu
- Text on the print pop-ups
- Column titles on pop-ups
- Text displayed by SET ACTION
- Column titles displayed by SET DISPLAY
- Pop-ups when SDSF us running under TSO

Note that the case of column titles has no effect on commands that accept column titles as parameters, such as LOCATE or SORT.

ASIS
preserves the case. It is the default.

UPPER
folds text to uppercase. Column titles are folded to uppercase regardless of how they are defined in field lists in ISFPARMS.

GROUP	ISFGRP
CURSOR (ON) | (OFF) | (TOP)	CURSOR=ON | OFF | TOP

Specifies how SDSF should control placement of the cursor on tabular panels (except OD).

ON causes the cursor to return to the NP column for the last row you worked with. If the row is not on the screen, because it would require a scroll or because system or user activity caused it to be removed from the display, the cursor is returned to the command line.

If CURSOR is omitted, the value is ON.

OFF
causes the cursor to return to the command line.

TOP
causes the last row you worked with to be scrolled to the top of the screen. The cursor returns to the command line.

GROUP	ISFGRP
CUSTOM (*proplist-name*)	Not supported

Names a PROPLIST statement that customizes certain SDSF properties. For information about the PROPLIST statement, see "Customized properties (PROPLIST)" on page 91.

GROUP	ISFGRP
DADFLT (*types-and-positions*)	DADFLT=(*types-and-positions*)

Indicates the default address space types and positions to be shown on the DA panel when members of this group enter a DA command without any parameters. If the list contains more than one item, separate the items with a comma.

If this parameter is not coded with at least one value for address space position (IN, OUT, TRANS, READY) and at least one value for address space type (STC, INIT, TSU, JOB), then no address spaces are displayed when the DA command is entered with no parameters.

The possible values for the parameter follow. When RMF is installed, SDSF uses RMF as the source of data for the panel.
IN Displays swapped-in address spaces
OUT Displays swapped-out address spaces
TRANS
Displays address spaces that are in transition
READY
Displays address spaces that are ready for execution
STC Displays started tasks
INIT Displays initiators
TSU Displays TSO users
JOB Displays batch jobs

GROUP	ISFGRP
DATE (MMDDYYYY) | (DDMMYYYY) | (YYYYMMDD)	DATE=MMDDYYYY | DMMYYYY | YYYYMMDD

Sets a date format for this group: *month day year*, *day month year*, or *year month day*. SDSF uses this format when displaying dates on tabular panels and on the title line of the log panels. Commands that accept dates (LOCATE, PRINT, and FILTER) use this format.

If DATE is omitted, SDSF uses MMDDYYYY.

Users can override the date format with the SET DATE command or pop-up.

Specify the separator to be used between month, day, and year with the DATESEP parameter.

GROUP	ISFGRP
DATESEP (/) | (-) | (.)	DATESEP=/ | - | .

Sets a date separator for this group: slash (/), dash (-), or period (.). SDSF uses this separator between the month, day, and year when displaying dates on tabular panels and on the title line of the log panels. Commands with dates as parameters (LOCATE, PRINT, and FILTER) accept this separator.

If DATESEP is omitted, SDSF uses the slash (/).

Users can override the date separator with the SET DATE command or pop-up.

GROUP	ISFGRP
DEST (*NTBL-statement-name*)	DEST=*ISFNTBL-macro-label*

Names an ISFNTBL macro or NTBL statement that can be used to limit a group member to particular jobs, printers, punches, action characters, and overtypeable fields for all SDSF panels, except the DA panel. The DEST parameter works with the IDEST, CMDAUTH, and DSPAUTH parameters.

You can use the ISFNTBL macro or NTBL statement that you name to perform two functions:

1. **To restrict the destination names that a group member can use with the DEST command.**

   The DEST command limits SDSF displays to jobs having the destination names it specifies.

   If either the DEST or IDEST parameter is not coded for a user's group, the group members can specify any of the installation's destination names on the DEST command, unless a member is not authorized to use a destination name through the SAF security scheme.

   When a SAF security scheme is not used, both the DEST and the IDEST parameters must be specified to restrict destination name usage on the DEST command. Only the destination names specified in either the DEST or IDEST lists are valid on the DEST command, and jobs for all destinations cannot be displayed.

   For more information, see “Destination names” on page 239.

2. **To restrict the jobs, printers or punches for which users can enter action characters and overtype fields on SDSF panels.**

   The DEST parameter works with the CMDAUTH and DSPAUTH parameters to perform this function. If CMDAUTH is set to DEST or DSPAUTH is set to

ADEST for a group, the members of the group can use action characters and overtypeable fields on SDSF panels only for jobs, printers or punches whose DESTID matches the destinations specified in the DEST parameter's ISFNTBL macro or NTBL statement.

Printers and Punches: To control printers and punches, the destination name must be coded on the ISFNTBL macro as follows:

- If the printer or punch is defined in the JES parameters as local, with PRT*nnnn* or PUN*nn*, use U*nnnn*.
- If the printer or punch is defined in the JES parameters as remote, with RMT*nnnn*, use R*nnnn*.

For example, the following would identify all local and remote printer and punches starting with 1: `ISFNTBL U1,1,R1,1`. Also, `ISFNTBL 'U1 ',1` identifies only local printer 1 and punch 1; `ISFNTBL 'R1 ',1` identifies only remote printer 1 and punch 1.

To allow members of a group to control printers from the Printer panel, or punches from the Punch panel, when relying on ISFPARMS for authorization, you must give sufficient command level authority with the CMDLEV parameter and must include CMDAUTH values of ALL or DEST in the group's *command-authorization-list*. You can authorize more flexible control of printers and punches by using a SAF security scheme. To specify DEST using the SAF security scheme, see "Destination names" on page 239.

Jobs: For jobs, the destination name must be coded on the ISFNTBL macro or NTBL statement to match the DEST field on the panel, for all panels except the JDS panel. For the JDS panel, the DEST value that SDSF uses for authority checking is the DEST value shown on the panel from which the JDS panel was invoked, such as ST, O, or I.

For example, consider a job on the Status panel showing a value of DEST1 in the DEST field, but having individual data sets within that job with DEST values of DEST2. When a user invokes the JDS panel by entering '?' against that job and then selects an individual data set with a DEST value of DEST2, the DEST value used for authority checking is DEST1 from the Status panel. Similarly, if the user selects the job from the Status panel by entering 'S' against the job, the DEST value used for authority checking is DEST1.

GROUP	**ISFGRP**
DFIELDS (*FLD-statement-name*)	DFIELDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the DA panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	**ISFGRP**
DFIELD2 (*FLD-statement-name*)	DFIELD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the DA panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	**ISFGRP**
DISPLAY (OFF) | (ON)	DISPLAY=OFF | ON

Specifies whether SDSF is to display the current values for DEST, OWNER, PREFIX, SORT, and FILTER on the SDSF tabular panels. The default is OFF.

Users can query and override the setting with the SET DISPLAY command or pull-down choice.

GROUP	ISFGRP
DSPAUTH (*authorization-list*)	DSPAUTH=(*authorization-list*)

Specifies for which jobs the members of this group can display SYSOUT and SYSIN data sets with the Output Data Set panel. See also the IDSP, IDSPD, XDSP, and XDSPD parameters.

Two other parameters (IDSPD and XDSPD) and several values for the DSPAUTH parameter (AMDEST, AMSG, and GRPMSG) refer to the ability of a user to display **messages only**. These parameters and values restrict the display of SYSOUT data sets with DSIDs less than 101, which are considered "message" data sets. By default, the message data sets are the non-spun JES2 job logs, allocation and termination messages, and the JCL listing.

If the user has issued the INPUT ON command, "messages" also includes input JCL, internal text, and the checkpoint/restart journal.

Note that when data is transferred to JES2 from other systems, such as JES3 or RSCS, DSID numbering cannot be compatible with JES2 numbering.

To specify DSPAUTH using the SAF security scheme, see Appendix B, "SAF equivalents for ISFPARMS," on page 561.

(*display-authorization-list***)**
specifies DSPAUTH values. If the list contains more than one value, the values must be separated by a comma.

ADEST
allows a group member to display output from jobs whose destination names match the value specified through the DEST parameter. This parameter cannot be used to authorize the display of output from the Display Active Users panel.

The destination name coded on the NTBL macro or NTBL statement for the DEST parameter must match the DEST field on the panel, for all panels except the JDS panel. For the JDS panel, the DEST value that SDSF uses for authority checking is the DEST value shown on the panel from which the JDS panel was invoked, such as ST, O, or I.

ALL
allows a member to display output from all jobs.

AMDEST
allows a member to display **messages only** from jobs whose destination names match the value specified with the DEST parameter. This parameter cannot be used to authorize the display of output from the DA panel.

The destination name coded on the NTBL macro or NTBL statement for the DEST parameter must match the DEST field on the panel, for all panels except the JDS panel. For the JDS panel, the DEST value that SDSF uses for authority checking is the DEST value shown on the panel from which the JDS panel was invoked, such as ST, O, or I.

AMSG
allows a member to display **messages only** from all jobs.

GROUP
allows a member to display output from jobs whose names begin with the group's prefix (see the GPREF and GPLEN parameters below). If NOTIFY is also specified, a member can issue commands for jobs whose NOTIFY matches the group prefix.

GRPMSG
allows a member can display **messages only** from jobs whose names begin with the group's prefix.

NOTIFY
allows a member to display output from any job on which the NOTIFY parameter of the job card contains the member's user ID. If GROUP is also specified, a member can issue commands for jobs whose NOTIFY matches the group prefix.

USERID
allows a member to display output from jobs whose names begin with the member's TSO user ID.

GROUP	ISFGRP
DYNXFLDS (*FLD-statement-name*)	DYNXFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the DYNX panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
DYNXFLD2 (*FLD-statement-name*)	DYNXFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **Alternate** variable field list for the DYNX panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
EMCSAUTH (MASTER | ALL)	EMCSAUTH=MASTER | ALL

Indicates the authority that will be used when activating the EMCS console. For a description of SDSF's use of the console, see "Issuing MVS and JES commands" on page 342.

MASTER
specifies MASTER authority. This is the default.

ALL
specifies SYS,IO,CONS authority. Note that profiles in the OPERCMDS class can be used to permit SDSF users to commands that require MASTER authority when EMCSAUTH=ALL is specified in ISFPARMS.

GROUP	ISFGRP
EMCSREQ (YES | NO)	EMCSREQ=YES | NO

Controls whether SDSF must use the EMCS console for system commands. For a description of SDSF's use of the console, see “Issuing MVS and JES commands” on page 342.

YES
specifies that SDSF must use the EMCS console.

NO specifies that the EMCS console is not required. SDSF will use console ID 0 (INTERNAL) to issue commands when an EMCS console is not active. This is the default.

GROUP	ISFGRP
ENCFLDS (*FLD-statement-name*)	ENCFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the ENC panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
ENCFLD2 (*FLD-statement-name*)	ENCFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the ENC panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
ENQFLDS (*FLD-statement-name*)	ENQFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the ENQ panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
ENQFLD2 (*FLD-statement-name*)	ENQFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the ENQ panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
GPLEN (*prefix-length*)	GPLEN=*prefix-length*

Defines a prefix for a group. This prefix can be used with the CMDAUTH parameter to determine which jobs members of a group can use action characters, or with the DSPAUTH parameter to determine which jobs the members can display SYSOUT and SYSIN data sets.

To create the prefix, SDSF takes as many characters as are specified by *group-prefix-length* from the members' TSO user IDs. *Group-prefix-length* can be 1 to 8.

For example, if you have operator IDs defined as OPER1, OPER2, and OPER3, you might put the operators in a group with a group membership parameter and set GPLEN to 4 to define a group prefix of OPER for that group.

Note: By specifying GROUP for both the CMDAUTH and the DSPAUTH parameter, you limit the operators' use of action characters to those jobs whose names begin with OPER.

You can code either GPLEN or GPREF, but not both. GPREF is described below. GPLEN works in conjunction with a value of GROUP for the PREFIX parameter.

GROUP	**ISFGRP**
GPREF (*group-prefix*)	GPREF=*group-prefix*

Specifies a prefix for an authorization group. This prefix can be used with the CMDAUTH parameter to determine for which jobs the members of a group can use action characters, or with the DSPAUTH parameter to determine for which jobs the members can display SYSOUT and SYSIN data sets. The group prefix can be 1 to 8 characters and can include the generic and placeholder characters (* and % by default).

Note: The generic search character must be appended to the group prefix in order for it to be treated like a prefix.
You can code either GPLEN or GPREF, but not both. GPREF works in conjunction with GROUP for the PREFIX parameter.

GROUP	**ISFGRP**
HFIELDS (*FLD-statement-name*)	HFIELDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the H panel. If this parameter is omitted, the default primary variable field list is used. (Applies to JES2 only.)

GROUP	**ISFGRP**
HFIELD2 (*FLD-statement-name*)	HFIELD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the H panel. If this parameter is omitted, the default alternate variable field list is used. (Applies to JES2 only.)

GROUP	**ISFGRP**
ICMD (*NTBL-statement-name*)	ICMD=*ISFNTBL-macro-label*

Indicates that a member of this group can issue action characters, at a level controlled by the CMDLEV parameter, for jobs whose names are in the list created by the specified ISFNTBL macro or NTBL statement. This will be in addition to those jobs for which the group is authorized to issue commands by virtue of the CMDAUTH parameter.

GROUP	**ISFGRP**
IDEST (*NTBL-statement-name*)	IDEST=*ISFNTBL-macro-label*

Names an ISFNTBL macro or NTBL statement that determines which jobs SDSF displays at session initialization to members of the group. This parameter does not affect the Display Active Users panel. See also the ISTATUS and XSTATUS parameters.

If the IDEST parameter is coded for a group, the SDSF panels are initialized with only those jobs having destination names listed in the NTBL macro or NTBL statement. The ISFNTBL macro or NTBL statement can contain from 1 to 4 valid destination names. Any of the names in this list that are invalid (not defined to the active JES subsystem), or to which the user is not authorized through SAF, are not used as initial destinations.

If the IDEST parameter is not coded, the SDSF panels are initialized with jobs for all destinations, unless a member is not authorized to a destination name through the SAF security scheme.

If both the IDEST and the DEST parameters are coded, members can use the DEST command to display only jobs with destination names contained in the ISFNTBL macro or NTBL statements for DEST and IDEST.

The members can use the DEST command to display jobs and outputs for *all* destinations, regardless of the user ID on the node. Also, if you are not using a SAF security scheme, and have coded both the DEST and IDEST parameters, you must list all destination names that are in the NTBL macro or NTBL statement for IDEST in the ISFNTBL macro or NTBL statement for DEST. If not, those IDEST destinations will not be used as initial destinations.

It is important to have an IDEST parameter for a user group that is denied view authority to all jobs through the SAF security scheme, regardless of destination. The IDEST parameter establishes a set of default destinations for a user that is used when the SDSF session is initialized, or that may be requested using the DEST command without specifying any destination names. To specify IDEST using the SAF security scheme, see "Destination names" on page 239.

To restrict destination name usage on the DEST command when a SAF security scheme is not used, both the DEST and the IDEST parameters must be specified. If either is missing, a user can display jobs for *all* destinations.

GROUP	**ISFGRP**
IDSP (*NTBL-statement-name*)	IDSP=*ISFNTBL-macro-label*

Indicates that a member of this group can browse the output of jobs whose names are in the list created by the specified ISFNTBL macro or NTBL statement. This is in addition to those jobs for which the group is authorized to display output by virtue of the DSPAUTH parameter.

GROUP	**ISFGRP**
IDSPD (*NTBL-statement-name*)	IDSPD=*ISFNTBL-statement-name*

Indicates that a member of this group can display **messages only** through the Output Data Set panel, from jobs whose names are in the list created by the specified ISFNTBL macro or NTBL statement. This is in addition to those jobs for which the group is authorized to display output by virtue of the DSPAUTH parameter.

The XDSPD parameter and the DSPAUTH parameter values AMDEST, AMSG, and GRPMSG also refer to the ability of a user to display **messages only**. These parameters and values restrict the display of SYSOUT data sets with DSIDs less than 101, which are considered "message" data sets. (When data is transferred to JES2 from other systems, such as JES3 or RSCS, DSID numbering may not be compatible with JES2 numbering.)

GROUP	**ISFGRP**
IFIELDS (*FLD-statement-name*)	IFIELDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the I panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	**ISFGRP**
IFIELD2 (*FLD-statement-name*)	IFIELD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the I panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	**ISFGRP**
ILOGCOL (1) | (*position*)	ILOGCOL=1 | *position*

Indicates which position (or column) of the SYSLOG or OPERLOG will be the first position displayed on the panel. *position-number* can be any number from 1 through 255.

This parameter is ignored if the screen on which the SYSLOG or OPERLOG is displayed can display the entire width of the SYSLOG/OPERLOG. Also, if the value for *position-number* is so high that less than a full screen of data is displayed on the SYSLOG or OPERLOG panel, SDSF adjusts the starting position number to display a full screen of data. For example, if the width of the screen on which the SYSLOG is displayed is 80 characters, SDSF adjusts the value of *position-number* to ensure that 80 characters of data are displayed.

GROUP	**ISFGRP**
INPUT (OFF) | (ON)	INPUT=OFF | ON

Sets an initial value to control whether SYSIN data sets are displayed when users browse a job.

OFF
specifies that SYSIN data sets should not be displayed.

ON specifies that SYSIN data sets should be displayed.

If INPUT is omitted, OFF is used.

Authorized users can override the INPUT value with the INPUT command or the associated pull-down choice.

GROUP	ISFGRP
INTFLDS (*FLD-statement-name*)	INTFLDS= *ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the Initiator panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
INTFLD2 (*FLD-statement-name*)	INTFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines **alternate** variable field list for the INIT panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
ISTATUS (*NTBL-statement-name*)	ISTATUS=*ISFNTBL-statement-name*

Indicates that jobs whose job names are in the list created by the specified ISFNTBL macro or NTBL statement are to always be displayed on the DA, H, I, O, PS and ST panels unless specifically excluded by the XSTATUS parameter.

There is an exception for the Held Output Queue. When the user enters the H command with no parameter, jobs in the ISTATUS list always appear, except when the user has PREFIX=*. In this case, jobs that don't match the user's user ID don't appear, even if they are on the ISTATUS list.

GROUP	ISFGRP
ISYS (LOCAL) | (NONE)	ISYS=LOCAL | NONE

Sets an initial value to limit the data, based on a system, that a group member will see on the sysplex panels (CK, DA, ENC, INIT, LI, NO, PR, PS, PUN, RDR, RM and SO). (Applies to JES2 only.)

LOCAL
indicates that the panels will show data for the system the user is logged on to.

NONE
indicates that data on the panels is not limited by system, that is, all systems in the sysplex will be shown.

If ISYS is omitted, LOCAL is used.

Authorized users can override the ISYS value with the SYSNAME command or pull-down choice.

GROUP	ISFGRP
JCFLDS (*FLD-statement-name*)	JCFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the Job Class panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
JCFLD2 (*FLD-statement-name*)	JCFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the Job Class panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
JDSFLDS (*FLD-statement-name*)	JDSFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the Job Data Set panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
JDSFLD2 (*FLD-statement-name*)	JDSFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the Job Data Set panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
J0FLDS (*FLD-statement-name*)	J0FLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the Job 0 panel. If this parameter is omitted, the default primary variable field list is used. (JES3 only)

GROUP	ISFGRP
J0FLD2 (*FLD-statement-name*)	J0FLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the Job 0 panel. If this parameter is omitted, the default alternate variable field list is used. (JES3 only)

GROUP	ISFGRP
LANG (ENGLISH) | (ENG) | (JAPANESE) | (JPN)	LANG=ENGLISH | ENG | JAPANESE | JPN

Selects, for an SDSF session under ISPF, the language to be used for the help and tutorial panels, provided the corresponding language feature is installed. The SET LANG command allows the user to select either English (ENGLISH or ENG) or Japanese (JAPANESE or JPN). SET LANG can be queried with the SET LANG ? command and, under ISPF, the LANG value is saved and restored across sessions.

GROUP	ISFGRP
LINEFLDS (*FLD-statement-name*)	LINEFLDS=*ISFFLD-statement-name*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the LI panel. If this parameter is omitted, the default primary variable field list is displayed. (Applies to JES2 only.)

GROUP	ISFGRP
LINEFLD2 (*FLD-statement-name*)	LINEFLD2=*ISFFLD-statement-name*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the LI panel. If this parameter is omitted, the default alternate variable field list is displayed. (Applies to JES2 only.)

GROUP	ISFGRP
LNKFLDS (*FLD-statement-name*)	LNKFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the LNK panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
LNKFLD2 (*FLD-statement-name*)	LNKFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the LNK panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
LOG (OPERACT) | (OPERLOG) | (SYSLOG)	LOG=OPERACT | OPERLOG | SYSLOG

Names the default Log panel. The default Log panel is displayed when the LOG command is entered with no parameters, or the Log choice of the Display pull-down is selected.

GROUP	ISFGRP
LPAFLDS (*FLD-statement-name*)	LPAFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the LPA panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
LPAFLD2 (*FLD-statement-name*)	LPAFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the LPA panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
MASFLDS (*FLD-statement-name*)	MASFLDS=*ISFFLD-statement-name*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the MAS (JES2) and JP (JES3) panels. If this parameter is omitted, the default primary variable field list is displayed.

GROUP	ISFGRP
MASFLD2 (*FLD-statement-name*)	MASFLD2=*ISFFLD-statement-name*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the MAS (JES2) and JP (JES3) panels. If this parameter is omitted, the default alternate variable field list is displayed.

GROUP	ISFGRP
NCFLDS (*FLD-statement-name*)	NCFLDS=*ISFFLD-statement-name*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the NC panel. If this parameter is omitted, the default primary variable field list is displayed.

GROUP	ISFGRP
NCFLD2 (*FLD-statement-name*)	NCFLD2=*ISFFLD-statement-name*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the NC panel. If this parameter is omitted, the default alternate variable field list is displayed.

GROUP	ISFGRP
NODEFLDS (*FLD-statement-name*)	NODEFLDS=*ISFFLD-statement-name*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the NODES panel. If this parameter is omitted, the default primary variable field list is displayed.

GROUP	ISFGRP
NODEFLD2 (*FLD-statement-name*)	NODEFLD2=*ISFFLD-statement-name*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the NODES panel. If this parameter is omitted, the default alternate variable field list is displayed.

GROUP	ISFGRP
NSFLDS (*FLD-statement-name*)	NSFLDS=*ISFFLD-statement-name*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the NS panel. If this parameter is omitted, the default primary variable field list is displayed.

GROUP	ISFGRP
NSFLD2 (*FLD-statement-name*)	NSFLD2=*ISFFLD-statement-name*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the NS panel. If this parameter is omitted, the default alternate variable field list is displayed.

GROUP	ISFGRP
ODFLDS (*FLD-statement-name*)	ODFLDS=*ISFFLD-statement-name*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the Output Descriptors panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
ODFLD2 (*FLD-statement-name*)	ODFLD2=*ISFFLD-statement-name*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the OD panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
OFIELDS (*FLD-statement-name*)	OFIELDS=*ISFFLD-statement-name*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the Output Queue panel. If this parameter is omitted, the default primary variable field list is used. (Applies to JES2 only.)

GROUP	ISFGRP
OFIELD2 (*FLD-statement-name*)	OFIELD2=*ISFFLD-statement-name*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the Output Queue panel. If this parameter is omitted, the default alternate variable field list is used. (Applies to JES2 only.)

GROUP	ISFGRP
OWNER (NONE) | (USERID)	OWNER=NONE | USERID

Limits the jobs that a group member will see on the DA, H, I, O, PS and ST panels.

It provides a default for the OWNER command.

USERID
indicates that only those jobs whose owner is the member's user ID are displayed.

NONE
is the default. Jobs displayed are not limited by owner.

Users who are authorized to issue the OWNER command (which can be protected only through SAF security) can override the OWNER parameter with the OWNER command or pull-down choice, or the SELECT command.

GROUP	ISFGRP
PAGFLDS (*FLD-statement-name*)	PAGFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the PAG panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
PAGFLD2 (*FLD-statement-name*)	PAGFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the PAG panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
PARMFLDS (*FLD-statement-name*)	PARMFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the PARM panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
PARMFLD2 (*FLD-statement-name*)	PARMFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the PARM panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
PREFIX (NONE) | (USERID) | (GROUP)	PREFIX=NONE | USERID | GROUP

Limits the jobs that a group member will see on the DA, H, I, O, PS and ST panels.

The possible values for the PREFIX parameter are:

USERID
indicates that only those jobs whose name begins with the member's user ID are displayed, unless this parameter is overridden by the ISTATUS parameter.

GROUP
indicates that only those jobs whose name begins with the group's prefix are displayed, unless overridden by the ISTATUS parameter.

Note: PREFIX=GROUP works in conjunction with GPLEN and GPREF.

NONE
is the default. All jobs are displayed. Only those jobs whose names begin with the member's user ID are displayed on the Held Output panel.

On the O panel, users will see netmail when their current PREFIX matches a job's netmail ID. The netmail ID is displayed as part of the DEST field. See also the ISTATUS and XSTATUS parameters.

Users who are authorized to issue the PREFIX command can override the PREFIX parameter with the PREFIX command or pull-down choice, or the SELECT command.

Specifying USERID or GROUP for end users of your system improves the performance of SDSF and makes more efficient use of system resources.

GROUP	**ISFGRP**
PRTFLDS (*FLD-statement-name*)	PRTFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the Printer panel. If this parameter is omitted, the default primary variable field list is used. (Applies to JES2 only.)

GROUP	**ISFGRP**
PRTFLD2 (*FLD-statement-name*)	PRTFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the Printer panel. If this parameter is omitted, the default alternate variable field list is used. (Applies to JES2 only.)

GROUP	**ISFGRP**
PSFLDS (*FLD-statement-name*)	PSFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the Process panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	**ISFGRP**
PSFLD2 (*FLD-statement-name*)	PSFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the Process panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	**ISFGRP**
PUNFLDS (*FLD-statement-name*)	PUNFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro that defines the **primary** field list for the Punch panel. If this parameter is omitted, the default primary variable field list is displayed.

GROUP	**ISFGRP**
PUNFLD2 (*FLD-statement-name*)	PUNFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro that defines the **alternate** field list for the Punch panel. If this parameter is omitted, the default alternate variable field list is displayed.

GROUP	ISFGRP
RDRFLDS (*FLD-statement-name*	RDRFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro that defines the **primary** field list for the Reader panel. If this parameter is omitted, the default primary variable field list is displayed.

GROUP	ISFGRP
RDRFLD2 (*FLD-statement-name*)	RDRFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro that defines the **alternate** field list for the Reader panel. If this parameter is omitted, the default alternate variable field list is displayed.

GROUP	ISFGRP
RESFLDS (*FLD-statement-name*)	RESFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the Resource panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
RESFLD2 (*FLD-statement-name*)	RESFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the Resource panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
RMFLDS (*FLD-statement-name*)	RMFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the RM panel. If this parameter is omitted, the default primary variable field list is used.(Applies to JES2 only.)

GROUP	ISFGRP
RMFLD2 (*FLD-statement-name*)	RMFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the RM panel. If this parameter is omitted, the default alternate variable field list is used.(Applies to JES2 only.)

GROUP	ISFGRP
RSYS (LOCAL) | (NONE)	RSYS=LOCAL | NONE

Sets an initial value to limit WTORs, based on system, that a group member will see on the Log panels.

LOCAL
indicates that only WTORS issued by the system the user is logged on to are displayed.

NONE
indicates that WTORs are not limited by system, that is, all WTORs for all systems are shown.

If RSYS is omitted, NONE is used.

GROUP	ISFGRP
SEFLDS (*FLD-statement-name*)	SEFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the Scheduling Environment panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
SEFLD2 (*FLD-statement-name*)	SEFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the Scheduling Environment panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
SOFLDS (*FLD-statement-name*)	SOFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the Spool Offload panel. If this parameter is omitted, the default primary variable field list is used.(Applies to JES2 only.)

GROUP	ISFGRP
SOFLD2 (*FLD-statement-name*)	SOFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the Spool Offload panel. If this parameter is omitted, the default alternate variable field list is used.(Applies to JES2 only.)

GROUP	ISFGRP
SPFLDS (*FLD-statement-name*)	SPFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the Spool Volumes panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
SPFLD2 (*FLD-statement-name*)	SPFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the Spool Volumes panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
SRCHFLDS (*FLD-statement-name*)	SRCHFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the SRCH panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
SRCHFLD2 (*FLD-statement-name*)	SRCHFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the SRCH panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
SRFLDS (*FLD-statement-name*)	SRFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the System Requests panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
SRFLD2 (*FLD-statement-name*)	SRFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the System Requests panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
STFLDS (*FLD-statement-name*)	STFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the Status panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
STFLD2 (*FLD-statement-name*)	STFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the Status panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
SYMFLDS (*FLD-statement-name*)	SYMFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the SYM panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
SYMFLD2 (*FLD-statement-name*)	SYMFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the SYM panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
SYSFLDS (*FLD-statement-name*)	SYSFLDS=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **primary** variable field list for the SYS panel. If this parameter is omitted, the default primary variable field list is used.

GROUP	ISFGRP
SYSFLD2 (*FLD-statement-name*)	SYSFLD2=*ISFFLD-macro-label*

Names an ISFFLD macro or FLD statement that defines the **alternate** variable field list for the SYS panel. If this parameter is omitted, the default alternate variable field list is used.

GROUP	ISFGRP
SYSID (*system-id*)	SYSID=*system-id*

Indicates the default system ID of the system log which a member of this group displays on the SYSLOG panel in a JES2 environment.. If this parameter is omitted, the default is the current system log. This parameter is useful in a JES2 multi-access spool environment. The setting of SYSID can be changed by the user through use of the SYSID command if the user is authorized to use it, through the AUTH parameter. (Applies to JES2 only.)

GROUP	ISFGRP
SYSID3 (*system-id*)	SYSID3=*system-id*

Indicates the default system ID of the system log which a member of this group displays on the SYSLOG panel in a JES3 environment. If this parameter is omitted, the default is the current system log. The setting of SYSID3 can be changed by the user through use of the SYSID command if the user is authorized to use it, through the AUTH parameter. (Applies to JES3 only.)

GROUP	ISFGRP
UPCTAB (TRTAB2) | (*TRTAB-statement-name*)	UPCTAB=TRTAB2 | *TRTAB-statement-name*

Assigns a name to the translation table that converts lowercase characters to uppercase. Use this parameter to request a code page other than the default code page for a group of users.

This parameter works with an ISFTR macro, TRTAB statement, or TRDEF statement. SDSF looks for:

- An ISFTR macro or TRTAB statement with the character string *TR-statement-name* in the UPCTAB parameter.
- A TRDEF statement with the character string *TR-statement-name* in the NAME parameter. Use TRDEF to define your own translation table.

TR-statement-name can be any character string that is a valid label for your assembler. The default is TRTAB2.

If you are using statements and omit UPCTAB, the code page defaults to **SDSF**. For more information, see "Code page (TRTAB/TRDEF or ISFTR)" on page 101.

GROUP	ISFGRP
VALTAB (TRTAB) | (*TRTAB-statement-name*)	VALTAB=TRTAB | *TRTAB-statement-name*

Assigns a name to the translation table that checks for valid characters. Use this parameter to request a code page other than the default code page for a group of users.

This parameter works with an ISFTR macro, TRTAB statement, or TRDEF statement. SDSF looks for:

- An ISFTR macro or TRTAB statement with the character string *TR-statement-name* in the VALTAB parameter.
- A TRDEF statement with the character string *TR-statement-name* in the NAME parameter. Use TRDEF to define your own translation table.

TR-statement-name can be any character string that is a valid label for your assembler. The default is TRTAB.

If you are using statements, and omit VALTAB, the code page defaults to **SDSF**. For more information, see "Code page (TRTAB/TRDEF or ISFTR)" on page 101.

GROUP	ISFGRP
VIO (SYSALLDA) | (*unit-name*)	VIO=SYSALLDA | *unit-name*

Specifies the unit name to be used for a temporary file when viewing page-mode output. (Applies to JES2 only.) If VIO is not specified, SDSF uses the default, SYSALLDA. Specification of a unit name that refers to a VIO device is strongly recommended for performance and security reasons.

GROUP	ISFGRP
XCMD (*NTBL-statement-name*)	XCMD=*ISFNTBL-macro-label*

Indicates that a member of this group *cannot* issue SDSF action characters at a level controlled by the CMDLEV parameter for jobs whose names are in the list created by the specified ISFNTBL macro or NTBL statement. This parameter overrides both the CMDAUTH and ICMD parameters.

GROUP	ISFGRP
XDSP (*NTBL-statement-name*)	XDSP=*ISFNTBL-macro-label*

Indicates that a member of this group *cannot* display, through the Output Data Set panel, messages and user output from jobs whose names are in the list created by

the specified ISFNTBL macro or NTBL statement. This parameter overrides the DSPAUTH, IDSP, IDSPD, and XDSPD parameters.

GROUP	ISFGRP
XDSPD (*NTBL-statement-name*)	XDSPD=*ISFNTBL-macro-label*

Indicates that a member of this group *cannot* display, on the Output Data Set panel, output for jobs whose names are in the list created by the ISFNTBL macro or NTBL statement. A member can display **messages only**. This parameter overrides the DSPAUTH, IDSPD, and IDSP parameters.

The IDSPD parameter and the DSPAUTH parameter values AMDEST, AMSG, and GRPMSG also refer to the ability of a user to display **messages only**. These parameters and values restrict the display of SYSOUT data sets with DSIDs less than 101, which are considered "message" data sets. (When data is transferred to JES2 from other systems, such as JES3 or RSCS, DSID numbering may not be compatible with JES2 numbering.)

GROUP	ISFGRP
XSTATUS (*NTBL-statement-name*)	XSTATUS=*ISFNTBL-macro-label*

Indicates that jobs whose names are in the list created by the specified ISFNTBL macro or NTBL statement will be excluded from all SDSF panels for members of this group. This parameter overrides all other parameters that control which jobs are displayed, including ISTATUS.

Action characters and overtypeable fields for each command level

The tables that follow are a reference for coding the CMDLEV parameter. The table shows the action characters that can be issued and the fields that can be overtyped for each command level.

The CMDLEV parameter is ignored in a JES3 environment.

Each command level is inclusive of all those with a lower number. For example, a user with a command level of 3 can perform the functions requiring a command level of 3, 2, 1, and 0.

If an action character or overtypeable field is valid on a panel, the column for `Where Valid` is marked with one or more panel names.

The SDSF panels are Address Space Memory (**AS**), Authorized Program Facility (**APF**), Health Checker (**CK**), Health Check History (**CKH**), Display Active Users (**DA**), Dynamic Exits (**DYNX**), Enclaves (**ENC**), Enqueues (**ENQ**), Held Output Queue (**H**), Input Queue (**I**), Initiator (**INIT**), Job Class (**JC**), Job Data Set (**JDS**), Lines (**LI**), Link List (**LNK**), Link Pack Area (**LPA**), Multi-Access Spool (**MAS**), Network Connection (**NC**), Nodes (**NO**), Network Server (**NS**), Output Queue (**O**), Output Descriptors (**OD**), Page (**PAG**), PARMLIB (**PARM**), Printer (**PR**), Process (**PS**), Punch (**PUN**), Reader (**RDR**), Resource (**RES**), Resource Monitor (**RM**), Scheduling Environment (**SE**), Spool Offload (**SO**), Spool Volumes (**SP**), Search (**SRCH**), System symbols (**SYM**), System Requests (**SR**), System (**SYS**), and Status (**ST**).

Some action characters and overtypeable columns have specific requirements, indicated by a superscript:

- RMF indicates that the DA panel must be obtaining its data from RMF.

Command level 0

Table 20. Command Level 0 Action Characters

Action Characters	SDSF Actions, or MVS and JES2 Commands Issued	Where Valid
+	Expand NP column width	All tabular panels except OD
//	Block	All tabular panels except OD
=	Repeat	All tabular panels except OD
?	List job data sets	DA H I O OD ST
I	Display more information	ENC I ST
M	Match a multisystem enclave	ENC
N	Display enqueues	DA
Q	Display output descriptors	DA H I JDS O ST
R	Display WLM resources. See note.	SE
S	Display check results or job data sets. See note.	CK CKH DA H I JDS O OD ST
SB SE	Display check results or job data sets. See note.	APF CK CKH DA H I JDS LNK LPA O OD PARM SRCH ST
SJ	Display check results or job data sets. See note.	DA H I JDS O OD ST
SBI	Browse REXX input data set using ISPF browse	CK
SBO	Browse REXX output data set using ISPF browse	CK
SEI	Edit REXX input data set using ISPF browse	CK
SEO	Edit REXX output data set using ISPF browse	CK
ST	Display associated jobs. See note.	JC SE
V	View page-mode data. See note.	JDS OD
X	Print data set. See note.	CK CKH DA H I JDS O OD ST
XC	Print data set and close print file	CK CKH DA H I JDS O OD ST
XD XDC	Print data set using Open Print Data Set panel	CK CKH DA H I JDS O OD ST
XF XFC	Print data set using Open Print File panel	CK CKH DA H I JDS O OD ST
XS XSC	Print data set using Open Print panel for SYSOUT	CK CKH DA H I JDS O OD ST

Table 20. Command Level 0 Action Characters (continued)

Action Characters	SDSF Actions, or MVS and JES2 Commands Issued	Where Valid

Note:

The R, S, ST on JC and SE, V and X action characters are not controlled by command level (CMDLEV parameter). See the DSPAUTH parameter for information on S and V. R and ST are controlled by access to the RES and ST commands. See the AUTH parameter for information.

Command level 1

Table 21. Command Level 1 Action Characters

Action Characters	SDSF Actions, or MVS and JES2 Commands Issued	Where Valid
D	$D (Display)	DA I ST INIT JC LI MAS NC NO NS PR PUN RDR RM SO SP
D	D (Display)	APF DYNX ENQ LNK LPA PAG PARM PS RES SE SR SYS
DA	$D (Display)	NS
D (all forms)	D (Display)	APF DYNX LNK LPA PAG PARM SYM SYS
DC	$D (Display)	NO
DL	$D (Display)	DA I ST INIT NC NS PR PUN RDR SP
DP	$D (Display)	NO
DS	$D (Display)	NS
J	$J	MAS
J	$D (Display)	SP
JD	$J	MAS
JH	$J	MAS
JJ	$J	MAS
JS	$J	MAS
L	$L (List) $DO Display check history	DA I H O ST CK
LL	$L (List) $DO	DA H O ST
ST	Display ST	JC

Command level 2

Table 22. Command Level 2 Action Characters

Action Characters	SDSF Actions, or MVS and JES2 Commands Issued	Where Valid
A	$TO (Set output)	H O
C	$C (Cancel) $CO (Cancel) See notes 1 and 2.	DA H I JDS O ST LI NO PR PUN RDR SO H O
CA	$C,ARMRESTART	DA I ST

Table 22. Command Level 2 Action Characters (continued)

Action Characters	SDSF Actions, or MVS and JES2 Commands Issued	Where Valid
CD	$C (Cancel, dump). See note 2.	DA I ST
CDA	$C,D,ARMRESTART	DA I ST
E	Erase output descriptors	OD
E	$E (Restart)	DA I ST LI NO PR PUN SO
EC	$E (Restart)	DA I ST
ES ESH	$E (Restart)	DA I ST
H	$H (Hold)	DA I ST
H	$TO (Set output)	H O
H	SSI	JDS
O	$O (Release). See note 1.	JDS ST
O	$TO (Release). See note 1.	H
OK	$TO (Release). See note 1.	H
P	$C (Purge) $CO (Purge) See notes 1 and 2.	DA H I JDS O ST H O
PP	$C (Purge protected). See note 2.	DA I ST
W	$T (Set)	DA I JDS ST

Notes:

1. SDSF uses the subsystem interface (SSI) when you enter a C, O, or P action character on the JDS panel. When all data sets are deleted by use of the C and P action characters on the H panel, SDSF issues $O.
2. When a TSU job is canceled or purged on the DA, I, or ST panels, SDSF issues the MVS command, C U=*userid* or C U=*userid*,*DUMP* rather than $C; a $C is used to cancel a TSU job on the DA panel. When an active APPC transaction program is canceled or purged on the DA panel, SDSF issues C *jobname*,A=*asid* rather than $C.

Table 23. Command Level 2 Overtypeable Fields

Overtypeable Fields	MVS and JES2 Commands Issued	Where Valid
ADDRESS		JDS OD
AFPPARMS		JDS OD
BUILDING		JDS OD
BURST	$TO (set Output)	H O
C	$TO (Set output). See note.	H O JDS
C	$T	I ST
CC	SSI	JDS
COLORMAP		JDS OD
COMSETUP		JDS OD
DEPARTMENT		JDS OD

Table 23. Command Level 2 Overtypeable Fields (continued)

Overtypeable Fields	MVS and JES2 Commands Issued	Where Valid
DEST	$TO (Set output). See note.	H O JDS
FCB	$TO (Set output)	H O
FLASH	$TO (Set output)	DA H O
FORMDEF		JDS OD
FORMLEN		JDS OD
FORMS	$TO (Set output). See note.	H O JDS
INTRAY		OD
IP DESTINATION		OD
ITY		JDS
NAME		JDS OD
NOTIFY		JDS OD
OCOPYCNT		JDS OD
ODISP	$TO (Set output)	H O
OFFSETXB		JDS OD
OFFSETXF		JDS OD
OFFSETYB		JDS OD
OFFSETYF		JDS OD
OUTBIN		OD
OUTBN		JDS
OVERLAYB		JDS OD
OVERLAYF		JDS OD
PAGEDEF		JDS OD
PORT		JDS
PORTNO		OD
PRTOPTNS		OD
PRTQUEUE		OD
PRMODE	$TO (Set Output)	H O
PRTDEST	$R (Route)	I ST
RETAINF		OD
RETAINS		OD
RETRYL		OD
RETRYT		OD
ROOM		JDS OD
TITLE		JDS OD
UCS	$TO (Set output)	H O
USERDATA		OD
USERDATA1		JDS
USERLIB		JDS OD

Table 23. Command Level 2 Overtypeable Fields (continued)

Overtypeable Fields	MVS and JES2 Commands Issued	Where Valid
WTR	$TO (Set Output). See note.	H O JDS

Note:

SDSF uses the subsystem interface (SSI) when you overtype fields on the JDS panel.

Command level 3

Table 24. Command Level 3 Action Characters

Action Characters	MVS and JES2 Commands Issued	Where Valid
A	F (Modify Activate)	CK
A	$A (Release)	DA I ST
A	$A (Release)	DA I ST
AI	SETAUTOR	SR
B*x*	$B (Back space)	PR PUN
C	*C	NC
C	C (Cancel)	PS
C	K C	SR
D	F (Modify Display)	CK
DL	F (Modify Display)	CK
DP	F (Modify Display)	CK
DPO	F (Modify Display)	CK
DS	F (Modify Display)	CK
E	F (Modify Refresh)	CK
E	$E (Restart)	NC NS
F*x*	$F (Forward space)	PR PUN
H	F (Modify Deactivate)	CK
H	$H (Hold)	JC
I	$I (Interrupt)	PR PUN
I	$T (Set)	LI
J	$SJ (Start)	I ST
K	C (Cancel)	DA
K	CANCEL	NS
K	F (Modify)	PR
K	F (Modify)	PS
KD	C (Cancel, dump)	DA
KD	CANCEL	NS
N	$N (Repeat)	PR PUN
P	F (Modify Delete)	CK
P	$P (Stop)	INIT LI PR PUN RDR SO SP

Table 24. Command Level 3 Action Characters (continued)

Action Characters	MVS and JES2 Commands Issued	Where Valid
P	$P (Stop)	NC NS
PC	$P (Stop)	SP
PF	F (Modify Delete,Force)	CK
R	F (Modify Run)	CK
R[RMF]	RESET	DA
R		ENC
R	R (Reply)	SR
RQ[RMF]	RESET	DA
RQ		ENC
Q	$T (Set)	LI
S	$S (Start)	INIT LI MAS PR PUN RDR SO SP
S	$S (Start)	NC NS
SN	$S (Start)	NC NO
SR	$S (Start)	SO
ST	$S (Start)	SO
T	F (Modify)	PS
U	F (Modify Update,Repcat)	CK
Y[RMF]	P (Stop)	DA
Y	STOP	NS
Z	$Z (Halt)	INIT PR PUN RDR SP
Z	FORCE	DA
Z	FORCE	NS

Table 25. Command Level 3 Overtypeable Fields

Overtypeable Fields	MVS and JES2 Commands Issued	Where Valid
ACTIVE	$T (Set)	JC
ADISC	$T (Set)	LI
ANODE	$T (Set)	LI NC
APPL	$T (Set)	NS
APPLID	$T (Set)	LI
ARCHIVE	$T (Set)	SO
ASIS	$T (Set)	PR
AUTH	$T (Set)	JC
AUTHORITY	$T (Set)	NO RDR
BLP	$T (Set)	JC
C	$T (Set)	I ST RDR
CATEGORY	F (Modify)	CK
CCTL	$T (Set)	PR PUN
CHAR1–4	$T (Set)	PR

Table 25. Command Level 3 Overtypeable Fields (continued)

Overtypeable Fields	MVS and JES2 Commands Issued	Where Valid
CKPTHOLD	$T (Set)	MAS
CKPTLINE	$T (Set)	PR PUN
CKPTMODE	$T (Set)	PR
CKPTPAGE	$T (Set)	PR PUN
CKPTSEC	$T (Set)	PR
CLASSES	$T (Set)	INIT
CLASS1-8	$T (Set)	INIT
CMPCT	$T (Set)	PR PUN
CODE	$T (Set)	LI
COMMAND	$T (Set)	JC
COMP	$T (Set)	LI PR PUN
COMPACT	$T (Set)	NC PR PUN
CONNECT	$T (Set)	LI NC
CONN-INT	$T (Set)	LI NC
COPYMARK	$T (Set)	PR
CP	$T (Set)	NO
CPR	$T (Set)	JC
CPY	$T (Set)	JC
CPYMOD	$T (Set)	PR
CRTIME	$T (Set)	SO
CTR	$T (Set)	LI NC NS
DEBUG	F (Modify)	CK
DFCB	$T (Set)	PR
DORMANCY	$T (Set)	MAS
DSENQSHR	$T (Set)	JC
DSNAME	$T (Set)	SO
DUPLEX	$T (Set)	LI
EINTERVAL	F (Modify)	CK
END	$T (Set)	NO
EXECNODE	$R (Route)	I ST
FCBL	$T (Set)	PR
FLS	$T (Set)	PUN
FSATRACE	$T (Set)	PR
FSSNAME	F (Modify)	PR
GROUP	$T (Set)	JC
HOLD	$T (Set)	NO RDR
HONORTRC	$T (Set)	PR
INTERVAL	F (Modify)	CK
INTF	$T (Set)	LI

Table 25. Command Level 3 Overtypeable Fields (continued)

Overtypeable Fields	MVS and JES2 Commands Issued	Where Valid
IPNAME	$T (Set)	NC NS
JCLIM	$T (Set)	JC
JESLOG	$T (Set)	JC
JRNL	$T (Set)	JC
JRNUM	$T (Set)	LI
JTNUM	$T (Set)	LI
JTR	$T (Set)	LI NC NS
K	$T (Set)	PR
LABEL	$T (Set)	SO
LIMIT	$T (Set)	RM
LINE	$T (Set)	NO NC
LINECCHR	$T (Set)	LI
LINE-LIMIT	$T (Set)	LI NC PR PUN SO
LOG	$T (Set)	LI NS
LOGMODE	$T (Set)	NC NO
LOGON	$T (Set)	NC NO
LRECL	$T (Set)	PR PUN
M	$T (Set)	PR
MAX-TIME	$T (Set)	JC
MBURST	$T (Set)	SO
MC	$T (Set)	RDR
MDEST	$T (Set)	SO
MFCB	$T (Set)	SO
MFLH	$T (Set)	SO
MFORMS	$T (Set)	SO
MHOLD	$T (Set)	SO
MODE	$T (Set)	PR
MODSP	$T (Set)	SO
MPRMODE	$T (Set)	SO
MSAFF	$T (Set)	SO
MSGLV	$T (Set)	JC
MUCS	$T (Set)	SO
MWRITER	$T (Set)	SO
NETSRV	$T (Set)	NC NO
NEWPAGE	$T (Set)	PR
NODE	$SN (Start)	LI
NODENAME	$T (Set)	NO
NOTIFY	$T (Set)	SO
NPRO	$T (Set)	PR

Table 25. Command Level 3 Overtypeable Fields (continued)

Overtypeable Fields	MVS and JES2 Commands Issued	Where Valid
ODISP	$T (Set)	JC
OUT	$T (Set)	JC
PAGE-LIMIT	$T (Set)	LI NC PR SO
PARAMETERS	F (Modify)	CK
PASSWORD	$T (Set)	LI NS
PAU	$T (Set)	PR PUN
PEN	$T (Set)	NO
PGN	E (Reset)	DA
PGN	$T (Set)	JC
PGNM	$T (Set)	JC
PI	$T (Set)	RDR
PL	$T (Set)	RDR
PMG	$T (Set)	NO
PORT	$T (Set)	NC NS
PROT	$T (Set)	SO
PRTDEST	$T (Set)	RDR
PRTY	$T (Set)	I ST
PRTY	$TO (Set output)	H O
PRV	$T (Set)	NO
PSEL	$T (Set)	PR
PUNDEST	$T (Set)	RDR
QHLD	$T (Set)	JC
QUIESCE	E (Reset)	DA
RECV	$T (Set)	NO
REGION	$T (Set)	JC
RESERVED	$T (Set)	SP
REST	$T (Set)	LI NC NO
RESTART	$T (Set)	LI NS
REST-INT	$T (Set)	LI NS
RST	$T (Set)	JC
RTPD	$T (Set)	SO
SAFF	$T (Set)	I SP ST
SAFF1	$T (Set)	RDR
SBURST	$T (Set)	PR SO
SCHEDULING-ENV	$T (Set)	I JC SO ST
SCN	$T (Set)	JC
SDISP	$T (Set)	SO
SECURE	$T (Set)	NC NS
SELECT	$T (Set)	PR PUN

Table 25. Command Level 3 Overtypeable Fields (continued)

Overtypeable Fields	MVS and JES2 Commands Issued	Where Valid
SENTRS	$T (Set)	NO
SEP	$T (Set)	PR PUN
SEPCHAR	$T (Set)	PR
SEPDS	$T (Set)	PR PUN
SETUP	$T (Set)	PR PUN
SEVERITY	F (Modify)	CK
SFCB	$T (Set)	PR SO
SFLH	$T (Set)	PR SO
SFORMS	$T (Set)	PR PUN SO
SHOLD	$T (Set)	SO
SJOBNAME	$T (Set)	PR PUN SO
SOCKET	$T (Set)	NS
SODSP	$T (Set)	SO
SODSP	$T (Set)	LI NC
SOWNER	$T (Set)	PR PUN SO
SPEED	$T (Set)	LI
SPRMODE1	$T (Set)	PR PUN SO
SRANGE	$T (Set)	PR PUN SO
SRNUM	$T (Set)	LI
SRVCLASS	E (Reset)	DA
SRVCLASS		ENC
SRVCLASS	$T (Set)	I ST
SSAFF	$T (Set)	SO
SSCHEDULIING-ENV	$T (Set)	SO
SSRVCLASS	$T (Set)	SO
STACK	$T (Set)	NS
STNUM	$T (Set)	LI
SUBNET	$T (Set)	NO
SUCS	$T (Set)	PR SO
SUS	$T (Set)	PR PUN
SVOL	$T (Set)	PUN SO
SVOL1	$T (Set)	PR
SWA	$T (Set)	JC
SWRITER	$T (Set)	PR PUN SO
SYNCTOL	$T (Set)	MAS
SYSSYM	$T (Set)	JC
TP6	$T (Set)	DA
TP26	$T (Set)	DA
TR	$T (Set)	LI NC NO NS PR PUN RDR

Table 25. Command Level 3 Overtypeable Fields (continued)

Overtypeable Fields	MVS and JES2 Commands Issued	Where Valid
TRANS	$T (Set)	NO
TRANSP	$T (Set)	LI
TRKCELL	$T (Set)	PR
UCSV	$T (Set)	PR
UJP	$T (Set)	JC
USERDATE	F (Modify)	CK
USO	$T (Set)	JC
VOLS	$T (Set)	SO
VALIDATE	$T (Set)	SO
VERBOSE	F (Modify)	CK
VTR	$T (Set)	LI NC NS
WARN%	$T (Set)	RM
WTOTYPE	F (Modify)	CK
XEQDEST	$T (Set)	RDR
XBM	$T (Set)	JC

Command level 4

Table 26. Command Level 4 Action Characters

Action Characters	MVS and JES2 Commands Issued	Where Valid
E	$E (Restart)	MAS
P	$P (Stop)	MAS
ZM	$J	MAS

Table 27. Command Level 4 Overtypeable Fields

Overtypeable Fields	MVS and JES2 Commands Issued	Where Valid
MCLASS	$T (Set)	SO
SCLASS	$T (Set)	PR PUN SO
System	F (Modify)	RES

Command level 5

Table 28. Command Level 5 Overtypeable Fields

Overtypeable Fields	MVS and JES2 Commands Issued	Where Valid
SDEST1	$T (Set)	PR PUN SO
SENDP	$T (Set)	NO
UNIT	$T (Set)	LI PR PUN RDR SO
VERIFYP	$T (Set)	NO

Command level 6

Table 29. Command Level 6 Overtypeable Fields

Overtypeable Fields	MVS and JES2 Commands Issued	Where Valid
WORK-SELECTION	$T (Set)	LI NC PR PUN SO

Command level 7

Command level 7 authorizes the user to all action characters and overtypeable fields.

Variable field lists (FLD or ISFFLD)

An FLD statement along with FLDENT statements, or an ISFFLD macro, defines the fields that are displayed on an SDSF panel. It is associated with the field list for a particular panel by an ISGRP macro or GROUP statement.

In ISFPARMS assembler macros, the ISFFLD macros must come after the ISFGRP macros. When you use statements, the statements can be in any order.

You can define a **primary** and **alternate** variable field list for each SDSF panel. The primary field list contains those fields that are shown upon entry into a panel. The alternate field list contains fields that can be displayed by use of the ? command.

For using SDSF interactively, it is important to locate overtypeable fields on the panel so that the entire field is visible on one screen. An overtypeable field can be overtyped only when the entire field is visible.

The fields that are available on a panel can also be affected by the JES level. The ARRANGE command allows users to change the order and widths of the fields in each field list.

With SDSF's support for REXX, users can develop REXX execs that have dependencies on specific columns. You should be aware when removing columns from a field list that this may impact REXX execs.

Example of the FLD statement and ISFFLD macro

FLD and FLDENT Statements	ISFFLD Macro
1 GROUP TSOAUTH(JCL,OPER,ACCT), 2 IFIELDS(DFLD) 3 FLD NAME(DFLD) TYPE(IN) FLDENT COLUMN(JNUM),TITLE(' JOBNUM'),WIDTH(7) 4 FLDENT COLUMN(JPRIO),TITLE(PRTY),WIDTH(4)	1 ISFGRP TSOAUTH=(JCL,OPER,ACCT), 2 IFIELDS=DFLD 3 DFLD ISFFLD JNUM,' JOBNUM',7, 4 JPRIO,'PRTY',4, 5 TYPE=IN

On line 2 of the example, the IFIELDS parameter refers to an ISFFLD macro (with the macro label) or FLD statement (with the NAME parameter).

The ISFFLD macro and FLD statement begin on the line marked with 3. Each defines a column for the JES job number, with a title of ' JOBNUM' and a width of 7 characters; and a column for the JES input queue priority, with a title of PRTY and width of 4 characters (line 4). The TYPE parameter identifies the panel as the IN or Input Queue panel (line 5 in the ISFGRP macro, line 3 of the FLD example).

FLD and ISFFLD syntax

FLD and FLDENT statements

FLD NAME(*FLD-statement-name*),TYPE(*panel-ID*)
FLDENT COLUMN(*column*),TITLE(*title*),WIDTH(*width*)

ISFFLD macro

label ISFFLD *column*,*title*, *width*,...,TYPE=*panel-ID*

label **or** *FLD-statement-name*
names the ISFFLD macro or FLD statement referenced by a group. The name can be alphabetic, numeric, or national characters (@, #, $) and must begin with an alphabetic character.

column
is a 2-to-8-character name, as defined by SDSF, for a column on an SDSF panel that displays tabular information. Chapter 4, "Columns on the SDSF panels," on page 133 includes tables of the columns for each panel.

You will achieve better SDSF performance if the primary field list contains only those fields that SDSF can obtain from in-storage control blocks. These are marked as having *immediate* access in the tables in Chapter 4, "Columns on the SDSF panels," on page 133. Those fields that require an I/O operation to the spool data set (*delayed* access) should be in the alternate field list.

title
is the title that appears on a panel for the column defined by *column*.

When you define a title using mixed case, enclose it in single quotation marks to ensure that it is displayed in mixed case. The case of the column titles does not affect commands that use titles as parameters, such as SORT and FILTER. The CTITLE parameter of the GROUP statement can be used to fold all column titles to uppercase.

If the title contains blanks, you ***must*** enclose it in single quotation marks. Similarly, users entering commands with column titles as parameters will be required to enclose those titles within quotation marks. For this reason, you may want to avoid coding titles that contain blanks.

A title must not be more than 18 characters long.

width
is the width of the column on the panel. The width must be at least as long as the title. Use `D` to get the SDSF default length.

When displaying numeric values that are too large for the column width, SDSF scales them using these abbreviations: T (thousands), M (millions), B (billions), KB (kilobytes), MB (megabytes), GB (gigabytes), TB (terabytes) and PB (petabytes).

panel-ID
is one of the following, corresponding to the SDSF tabular panel for which this variable field list was designed:
APF Authorized Program Facility panel
AS Address Space Memory panel
CK Health Checker panel
CKH Health Checker History panel
DA Display Active Users panel
DYNX Dynamic Exits panel

ENC Enclaves panel
ENQ Enqueues panel
HOLD
Held Output Queue panel
IN Input Queue panel
INT Initiator panel
JC Job Class panel
JDS Job Data Set panel
J0 Job 0 panel
LINE Lines panel
LNK Link List panel
LPA Link Pack Area panel
MAS Multi-Access Spool panel (JES2) and JESPLEX panel (JES3)
NC Network connection panel
NODE
Nodes panel
NS Network server panel
OD Output Descriptors panel
OUT Output Queue panel
PAG Page panel
PARM PARMLIB panel
PRT Printer panel
PS Process panel
PUN Punch panel
RDR Reader panel
RES Resource panel
RM Resource Monitor panel
SE Scheduling Environment panel
SO Spool Offload panel
SP Spool Volumes panel
SR System Requests panel
SRCH Search panel
STAT Status panel
SYM System Symbols panel
SYS System Information panel

Table 30 shows, for each SDSF panel, the ISFGRP and GROUP parameters that name the primary and alternate field lists, and where to find a complete list of fields.

Table 30. Field List Parameters

Panel	ISFGRP or GROUP Parameter	Reference for Field List
APF	APFFLDS, APFFLD2	"Authorized Program Facility panel (APF)" on page 135
AS	ASFLDS, ASFLD2	"Address Space Memory panel (AS)" on page 133
CK	CKFLDS, CKFLD2	"Health Checker panel (CK)" on page 142
CKH	CKHFLDS, CKHFLD2	"Health Check History panel (CKH)" on page 141
DA	DFIELDS, DFIELD2	"Display Active Users panel (DA)" on page 135
DYNX	DYNXFLDS, DYNXFLD2	"Dynamic Exits panel (DYNX)" on page 138

Table 30. Field List Parameters (continued)

Panel	ISFGRP or GROUP Parameter	Reference for Field List
ENC	ENCFLDS, ENCFLD2	"Enclaves panel (ENC)" on page 139
ENQ	ENQFLDS, ENQFLD2	"Enqueue panel (ENQ)" on page 140
H	HFIELDS, HFIELD2	"Held Output panel (H)" on page 144
I	IFIELDS, IFIELD2	"Input Queue panel (I)" on page 147
INIT	INTFLDS, INTFLD2	"Initiator panel (INIT)" on page 146
JC	JCFLDS, JCFLD2	"Job Class panel (JC)" on page 149
JDS	JDSFLDS, JDSFLD2	"Job Data Set panel (JDS)" on page 151
JP	MASFLDS, MASFLD2	"JESPLEX panel (JP)" on page 149 and "Multi-Access Spool panel (MAS) and JESPLEX (JP) panel" on page 158
J0	J0FLDS, J0FLD2	"Job 0 (J0)" on page 154
LI	LINEFLDS, LINEFLD2	"Lines panel (LI)" on page 155
LNK	LNKFLDS, LNKFLD2	"Link List panel (LNK)" on page 157
LPA	LPAFLDS, LPAFLD2	"Link Pack Area panel (LPA)" on page 158
MAS	MASFLDS, MASFLD2	"Multi-Access Spool panel (MAS) and JESPLEX (JP) panel" on page 158 and "JESPLEX panel (JP)" on page 149
NC	NCFLDS, NCFLD2	"Network Connections (NC)" on page 160
NO	NODEFLDS, NODEFLD2	"Nodes panel (NO)" on page 162
NS	NSFLDS, NSFLD2	"Network Servers (NS)" on page 161
O	OFIELDS, OFIELD2	"Output Queue panel (O)" on page 166
OD	ODFLDS, ODFLD2	"Output Descriptors panel (OD)" on page 164
PAG	PAGFLDS, PAGFLD2	"Page panel (PAG)" on page 168
PARM	PARMFLDS, PARMFLD2	"PARMLIB panel (PARM)" on page 169
PR	PRTFLDS, PRTFLD2	"Printer panel (PR)" on page 169

Table 30. Field List Parameters (continued)

Panel	ISFGRP or GROUP Parameter	Reference for Field List
PS	PSFLDS, PSFLD2	"Processes panel (PS)" on page 174
PUN	PUNFLDS, PUNFLD2	"Punch panel (PUN)" on page 174
RDR	RDRFLDS, RDRFLD2	"Reader panel (RDR)" on page 177
RES	RESFLDS, RESFLD2	"Resource panel (RES)" on page 178
RM	RMFLDS, RMFLD2	"Resource Monitor (RM) panel" on page 179
SE	SEFLDS, SEFLD2	"Scheduling Environment panel (SE)" on page 179
SO	SOFLDS, SOFLD2	"Spool Offload panel (SO)" on page 180
SP	SPFLDS, SPFLD2	"Spool Volumes panel (SP)" on page 182
SR	SRFLDS, SRFLD2	"System Requests panel (SR)" on page 189
SRCH	SRCHFLDS, SRCHFLD2	"Search panel (SRCH)" on page 184
ST	STFLDS, STFLD2	"Status panel (ST)" on page 188
SYM	SYMFLDS, SYMFLD2	"System Symbols panel (SYM)" on page 185
SYS	SYSFLDS, SYSFLD2	"System panel (SYS)" on page 186

Name tables (NTBL or ISFNTBL)

An NTBL statement along with NTBLENT statements, or an ISFNTBL macro, works in conjunction with an ISFGRP macro or GROUP statement in placing an SDSF user into a group, or in determining which SDSF functions are available to a member of a group.

In ISFPARMS assembler macros, the ISFNTBL macros must follow the ISFGRP macros.

Examples of the NTBL statement and ISFNTBL macro

NTBL and NTBLENT Statements	ISFNTBL Macro
1 GROUP TSOAUTH(JCL,OPER,ACCT), **2** XUID(XLIST) **3** NTBL NAME(XLIST) NTBLENT STRING($S),OFFSET(1) NTBLENT STRING(OPER),OFFSET(3)	**1** ISFGRP TSOAUTH=(JCL,OPER,ACCT), **2** XUID=XLIST **3** XLIST ISFNTBL $S,1,OPER,3

On line **1** of the example, the TSOAUTH parameter of the ISFGRP macro or GROUP statement defines a group of users with TSO authority of JCL, OPER, and ACCT.

On line **2**, the XUID parameter works with the ISFNTBL macro, or the combination of NTBL and NTBLENT statements, beginning on line **3**, to exclude from the user group any user with an ID that has the character string *$S* beginning in the first position, or the character string *OPER* beginning in the third position.

NTBL and NTBLENT Statements	ISFNTBL Macro
1 GROUP TSOAUTH(JCL), **2** PREFIX(USERID), **3** XSTATUS(EXCLUDE) **4** NTBL NAME(EXCLUDE) NTBLENT STRING(RSCS)	**1** ISFGRP TSOAUTH=(JCL), **2** PREFIX=USERID, **3** XSTATUS=EXCLUDE **4** EXCLUDE ISFNTBL RSCS,1

On line **1** of the example, the TSOAUTH parameter of the ISFGRP macro or GROUP statement defines a group of users with TSO authority of JCL. Line **2** defines which jobs will be included on SDSF panels.

On line **3**, the XSTATUS parameter works with the ISFNTBL macro, or the combination of NTBL and NTBLENT statements, beginning on line **4**, to exclude from the SDSF panels any job whose name begins with the characters *RSCS*. The OFFSET parameter is omitted and defaults to 1.

For more examples, see samples ISFPRM00 and ISFPRM01 in ISF.SISFJCL.

NTBL and ISFNTBL syntax

NTBL and NTBLENT Statements

NTBL NAME(*NTBL-statement-name*) TYPE(DEST)
NTBLENT STRING(*string*) OFFSET(*beginning-column-of-string*)

ISFNTBL Macro

label ISFNTBL *string*, *beginning-column-of-string*,... [,TYPE=DEST]

label **or** *NTBL-statement-name*
names the ISFNTBL macro or NTBL statement. The name must be 2-8 alphabetic, numeric, or national characters (@, #, $) and must begin with an alphabetic character.

string
is a character string.

If a character string contains blanks, it must be enclosed in single quotation marks.

beginning-column-of-string
is the beginning column number of the character string. In the NTBLENT statement, OFFSET(*beginning-column-of-string*) is optional. If it is omitted, *beginning-column-of-string* defaults to 1.

TYPE
is an optional parameter. The value of DEST indicates that this definition contains enhanced destination names. If you are using these longer destination names, you must specify the TYPE parameter, with a value of DEST.

Usage notes

If you code name tables for destination names, you may want to put the installation-defined destination names last. Installation-defined names may be most

likely to cause an error, and when SDSF encounters an error during initialization, it continues initialization with the destination names that were processed successfully before the error.

An ISFNTBL macro or NTBL statement can be referred to by the following parameters of one or more ISFGRP macros or GROUP statements:

DEST
ICMD
IDEST
IDSP
IDSPD
ILPROC
ISTATUS
ITNAME
IUID
XCMD
XDSP
XDSPD
XLPROC
XSTATUS
STNAME
XUID

Customized properties (PROPLIST)

A PROPLIST statement, along with PROPERTY statements, defines customized values for certain SDSF properties. It provides an alternative to writing user exit routines to customize those properties. A user exit routine that customizes the same property as a PROPERTY statement overrides the value on the PROPERTY statement.

The PROPLIST statement is associated with a group of users through the CUSTOM parameter on the GROUP statement.

The PROPLIST statement has no equivalent in ISFPARMS assembler macros.

Example of the PROPLIST and associated statements

PROPLIST and Associated Statements

```
1 GROUP NAME(DEPTA),
2   CUSTOM(USERPROP)
    .
    .
    .
3 PROPLIST NAME(USERPROP)
    4    PROPERTY NAME(Security.Browse.LogNOFAIL) VALUE(TRUE)
```

On line 2 of the example, the CUSTOM parameter refers to a PROPLIST statement with the NAME parameter.

The PROPLIST statement with the appropriate name begins on the line marked with **3** . It consists of one PROPERTY statement, on the line marked with **4** , which specifies the Security.Browse.LogNOFAIL property.

PROPLIST syntax

PROPLIST and PROPERTY statements

PROPLIST NAME(*proplist-statement-name*),
PROPERTY NAME(*property-name*) VALUE(value)

proplist-statement-name
names the PROPLIST statement referenced by the CUSTOM parameter in a GROUP statement. The name can be 1 to 8 alphabetic, numeric, or national characters (@, #, $) and must begin with an alphabetic or national character.

property-name
names the property. The properties are described in Table 31.

value
specifies the setting for the property.

Table 31 shows the properties that you can specify with the PROPERTY statement, and the corresponding flag that you could set in a user exit routine to achieve the same result. The user exit overrides the PROPERTY statement.

Table 31. Properties to Specify with the PROPERTY Statement

Name	Values	Description	Corresponding Field for User Exit
Browse.CoreBuf.NoSwap	TRUE or FALSE	Affects the browsing of job data sets. A value of TRUE requests that SDSF not attempt to gather data not yet written to spool if the job is swapped out. This is ignored for systems other than the one you are logged onto. FALSE is the default.	UPRSFLG3.UPRS3SWP
Browse.Suppress.DupDS	TRUE or FALSE	Controls whether duplicate SYSOUT data sets are included when you browse or print a job. A value of TRUE requests that duplicate SYSOUT data sets not be included. FALSE is the default.	UPROFLG3.UPRO3NOD

Table 31. Properties to Specify with the PROPERTY Statement (continued)

Name	Values	Description	Corresponding Field for User Exit
Comm.Release.Mode	1 or 2	Sets the mode that SDSF uses for communication to provide sysplex-wide data on SDSF panels. For more information, refer to "Using the server for sysplex data" on page 110. A value of 1 sets the communication mode to Z12, which requests that SDSF revert to using WebSphere MQ for communications if one or more systems is z/OS V1R12 or lower. Systems must be in the server group. A value of 2 sets the communication mode to Z13, which requests that SDSF use the sysplex support that was introduced in z/OS V1R13 SDSF. SDSF uses XCF for communications and does not use the server group. Systems that you wish to be included must be at least z/OS V1R13. This is the default.	UPRCMODE
Command.FILTER.SymbolsDisabled	TRUE or FALSE	Controls the use of system symbols with filtering. If the value is TRUE, any symbols in a string are not resolved. If the value is FALSE, symbols are resolved. FALSE is the default.	UPRS6FSY
Command.HOLD.AddGenChar	TRUE or FALSE	Affects the job name parameter on the H command. If the value is TRUE, SDSF appends a generic pattern-matching character to the job name specified with the H command, unless the job name already ends with a generic character or is already the maximum length (8 characters). For example, the command H GREER would result in H GREER*. FALSE is the default.	UPROFLG1.UPRO1GHO
Command.INIT. DefaultJESManaged	TRUE or FALSE	Controls the rows that are shown on the initiator panel by default. If the value is TRUE, only JES-managed initiators are shown by default. FALSE is the default.	UPROFLG2.UPRO2IDJ

Table 31. Properties to Specify with the PROPERTY Statement (continued)

Name	Values	Description	Corresponding Field for User Exit
Command.PREFIX.AddGenChar	TRUE or FALSE	Affects the PREFIX command. If the value is TRUE, SDSF appends a generic pattern-matching character to the prefix specified with the PREFIX command, unless the prefix already ends with a generic character or is already the maximum length (8 characters). For example, the command PREFIX JONES would result in a prefix of JONES*. FALSE is the default.	UPROFLG1.UPRO1GPF
Command.SLASH.Name	/, (or)	Specifies a single character to use when issuing system commands through SDSF (usually referred to as the slash command). You would use this character with all forms of the slash command, including I/ and W/. Enclose the character in single quotation marks, for example VALUE(')'). The default is /. This also affects the character used with the REXX ISFEXEC command. The REXX ISFSLASH command is preferred, as it does not require the character to be coded with the command.	UPRSLCMD UPRSLCIC UPRSLCWC
Command.STAT.AddGenChar	TRUE or FALSE	Affects the job name parameter on the ST command. If the value is TRUE, SDSF appends a generic pattern-matching character to the job name specified with the ST command, unless the job name already ends with a generic character or is already the maximum length (8 characters). For example, the command ST GREER would result in ST GREER*. FALSE is the default.	UPROFLG1.UPRO1GST
Console.EMCS.ConModChars	String of up to 32 characters consisting of A-Z, 0-9, @, #, $.	Names the list of suffixes to use when modifying the console name when the console activation fails due to the console being in use. The default is $#@12345.	UPXCONSF

Table 31. Properties to Specify with the PROPERTY Statement (continued)

Name	Values	Description	Corresponding Field for User Exit
Console.EMCS.CrossShare	TRUE or FALSE	A value of TRUE allows sharing of an EMCS console if it is in use but was activated in a different address space than the user. Console sharing means that commands will be issued using that console, and any responses will be directed to the ULOG for the task that has activated the console. A value of FALSE specifies that console can be shared only if it has been activated in the same address space as the user. The option to allowing sharing is effective only when console sharing is permitted. See Console.EMCS.NoShare. FALSE is the default.	UPRSFLG5.UPRS5CSX
Console.EMCS.DataSpaceSize	1 - 2048	Controls the size of the dataspace used when the EMCS console is activated. The data space size controls the number of messages that may be queued to the console prior to them being retrieved. The value indicates the size in megabytes. 2048 is the default.	UPRCONSZ
Console.EMCS.NoConMod	TRUE or FALSE (the default)	Disables modification of the console name when console activation fails due to the console being in use. A value of TRUE disables the function and a value of FALSE enables it. FALSE is the default.	UPROFLG2.UPRO2NMD
Console.EMCS.NoShare	TRUE or FALSE	A value of TRUE specifies that an EMCS console can be shared if it is already active. See Console.EMCS.CrossShare for controlling sharing of the EMCS console across address spaces. FALSE is the default.	UPRSFLAG.UPRSNOCS
Log.Operlog.ViewAll	TRUE or FALSE	Controls the lines shown on the OPERLOG panel. If the value is TRUE, the OPERLOG panel includes data from the inactive portion of the log stream. FALSE is the default.	UPROFLG2.UPRO2OVW

Table 31. Properties to Specify with the PROPERTY Statement (continued)

Name	Values	Description	Corresponding Field for User Exit
Log.Syslog.UseHaspIndx	TRUE or FALSE	Controls the use of the HASPINDX-based SYSLOG in a JES2 environment, when the logical SYSLOG is available. A value of TRUE indicates that the HASPINDX-based SYSLOG is to be used even though the logical SYSLOG is available. This value is intended to be used when a critical migration problem occurs using the logical log such that the HASPINDX-based SYSLOG is needed. IBM's intent is to eventually remove the HASPINDX-based SYSLOG when all systems support the logical log. FALSE is the default.	UPROFLG1.UPRO1HDX
Panel.All.JESPlexScope	TRUE or FALSE	Controls the scope of the APF, AS, CK, DA, DYNX, ENC, ENQ, LNK, LPA, PAG, PARM, PS, SYM, and SYS panels. If the value is TRUE, the scope of the panels is JESPlex-wide. If the value is FALSE, the scope of the panels is sysplex-wide. FALSE is the default.	UPROFLG3.UPRO3JPC, UPROFLG3.UPRO3JPD, UPROFLG3.UPRO3JPE, UPROFLG3.UPRO3JPP, UPROFLG4.UPRO4JAP, UPROFLG4.UPRO4JLN, UPROFLG4.UPRO4JLP, UPROFLG4.UPRO4JPA, UPROFLG4.UPRO4JPM, UPROFLG4.UPRO4JSM, UPROFLG4.UPRO4JSY, UPROFLG5.UPRO5JEN, UPROFLG5.UPRO5JAS, UPROFLG5.UPRO5JDY
Panel.APF.JESPlexScope	TRUE or FALSE	Controls scope of the APF panel. If the value is TRUE, the scope of the APF panel is JESPlex-wide. If the value is FALSE, the scope of the APF panel is sysplex-wide. FALSE is the default.	UPROFLG4.UPRO4JAP
Panel.AS.JESPlexScope	TRUE or FALSE	Controls scope of the AS panel. If the value is TRUE, the scope of the AS panel is JESPlex-wide. If the value is FALSE, the scope of the AS panel is sysplex-wide. FALSE is the default.	UPROFLG5.UPRO5JAS
Panel.CK.JESPlexScope	TRUE or FALSE	Controls the scope of the CK panel. If the value is TRUE, the scope of the CK panel is JESPLex-wide. If the value is FALSE, the scope of the CK panel is sysplex-wide. FALSE is the default.	UPROFLG3.UPRO3JPC

Table 31. Properties to Specify with the PROPERTY Statement (continued)

Name	Values	Description	Corresponding Field for User Exit
Panel.CKH.DefaultCKLim	1–999999	Sets the default maximum number of instances for a check for IBM Health Checker for z/OS that will be read from the logstream for the CKH panel. Users can override this with the SET CKLIM command. The default is 10.	UPRCKLIM
Panel.DA.CPUPctBasedLPAR	TRUE or FALSE	Affects normalization of the CPU% column on the DA panel. If the value is TRUE, the CPU% column is normalized using the LPAR value for CPU busy for the system. If the value is FALSE, the CPU% column is normalized with the MVS value for CPU busy for the system. The LPAR value takes into account several states related to PR/SM. The LPAR value requires RMF. If the LPAR value is not available, SDSF uses the MVS value to normalize the CPU% colunm. FALSE is the default.	UPRSFLG6.UPRS6DNL
Panel.DA.DynamiczAAPCols	TRUE or FALSE	Affects the display of columns on the DA panel. If the value is TRUE, SDSF includes the columns related to a zAAP only if a zAAP is defined in the set of systems being shown, and includes the columns related to a zIIP only if a zIIP is defined in the set of systems being shown. If the value is FALSE, the inclusion of the columns does not depend on whether the special processor is defined. FALSE is the default.	UPROFLG1.UPRO1DYZ
Panel.DA.JESPlexScope	TRUE or FALSE	Controls the scope of the DA panel. If the value is TRUE, the scope of the DA panel is JESPLex-wide. If the value is FALSE, the scope of the DA panel is sysplex-wide. FALSE is the default.	UPROFLG3.UPRO3JPD
Panel.DA.ShowTitleSIO	TRUE or FALSE	Affects the contents of the title line on the DA panel. If the value is TRUE, the system SIO rate is included, but the system zAAP use is not. If the value is FALSE, the SIO rate is omitted, and the system zAAP use is shown if a zAAP is defined on the local system. FALSE is the default.	UPRSFLG5.UPRS5DSI

Table 31. Properties to Specify with the PROPERTY Statement (continued)

Name	Values	Description	Corresponding Field for User Exit
Panel.DYNX.JESPlexScope	TRUE or FALSE	Controls scope of the DYNX panel. If the value is TRUE, the scope of the DYNX panel is JESPlex-wide. If the value is FALSE, the scope of the DYNX panel is sysplex-wide. FALSE is the default.	UPROFLG5.UPRO5JDY
Panel.ENC.JESPlexScope	TRUE or FALSE	Controls the scope of the ENC panel. If the value is TRUE, the scope of the ENC panel is JESPLex-wide. If the value is FALSE, the scope of the ENC panel is sysplex-wide. FALSE is the default.	UPROFLG3.UPRO3JPE
Panel.ENQ.JESPlexScope	TRUE or FALSE	Controls scope of the ENQ panel. If the value is TRUE, the scope of the ENQ panel is JESPlex-wide. If the value is FALSE, the scope of the ENQ panel is sysplex-wide. FALSE is the default.	UPROFLG5.UPRO5JEN
Panel.LNK.JESPlexScope	TRUE or FALSE	Controls scope of the LNK panel. If the value is TRUE, the scope of the LNK panel is JESPlex-wide. If the value is FALSE, the scope of the LNK panel is sysplex-wide. FALSE is the default.	UPROFLG4.UPRO4JLN
Panel.LPA.JESPlexScope	TRUE or FALSE	Controls scope of the LPA panel. If the value is TRUE, the scope of the LPA panel is JESPlex-wide. If the value is FALSE, the scope of the LPA panel is sysplex-wide. FALSE is the default.	UPROFLG4.UPRO4JLP
Panel.PAG.JESPlexScope	TRUE or FALSE	Controls scope of the PAG panel. If the value is TRUE, the scope of the PAG panel is JESPlex-wide. If the value is FALSE, the scope of the PAG panel is sysplex-wide. FALSE is the default.	UPROFLG4.UPRO4JPA
Panel.PARM.JESPlexScope	TRUE or FALSE	Controls scope of the PARM panel. If the value is TRUE, the scope of the PARM panel is JESPlex-wide. If the value is FALSE, the scope of the PARM panel is sysplex-wide. FALSE is the default.	UPROFLG4.UPRO4JPM

Table 31. Properties to Specify with the PROPERTY Statement (continued)

Name	Values	Description	Corresponding Field for User Exit
Panel.PR.DevNameAlwaysShort	TRUE or FALSE	Controls how device names are formatted on the PR panel. If the value is TRUE, the device names are shown in a shortened format. Otherwise, the name is shown with dots between subtypes. FALSE is the default.	UPROFLG2.UPRO2DF8
Panel.PS.JESPlexScope	TRUE or FALSE	Controls the scope of the PS panel. If the value is TRUE, the scope of the PS panel is JESPLex-wide. If the value is FALSE, the scope of the PS panel is sysplex-wide. FALSE is the default.	UPROFLG3.UPRO3JPP
Panel.PUN.DevNameAlwaysShort	TRUE or FALSE	Controls how device names are formatted on the PUN panel. If the value is TRUE, the device names are shown in a shortened format. Otherwise, the name is shown with dots between subtypes. FALSE is the default.	UPROFLG2.UPRO2DU8
Panel.RDR.DevNameAlwaysShort	TRUE or FALSE	Controls how device names are formatted on the RDR panel. If the value is TRUE, the device names are shown in a shortened format. Otherwise, the name is shown with dots between subtypes. FALSE is the default.	UPROFLG2.UPRO2DR8
Panel.Settings.DisablePointAndShoot	TRUE or FALSE	Controls the use of point-and-shoot fields on the SDSF primary option menu and the column titles of tabular panels. If the value is TRUE, the fields are not conditioned for point-and-shoot. FALSE is the default.	UPROFLG2.UPRO2PNS
Panel.SYM.JESPlexScope	TRUE or FALSE	Controls scope of the SYM panel. If the value is TRUE, the scope of the SYM panel is JESPlex-wide. If the value is FALSE, the scope of the SYM panel is sysplex-wide. FALSE is the default.	UPROFLG4.UPRO4JSM
Panel.SYS.JESPlexScope	TRUE or FALSE	Controls scope of the SYS panel. If the value is TRUE, the scope of the SYS panel is JESPlex-wide. If the value is FALSE, the scope of the SYS panel is sysplex-wide. FALSE is the default.	UPROFLG4.UPRO4JSY

Table 31. Properties to Specify with the PROPERTY Statement (continued)

Name	Values	Description	Corresponding Field for User Exit
Print.CCTL.AlwaysUseASA	TRUE or FALSE	Specifies how SDSF's print function handles carriage control. A value of TRUE causes SDSF to always use ASA carriage control when printing, regardless of the record format of the output data set. A value of FALSE causes SDSF to handle carriage control based on the record format of the output, as follows: • If the record format includes A, then the print function uses ASA (ANSI) carriage control. • If the record format includes M, then the print function uses machine carriage control. • Otherwise, SDSF removes carriage control characters if they are present in the source. TRUE is the default.	
Security.Browse.LogNOFAIL	TRUE or FALSE	Specifies the SAF logging option to use when a job's data sets are browsed from an SDSF panel, with the exceptions of the JDS and OD panels. If the value is TRUE, the SAF logging setting is LOG=NOFAIL (rather than the default, LOG=ASIS). FALSE is the default.	UPROFLG1.UPRO1LNF
Security.SAFNoDec.WarnMsg	TRUE or FALSE	Specifies the SAF no-decision option in a JES3 environment. If the value is TRUE, an SDSF message is issued whenever a SAF no-decision result (return code 04) is converted to a failure. The message includes the class name, resource name and access level being checked. This setting can be helpful during a conversion period; once you have defined the SAF profiles, set the value to FALSE. FALSE is the default.	UPROFLG1.UPRO1SFW
Security.Syslog.UseSAFRecvr	TRUE or FALSE	Controls the use of RECVR when processing the logical SYSLOG. A value of TRUE indicates that a RECVR equal to the current user ID will be used when the logical SYSLOG is opened. This causes the authorization check to the logical SYSLOG to always succeed (see note). FALSE is the default.	UPROFLG1.UPRO1RCV

Table 31. Properties to Specify with the PROPERTY Statement (continued)

Name	Values	Description	Corresponding Field for User Exit
Note: The resource is *nodeid*.+MASTER+.SYSLOG.SYSTEM.*sysname*.			

Code page (TRTAB/TRDEF or ISFTR)

A TRTAB statement or ISFTR macro specifies the code page that SDSF uses for a group of users. SDSF uses the code page to ensure that it displays valid characters on the terminal and to convert lowercase characters to uppercase.

A code page consists of two translation tables. One table contains the character set that is valid for a group of users and the other contains the uppercase characters for that character set. SDSF folds all input data, such as action characters, to uppercase and verifies all the data it displays, such as field titles, for valid characters. If SDSF encounters a character that is not in the valid character set table, it displays that character as a blank.

The code page you specify does not apply to the pull-downs and pop-ups displayed by ISPF. For them, ISPF uses the code page defined for the terminal type currently in effect.

If none of the code pages that can be specified with the CODPAG parameter match the needs of your installation, you can code your own translation tables in ISFPARMS or in your statements. See “Coding a translate table” on page 105 for more information.

The ISFTR macro in ISFPARMS must follow all ISFGRP macros.

Examples of the TRTAB statement and ISFTR macro

TRTAB Statement	ISFTR Macro
No TRTAB coded	ISFTR

This example shows the minimum coded parameters for the ISFTR macro and the TRTAB statement. The ISFTR macro includes no parameters. THE TRTAB statement is omitted altogether. In this case, the VALTAB and UPCTAB parameters are not coded here, nor in the group definitions (ISFGRP macros or GROUP statements). All SDSF users are assigned the default code page, SDSF.

TRTAB Statement	ISFTR Macro
1 GROUP TSOAUTH(JCL,OPER,ACCT), 2 AUTH(LOG,I,O,H,DA,DEST,PREF, 3 ACTION,INPUT,ST,INIT,PR,SE), 4 CMDAUTH(ALL), 5 CMDLEV(7), 6 DSPAUTH(ALL), 7 VALTAB(VAL500), 8 UPCTAB(UPC500) 9 GROUP TSOAUTH(JCL), 10 AUTH(I,O,H,DA,ST), 11 CMDAUTH(USERID,NOTIFY), 12 CMDLEV(2), 13 DSPAUTH(USERID,NOTIFY), 14 PREFIX(USERID) 15 TRTAB CODPAG(CP00500),VALTAB(VAL500), 16 UPCTAB(UPC500) 17	1 ISFGRP TSOAUTH=(JCL,OPER,ACCT), 2 AUTH=(LOG,I,O,H,DA,DEST,PREF, 3 ACTION,INPUT,ST,INIT,PR,SE), 4 CMDAUTH=(ALL), 5 CMDLEV=7, 6 DSPAUTH=(ALL), 7 VALTAB=VAL500, 8 UPCTAB=UPC500 9 ISFGRP TSOAUTH=(JCL), 10 AUTH=(I,O,H,DA,ST), 11 CMDAUTH=(USERID,NOTIFY), 12 CMDLEV=2, 13 DSPAUTH=(USERID,NOTIFY), 14 PREFIX=USERID 15 ISFTR CODPAG=CP00500,VALTAB=VAL500, 16 UPCTAB=UPC500 17 ISFTR

On line **1** of the example, the TSOAUTH parameter defines a group of users with TSO authority of JCL, OPER, and ACCT.

On lines **2**, **3**, **4**, **5**, and **6** the parameters define authorization levels for members of the group.

On line **7**, the VALTAB parameter specifies VAL500 as the name of the translation table that checks for valid characters.

On line **8**, the UPCTAB parameter specifies UPC500 as the name of the translation table that converts lowercase characters to uppercase.

The translation tables are generated by an ISFTR macro or TRTAB statement that has VALTAB and UPCTAB parameters that name the same translation tables, which is found on line **15**. The CODPAG parameter specifies the code page, CP00500, that is to be used fo the group of users.

On line **9** of the example, the TSOAUTH parameter defines a group of users with TSO authority of JCL. It includes no VALTAB or UPCTAB parameters. This tells SDSF to use the default code page, SDSF. Lines **10** through **14** define authority.

On line **17** in the assembler example is the ISFTR macro with no parameters. This assigns the default code page, SDSF, which will be used with the second group of users. Assigning the default code page in this manner is not required with statements.

TRTAB and ISFTR syntax

TRTAB Statement

TRTAB CODPAG (*code-page*),
VALTAB (*valid-character-translation-table-name*),
UPCTAB (*uppercase-translation-table-name*)

ISFTR Macro

ISFTR CODPAG=*code-page*,
VALTAB= *valid-character-translation-table-name*
UPCTAB= *uppercase-translation-table-name*

CODPAG

Specifies an alternate code page, *code-page*, that SDSF will use for a group of users. The valid character and uppercase translation tables generated by SDSF correspond to the CODPAG you specify.

If you omit this parameter, SDSF uses code page **SDSF** (or CP00037, when running SDSF in batch with program name ISFAFD).

code-page can be:

SDSF USA WP, Original.

SDSF consists of CP00001 plus three optical character reader (OCR) characters, which results in mixed-case characters in the help and tutorial panels, SDSF panels, and the SDSF Primary Option menu.

CASE Same as SDSF, but characters are folded to uppercase.

CP00037
USA/Canada – CECP

CP00273
Germany F.R./Austria – CECP

CP00275
Brazil – CECP

CP00277
Denmark, Norway – CECP

CP00278
Finland, Sweden – CECP

CP00280
Italy – CECP

CP00281
Japan (Latin) – CECP

CP00284
Spain/Latin America – CECP

CP00285
United Kingdom – CECP

CP00290
Japanese (Katakana) Extended

CP00297
France – CECP

CP00420
Arabic, Bilingual

CP00424
Israel (Hebrew) Extended

CP00500
International #5

CP00803
Hebrew Character Set A

CP00833
Korean Extended

CP00836
Simplfied Chinese Extended

CP00870
Latin 2/Multilingual/ROECE

CP00871
Iceland – CECP
CP00875
Greece
CP01025
Cyrillic, Multilingual
CP01026
Latin 5/Turkey
CP01027
Japanese (Latin) Extended
CP01047
Latin 1/Open systems
CP01112
Baltic, Multilingual
CP01122
Estonia
CP01140
ECECP USA, Canada, Netherlands, Portugal, Brazil, Australia, New Zealand
CP01141
ECECP Austria, Germany
CP01142
ECECP Denmark, Norway
CP01143
ECECP Finland, Sweden
CP01144
ECECP Italy
CP01145
ECECP Spain, Latin America (Spanish)
CP01146
ECECP UK
CP01147
ECECP France
CP01148
ECECP Belgium, Canada, Switzerland
CP01149
ECECP Iceland
CP01153
EBCDIC Latin 2 Multilingual with Euro Extended
CP01159
T-Chinese EBCDIC

VALTAB
Specifies the name of the valid character set translation table. If omitted, SDSF uses TRTAB for the name. TRTAB cannot be used as a default name more than once.

Use the same value for *valid-character-translation-table-name* that you used in the VALTAB parameter of the ISFGRP macro or GROUP statement for the group. If you have more than one ISFTR macro in ISFPARMS, you must use a unique name for each *valid-character-translation-table-name*.

UPCTAB
Specifies the name of the uppercase translation table. If omitted, SDSF uses TRTAB2 for the name. TRTAB2 cannot be used as a default name more than once.

Use the same value for *uppercase-translation-table-name* that you used in the UPCTAB parameter of the ISFGRP macro or GROUP statement for the group. If you have more than one ISFTR macro in ISFPARMS, you must use a unique name for each *uppercase-translation-table-name*.

Coding a translate table

To code your own translate table, use the VALTAB and UPCTAB parameters of an ISFGRP macro or GROUP statement to assign the translate tables to a group of users. Then, if you are using ISFPARMS assembler macros, code the translate table in the ISFPARMS module, after the ISFGRP macros. If you are using statements, define the translate table with the TRDEF statement.

The translate tables must be 256 bytes each.

TRDEF syntax

TRDEF Statement

TRDEF NAME(*table-name*),
DATA(*hex-characters*)

NAME(*table-name***)**
names the translate table being defined. The name is referenced in the UPCTAB or VALTAB parameter of a GROUP statement.

DATA(*hex-characters***)**
specifies the translate table, which must be 256 bytes.

Example of the TRDEF statement

```
1  GROUP TSOAUTH(JCL,OPER,ACCT),
          AUTH(LOG,I,O,H,DA,DEST,PREF,SYSID,
          ACTION,INPUT,FINDLIM,ST,INIT,PR),
          CMDAUTH(ALL),
          CMDLEV(7),
          DSPAUTH(ALL),
2         VALTAB(UVALTAB),
3         UPCTAB(UUPCTAB)
4  TRDEF NAME(UVALTAB),    /* Valid character table */
5        DATA(000102030405060708090A0B0C0D0E0F,    /* 00-0F */
             101112131415161718191A1B1C1D1E1F,    /* 10-1F */
             202122232425262728292A2B2C2D2E2F,    /* 20-2F */
             303132333435363738393A3B3C3D3E3F,    /* 30-3F */
             404142434445464748494A4B4C4D4E4F,    /* 40-4F */
             505152535455565758595A5B5C5D5E5F,    /* 50-5F */
             606162636465666768696A6B6C6D6E6F,    /* 60-6F */
             707172737475767778797A7B7C7D7E7F,    /* 70-7F */
             808182838485868788898A8B8C8D8E8F,    /* 80-8F */
             909192939495969798999A9B9C9D9E9F,    /* 90-9F */
             A0A1A2A3A4A5A6A7A8A9AAABACADAEAF,    /* A0-AF */
             B0B1B2B3B4B5B6B7B8B9BABBBCBDBEBF,    /* B0-BF */
             C0C1C2C3C4C5C6C7C8C9CACBCCCDCECF,    /* C0-CF */
             D0D1D2D3D4D5D6D7D8D9DADBDCDDDEDF,    /* D0-DF */
             E0E1E2E3E4E5E6E7E8E9EAEBECEDEEEF,    /* E0-EF */
             F0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFF)    /* F0-FF */
6  TRDEF NAME(UUPCTAB),    /* Upper case table */
7        DATA(000102030405060708090A0B0C0D0E0F,    /* 00-0F */
             101112131415161718191A1B1C1D1E1F,    /* 10-1F */
             202122232425262728292A2B2C2D2E2F,    /* 20-2F */
             303132333435363738393A3B3C3D3E3F,    /* 30-3F */
             404142434445464748494A4B4C4D4E4F,    /* 40-4F */
             505152535455565758595A5B5C5D5E5F,    /* 50-5F */
             606162636465666768696A6B6C6D6E6F,    /* 60-6F */
             707172737475767778797A7B7C7D7E7F,    /* 70-7F */
```

```
8081828384858687888 98A8B8C8D8E8F,    /* 80-8F */
9091929394959697989 99A9B9C9D9E9F,    /* 90-9F */
A0A1A2A3A4A5A6A7A8A 9AAABACADAEAF,    /* A0-AF */
B0B1B2B3B4B5B6B7B8B 9BABBBCBDBEBF,    /* B0-BF */
C0C1C2C3C4C5C6C7C8C 9CACBCCCDCECF,    /* C0-CF */
D0D1D2D3D4D5D6D7D8D 9DADBDCDDDEDF,    /* D0-DF */
E0E1E2E3E4E5E6E7E8E 9EAEBECEDEEEF,    /* E0-EF */
F0F1F2F3F4F5F6F7F8F 9FAFBFCFDFEFF)    /* F0-FF */
```

On the line marked with **1**, a GROUP statement begins the definition of a group. On the line marked with **2**, the VALTAB parameter gives the valid character translation table the name UVALTAB. On the line marked with **3**, the UPCTAB parameter gives the uppercase translation table the name UUPCTAB. The names UVALTAB and UUPCTAB are used to associate these parameters with TRDEF statements on lines **4** and **6**. The valid character translate table is defined beginning on line **5**. The uppercase translate table is defined beginning on line **7**.

Chapter 3. Using the SDSF server

The SDSF server is an address space that SDSF uses to:

- Process ISFPARMS statements.
- Provide sysplex support. This consists of sysplex-wide data for JES2 devices and for system resources (CK, ENC, INIT, LI, NO, PR, PS, PUN, RDR, RM and SO panels) as well as the most recent SYSLOG data for remote systems (SYSLOG panel). For more information, refer to "Using the server for sysplex data" on page 110.
- Manage the starting and stopping of the SDSFAUX address space. SDSFAUX is used to provide data gathering support and other services for SDSF panels.

To process ISFPARMS, the server must be active on each system that contains SDSF users. To provide sysplex data, the server must be active on each system that is to be included on SDSF panels. Use the WHO command or pop-up to verify that the server is in use.

Multiple SDSF servers may be run on the same system; however, you must assign them unique names. Only one server with a particular name can be active on the system. The level of the server must match the level of the SDSF application.

Note: There can be only a single SDSFAUX address space active in the system. If a second SDSF server tries to start SDSFAUX, the start is rejected because SDSFAUX is already active. However, the active SDSFAUX is available from any SDSF server in the same system.

You control the server through the MVS operator START, STOP, and MODIFY commands. (The START command names the server; the MODIFY command refreshes the ISFPARMS statements, changes server options, and displays and controls server communications.) For details on the commands, see "Server operator commands" on page 118.

Sample JCL for the server is in member ISFSRJCL in SDSF's SISFJCL data set.

Sample JCL for SDSFAUX is in member HSFSRJCL of data set ISF.SISFJCL.

Note: SDSFAUX uses services to load modules that require SISFLOAD to be present in the system lnklst. If SISFLOAD is not in the lnklst, SISFLOAD must be added as a //STEPLIB in the SDSFAUX JCL procedure and APF authorized.

When SISFLOAD is in the lnklst and LNKAUTH=APFTAB, then SISFLOAD must be added to the APF list.

Defining the input

The input to the SDSF server is the ISFPARMS statements. By default, SDSF assumes the statements reside in PARMLIB, in member ISFPRM00. You can use a PARMLIB member with a different suffix by specifying that suffix on the command you use to start the server. See "Start the SDSF server" on page 118. Or you can use your own partitioned data set, rather than PARMLIB, by defining it using ddname SDSFPARM in the server JCL.

© Copyright IBM Corp. 1997, 2016

For details on defining the ISFPARMS statements, see Chapter 2, “Using ISFPARMS for customization and security,” on page 15.

Starting the server

You start the server using the START command. The command takes the server name as a parameter. Optional parameters identify the suffix of PARMLIB member ISFPRM*xx* that contains the statements to be read, as well as other options. For details, see “Start the SDSF server” on page 118.

Starting the SDSFAUX server

The SDSFAUX address space is automatically started by the SDSF server address space when the server starts. Conversely, SDSFAUX is automatically stopped when the SDSF server is stopped.

Perform the following steps:

1. Ensure that the SAF SDSF class is RACLISTed. SDSFAUX fails all authorization requests if the SAF SDSF class is not RACLISTed. For more information on RACLIST, see Chapter 6, “SDSF and RACF,” on page 205.
2. Define a user ID associated with the SDSFAUX started task by adding a profile in the SAF STARTED class. You can use the same user ID you used for the SDSF server. For example, the profile SDSF*.** applies to both the SDSF and SDSFAUX started tasks.

   The SDSF server user ID must have access to the appropriate OPERCMDS resources so that the START and STOP operator commands can be issued to start and stop the SDSFAUX address space.

Keep the following additional considerations in mind:

- By default, the SDSF server starts SDSFAUX with the S SDSFAUX operator command. You can change the SDSFAUX procedure and job names, or not start SDSFAUX by default, using the AUXPROC and AUXNAME keywords of the CONNECT statement as described in the “CONNECT statement” on page 34.
- If SDSFAUX is already active, any changes to parameters related to SDSFAUX on the CONNECT statement such as AUXPROC, AUXNAME, and AUXSAF are ignored. If you make changes to the CONNECT statement related to SDSFAUX, stop the SDSF server and wait for SDSFAUX to end. Then, restart the SDSF server for the changes to take effect.
- If SDSFAUX does not automatically start or stop, you can use the START and STOP commands, respectively. For example, S SDSFAUX starts SDSFAUX and P SDSFAUX stops it. In addition, SDSFAUX is also started when you refresh ISFPRM*xx*, such as with the F SDSF,REFRESH command.

Processing the statements

When the server is started, it reads the statements from the input data set.

You can activate new parameters at any time with the MODIFY command, which you can enter from the console or from SDSF by users that are authorized to use the slash (/) command. Changes take effect the *next* time users access SDSF. A TEST parameter allows you to check the syntax of the statements without activating them. See “Refresh ISFPARMS” on page 127 for more information.

A server supports a single ISFPARMS. To run multiple levels of ISFPARMS, you must have a unique server for each level. You may, for example, have a unique ISFPARMS corresponding to each JESNAME. You access a specific ISFPARMS by using the SERVER keyword on the SDSF command. By controlling access to the server, you can control the statements that apply to the user. See "Accessing the server" for more information. Note that conditional processing and support for system symbols reduces the need for multiple levels of ISFPARMS. See "Conditional processing" on page 20 for details.

Accessing the server

Note: SDSF V2R1 and higher makes use of 64-bit memory wherever possible. If you use the z/OS default of 2GB for all address spaces, then no action is required. If you have set a MEMLIMIT default for TSO users and batch jobs that is below 512MB, consider increasing the value to avoid any problems relating to SDSF use of 64-bit memory.

When the user accesses SDSF, the SDSF client attempts to connect to the SDSF server. The server that the client connects to is known as the *local* server. The following determine which server the client connects to, in this order:

1. The SERVER parameter of the SDSF command used to invoke SDSF (described below)
2. The SERVER keyword on the ISFPMAC macro of ISFPARMS.
3. The default server as defined by the CONNECT statement in ISFPARMS. See "CONNECT statement" on page 34.

If none of those names the server to be used, the client attempts to connect to server SDSF.

The user can request a specific server by adding a SERVER parameter to the SDSF command that invokes SDSF. The format of the server parameter is `SERVER(`*server-name*`)`. For example, you might enter

- `SDSF SERVER(SDSFT)` from the TSO READY prompt or
- `S.SERVER(SDSFT)` from the ISPF command line.

When you use the SERVER parameter with the SDSF command, you cannot add any other SDSF commands. To use the SERVER parameter you must have READ access to the ISFCMD.OPT.SERVER SAF resource in the SDSF class. (The SERVER parameter can be protected only through SAF, not ISFPARMS.)

Logging

The SDSF server logs all statements processed, and any associated error messages, to a log file. With the server START command, you can control the destination of the log file (SYSOUT or the hardcopy log). When the destination is SYSOUT, SDSF uses the class specified in the server JCL if one is specified there, or the class specified in the LOGCLASS option on the START command. If no SYSOUT class is specified, SDSF uses class A. When SDSF dynamically allocates the log, it is freed when it is closed. In the event of an error allocating the log, SDSF redirects any log messages to the hardcopy log. Messages issued by the server are documented in Chapter 15, "SDSF messages and codes," on page 475.

The SDSFAUX log is written to the HSFLOG data set allocated by the SDSFAUX address space. It contains messages related to SDSFAUX processing for use by IBM service personnel.

Security

Security for the SDSF server is provided with SAF resources. You can protect these aspects of the server related to processing ISFPARMS statements:

- Use of the SERVER parameter on the SDSF command, which specifies a server name that overrides the default server name defined in ISFPARMS.
- Reverting from ISFPARMS in statement format to ISFPARMS in assembler macro format, when the server is not available or no ISFPARMS statements are defined.
- Use of the server operator commands.

For details on these aspects of server security, see "SDSF server" on page 300.

If you are using the server to provide sysplex data, you must also protect the WebSphere MQ queues used by SDSF. For details, see "WebSphere MQ" on page 306.

SDSF's sysplex support provided by the SDSF server and WebSphere MQ cannot be used in the multilevel security environment. If you want to implement multilevel security and are already using this sysplex support, you should disable the sysplex support by removing the server group definitions from ISFPARMS. For more information on multilevel security, see *z/OS Planning for Multilevel Security and the Common Criteria*.

Using the server for sysplex data

The SDSF server is used to provide sysplex-wide data on SDSF panels.

Device panels

These device panels are sysplex-wide by default in all supported environments: INIT, LI, NO, PR, PUN, RDR and SO.

The SDSF server is never required for sysplex-wide data in a JES3 environment.

Sysplex-wide panels

An SDSF server is required on each system for sysplex-wide CK, ENC, PS and RM panels, in both JES2 and JES3 environments. Similarly, an SDSF server with the SDSFAUX address space started is required on each system for sysplex-wide APF, AS, DYNX, ENQ, LNK, LPA, PAG, PARM, SYM, and SYS panels.

When one or more systems that you want to include is at the z/OS V1R12 level, the server group defined in ISFPARMS is also required, along with WebSphere MQ. Additional configuration may be required. Refer to "Using z/OS V1R12 compatibility mode" on page 111.

When all systems that you want to include are at the z/OS V1R13 level, SDSF uses XCF to communicate between SDSF servers, and does not use a server group defined in ISFPARMS. XCF communication between SDSF servers requires a common server name for all systems. If all of your SDSF servers do not have the same name, you can use the XCFSRVNM parameter on the CONNECT statement in ISFPARMS to meet this requirement. For more information, refer to "Server

connection (CONNECT)" on page 34. You can exclude a system from a sysplex-wide panel by specifying XCFSRVNM(NONE) on the CONNECT statement in the ISFPARMS for that system.

The APF, AS, DYNX, ENQ, LNK, LPA, PAG, PARM, SYM, and SYS panels are available only in the SDSF V2R1 environment and above. Sysplex-wide data is not shown for systems below the V2R1 level.

Using z/OS V1R12 compatibility mode

When one or more systems that you want to include is at the z/OS V1R12 or lower level, you must obtain sysplex-wide data for the CK, ENC, PS and RM panels as in z/OS V1R12 SDSF and earlier releases. This requires an SDSF server on each system, a server group defined in ISFPARMS and WebSphere MQ for communication between servers. Perform these steps:

1. Configure SDSF's sysplex support as described in "Servers with server groups and WebSphere MQ."
2. If both the system you log on to and at least one other system are at the z/OS V1R12 or lower level, request that SDSF run in Z12 compatibility mode, which causes SDSF to revert to using the server group and WebSphere MQ for communications. To request Z12 compatibility mode:

Who	Method	Details
Users	Command: `SET CMODE Z12`	Refer to the online help.
System programmers	Custom property in ISFPARMS: Comm.Release.Mode	"Customized properties (PROPLIST)" on page 91

If you do not set the compatibility mode to Z12 and both the system you log on to and at least one other system are at the z/OS V1R13 level, then SDSF will use XCF for communications and any lower level systems will be omitted.

SYSLOG panel

Displaying the latest data (data not yet written to spool) on the SYSLOG panel when you browse the SYSLOG for a remote system that is z/OS V1R11 JES2 requires SDSF servers with WebSphere MQ and z/OS V1R12 compatibility mode. Refer to "Using z/OS V1R12 compatibility mode" for details.

Websphere MQ and z/OS V1R12 compatibility mode are also required if you have customized SDSF to not use the logical log. Refer to "SYSLOG" on page 349 for details about the logical log.

Servers with server groups and WebSphere MQ

A server group is the group of SDSF servers that the local SDSF server communicates with, using WebSphere MQ, to provide sysplex data. All SDSF servers must be in the same sysplex, and all associated JESes must be in the same MAS. Communication between servers with WebSphere MQ is described in "Server communications with WebSphere MQ" on page 116.

In defining a server group, you identify the systems and their related primary or secondary JESes that will participate in a sysplex-wide request. Each different combination of systems and JESes requires a separate instance of the SDSF server. For details on defining a server group, see "Server group definition parameters (SERVERGROUP, SERVER, COMM)" on page 30.

Note that if a server group is not defined, SDSF does not use the server and WebSphere MQ to gather the data.

The following figures illustrate some server groups.

For details on the statements you use to define a server group, see “Server group definition parameters (SERVERGROUP, SERVER, COMM)” on page 30.

Example — two primary JES2s

Figure 1 shows a sysplex with MVS systems SY1 and SY2. An SDSF user logs on to SY1, invokes SDSF and is connected to the server SDSF. This is the user's local server. The server processes ISFPARMS, which contains a server group definition consisting of the two servers named SDSF, shown with shading. These servers each gather data for JES2 subsystems name JES2. The user’s panels will show data from the two JES2s.

Other SDSF servers are running on these systems, each named SDSFA. These servers, which gather data for alternate JES2s, are not part of the server group. Data from the alternate JES2s will not be included on the user's panels.

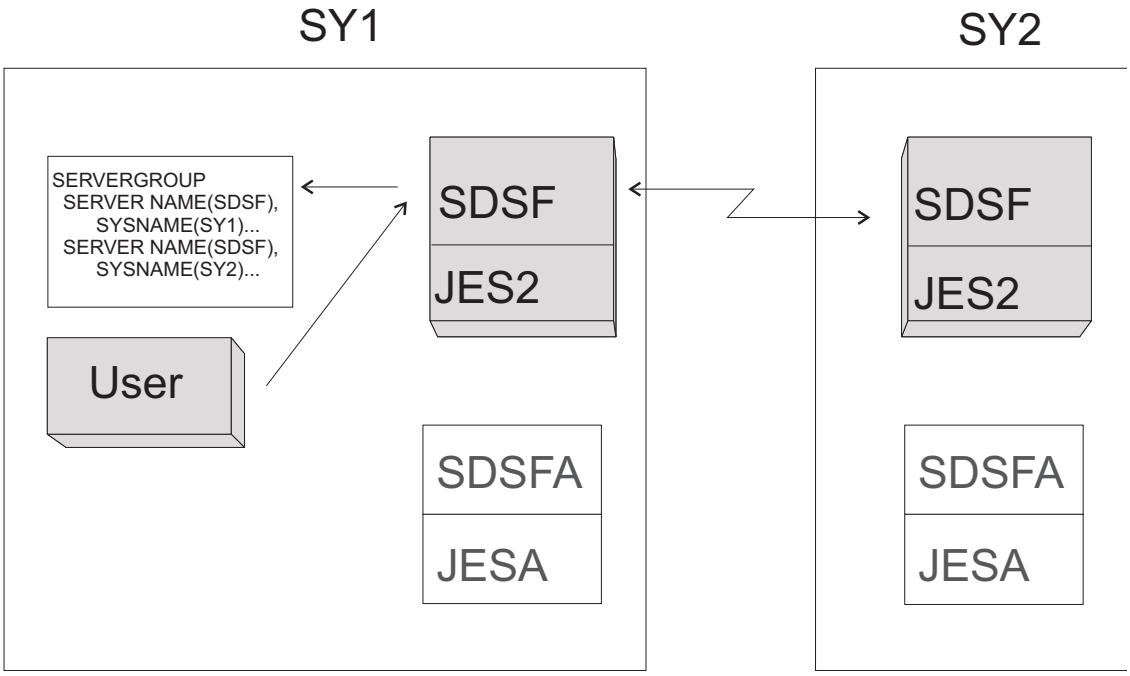


Figure 1. Server Group Example

Example — two alternate JES2s

Figure 2 on page 113 shows the same sysplex. This time, the SDSF user is connected to the SDSF server SDSFA that collects data for the alternate JES. (The user might, for example, have invoked SDSF with the command `s.server(sdsfa)`.) The server group defined in the ISFPARMS processed by that server consists of the two servers named SDSFA. The user’s panels will show data from the two JESAs.

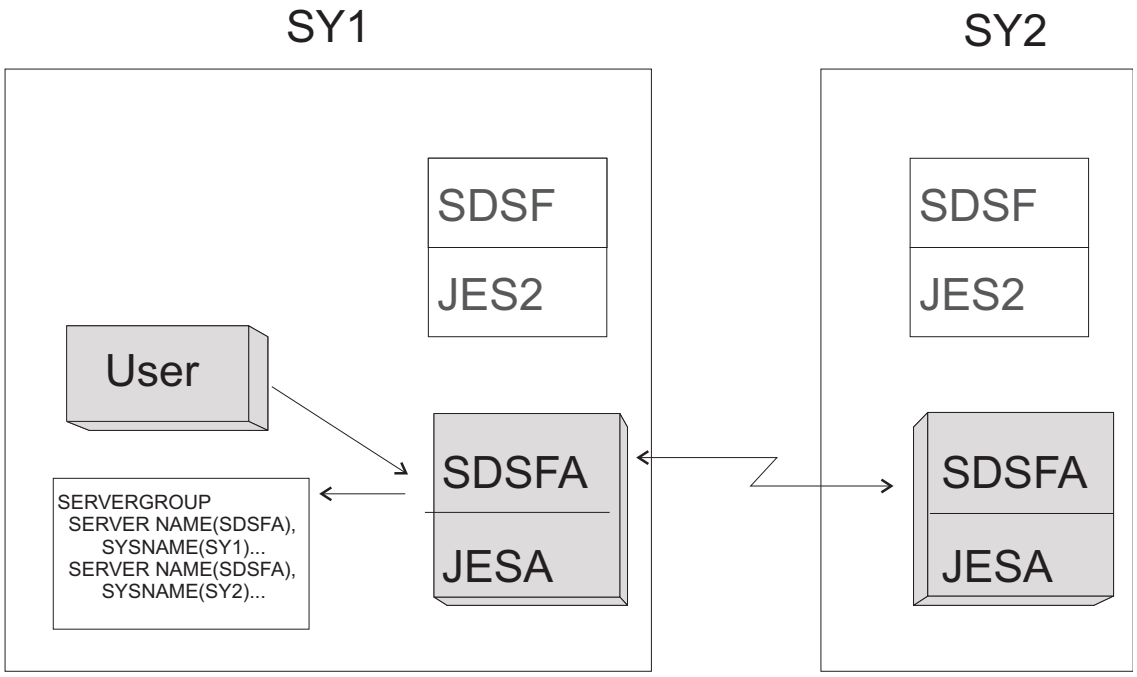


Figure 2. Server Group Example–Alternate JES2s

Example — all JES2s

Figure 3 on page 114 shows the same sysplex. All SDSF servers are in the sysplex, and all associated JESes are in the same MAS. The SDSF user is connected to the SDSF server SDSF. The server group defined in the ISFPARMS processed by that server is made up of all four servers. The user's panels will show data from the two primary and alternate JESs.

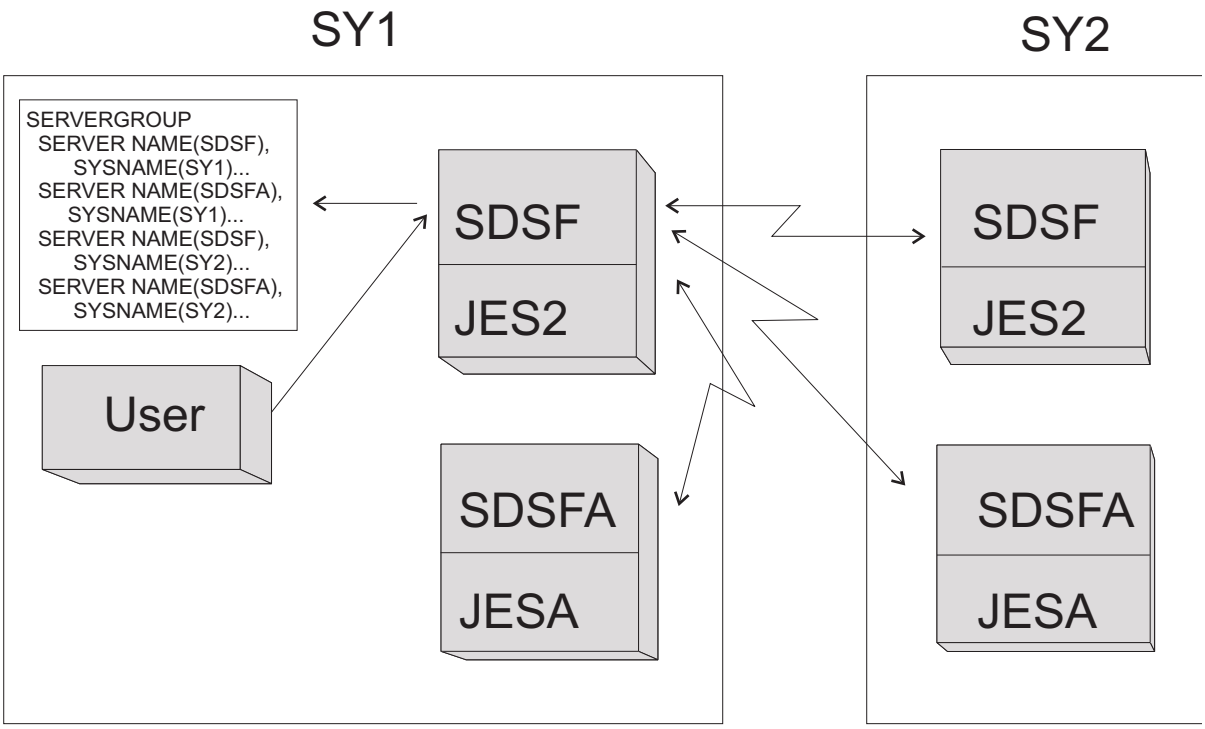


Figure 3. Server Group Example–All Servers

Example — server group on SY2

Note the server groups defined for each server are independent of each other. In the previous example, server SDSF3 on SY1 and server SDSF on SY2 are in the same server group. However, server SDSF on SY2 may have a different server group defined in the ISFPARMS it processes than server SDSF3 has in the ISFPARMS it processes. Figure 4 on page 115 shows a user logged on to SY2 and connected to server SDSF there, which processes an ISFPARMS with a server group consisting of three servers. All SDSF servers are in the sysplex, and all associated JESes are in the same MAS.

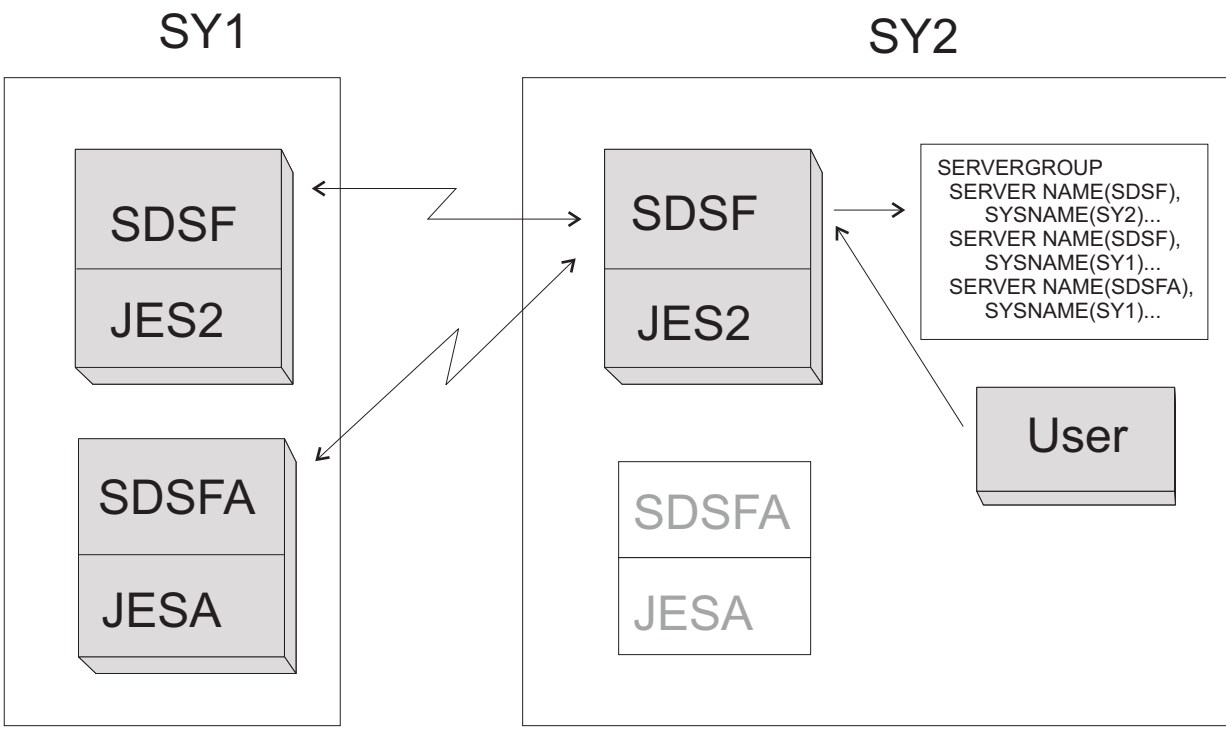


Figure 4. Server Group Example—Server Group on SY2

Adding or modifying a server group

About this task

If a server group has been defined and is in use, use the following procedure to add or change the server group:

1. Make the change to ISFPARMS.
2. End server communications with the `MODIFY server-name,STOP,C,TERM` command.
3. Use the `MODIFY server-name,REFRESH` command to cause the new ISFPARMS to be processed.

If no server group was defined when the server was started, you can:

1. Add a server group to the ISFPARMS .
2. Use the `MODIFY server-name,REFRESH` command to refresh the statements and add the group.

Alternatively, you could, after modifying the ISFPARMS, restart the server with the START command.

SDSF's syntax checking of initialization statements does not check for the presence of a server group's servers or systems. This means that you can use the initialization statements to define a server group, and start the server, even if the server group you have defined includes servers or systems that are not yet present. You can then start the servers or systems at a later time.

Server communications with WebSphere MQ

To provide sysplex data, SDSF servers can communicate through WebSphere MQ, which must be up and operational on each system in the server group. The WebSphere MQ queue managers must be configured to communicate using channels, as described in *WebSphere MQ Intercommunication*, SC33–1872. This section describes the WebSphere MQ queues used by SDSF. For information on configuration tasks required to use WebSphere MQ with SDSF, see "WebSphere MQ considerations" on page 344. Server communication for sysplex data does not apply in the JES3 environment.

Queues used by SDSF:
SDSF uses several WebSphere MQ queues, including:

- A model queue, which is used in creating other queues. This is defined by the SDSF server.
- Temporary, dynamic queues used to communicate between the SDSF server and the user. These are created dynamically by WebSphere MQ.

The queues are shown in Table 121 on page 307.

Table 32. WebSphere MQ Queues Used by SDSF

		Required Access	
Queue	**Queue Name**	**Server**	**Client**
Server request queue	*queue-prefix*.SERVER.*server.system*.REQUESTQ	Yes	None
Client request queue, used to send work to the server, and to send work from the server to remote servers	*queue-prefix*.CLIENT.*server.system*.REQUESTQ	Yes	Put only
ReplyTo queue, used by the client to receive server responses	*queue-prefix*.USER.*userid*.* (the final qualifier is a unique string generated by WebSphere MQ)	Yes	Yes
Model queue, used to create dynamic queues	*queue-prefix*.MODEL.QUEUE	Yes	Yes

SDSF uses both a server and a client request queue. The client request queue is actually an alias for the server request queue. The two queues work together so that SDSF users can put requests on the server queue, but cannot read the queue. The client request queue is defined by SDSF so that users' reading of the queue is prohibited by WebSphere MQ.

The WebSphere MQ DEFINE commands that the SDSF server uses to define the model queue (and the client request queue) are logged in the server log. See "Logging" on page 109 for details.

It is possible, through the use of QDEFINE(NO) on the COMM statement in ISFPARMS, to request that SDSF not dynamically define the model queue and client request queue at initialization. In this case, the queues must already exist. See "COMM statement" on page 32 for more information.

Defining queues:
SDSF uses queues provided by WebSphere MQ to create a temporary ReplyTo queue and to submit the WebSphere MQ DEFINE commands that define queues. The queues used in this process are shown in Table 33 on page 117.

Table 33. Queues Used to Define Queues

Queue	Queue Name	Required Access	
		Server	**Client**
Model queue, used to create the temporary server RreplyTo queue	SYSTEM.COMMAND.REPLY.MODEL	Yes	No
Command input queue, used to submit DEFINE commands	SYSTEM.COMMAND.INPUT	Yes	No

WebSphere MQ clustering: SDSF recommends the use of clustering with the WebSphere MQ queue managers. Clustering is a configuration technique that provides these benefits in SDSF:

- Significant reduction in the WebSphere MQ definitions required to link queue managers together
- Improved awareness of the status of SDSF servers in the server group, eliminating the need for the client to wait for a timeout when a remote server in the server group is not active.

Clusters are groups of queue managers that can make their associated queues available to every other queue manager in the cluster. Communication between queue managers is simplified, reducing the number of channels that need to be defined. Without clustering, an installation must define a sender and a receiver channel for every queue manager with which a given queue manager will communicate. With clustering, the installation must define only a sender channel and a receiver channel for each cluster. Channels are not required for communicating between queue managers within the cluster of queue managers.

Clustering is enabled by defining a queue with the cluster attribute, or, if the queue is part of more than one cluster, a cluster name list. The queue manager must also have a repository defined for the cluster (with the WebSphere MQ `alter qmgr` command) and appropriate cluster sender and receiver channels must have been activated between the participating queue managers.

Clustering is described in *WebSphere MQ Queue Manager Clusters*.

When you use clustering, you specify the cluster name or namelist with the CLUSTER or CLUSNL parameters on the COMM statement in ISFPARMS.

If you do not use WebSphere MQ clustering, the addition of queues used by SDSF may require you to perform some WebSphere MQ configuration so that the queue managers for those queues can communicate, including defining a queue manager alias. Communication between queue managers is described in *WebSphere MQ Intercommunication*. The use of the queue manager alias with SDSF is described in "Communication between queue managers in a non-clustered environment" on page 346.

Solving communication problems:
For possible explanations and solutions to problems with communication between SDSF servers, see "Communication problems within a server group" on page 558.

Security:
You can protect the queues with SAF. See "WebSphere MQ" on page 306.

Server operator commands

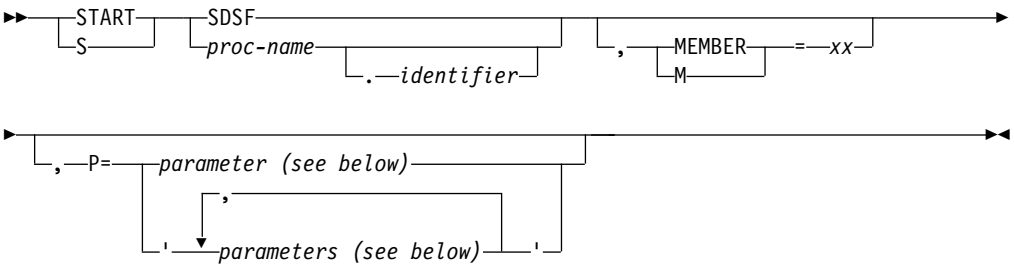
You control the server with the MVS operator commands described on the pages that follow.

Start the SDSF server

Use the server START command to initialize the SDSF server address space, and to control server options. When the server is initialized, the server is ready to process requests from the SDSF application.

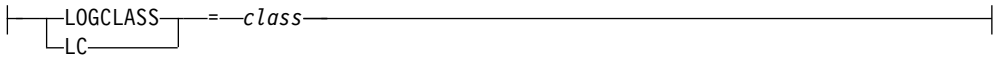
Format

Server START Command

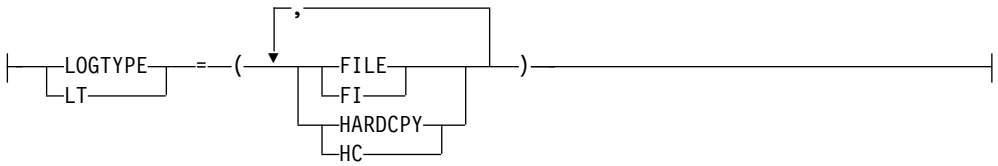


Message Folding:

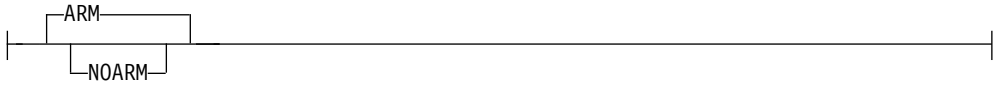
Log Class:



Log Type:

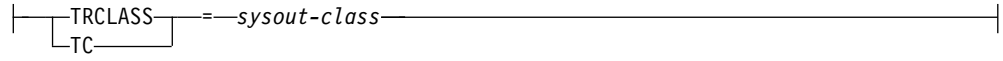


ARM:



Trace:

Trace SYSOUT Class:



Debug:

proc-name
is the name of the SDSF server to be started. The SDSF server name is the same as the procedure name; the server must run as a started task.

identifier
is an identifier that is used as the server name, instead of the procedure name.

MEMBER or M=*xx*
specifies the suffix of member name ISFPRM*xx*, which contains the statements to be read. The default for *xx* is 00. The dataset is either PARMLIB or a dataset defined in the server JCL using ddname SDSFPARM.

parameters
are the following:

NOFOLDMSG or NOFM
specifies that server messages should not be folded to uppercase; they are in mixed case. This is the default.

FOLDMSG or FM
specifies that server messages should be folded to uppercase.

LOGCLASS or LC (*class***)**
specifies the default SYSOUT class for the server log. If no SDSFLOG is defined in the JCL, SDSF will dynamically allocate a log to this class. The default is A.

LOGTYPE or LT
specifies the destination of the server log. The options are as follows:

FILE or FI
specifies that the report will be written to file with the ddname SDSFLOG. This is the default, unless the SDSF server is running under MSTR.

HARDCPY or HC
specifies that messages issued during processing of ISFPARMS will be written to the hardcopy log (syslog). This is the default if the SDSF server is running under MSTR.

ARM
specifies that ARM registration will be done if ARM is active in the system. The server will register using the following values:

- element name: ISF*server-name*@&*sysclone*
- element type: SYSSDSF
- termtype: ELEMTYPE

NOARM
specifies that ARM registration will not be done.

TRACE or TR
specifies the trace option. Tracing should be used under the direction of IBM service personnel. The options are as follows:

ALL
is equivalent to a mask of X'FFFFFFFF'.

NONE
is equivalent to a mask of X'00000000'.

ON starts tracing

trace-mask
specifies the event mask to be used. You can trace several events at one time by combining the mask values (in hexadecimal). The *mask* is a hexadecimal number that is 2, 4, 6, or 8 characters long. Each bit in the number represents a specific SDSF event to be traced. Leading zeros are not required, but the resulting mask must have an even number of digits. Possible values of *mask* are:

Mask	Description
FFFFFFFF	Unconditional trace
00800000	Message service
00400000	Communications events
00200000	ISFPARMS statements
00100000	Filter
00080000	Log processing
00040000	Internal interfaces
00020000	ISPF services
00010000	RMF processing
00008000	SDSF initialization
00004000	SDSF JES initialization
00002000	Call
00001000	Return
00000800	TSO data stream, ISPF buffers, batch input and output
00000400	Device and node processing
00000200	GDDM processing
00000100	SJF processing
00000080	SAF processing
00000040	Spool I/O and SRB processing
00000020	SSI processing, MVS/JES commands and job classes

Mask	Description
00000010	Data set processing
00000008	External interfaces, WLM scheduling environments and WLM resources
00000004	User exit call, return, and parameter list
00000002	ULOG functions
00000001	Reserved
00000000	No trace

TRCLASS or TC (*sysout-class*).
specifies the SYSOUT class to be used when dynamically allocating a trace file. If no ISFTRACE ddname is present in the server JCL, a trace will be allocated to SYSOUT using this class.

NODEBUG or NODB
specifies that the server should not run in diagnostic mode. This is the default.

DEBUG or DB
specifies that the server should run in diagnostic mode. This parameter is intended for use by IBM Service.

Notes to users

1. You must start the server before any users access SDSF, so that the statements can be read.
2. You can start multiple SDSF servers, as long as the server names are unique on a system.
3. When tracing is active, significant performance degradation may occur. A significant amount of trace output may be generated.
4. If the installation has defined an SDSFLOG DD statement in the server proc and SDSF is running under MSTR, you must specify LOGTYPE=FILE. The default value of HARDCPY will cause the server log not to be written to SDSFLOG.
5. The SDSFAUX log is written to the HSFLOG data set allocated by the SDSFAUX address space. It contains messages related to SDSFAUX processing.

Examples

1. `S SDSF`

   This command starts the SDSF server address space with the name SDSF.

2. `S SDSF,M=01`

   This command starts the SDSF server address space with the name SDSF. Statements will be read from member ISFPRM01 of the data set defined in the server JCL. Member ISFPRM01 is made the default member for any subsequent MODIFY *server*,REFRESH commands.

3. `S SDSF,M=01,P='FM,LC(H)'`

   This command starts the SDSF server address space, with the name SDSF. Statements will be read from member ISFPRM01 of the data set defined in the server JCL. Server messages will be folded to uppercase. The default SYSOUT class for the server log is H.

4. `S SDSFT`

   This command starts the SDSF server address space with the name SDSFT.

5. `S SDSF.SDSFA`

This command starts the SDSF server address space with proc SDSF and server name SDSFA. The server name, SDSFA, corresponds to the name coded in the ISFPMAC macro of ISFPARMS, or on the SDSF command.

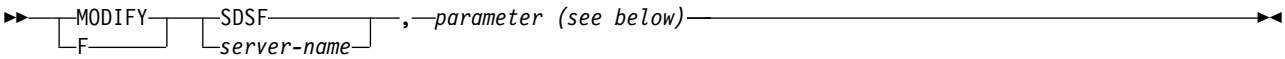
Change server options

Use the MODIFY command to dynamically change server options. You can specify a test mode to cause the syntax of the statements to be checked without activating the statements.

Format

The syntax is shown in Figure 5 on page 123.

Change Server Options



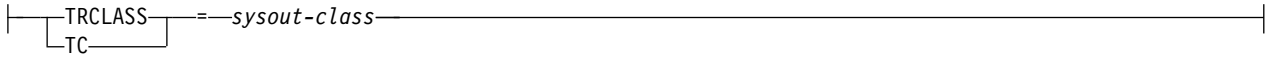
Message Folding:

Log Class:

Log Type:

Trace:

Trace SYSOUT Class:



Debug:

Figure 5. Change Server Options — Syntax

server-name
is the name of the SDSF server to be modified.

TEST
indicates that the syntax of the statements is to be syntax checked, but the statements are not to be activated.

parameter
is one of the following:

NOFOLDMSG or NOFM
specifies that server messages should not be folded to uppercase; they are in mixed case. This is the default.

FOLDMSG or FM
specifies that server messages be folded to uppercase.

LOGCLASS or LC (*class***)**
specifies the default SYSOUT class for the server log. If no SDSFLOG is defined in the JCL, SDSF will dynamically allocate a log to this class. The default is A.

LOGTYPE or LT
specifies the destination of the server log. The options are as follows:

FILE or FI
specifies that the report will be written to file with the ddname SDSFLOG.

HARDCPY or HC
specifies that messages issued during processing of ISFPARMS will be written to the hardcopy log (syslog)

TRACE or TR
Sets the trace option. Tracing should be used under the direction of IBM service personnel. The options are as follows:

ALL
is equivalent to a mask of X'FFFFFFFF'.

NONE
is equivalent to a mask of X'00000000'.

ON starts tracing.

OFF
stops tracing.

trace-mask
specifies the event mask to be used. You can trace several events at one time by combining the mask values (in hexadecimal). The *mask* is a hexadecimal number that is 2, 4, 6, or 8 characters long. Each bit in the number represents a specific SDSF event to be traced. Leading zeros are not required, but the resulting mask must have an even number of digits. Possible values of *mask* are:

Mask	Description
FFFFFFFF	Unconditional trace
00800000	Message service
00400000	Communications events
00200000	ISFPARMS statements
00100000	Filter
00080000	Log processing

Mask	Description
00040000	Internal interfaces
00020000	ISPF services
00010000	RMF processing
00008000	SDSF initialization
00004000	SDSF JES initialization
00002000	Call
00001000	Return
00000800	TSO data stream, ISPF buffers, batch input and output
00000400	Device and node processing
00000200	GDDM processing
00000100	SJF processing
00000080	SAF processing
00000040	Spool I/O and SRB processing
00000020	SSI processing, MVS/JES commands and job classes
00000010	Data set processing
00000008	External interfaces, WLM scheduling environments and WLM resources
00000004	User exit call, return, and parameter list
00000002	ULOG functions
00000001	Reserved
00000000	No trace

TRCLASS or TC (*sysout-class*)
specifies the SYSOUT class to be used when dynamically allocating a trace file. If no ISFTRACE ddname is present in the server JCL, a trace will be allocated to SYSOUT using this class.

NODEBUG or NODB
specifies that the server should not run in diagnostic mode.

DEBUG or DB
specifies that the server should run in diagnostic mode. This parameter is intended for use by IBM Service.

Note to users
When tracing is active, significant performance degradation may occur. A significant amount of trace output may be generated.

Example
```
F SDSFK,LC(H)
```

This command changes the default SYSOUT class for the server log to H.

Display server options
Use the MODIFY,D command to display options for the SDSF server.

Format

The syntax is shown in Figure 6.

Display Server Options

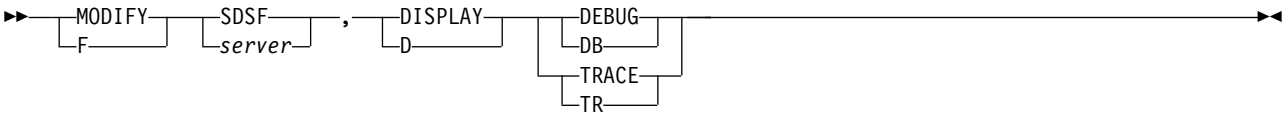


Figure 6. Display Server Options — Syntax

DEBUG or DB
displays the current setting for diagnostic mode.

TRACE or TR
displays the current setting for trace.

Example

```
F SDSF,D,TRACE
```

This command displays the current setting for trace.

Display information about server communications

Use this command to display information about the servers and the communication between SDSF servers.

Format

The syntax is shown in Figure 7.

Display Information About Server Communications

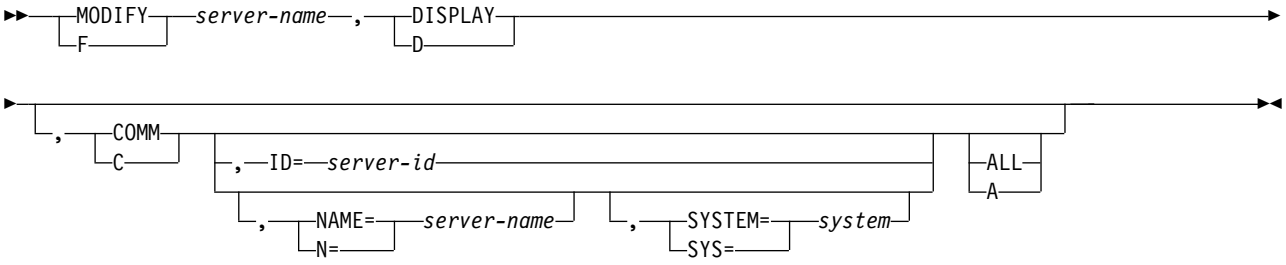


Figure 7. Display Information About Server Communications — Syntax

server-name
is the name of the SDSF server.

DISPLAY or D
displays information about the server, including the status of the server and server communications

COMM or C
displays information about the servers, including their ID, status and the system they are processing.

ID=*server-id*
displays information for the server with the indicated ID. The server ID can be displayed with the `F server-name,D` command. Leading zeros are not required.

NAME or N=*server-name*
displays information for the server with the indicated name. The server name can be a pattern.

SYSTEM or SYS=*system-name*
is the system on which the server runs. The system name can be a pattern.

ALL or A
displays information about WebSphere MQ, including the queue manager name and a count of requests processed by the server.

Refresh ISFPARMS

Use this command to refresh ISFPARMS statements. You can specify a test mode to cause the syntax of the statements to be checked without activating the statements.

Format

The syntax is shown in Figure 8.

Refresh ISFPARMS

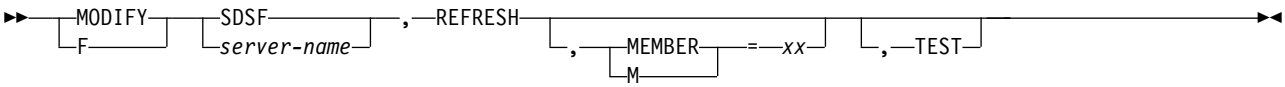


Figure 8. Refresh ISFPARMS — Syntax

server-name
is the name of the SDSF server to be modified.

REFRESH
indicates that a new set of statements is to be processed.

MEMBER or M(*xx***)**
specifies the suffix of member name ISFPRM*xx*, which contains the statements to be read. The dataset is either PARMLIB or a dataset defined in the server JCL using ddname SDSFPARM. The default for *xx* is whatever was used to start the server. For example, if you start the server with `S SDSF,M=01`, then refresh it with `F SDSF,REFRESH`, the member suffix used for the refresh is 01. If no suffix was specified on the START command, the suffix default is 00.

TEST
indicates that the syntax of the statements is to be syntax checked, but the statements are not to be activated.

Notes to users

1. A MODIFY REFRESH command processes only the statements defined in the current input stream. Any statements processed prior to the refresh are discarded when the new parameters are activated. If an error occurs, the current ISFPARMS remain in effect.

2. When SDSF is running on multiple systems in either a MAS or a sysplex, the SDSF server must be active on each system. Although the servers can share the same parameter data set, a MODIFY REFRESH command must be issued against each server.
3. If a server group has been defined and is in use, you cannot dynamically modify or delete it with the `MODIFY server-name,REFRESH` command. SERVERGROUP and SERVER statements are ignored. Instead, you should
   a. Make the change to ISFPARMS
   b. End server communications with the `MODIFY server-name,STOP,C,TERM` command
   c. Use the `MODIFY server-name,REFRESH` command to cause the new ISFPARMS to be processed

Examples

1. `F SDSF,REFRESH`

   This command activates a new set of statements for server SDSF. Because no member is specified, SDSF uses the member that was used when the server was started.

2. `F SDSFK,REFRESH,TEST`

   This command causes the syntax of statements to be checked for server SDSFK. The statements will not be activated.

3. `F SDSFT,REFRESH,M=01,TEST`

   This command causes the syntax of statements to be checked for server SDSFT. Statements will be read from member ISFPRM01 of the data set defined in the server JCL. The statements will not be activated.

Start communications

Use this command to logically start communications between SDSF servers. You might use it if a server has been previously stopped with the STOP command or was defined with the STOP(YES) on the SERVER statement in ISFPARMS, or if XCF has been stopped .

Format

The syntax is shown in Figure 9.

Start Communications

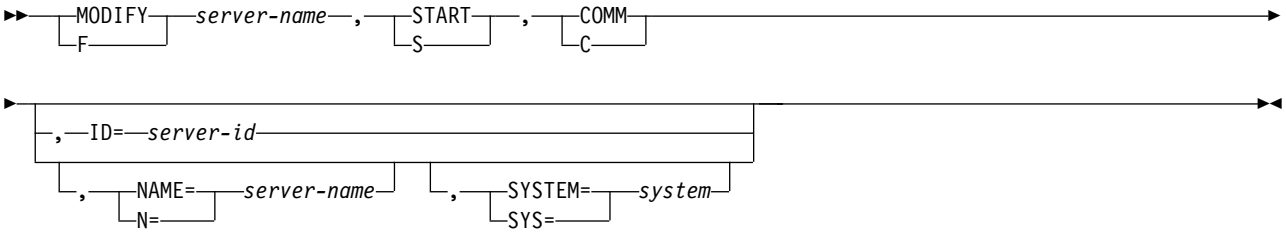


Figure 9. Start Communications — Syntax

server-name
is the name of the SDSF server.

START or S
indicates that the action is start.

COMM or C
causes communication between servers to be started. For WebSphere MQ communications, if no parameters follow, this causes communication for all servers in the server group to be started. The server group is the one defined for *server-name*.

ID=*server-id*
causes communication for the server with the indicated ID to be started. The server ID can be displayed with the `F server-name,D,C` command. Leading zeros are not required.

This applies only when WebSphere MQ is being used for communication between servers defined in the server group in ISFPARMS. It does not apply when XCF is being used for communication.

NAME or N=*server-name*
is the name of the server for which communication is to be started. The server name can be a pattern.

This applies only when WebSphere MQ is being used for communication between servers defined in the server group in ISFPARMS. It does not apply when XCF is being used for communication.

SYSTEM or SYS=*system-name*
is the system on which the server runs. The system name can be a pattern.

This applies only when WebSphere MQ is being used for communication between servers defined in the server group in ISFPARMS. It does not apply when XCF is being used for communication.

Stop communications

Use this command to stop communications between SDSF servers. You might use this command if a server is known to be unavailable, so that SDSF does not send requests to that server or wait for responses from it.

Format

The syntax is shown in Figure 10.

Stop Communications

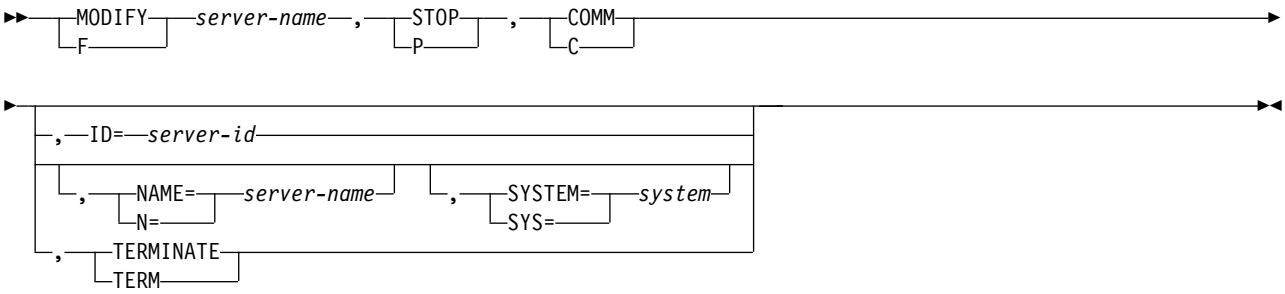


Figure 10. Stop Communications — Syntax

server-name
is the name of the SDSF server.

STOP or P
indicates that the action is stop.

COMM or C
causes communication between servers to be stopped. For WebSphere MQ communications, if no parameters follow, this causes communication for all servers in the server group to be stopped. The server group is the one defined for *server-name*.

ID=*server-id*
causes communication for the server with the indicated ID to be stopped. The server ID can be displayed with the `F server-name,D,C` command. Leading zeros are not required.

This applies only when WebSphere MQ is being used for communication between servers defined in the server group in ISFPARMS. It does not apply when XCF is being used for communication.

NAME or N=*server-name*
is the name of the server for which communication is to be stopped. The server name can be a pattern.

This applies only when WebSphere MQ is being used for communication between servers defined in the server group in ISFPARMS. It does not apply when XCF is being used for communication.

SYSTEM or SYS=*system-name*
is the system on which the server runs. The system name can be a pattern.

This applies only when WebSphere MQ is being used for communication between servers defined in the server group in ISFPARMS. It does not apply when XCF is being used for communication.

TERMINATE or TERM
ends communications. With WebSphere MQ communications, after this command, you can issue a MODIFY command with the REFRESH parameter to change the server group, and then re-activate communications. TERM can also be used to stop communications initialization.

Note to users
The initial status of the server can be defined as stopped with the STOP keyword of the SERVER statement in ISFPARMS. See "Server group definition parameters (SERVERGROUP, SERVER, COMM)" on page 30 for more information.

Stop the SDSF server
Use the STOP command to end the server.

Format
The syntax of the STOP command is shown in Figure 11.

Server STOP Command

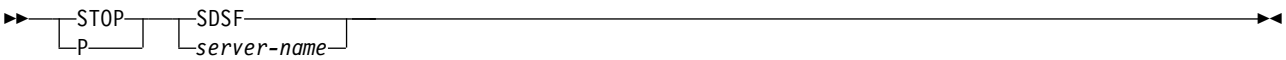


Figure 11. STOP the SDSF Server — Syntax

server-name
is the name of the SDSF server to be stopped.

Example

```
P SDSF
```

This command stops server SDSF.

Chapter 4. Columns on the SDSF panels

This topic describes the columns on SDSF panels that display data in a tabular format. Use this information when coding:

- FLD statements or ISFFLD macros, to customize which columns are included on a tabular panel, as well as their order, titles and widths.
- REXX execs or Java programs. Reference columns by their *names* rather than by their *titles*.

End users can use the ARRANGE command to reorder or change the widths of the columns. The ARRANGE command cannot modify the column titles or specify which columns are displayed. ARRANGE is described in the online help.

When displaying numeric values that are too large for the column width, SDSF scales them using these abbreviations: T (thousands), M (millions), B (billions), KB (kilobytes), MB (megabytes), GB (gigabytes), TB (terabytes) and PB (petabytes).

The fields on the title lines of SDSF panels cannot be customized. They are described in the online help.

In the tables that follow, an X in the *Delay* column indicates that obtaining the data may require an I/O operation. These columns are typically in the alternate field list. I/O operations are performed only when the columns are visible on the screen or being sorted. SDSF performance is best when columns that require an I/O operation are at the end of the field list. If there are no columns requiring I/O, the Delay column is not included.

Address Space Memory panel (AS)

The Address Space Memory (AS) panel shows system storage utilization for all address spaces in the sysplex. It provides a convenient means for identifying address spaces that are consuming the most common storage area (CSA) and system queue area (SQA). It also shows memory object usage, such as the number of memory objects owned, the current size of the memory object, and the highest size used.

You can use the fast path select (S) command to filter results, as follows. Leading zeros are not required when specifying the job number.

- **jobname** *jobid*, where *jobid* is optional and is the job type (JOB, TSU, STC, J, T, S) followed by the job number.
- **jobname** *job-number*, where *job-number* is optional
- *job-number*

In REXX execs and Java programs, reference columns by name rather than by title.

Table 34. Columns on the AS Panel

Column name	Title (Displayed)	Width	Description
JNAME	JOBNAME	8	Job name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
ASIDX	ASIDX	5	Address space identifier in hexadecimal

© Copyright IBM Corp. 1997, 2016

Table 34. Columns on the AS Panel (continued)

Column name	Title (Displayed)	Width	Description
REAL	Real	5	Current utilization of real storage in frames
FIXED	Fixed	5	Number of fixed real storage frames
CSA	CSA	8	CSA storage below the 16MB line in bytes
CSAPCT	CSA%	6	Percentage of CSA storage below the line being used
ECSA	ECSA	8	CSA storage above the 16MB line in bytes
ECSAPCT	ECSA%	6	Percentage of CSA above the 16MB line being used
SQA	SQA	8	SQA storage below the 16MB line in bytes
SQAPCT	SQA%	6	Percentage of SQA below the line being used
ESQA	ESQA	8	SQA storage above the 16MB line in bytes
ESQAPCT	ESQA%	6	Percentage of SQA above the line being used
AUX	Aux	6	Non-VIO slots being used
MEMLIMIT	MemLimit	8	Memory limit for 64-bit storage objects
MOBJNUM	MemObjNum	9	Number of memory objects for address space
MOBJ	MemObjUsed	10	Total allocated memory object size in MB
MOBJHWM	MemObjHWM	9	High-water mark allocated to memory objects in MB
HVCOMNUM	HVComNum	8	Number of high virtual common memory objects
HVCOM	HVComUsed	9	High virtual common memory size in MB
HVCOMHWM	HVComHWM	8	High virtual common memory high-water mark in MB
SHRMONUM	ShrMONum	8	Number of shared memory objects for address space
SHRMO	ShrMOUsed	9	Total size of shared memory objects in MB
SHRMOHWM	ShrMOHWM	8	Shared memory objects high-water mark in MB
FIXEDB	FixedB	6	Number of fixed frames below 16MB line
STEPN	StepName	8	Step name
PROCS	ProcStep	8	Procedure step name
JOBID	JobID	8	JES job ID, or work ID
OWNERID	Owner	8	User ID of job creator
POS	Pos	3	Address space position. For example: swapped in, swapped out, non-swappable, in transition
SWAPR	SR	2	Swap-out reason code
JTYPE	Type	4	Job type (STC, TSU, JOB)
ASID	ASID	5	Address space identifier
SUBSYS	SSName	6	Subsystem name
SYSNAME	SysName	8	System name
SYSLEVEL	SysLevel	25	Level of the operating system

Authorized Program Facility panel (APF)

The APF panel shows the data sets defined to the authorized program facility (APF) for each system in the sysplex.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 35. Columns on the APF Panel

Column name	Title (Displayed)	Width	Description
DSNAME	DSNAME	13-44 (Varies based on longest name.)	Data set name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
SEQ	Seq	3	Sequence number
VOLSER	VolSer	6	Volume serial
STATUS	Status	8	Data set status
BLKSIZE	BlkSize	7	Data set block size
EXTENT	Extent	6	Number of extents
SMS	SMS	3	SMS indicator. YES if the data set is SMS managed. Otherwise, NO
LRECL	LRecL	5	Logical record length
DSORG	DSOrg	5	Data set organization
RECFM	RecFm	5	Record format
DEFVOL	DefVol	6	Defined volume
CRDATE	CrDate	8	Data set creation date
REFDATE	RefDate	8	Data set last referenced date
SYSNAME	SysName	8	System name
SYSLEVEL	SysLevel	25	Operating system level

Display Active Users panel (DA)

The DA panel shows information about MVS address spaces (jobs, started tasks, and TSO users) that are running. SDSF obtains the information from RMF when it is installed. Columns for which RMF is required are indicated by [RMF].

The N action character invokes the ENQ panel as a secondary display to show all enqueues associated with the ASID for the row. To protect the N action character to display enqueues from the DA panel, protect the ENQ command. This is described in "Authorized SDSF commands" on page 235.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 36. Columns on the DA Panel

Column Name	Title (Displayed)	Width	Description	Delay
JNAME	JOBNAME	8	Job name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.	
STEPN	StepName	8	Job step name (TSO logon procedure name for TSO users)	

Table 36. Columns on the DA Panel (continued)

Column Name	Title (Displayed)	Width	Description	Delay
PROCS	ProcStep	8	Procedure step name (terminal ID for TSO users)	
JTYPE	Type [1]	4	Type of address space	
JNUM	JNum[1]	6	JES job number	
JOBID	JobID	8	JES job ID	
OWNERID	Owner	8	User ID of job owner, or default values of ++++++++ or ????????, if user ID not defined to RACF	
JCLASS	C	1 or 8	JES input class at the time the job was selected for execution. Default width expands to 8 if there are long class names in the MAS.	
POS	Pos	3	Address space position	
DP	DP	2	Address space dispatching priority in hexadecimal	
REAL	Real	4	Current real storage usage in frames	
PAGING	Paging	6	Demand paging rate for address space	
EXCPRT	SIO	6	EXCP rate in EXCPs per second for address space. The value is approximate, and derived from this calculation: the job delta EXCP count (from RMF or the ASCB) divided by the total time interval.	
CPUPR	CPU% [2]	6	Percent of CPU time consumed by and on behalf of the address space during the most recent interval measured	
ASID	ASID	4	Address space identifier	
ASIDX	ASIDX	5	Address space identifier in hexadecimal	
EXCP	EXCP-Cnt	9	Accumulated EXCP count for the current job step for the address space. Uses hexadecimal scaling.	
CPU	CPU-Time	10	Accumulated CPU time consumed by and on behalf of the address space, for the current job step, in seconds	
SWAPR	SR	2	Swap out reason code	
STATUS	Status	6	JES job status	
SYSNAME [RMF]	SysName	8	System name where job is executing	
SPAGING [RMF]	SPag	4	System demand paging rate for system that the job is executing on. The value is the same for all rows for a system.	
SCPU [RMF]	SCPU%	5	System CPU percentage for system that is processing the job. The value is the same for all rows for a system.	
WORKLOAD [RMF]	Workload	8	Workload name	
SRVCLASS [RMF]	SrvClass	8	Service class name	
PERIOD [RMF]	SP	2	Service class period	
RESGROUP [RMF]	ResGroup	8	Resource group name	

Table 36. Columns on the DA Panel (continued)

Column Name	Title (Displayed)	Width	Description	Delay
SERVER RMF	Server	8	Server indicator (resource goals are not being honored)	
QUIESCE RMF	Quiesce	7	Quiesce indicator (address space is quiesced)	
ECPU RMF	ECPU-Time	10	Total CPU time consumed by and within the address space, for the current job step, in seconds	
ECPUPR RMF	ECPU%	6	CPU usage by and within the address space	
CPUCRIT RMF	CPUCrit	7	Current address space CPU-protection	
STORCRIT RMF	StorCrit	8	Current address space storage protection	
RPTCLASS RMF	RptClass	8	Report class	
MEMLIMIT RMF	MemLimit	8	Memory limit	
TRANACT RMF	Tran-Act	10	Elapsed time the transaction has been active	
TRANRES RMF	Tran-Res	10	Elapsed time the transaction was swapped in	
SPIN RMF	Spin	4	Indicator of whether job can be spun	X
SECLABEL	SecLabel	8	Security label of the address space	
GCPTIME RMF	GCP-Time	8	Accumulated general processor service time, in seconds	
ZAAPTIME RMF	zAAP-Time	9	Accumulated zAAP service time, in seconds	
ZAAPCPTM RMF	zACP-Time	9	CPU time consumed on general processors by work that was eligible for a zAAP, in seconds	
GCPUSE RMF	GCP-Use%	8	Percent of the total general processor time used by the address space in the most recent interval	
ZAAPUSE RMF	zAAP-Use%	9	Percent of the total zAAP time used by the address space in the most recent interval	
SZAAP RMF	SzAAP%	6	zAAP view of CPU use for the system, in the most recent interval. The value is the same for all rows for a system.	
SZIIP RMF	SzIIP%	6	System zIIP utilization for the system that is processing the job. The value is the same for all rows for a system.	
PROMOTED RMF	Promoted	8	Indicates whether the address space is currently promoted due to a chronic resource contention	
ZAAPNTIM RMF	zAAP-NTime	10	Normalized zAAP service time, in seconds	
ZIIPTIME RMF	zIIP-Time	9	CPU time consumed on zIIPs, in seconds	
ZIIPCPTMRMF	zICP-Time	9	CPU time consumed on general processors by work that was eligible for a zIIP, in seconds	
ZIIPNTIM RMF	zIIP-NTime	10	Normalized zIIP service time, in seconds	
ZIIPUSE RMF	zIIP-Use%	9	Percent of the total zIIP time used by the address space in the most recent interval	
SLCPU RMF	SLCPU%	6	Percentage of time the LPAR is busy for the system, in the most recent interval. The value for SLCPU% is the same for all rows for a system.	
IOPRIOGRPRMF	IOPrioGrp	9	WLM I/O priority group	
JOBCORR	JobCorrelator	32	User portion of the job correlator (JES2 only)	

Table 36. Columns on the DA Panel (continued)

Column Name	Title (Displayed)	Width	Description	Delay

[1] Not included in the default field list.

[2] SDSF calculates the value for the CPU% column. It is the ratio between the CPU time used by one job and the CPU time used by all jobs, in the interval between times that the user presses Enter.

[3] Columns with information for zAAPs and zIIPs are shown only if at least one of the appropriate specialized processors (zAAP or zIIP) has been configured for a system that is within the scope of the systems being shown on the panel. Note that changing the systems being shown (with the SYSNAME or FILTER commands) once the DA panel is displayed does not affect whether SDSF includes or omits the column.

Dynamic Exits panel (DYNX)

The Dynamic Exits (DYNX) panel shows all of the dynamic exits in the sysplex, their status, and the modules that implement the exit.

You can use the fast path select (S) command with an EXITNAME to filter results.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 37. Columns on the DYNX Panel

Column name	Title (Displayed)	Width	Description
EXITNAME	EXITNAME	16	Dynamic exit name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
SEQ	Seq	3	Sequence number for module in list
MODNAME	ModName	8	Module name implementing exit
ACTIVE	Active	6	Exit active (YES or NO)
FASTPATH	FastPath	8	Exit FASTPATH option (YES or NO). FASTPATH processing means that the system does not provide as much function, and therefore the overall processing time is less.
MODEPA	ModEPA	8	Module entry point address
MODLOADPT	LoadPt	8	Module load point address if available
MODSIZE	ModLen	8	Module length if available
JNAME	FiltJob	8	Jobname for which exit is to get control
STOKEN	FiltSTok	16	Address space token (STOKEN) for which exit is to get control
ABENDNUM	NumAbend	8	Number of abends before exit inactivates
ABENDCON	ConAbend	8	Consecutive abend option (YES – consecutive abends before inactivation, NO – cumulative abends before inactivation)
SEQMAX	SeqMax	6	Maximum module sequence number
SYSNAME	SysName	8	System name
SYSLEVEL	SysLevel	25	Level of the operating system

Enclaves panel (ENC)

The Enclaves panel shows enclaves.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 38. Columns on the ENC Panel

Column name	Title (Displayed)	Width	Description
NAME	NAME	16	Token that identifies the enclave. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
SSTYPE	SSType	6	Subsystem type (for example, DB2®, MQ)
STATUS	Status	8	Active or inactive
ESRVCLS	SrvClass	8	Service class
PERIOD	Per	3	Period number
PGN	PGN	3	Performance group
RPTCLS	RptClass	8	Report class
RESGROUP	ResGroup	8	Resource group
CPU	CPU-Time	10	Total CPU time
OWNSYS	OwnerSys	8	Enclave owner system
JNAME	OwnerJob	8	Enclave owner jobname
ASID	OwnerAS	7	Enclave owner ASID (displayed only if this enclave is the original)
ASIDX	OwnerASXA	8	Enclave owner ASID in hexadecimal (displayed only if this enclave is the original)
ORIGINAL	Original	8	Indicates, for an enclave that has been exported, if this is the original. Value is YES or NO.
ESCOPE	Scope	8	Scope of the enclave; LOCAL (single-system) or MULTISYS (multisystem capable; there is an export token for the enclave)
TYPE	Type	4	IND (Independent) or DEP (dependent)
WORKLOAD	Workload	8	Workload name
QUIESCE	Quiesce	12	Indicates if the enclave is in a quiesce delay, which occurs if the address space has been reset with the MVS RESET,QUIESCE command. Value is YES, YES-IMPLICIT (quiesced through enclave server quiesce) or NO.
SYSNAME	SysName	8	Name of the system that provided the data
SYSLEVEL	SysLevel	25	Level of the operating system
SUBSYS	Subsys	8	Subsystem name
ZAAPTIME	zAAP-Time	9	Cumulative zAAP time consumed by dispatchable units running in the enclave on the local system. See note below.
ZAAPCPTM	zACP-Time	9	Cumulative zAAP on CP time consumed by dispatchable units running in the enclave on the local system. See note below.

Table 38. Columns on the ENC Panel (continued)

Column name	Title (Displayed)	Width	Description
ZIIPTIME	zIIP-Time	9	Cumulative zIIP time consumed by dispatchable units running in the enclave on the local system. See note below.
ZIIPCPTM	zICP-Time	9	Cumulative zIIP on CP time consumed by dispatchable units running in the enclave on the local system. See note below.
PROMOTED	Promoted	8	Indicates whether the address space is currently promoted due to a chronic resource contention
ZAAPNTIM RMF	zAAP-NTime	10	zAAP service time, in seconds, normalized for the slower CP
ZIIPNTIM RMF	zIIP-NTime	10	zIIP service time, in seconds, normalized for the slower CP
ARRTIME	Arrival-Time	19	Date and time the enclave was created
ARRINTV	Arrival-Int	11	Interval since the enclave was created (*hh:mm:ss*)
CPUCRIT	CPUCrit	7	CPU protection
IOPRIOGRP	IOPrioGrp	9	WLM I/O priority group

Note: This column shows time consumed by dispatchable units running in the enclave on the local system. For a multisystem enclave, time consumed on other systems is not included. The value may decrease between invocations if the transaction is restarted to avoid an overflow of internal accumulators.

Enqueue panel (ENQ)

Enqueuing is the mechanism by which a program requests control of a serially reusable resource. The Enqueue (ENQ) panel allows authorized users to display active system enqueues. The panel shows the major and minor names for the enqueuer, as well as the job name waiting for or holding the enqueue. Parameters on the ENQ command control which major and system names are shown. By default, only major SYSDSN enqueues on the local system are shown.

You can also access the ENQ panel from the DA panel using the N action character. When ENQ is accessed in this way, all enqueues used by the selected address space are shown.

Note: Major and minor names can contain hexadecimal characters that cannot be displayed by SDSF. SDSF translates control characters (0x00 through 0x3F) to periods. Other characters are not translated and their display varies based on factors such as the emulator. You can use the D action character to display major and minor names in hexadecimal, but the length is limited by the message text in the response.

The **ENQC** command provides a convenient means of showing all enqueues with contention. That is, **ENQC** shows currently held enqueues that are required by another job. **ENQC** does not accept any parameters.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 39. Columns on the ENQ Panel

Column name	Title (Displayed)	Width	Description
MINOR	MINOR	52	Minor name (RNAME). This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro. Control characters are translated to periods.
MAJOR	Major	8	Major name (QNAME). Control characters are translated to periods.
REQTYPE	Req	3	Request type (SHR or EXC)
JOBNAME	JobName	8	Job name holding or requesting enqueue
ASID	ASID	4	Job name ASID (decimal)
ASIDX	ASIDX	6	Job name ASID (hexadecimal)
LEVEL	Level	10	Request level: ENQ-normal enqueuer, Reserve-hardware reserve, Global enq-hardware reserve converted to global enqueue
SMC	SMC	3	Step must complete indicator
SCOPE	Scope	8	Enqueue scope (step, system, systems, global)
STATUS	Status	6	Resource status (own, wait)
OWNERS	Owners	6	Number of resource owners for enqueuer
WAITERS	Waiters	7	Number of tasks waiting for enqueue
WAITEXC	WaitExc	7	Number of tasks waiting for exclusive use
WAITSHR	WaitShr	7	Number of tasks waiting for shared use
UNIT	Unit	4	Device address for reserves
USERDATA	UserData	32	User data passed on ISGENQ
REQTIME	ReqTime	19	Date and time of request
ENQTOKEN	EnqToken	64	Enqueue token
RNAMEL	RNameLong	127	Longer version of minor name, up to 127 characters. Control characters are translated to periods.
SYSNAME	SysName	8	System name

Health Check History panel (CKH)

The CKH panel shows information about instances of a check selected from the CK panel.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 40. Columns on the CKH Panel

Column name	Title (Displayed)	Width	Description
COUNT	Count	17	Count of this instance of the check
OWNER	CheckOwner	16	Check owner
STATUS	Status	18	Check status
RESULT	Result	6	Result code from the check
DIAG1	Diag1	8	Diagnostic data from check, word 1
DIAG2	Diag2	8	Diagnostic data from check, word 2
DATEE	Start-Date-Time	19	Date and time the check started (YYYY.DDD HH:MM:SS)

Table 40. Columns on the CKH Panel (continued)

Column name	Title (Displayed)	Width	Description
DATEN	End-Date-Time	19	Date and time the check ended (YYYY.DDD HH:MM:SS)
SYSPLEX	Sysplex	8	Sysplex name for the sysplex on which the check ran
SYSNAME	SysName	8	System name for the system on which the check ran
NAME	Name	32	Check name

Health Checker panel (CK)

The CK panel shows information from IBM Health Checker for z/OS about the active checks.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 41. Columns on the CK Panel

Column name	Title (Displayed)	Width	Description
NAME	NAME	32	Check name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
OWNER	CheckOwner	16	Check owner
STATE	State	18	Check state
STATUS	Status	18	Check status
RESULT	Result	6	Result code from the last invocation of the check
DIAG1	Diag1	8	Diagnostic data from check, word 1
DIAG2	Diag2	8	Diagnostic data from check, word 2
DIAGFROM	DiagFrom	8	Source of the diagnostic data, words 1 and 2: ABEND, HCHECKER or CHECKRTN
GLOBAL	Global	6	Indicator of whether the check is global
GLOBALSYS	GlobalSys	9	Name of the system on which the global check is running
EXCOUNT	ExcCount	8	Number of exceptions detected by this check on the last iteration
COUNT	RunCount	8	Number of times the check has been invoked
FAIL	Fail	4	Number of times the check failed
SEVERITY	Severity	8	Severity level of the check (HIGH, MEDIUM, LOW, NONE)
SEVCODE	SevCode	7	Numeric severity level of the check
WTOTYPE	WTOType	9	WTO type issued when an exception is found (EVENTUAL, CRITICAL, INFO, HC, NONE or a descriptor code)
MODIFIED	ModifiedBy	26	How the check was modified
POLSTAT	PolicyStatus	18	Policy error status
WTONUM	WTONum	6	Number of WTOs issued by the check
NUMCAT	NumCat	6	Number of categories in which the check is defined
CATEGORY	Category	16	Category name. Users can view the complete set of categories by typing + alone in this column.

Table 41. Columns on the CK Panel (continued)

Column name	Title (Displayed)	Width	Description
CATEGORY2 -CATEGORY4	Category2 – Category4	16	Category names 2 to 4.
CATEGORY5 -CATEGORY16	Category5 – Category16	16	Category names 5 to 16. By default, these appear only in the alternate field list.
EXITNAME	ExitName	8	Exit modname that added the check
MODNAME	ModName	8	Check module name
MSGNAME	MsgName	8	Message load module name
USERDATE	UserDate	8	Current date of the check
DEFDATE	DefDate	8	Default date of the check
DEBUG	Debug	5	Debug mode indicator
DATEE	Start-Date-Time	19	Date and time the check last started (YYYY.DDD HH:MM:SS)
INTERVAL	Interval	8	Time interval at which the check runs (HHH:MM)
SCHDATE	NextSch-Date-Time	19	Date and time the check is next scheduled to run (YYYY.DDD HH:MM:SS)
SCHINT	NextSch-Int	11	Time remaining to the date and time the check is next scheduled to run, in HHHHH:MM:SS
LOGDATE	Log-Date-Time	19	Date and time of the last successful write to System Logger
DELDATE	Deleted-Date-Time	19	Date and time the check was deleted
PROCNAME	ProcName	8	Health Checker procedure name
STCID	TaskID	8	Health Checker started task ID
REASON	Reason	126	Description of the reason for check
UPDREAS	UpdateReason	48	Description of updates to the check. The width can be increased to 126.
PARMLEN	ParmLen	7	Length of the check parameters
PARM	Parameters	32	Check parameters. Only characters A-Z, 0-9, #, @, $ and blanks are shown. Any other value is translated to a period.
SYSLEVEL	SysLevel	25	Level of the operating system
SYSNAME	SysName	8	System name
EINTERVAL	EInterval	9	Interval at which the check will run when it has raised an exception
EXECNAME	ExecName	8	Name of the exec to run
LOCALE	Locale	8	Where the check is running
ORIGIN	Origin	8	Origin of the check
VERBOSE	Verbose	7	Verbose mode for the check
REXXIN	RexxIn	44	REXX input data set name
REXXOUT	RexxOut	44	REXX output data set name
LOGSTREAM	LogStream	26	Name of the logstream used to record this check

Held Output panel (H)

The Held Output panel shows the user information about SYSOUT data sets for jobs, started tasks, and TSO users on any *held* JES output queue.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 42. Columns on the H Panel

Column name	Title (Displayed)	Width	Description	Delay
JNAME	JOBNAME	8	Job name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.	
JNUM	JNum[1]	6	JES job number	
JOBID	JobID	8	JES job ID	
OWNERID	Owner	8	User ID of SYSIN/SYSOUT owner, or default values of ++++++++ or ????????, if user ID not defined to RACF	
DPRIO	Prty	4	JES output group priority	
OCLASS	C	1	JES output class	
OUTDISP	ODisp	5	JES output disposition (JES2 only)	
DESTN	Dest	18	JES print destination name	
RECCNT	Tot-Rec	9	Output total record count (lines). Blank for page-mode data.	
PAGECNT	Tot-Page	9	Output page count (lines). Blank if not for page-mode data.	
FORMS	Forms	8	Output form number	
FCBID	FCB	4	Output FCB ID	
STATUS	Status	16	JES job status	
UCSID	UCS	4	Output UCS ID (print train required)	
WTRID	Wtr	8	Output external writer name	
FLASHID	Flash	5	Output flash ID	
BURST	Burst	5	3800 burst indicator	
PRMODE	PrMode	8	Printer process mode	
DEST	Rmt	5	JES print routing. Remote number if routing is not local. (JES2 only)	
NODE	Node	5	JES print node (JES2 only)	
SECLABEL	SecLabel	8	Security label of data sets	
OGNAME	O-Grp-N	8	Output group name (JES2 only)	
OGID	OGID1	5	Output group ID 1 (JES2 only)	
OGID2	OGID2	5	Output group ID 2 (JES2 only)	
JPRIO	JP	2	Job priority	
DSDATE	CrDate	10	Data set creation date. The installation can change the CRDATE column to 19, so that the date and time is included. (JES2 only)	
OHREASON	OHR	3	Output hold reason code	
OHRSNTXT	Output-Hold-Text	37	Output hold reason text	
DEVID	Device	18	Output device name	

Table 42. Columns on the H Panel (continued)

Column name	Title (Displayed)	Width	Description	Delay
DSYSID	SysID	5	Printing system (JES2 only)	
OFFDEVS	Offs	4	List of offload devices for a job or output that has been offloaded (JES2 only)	
RETCODE	Max-RC	10	Return code information for the job	
JTYPE	Type	4	Type of address space	
ROOMN	RNum	8	JES job room number	X
PNAME	Programmer-Name	20	JES programmer name	X
ACCTN	Acct	4 (JES2) 8 (JES3)	JES account number	X
NOTIFY	Notify	8	TSO user ID from NOTIFY parameter on job card	X
ISYSID	ISys	4 (JES2) 8 (JES3)	JES input system ID	X
TIMER	Rd-Time	8	Time that the job was read in	X
DATER	Rd-Date	8	Date that the job was read in	X
ESYSID	ESys	4 (JES2) 8 (JES3)	JES execution system ID	X
TIMEE	St-Time	8	Time that execution began	X
DATEE	St-Date	8	Date that execution began	X
TIMEN	End-Time	8	Time that execution ended	X
DATEN	End-Date	8	Date that execution ended	X
ICARDS	Cards	5	Number of cards read for job	X
JCLASS	JC	1 or 8	JES input job class. Default width expands to 8 if there are long class names in the MAS.	
MCLASS	MC	2	Message class of job	X
SUBGROUP	SubGroup	8	Submittor group	X
JOBACCT1	JobAcct1[1]	20	Job accounting field 1	X
JOBACCT2	JobAcct2[1]	20	Job accounting field 2	X
JOBACCT3	JobAcct3[1]	20	Job accounting field 3	X
JOBACCT4	JobAcct4[1]	20	Job accounting field 4	X
JOBACCT5	JobAcct5[1]	20	Job accounting field 5	X
JOBCORR	JobCorrelator	32	User portion of the job correlator (JES2 only)	

Notes:

[1] This column is not included in the default field list.

Initiator panel (INIT)

The Initiator panel allows users to display information about JES initiators that are defined in the active JES on their CPUs.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 43. Columns on the INIT Panel

Column name	Title (Displayed)	Width	Description
INTNAME	ID	4 (JES2) 8 (JES3)	Initiator ID (JES2) or group or class name (JES3). This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
STATUS	Status	10	Initiator status
ICLASS	Classes	8	JES2 initiator classes (JES2 only). Multi-character classes and groups shows as periods (.).
JNAME	JobName	8	Job name
STEPN	StepName	8	Job step name
PROCS	ProcStep	8	Procedure step name (JES2 only)
JTYPE	Type	4	Type of address space
JNUM	JNum[1]	6	JES job number
JOBID	JobID	8	JES job ID or work ID
JCLASS	C	8	JES input class at time job was selected for execution
ASID	ASID	4	Address space identifier
ASIDX	ASIDX	5	Address space identifier in hexadecimal
OWNERID	Owner	8	User ID of the owner of the active job
SYSNAME	SysName	8	System name
DSYSID	SysID	5 (JES2) 8 (JES3)	JES member name (JES2) or the system on which the job is active under the class (JES3, resource type of INIT)
JESNAME	JESN	4	JES subsystem name
JESLEVEL	JESLevel	8	JES level
SECLABEL	SecLabel	8	Security label of the job
SRVCLASS	SrvClass	8	For JES-managed initiators, shows the service class of the active job. For WLM-managed initiators, shows the service class the initiator is running.
IMODE	Mode	4	Initiator mode (group rows only)
BARRIER	Barrier	7	Group scheduling barrier (JES3 only, group rows only)
DEFAULT	Default	7	Default group indicator (JES3 only)
DEFCNT	DefCount	8	Defined initiator count (JES3 only, group rows only)
ALLOCCNT	AllocCount	10	Allocated initiator count (JES3 only)
USECOUNT	UseCount	8	In-use initiator count
ALLOC	Alloc	5	Allocation option (JES3 only, group rows only), which determines when the execution resources are to be allocated to the JES-managed group
UNALLOC	Unalloc	7	Unallocation indicator (JES3 only, group rows only)
GROUP	Group	8	Group name
RESTYPE	ResType	7	Resource type (group or class)

Table 43. Columns on the INIT Panel (continued)

Column name	Title (Displayed)	Width	Description
ICLASS1-8	Class1-8	8	JES2 initiator classes 1-8, including multi-character classes and groups (JES2 only)

Notes:

[1] JNUM is not included in the default field list.

Input Queue panel (I)

The Input Queue panel allows the user to display information about jobs, started tasks, and TSO users on the JES input queue or executing.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 44. Columns on the I Panel

Column name	Title (Displayed)	Width	Description	Delay
JNAME	JOBNAME	8	Job name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.	
JOBID	JobID	8	JES job ID	
JTYPE	Type	4	Type of address space	
JNUM	JNum[1]	6	JES job number	
OWNERID	Owner	8	User ID of job owner, or default values of ++++++++ or ????????, if user ID not defined to RACF 1.9 and later	
JPRIO	Prty	4	JES2 input queue priority	
JCLASS	C	1 or 8	JES input class. Default width expands to 8 if there are long class names in the MAS.	
POS	Pos	5	Position within JES input queue class	
PRTDEST	PrtDest	18	JES print destination name	
ROUTE	Rmt	5	JES print routing. Remote number if routing is not local. (JES2 only)	
NODE	Node	5	JES print node (JES2 only)	
SYSAFF	SAff	5 (JES2) 8 (JES3)	JES execution system affinity (if any)	
ACTSYS	ASys	4 (JES2) 8 (JES3)	JES execution system ID (for logged-on users only)	
STATUS	Status	17	Status of job	
SECLABEL	SecLabel	8	Security label of job	
TGNUM	TGNum	5	Track groups used by job	
TGPCT	TGPct	6	Percentage of total track group usage	
ORIGNODE	OrigNode	8	Origin node name	
EXECNODE	ExecNode	8	Execution node name	
DEVID	Device	18	JES device name	
SRVCLS	SrvClass	8	Service class	
WLMPOS	WPos	5	Position on the WLM queue	
SCHENV	Scheduling-Env	16	Scheduling environment for the job	

Table 44. Columns on the I Panel (continued)

Column name	Title (Displayed)	Width	Description	Delay
DELAY	Dly	3	Indicator that job processing is delayed	
SSMODE	Mode	4	Subsystem managing the job (JES or WLM)	
ROOMN	RNum	8	JES job room number	X
PNAME	Programmer-Name	20	JES programmer name field	X
ACCTN	Acct	4 (JES2) 8 (JES3)	JES account number field	X
NOTIFY	Notify	8	TSO user ID from NOTIFY parameter on job card	X
ISYSID	ISys	4 (JES2) 8 (JES3)	JES input system ID	X
TIMER	Rd-Time	8	Time that the job was read in	X
DATER	Rd-Date	8	Date that the job was read in	X
ESYSID	ESys	4 (JES2) 8 (JES3)	JES execution system ID	X
TIMEE	St-Time	8	Time that execution began	X
DATE	St-Date	8	Date that execution began	X
ICARDS	Cards	5	Number of cards read for job	X
MCLASS	MC	2	MSGCLASS of job	X
TSREC	Tot-Lines	10	Total number of spool records for job	X
SPIN	Spin	4	Indicator of whether the job is eligible to be spun	
SUBGROUP	SubGroup	8	Submitter group	X
PHASENAME	PhaseName	20	Name of the phase the job is in	
PHASE	Phase	8	Number of the phase the job is in	
JOBACCT1	JobAcct1[1]	20	Job accounting field 1	X
JOBACCT2	JobAcct2[1]	20	Job accounting field 2	X
JOBACCT3	JobAcct3[1]	20	Job accounting field 3	X
JOBACCT4	JobAcct4[1]	20	Job accounting field 4	X
JOBACCT5	JobAcct5[1]	20	Job accounting field 5	X
SUBUSER	SubUser	8	Submitting user ID	
DELAYRSN	DelayRsn	32	Reason for the job delay (JES2 only). The width can be expanded to 127.	
JOBCORR	JobCorrelator	32	User portion of the job correlator (JES2 only)	
ASID	ASID	5	ASID of the active job	
ASIDX	ASIDX	5	ASID of the active job, in hexadecimal	
SYSNAME	SysName	8	MVS system name where the job is executing	
JOBGROUP	JobGroup	8	Name of the job group associated with job (JES2 only)	
JOBGRPID	JobGrpId	8	JES2 job group job ID	
JOBSET	JobSet	8	Job set within the job group to which this job belongs (JES2 only)	
JGSTATUS	JGStatus	8	Status of the job within the dependency network (JES2 only)	

Table 44. Columns on the I Panel (continued)

Column name	Title (Displayed)	Width	Description	Delay
FLUSHACT	FlushAct	8	Flush action indicator (JES2 only)	
HOLDUNTIL	HoldUntil	19	HOLDUNTIL date and time (JES2 only)	
STARTBY	StartBy	19	STARTBY date and time (JES2 only)	
WITH	With	19	Name of the job or started task that the job must run with (on the same system) (JES2 only)	
DATETIMER	Rd-DateTime	19	Date and time that the job was read in. This column is displayed only with the SDSF task of z/OSMF. It combines the information in the Rd-Date and Rd-Time columns.	X
DATETIMEE	St-DateTime	19	Date and time that execution began. This column is displayed only with the SDSF task of z/OSMF. It combines the information in the St-Date and St-Time columns.	X

Notes on the table:

1. This column is not included in the default field list.

JESPLEX panel (JP)

The JESPLEX (JP) panel simplifies the display and control of members in a JES3 JESPLEX. It is analogous to the JES2 MAS panel, and they share a common field list. For a description of the columns, see "Multi-Access Spool panel (MAS) and JESPLEX (JP) panel" on page 158.

Job Class panel (JC)

The JC panel allows the user to display information about job classes.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 45. Columns on the JC Panel

Column name	Title (Displayed)	Width	Description
JOBCL	CLASS	8	Job class. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
JSTATUS	Status	8	Class status
MEMBER	Member	8	Member name (JES3 only)
GROUP	Group	8	Group name
JMODE	Mode	4	Manager of the class
WAITCNT	Wait-Cnt	8	Number of jobs waiting for execution (non-WLM jobs only) (JES2 only)
XEQCNT	Xeq-Cnt	8	Number of active jobs
HOLDCNT	Hold-Cnt	8	Number of held jobs (JES2 only)
JCODISP	ODisp	13	Output disposition for normal and abnormal end of the job (JES2 only)
QHELD	QHld	4	Job class hold indicator (JES2 only)
JHOLD	Hold	4	Job hold indicator (JES2 only)

Table 45. Columns on the JC Panel (continued)

Column name	Title (Displayed)	Width	Description
XBM	XBM	8	Name of the execution batch monitor (XBM) procedure to be executed by jobs running in the class (JES2 only)
JCLIM	JCLim	5	Job class limit for the system (JES2 only)
TDEPTH	TDepth	6	Maximum job count for the class (JES3 only). This is analagous to the JCLim column for JES2.
JPGN	PGN	3	Default performance-group number (JES2 only)
JAUTH	Auth	4	MVS operator command groups that are to be executed (JES2 only)
BLP	BLP	3	Perform bypass label processing (JES2 only)
COMMAND	Command	7	Disposition of commands read from the input stream (JES2 only)
JLOG	Log	3	Job log indicator
MSGLEVEL	MsgLV	5	Message level value (JES2 only)
OUTPUT	Out	3	SYSOUT write indicator (JES2 only)
PROCLIB	PL	2	Default procedure library number (JES2 only)
REGION	Region	6	Default region size assigned to each job step (JES2 only)
SWA	SWA	5	Placement of SWA control blocks created for jobs, in relation to 16 megabytes in virtual storage (JES2 only)
TIME	Max-Time	11	Default for the maximum time that each job step may run (JES2 only)
ACCT	Acct	4	Requirement for the account number on a JCL JOB statement (JES2 only)
COPY	Cpy	3	Queue jobs for output processing as though TYPRUN=COPY were specified on the JOB statement (JES2 only)
JOURNAL	Jrnl	4	Save job-related information in a job journal
PGMRNAME	PgNm	4	Programmer name required on a JCL JOB statement (JES2 only)
RESTART	Rst	3	Requeue for execution jobs that had been executing before the IPL of the system was repeated and a JES2 warm start was performed
SCAN	Scn	3	Queue jobs for output processing immediately after JCL conversion (JES2 only)
IEFUJP	UJP	3	Take the IEFUJP exit when a job is purged (JES2 only)
IEFUSO	USO	3	Take the IEFUSO installation exit when the SYSOUT limit is reached for a job (JES2 only)
TYPE6	Tp6	3	Produce type 6 SMF records (JES2 only)
TYPE26	Tp26	4	Produce type 26 SMF records (JES2 only)
CONDPURG	CPr	3	Conditionally purge system data sets in this time-sharing user class (JES2 only)
JMCLASS	MC	2	Message class for all time-sharing sessions (default logon message class for all TSO/E logons) (JES2 only)
SCHENJC	Scheduling-Env	16	Scheduling environment for the job (JES2 only)
JESLOG	JESLog	13	Spin options for the jobs' JES2 joblog and messagelog

Table 45. Columns on the JC Panel (continued)

Column name	Title (Displayed)	Width	Description
XBMPROC	XBMProc	8	Procedure name for XBM/2 job (JES2 only)
DUPJOB	DupJob	6	Duplicate job names acceptable for this class (JES2 only)
SDEPTH	SDepth	6	Setup depth (JES3 only)
PARTNAM	PartName	8	Spool partition name (JES3 only)
PRITRK	PriTrk	6	Primary track group allocation (JES3 only)
SECTRK	SecTrk	6	Secondary track group allocation (JES3 only)
PRIO	Prio	4	Priority (JES3 only)
JOBRC	JobRC	6	Indicates whether the last (LASTRC) or max (MAXRC) step completion code is reported as the job completion code (JES2 only)
CLACTIVE	Active	6	Indicates if the class is currently active (JES2 only)
DSENQSHR	DSEnqShr	8	Indicates if JES should change data set enqueues to shared access when exclusive access is not required (JES2 only)
SYSSYM	SysSym	8	Indicates if system symbols are allowed in batch jobs (JES2 only)

Job Data Set panel (JDS)

The Job Data Set panel allows the user to display information about SYSOUT data sets for a selected job, started task, and TSO user.

When the JDS panel is accessed from the DA, I, or ST panel, the values for all the columns are obtained from the spool data set. When the JDS panel is accessed from the H or O panel, the values for some columns are obtained from in-storage control blocks.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 46. Columns on the JDS Panel

Column name	Title (Displayed)	Width	Description	Delay
DDNAME	DDNAME	8	DD name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.	
STEPN	StepName	8	Job step name	
PROCS	ProcStep	8	Procedure step name	
DSID	DSID	4	Data set ID number	
OWNERID	Owner	8	User ID of SYSIN/SYSOUT owner, or default values of ++++++++ or ????????, if user ID not defined to RACF 1.9 and later	
OCLASS	C	1	JES output class	
DESTN	Dest	18	JES print destination name	
RECCNT	Rec-Cnt	7	Data set record count	
PAGECNT	Page-Cnt	8	Data set page count. Blanks if not page-mode data.	
BYTECNT	Byte-Cnt	8	Data set byte count	

Table 46. Columns on the JDS Panel (continued)

Column name	Title (Displayed)	Width	Description	Delay
COPYCNT	CC	2	Data set copy count	
DEST	Rmt	5	JES2 print routing. Remote number if routing is not local (JES2 only).	
NODE	Node	5	JES2 print node (JES2 only)	
OGNAME	O-Grp-N	8	Output group name (JES2 only)	
SECLABEL	SecLabel	8	Security label of data sets	
PRMODE	PrMode	8	Data set process mode	
BURST	Burst	5	Data set burst indicator	
DSDATE	CrDate-CrTime	19	Data set creation date and time, or, if ***** N/A *****, the creation date and time were not available.	
FORMS	Forms	8	Output form number	
FCBID	FCB	4	Output FCB ID	
UCSID	UCS	4	Output UCS ID	
WTRID	Wtr	8	Output special writer ID or data set ID	
FLASHID	Flash	5	Output flash ID	
FLASHC	FlashC	6	Flash count	
SEGID	SegID	5	Data set segment number	
DSNAME	DSName	44	Output data set name	
CHARS	Chars	20	Character arrangement table names	
CPYMOD	CpyMod	6 (JES2) 8 (JES3)	Copy modification module name	
CPYMODFT	CpyModFT	8	Copy modification table reference character (JES2 only)	
PAGEDEF	PageDef	7	Library member used by PSF to specify print characteristics such as page width	X
FORMDEF	FormDef	7	Library member used by PSF to specify print characteristics such as overlays	X
ODTITLE	Title	20	Report title to be printed on separator pages . This column can be expanded to 60.	X
ODNAME	Name	20	Name to be printed on separator pages . This column can be expanded to 60.	X
ODBLDG	Building	10	Building identification to be printed on separator pages . This column can be expanded to 60.	X
ODDEPT	Department	10	Department identification to be printed on separator pages . This column can be expanded to 60.	X
ODROOM	Room	10	Room identification to be printed on separator pages. This column can be expanded to 60.	X
ODADDR	Address-Line1	20	Address to be printed on separator pages . This column can be expanded to 60	X
ODADDR2	Address-Line2	20	Output address line 2. This column can be expanded to 60.	X

Table 46. Columns on the JDS Panel (continued)

Column name	Title (Displayed)	Width	Description	Delay
ODADDR3	Address-Line3	20	Output address line 3. This column can be expanded to 60.	X
ODADDR4	Address-Line4	20	Output address line 4. This column can be expanded to 60.	X
OUTBIN	OutBn	5	Output bin	X
COMSETUP	ComSetup	8	Setup options for microfiche printers	X
FORMLEN	FormLen	10	Form length	X
COLORMAP	ColorMap	8	AFP resource for the data set containing color translation information	X
INTRAY	ITy	3	Paper source	X
OVERLAYB	OverlayB	8	Overlay for the back of each sheet	X
OVERLAYF	OverlayF	8	Overlay for the front of each sheet	X
OFFSETXB	OffsetXB	13	Offset in the x direction from the page origin for the back of each page	X
OFFSETXF	OffsetXF	13	Offset in the x direction from the page origin for the front of each page	X
OFFSETYB	OffsetYB	13	Offset in the y direction from the page origin for the back of each page	X
OFFSETYF	OffsetYF	13	Offset in the y direction from the page origin for the front of each page	X
PORTNO	Port	5	Number of the TCP/IP port where the FSS connects to the printer	X
ODNOTIFY	Notify	17	Print complete notification message	X
ODUSRLIB	UserLib	44	Libraries containing Advanced Function Printing (AFP) resources to be used by Print Services (PSF) when processing SYSOUT data sets.	X
USERDATA	UserData1	60	User data. Access values 2-16 by typing + alone in the column.	X
AFPPARMS	AFPParms	54	Names a data set that contains the parameters to be used by the AFPPrint Distributor	X
QUEUE	Queue	5	Names the JES3 queue the dataset is on (TCP, BDT, HOLD, WTR) (JES3 only)	
SPIN	Spin	4	Indicates whether this is a spin data set	
SELECT	Sel	3	Indicates whether the data set is selectable	
TP	TP	3	Indicates whether SYSOUT was created by a transaction program.	
TPJNAME	TPJName	8	Job name of the transaction program that created the data set	
TPJOBID	TPJobID	8	Job ID of the transaction program that created the data set	
TPTIMER	TRd-Time	8	Start time for entry of the transaction program	
TPDATER	TRd-Date	8	Start date for entry of the transaction program	
TPTIMEE	TSt-Time	8	Start time for execution of the transaction program	

Table 46. Columns on the JDS Panel (continued)

Column name	Title (Displayed)	Width	Description	Delay
TPDATEE	TSt-Date	8	Start date for execution of the transaction program	
TPACCT	TPAcct	8	Account number of the transaction program	
RECFM	RecFm	5	Record format	
SPINNABLE	W	3	Indicates if the data set is open and spinnable (JES2 only)	
OCOPYCNT	OCopyCnt	8	Copy count specified with COPYCNT. Used by InfoPrint printers.	X
LRECL	LRecL	5	Logical record length	

Job 0 (J0)

The Job 0 panel allows the user to display information about SYSOUT data sets for a JES3 job 0.

The values for all the columns are obtained from the spool data set.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 47. Columns on the J0 Panel

Column name	Title (Displayed)	Width	Description
NAME	DSPNAME	8	DSP that created the data. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
DSID	DSID	4	Data set ID number
OWNERID	Owner	8	User ID of SYSIN/SYSOUT owner, or default values of ++++++++ or ????????, if user ID not defined to RACF 1.9 and later
OCLASS	C	1	JES3 output class
COPYCNT	CC	2	Data set copy count
PRMODE	PrMode	8	Data set process mode
BURST	Burst	5	Data set burst indicator
FORMS	Forms	8	Output form number
FCBID	FCB	4	Output FCB ID
UCSID	UCS	4	Output UCS ID
WTRID	Wtr	8	External writer name
FLASHID	Flash	5	Output flash ID
FLASHC	FlashC	6	Flash copies
SEGID	SegID	5	Data set segment number
CHARS	Chars	21	Character arrangement table names
CPYMOD	CpyMod	8	Copy modification module name
QUEUE	Queue	5	Queue the dataset is on (TCP, BDT, HOLD, WTR)
DESTN	Dest	18	SYSOUT destination
SECLABEL	SecLabel	8	Security label

Table 47. Columns on the J0 Panel (continued)

Column name	Title (Displayed)	Width	Description
DSDATE	CrDate-CrTime	19	Data set creation date and time, or, if ***** N/A *****, the creation date and time were not available.
SPIN	Spin	4	Indicates whether this is a spin data set
SELECT	Sel	3	Indicates whether the data set is selectable
RECCNT	Rec-Cnt	7	Data set record count
PAGECNT	Page-Cnt	8	Data set page count. Blank if not page-mode data.
BYTECNT	Byte-Cnt	8	Data set byte count
RECFM	RecFm	5	Record format
DDNAME	DDName	8	DD name
DSNAME	DSName	44	Data set name
STEPN	StepName	8	Job step that created the SYSOUT
PROCS	ProcStep	8	Procedure step that created the SYSOUT

Lines panel (LI)

The Lines panel allows the user to display information about JES lines and their associated transmitters and receivers.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 48. Columns on the LI Panel

Column name	Title (Displayed)	Width	Description
DEVNAME	DEVICE	12	Device name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
STATUS	Status	8	Line status
UNIT	Unit	5	Line address or type
NNODE	Node	8	Node that the line is connected to
JNAME	JobName	8	Job name
JOBID	JobID	8	JES2 job ID
JTYPE	no default	4	Type of address space
JNUM	JNum	6	JES2 job number
OWNERID	Owner	8	User ID of owner
RECPRT	Proc-Lines	10	Number of lines processed for the job.
RECCNT	Tot-Lines	10	Number of lines in the job.
TYPE	Type	4	Type of line
LINELIM	Line-Limit	13	Line limit for the line (JES2 only)
PAGELIM	Page-Limit	13	Page limit for the line (JES2 only)
PRTWS	Work-Selection	14	Line work selection criteria (JES2 only)
SESSION	Session	8	Session name (JES2 only)
TOTERRS	Tot-Errs	8	Error count (JES2 only)
AUTODISC	ADisc	5	Line disconnect option
CODE	Code	4	BSC adaptor code

Table 48. Columns on the LI Panel (continued)

Column name	Title (Displayed)	Width	Description
COMPRESS	Comp	4	BSC data compression option
APPLID	ApplID	8	Application name for NJE line (JES2 only)
DUPLEX	Duplex	6	BSC line mode
INTERFAC	Intf	4	BSC adapter interface
LINECCHR	LineCChr	8	BSC line control characters configuration (JES2 only)
LOG	Log	3	Message logging option (JES2 only)
REST	Rest	4	Resistance rating of line (JES2 only)
SPEED	Speed	5	Speed of the line
PTRACE	Tr	3	Trace I/O option
TRANSPAR	Transp	6	BSC transparency feature
PSWD	Password	8	Password
DISC	Discon	9	Disconnect status: NO, INTERRUPT, or QUIESCE (only for active lines).
RMTSHR	RmtShr	6	Indicates whether the line is allowed to be dedicated (JES2 only)
JRNUM	JRNum	7	Job receivers associated with the line, either a count or D, for default (JES2 only)
JTNUM	JTNum	7	Job transmitters associated with the line, either a count or D, for default (JES2 only)
SRNUM	SRNum	7	SYSOUT receivers associated with the line, either a count or D, for default (JES2 only)
STNUM	STNum	7	SYSOUT transmitters associated with the line, either a count or D, for default (JES2 only)
SYSNAME	SysName	8	System Name
DSYSID	SysID	5	JES2 member name (JES2 only)
JESNAME	JESN	4	JES subsystem name
JESLEVEL	JESLevel	8	z/OS JES2 level
DEVSECLB	DSecLabel	9	Security label of the device (JES2 only)
SOCKETN	SocketN	8	Socket name (JES2 only)
IPADDR	IPAddr	24	IP address (JES2 only)
IPNAME	IPName	32	IP name (JES2 only)
PORT	Port	5	TCP/IP port number (JES2 only)
PORTNAME	PortName	8	TCP/IP port name. Blank if a port number has been set explicitly. (JES2 only)
SECURE	Secure	6	Secure socket (JES2 only)
NSNAME	NSName	8	Network server name (JES2 only)
ANODE	ANode	8	Adjacent node (JES2 only)
LINELIML	Line-Lim-Lo	11	Line limit, minimum (JES2 only)
LINELIMH	Line-Lim-Hi	11	Line limit, maximum (JES2 only)
PAGELIML	Page-Lim-Lo	11	Page limit, minimum (JES2 only)
PAGELIMH	Page-Lim-Hi	11	Page limit, maximum (JES2 only)
CTRACE	CTr	3	Common tracing (JES2 only)

Table 48. Columns on the LI Panel (continued)

Column name	Title (Displayed)	Width	Description
VTRACE	VTr	3	Verbose tracing (JES2 only)
JTRACE	JTr	3	JES tracing (JES2 only)
CONNECT	Connect	7	Connect line automatically (JES2 only)
CTIME	Conn-Int	10	Connection interval in minutes (JES2 only)
RESTART	Restart	8	Restart line automatically (JES2 only)
RTIME	Rest-Int	10	Restart interval, in minutes (JES2 only)
SODISP	SODsp	5	Selection output disposition 1 (JES2 only)
SODISP2	SODsp2	5	Selection output disposition 2 (JES2 only)
SODISP3	SODsp3	5	Selection output disposition 3 (JES2 only)
SODISP4	SODsp4	5	Selection output disposition 4 (JES2 only)

Note:

[1] JNUM is not included in the default field list.

Link List panel (LNK)

The LNK panel displays the data sets in the link list (lnklst) for each system in the sysplex. Only data sets in the current lnklst set are shown.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 49. Columns on the LNK Panel

Column name	Title (Displayed)	Width	Description
DSNAME	DSNAME	13-44 (Varies based on longest name.)	Data set name.This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
SEQ	Seq	3	Sequence number
VOLSER	VolSer	6	Volume serial
BLKSIZE	BlkSize	7	Data set block size
EXTENT	Extent	6	Number of extents
SMS	SMS	3	SMS indicator. YES if the data set is SMS managed. Otherwise, NO.
APF	APF	3	APF indicator. YES if the data set is APF authorized. Otherwise, NO.
LRECL	LRecL	5	Logical record length
DSORG	DSOrg	5	Data set organization
RECFM	RecFm	5	Record format
CRDATE	CrDate	8	Data set creation date
REFDATE	RefDate	8	Data set last referenced date
SETNAME	SetName	16	Link list set name
SYSNAME	SysName	8	System name
SYSLEVEL	SysLevel	25	Operating system level

Link Pack Area panel (LPA)

The LPA panel shows the data sets in the link pack area (LPA) for each system in the sysplex.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 50. Columns on the LPA Panel

Column name	Title (Displayed)	Width	Description
DSNAME	DSNAME	13-44 (Varies based on longest name.)	Data set name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
SEQ	Seq	3	Sequence number
VOLSER	VolSer	6	Volume serial
BLKSIZE	BlkSize	7	Data set block size
EXTENT	Extent	6	Number of extents
SMS	SMS	3	SMS indicator. YES if the data set is SMS managed. Otherwise, NO.
APF	APF	3	APF indicator: YES if the data set is APF authorized. Otherwise, NO.
LRECL	LRecL	5	Logical record length
DSORG	DSOrg	5	Data set organization
RECFM	RecFm	5	Record format
CRDATE	CrDate	8	Data set creation date
REFDATE	RefDate	8	Data set last referenced date
SYSNAME	SysName	8	System name
SYSLEVEL	SysLevel	25	Operating system level

Multi-Access Spool panel (MAS) and JESPLEX (JP) panel

The Multi-Access Spool (MAS) panel simplifies the display and control of members in a JES2 MAS. The analogous JES3 JESPLEX panel simplifies the display and control of members in a JES3 JESPLEX. They share a single field list.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 51. Columns on the MAS and JP Panel

Column name	Title (Displayed)	Width	Panel	Description
NAME	NAME	4 (JES2) 8 (JES3)	MAS, JP	Member name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
STATUS	Status	12	MAS, JP	Member status
SYSID	SID	3	MAS	The system ID number
PREVCKPT	PrevCkpt	8	MAS	Number of seconds elapsed since the previous checkpoint (ss.hh format)
CKPTHOLD	Hold	8	MAS	Checkpoint hold in hundredths of seconds

Table 51. Columns on the MAS and JP Panel (continued)

Column name	Title (Displayed)	Width	Panel	Description
ACTHOLD	ActHold	8	MAS	Actual checkpoint hold in hundredths of seconds
DORMANCY	Dormancy	11	MAS	Checkpoint dormancy (minimum,maximum). Format in hundredths of seconds.
ACTDORM	ActDorm	7	MAS	Actual checkpoint dormancy in hundredths of seconds
SYNCTOL	SyncTol	7	MAS	Checkpoint synchronization tolerance in seconds
SYSMODE	Ind	3	MAS	Independent mode
RSYSID	RSID	4	MAS	Name of member performing a $ESYS
SYSNAME	SysName	8	MAS, JP	System name of the MVS image on which this JES system is active
VERSION	Version	8	MAS, JP	JES version the system is running
LASTCKPT	Last-Checkpoint	22	MAS	Last date and time checkpoint was taken
COMCHAR	C	1 (JES2) 8 (JES3)	MAS, JP	Command character
JESNAME	JESN	4	MAS, JP	JES subsystem name
SLEVEL	SLevel	6	MAS, JP	JES service level
BOSS	Boss	4	MAS	Indicates if this member is a manager or "boss" of WLM service class queues
GLOBAL	Global	6	JP	JES3 Global member indicator
COMMAND	Command	8	MAS	Command in progress
TYPE	Start-Type	18	MAS, JP	Last start type for the member
DATEE	Start-Date-Time	19	MAS, JP	Date and time the member was started
LASTGCON	LastGCon-Date-Time	18	JP	Last time the global was contacted
PTRACK	PrimTG	6	JP	Primary track group allocation
STRACK	SecTG	6	JP	Secondary track group allocation
WTOLIM	WTOLim	6	JP	WTO message limit
WTOINT	WTOInt	6	JP	WTO message interval
PCSALIM	PBufCSA	7	JP	Protected buffer CSA limit
PAUXLIM	PBufAux	7	JP	Protected buffer JES3 auxiliary limit
PFIXED	PBufFixed	9	JP	Fixed protected buffers
USRPAGE	UserPages	9	JP	User pages per open SYSOUT dataset
SELMNAME	SelectModeName	14	JP	Selection mode name
SPARTN	PartName	8	JP	Spool partition name
MSGPRF	MsgPrefix	11	JP	Message prefix
MSGDEST	MsgDest	7	JP	Message destination
CONSTAT	ConnStat	13	JP	Connect status
ATTSTAT	AttStat	11	JP	Attach status

Network Connections (NC)

The Network Connections panel allows the user to display information about JES networking connections to an adjacent node.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 52. Columns on the NC Panel

Column name	Title (Displayed)	Width	Description
DEVNAME	DEVICE	10	Name of the connection, transmitter or receiver. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
STATUS	Status	8	Device status
TYPE	Type	4	Connection type (SNA, BSC, TCP)
ANODE	ANode	8	Adjacent node
JNAME	Jobname	8	Job name of job being processed
JOBID	JobID	8	JES job ID of job being processed
JTYPE	JType	8	Type of address space being processed
OWNERID	Owner	8	User ID of job creator
RECPRT	Proc-Lines	10	Number of lines processed for the job
RECCNT	Tot-Lines	10	Number of lines in the job
LINE	Line	5	Number of line to use (JES2 only)
UNIT	Unit	5	Unit associated with line
JRNUM	JRNum	5	Job receiver count
JTNUM	JTNum	5	Job transmitter count
SRNUM	SRNum	5	SYSOUT receiver count
STNUM	STNum	5	SYSOUT transmitter count
CONNECT	Connect	7	Connect automatically (JES2 only)
CTIME	Conn-Int	8	Connection interval (JES2 only)
PTRACE	Tr	3	Tracing (JES2 only)
CTRACE	CTr	3	Common tracing
JTRACE	JTr	3	JES tracing
VTRACE	VTr	3	Verbose tracing
LOGMODE	LogMode	8	Logon mode table entry (JES2 only)
REST	Rest	5	Resistance of the connection (JES2 only)
COMPACT	Compact	8	Compaction table name (JES2 only)
IPADDR	IPAddr	24	IP address (JES2 only)
IPNAME	IPName	32	IP host name
PORT	Port	5	TCP/IP port number
PORTNAME	PortName	16	TCP/IP port name (JES2 only)
SECURE	Secure	6	Secure (TLS) connection
LOGON	Logon	5	Number of the associated LOGON device (JES2 only)
NETSRV	Netsrv	5	Number of the associated NETSRV device (JES2 only)
RELCONN	RelConn	8	Related connection name

Table 52. Columns on the NC Panel (continued)

Column name	Title (Displayed)	Width	Description
SRVNAME	SrvName	10	Name of the associated server device
DSECLABEL	DSecLabel	9	Security label of the adjacent node (JES2 only)
SYSNAME	SysName	8	System name
DSYSID	SysID	5	JES2 member name (JES2 only)
JESNAME	JESN	4	JES subsystem name
JESLEVEL	JESLevel	8	z/OS JES version and release
PRTWS	Work-Selection	14	Work selection criteria (JES2, transmitters and receivers)
LINELIM	Line-Limit	13	Line limit for selection (JES2, transmitters and receivers)
PAGELIM	Page-Limit	13	Page limit for selection (JES2, transmitters and receivers)
LINELIML	Line-Lim-Lo	11	Line limit, minimum (JES2 only)
LINELIMH	Line-Lim-Hi	11	Line limit, maximum (JES2 only)
PAGELIML	Page-Lim-Lo	11	Page limit, minimum (JES2 only)
PAGELIMH	Page-Lim-Hi	11	Page limit, maximum (JES2 only)
SODISP	SODsp	5	Selection output disposition (JES2 only)
SODISP2-4	SODsp2-4	6	Selection output disposition 2-4 (JES2 only)

Network Servers (NS)

The Network Servers panel allows the user to display information about JES server-type networking devices on the node.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 53. Columns on the NS Panel

Column name	Title (Displayed)	Width	Description
DEVNAME	DEVICE	10	Name of the network server. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
STATUS	Status	8	Device status
DSPNAME	DSPName	8	Dynamic support program name (JES3 only)
APPL	Appl	8	Application name (JES2 only)
SOCKET	Socket	8	Socket name (JES2 only)
STACK	Stack	8	Name of the TCP/IP stack
RESTART	Restart	8	Restart the device automatically (JES2 only)
RTIME	Rest-Int	10	Restart interval (minutes) (JES2 only)
PTRACE	Tr	3	Tracing (JES2 only)
CTRACE	CTr	3	Common tracing
VTRACE	VTr	3	Verbose tracing
JTRACE	JTr	3	JES tracing
LOG	Log	3	Log activity (JES2 only)

Table 53. Columns on the NS Panel (continued)

Column name	Title (Displayed)	Width	Description
ASID	ASID	5	ASID of the network server
SRVJOBNM	SrvJobNm	8	Job name of the network server address space
PASSWORD	Password	8	Password (SET or NOTSET) (JES2 only)
IPNAME	IPName	32	Local TCP/IP host name
PORT	Port	5	Local TCP/IP port number
PORTNAME	PortName	16	Local TCP/IP port name (JES2 only)
SECURE	Secure	6	Secure (TLS) socket
SYSNAME	SysName	8	System name
DSYSID	SysID	5	JES2 member name (JES2 only)
JESNAME	JESN	4	JES subsystem name
JESLEVEL	JESLevel	8	z/OS JES level
DEVSECLB	DSecLabel	9	Security label of the device (JES2 only)

Nodes panel (NO)

The Nodes panel allows the user to display information about JES nodes.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 54. Columns on the NO Panel

Column name	Title (Displayed)	Width	Description
NUMBER	NUMBER	5	Node number (JES2 only). For JES2, this is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
NODENAME	NodeName	8	Node name. For JES3, this is the fixed field, and is ignored if coded on an FLD statement or ISFFLD macro.
STATUS	Status	21	Node status, By default, this shows status for the first path. Increase the width (up to 43) to show the status for the second path.
AUTH	Authority	17	Authority of the node (JES2 only)
TRANS	Trans	6	What the local node transmits to the specified node (JES2 only)
RECV	Recv	6	What the local node receives from the specified node (JES2 only)
HOLD	Hold	4	Job hold indicator for the local node
PENCRYPT	PEn	3	Password encryption indicator (JES2 only)
ENDNODE	End	3	Eligibility for store-and-forward operations (JES2 only)
RESIST	Rest	4	Resistance rating of the connection (JES2 only)
SENTREST	SentRs	6	Whether the resistance from an adjacent node is used in calculating the resistance of an adjacent connection (JES2 only)
COMPACT	Cp	2	Compaction table number for outbound compaction when communicating with this node (JES2 only)

Table 54. Columns on the NO Panel (continued)

Column name	Title (Displayed)	Width	Description
LINE	Line	4	Line dedicated to the NJE session for with this application (JES2 only)
LNAME	LineName	8	Line dedicated to NJE for this node (JES3 only)
LOGMODE	LogMode	8	Logon mode table entry for this application (JES2 only)
PATHMGR	PMg	3	Indicator of whether NCC records relevant to the path manager should be sent to this node (JES2 only)
PRIVATE	Prv	3	Private indicator for the connection between this node and an adjacent node (JES2 only)
SUBNET	Subnet	8	Name of the subnet that should include this node (JES2 only)
NTRACE	Tr	3	Trace option (JES2 only)
VERIFYP	VerifyP	8	Password received from the node
SENDP	SendP	8	Password sent to the node
LOGON	Logon	5	Number of the local logon DCT (1-999) which should be use when specifying connections to the application. The default value of 0 indicates that the logon DCT defined with the lowest number is to be. (JES2 only)
SYSNAME	SysName	8	System name
DSYSID	SysID	5	JES2 member name (JES2 only)
JESNAME	JESN	4	JES subsystem name
JESLEVEL	JESLevel	8	JES version and release
NETSRV	NetSrv	6	Network server number (JES2 only)
DEVSECLB	DSecLabel	9	Security label of the device (JES2 only)
MAXRETR	MaxRetries	6	Number of retries to attempt before ending the BSC NJE line (JES3 only)
PATH	Path	8	Name of the adjacent node in the path (JES3 only)
PTYPE	PType	5	Protocol type (JES3 only)
BDTNAME	BDTName	8	Bulk Data Transfer (BDT) ID (JES3 only)
PARTNAM	PartName	8	Name of the spool partition to which JES3 writes spool data for all jobs from that node (JES3 Only)
MAXLINES	MaxLines	3	Maximum number of lines for the node. (JES3 Only)
DIRECT	Direct	6	Specifies whether the node can be directly attached only
SSIGNON	SSignon	7	Specifies whether secure signon protocol is to be used
JTNUM	JTNum	5	Number of job transmitters associated with the TCP/IP node (JES3 only)
JRNUM	JRNum	5	Number of job receivers associated with the TCP/IP node (JES3 only)
STNUM	STNum	5	Number of SYSOUT transmitters associated with the TCP/IP node (JES3 only)
SRNUM	SRNum	5	Number of SYSOUT receivers associated with the TCP/IP node (JES3 only)
SECURE	Secure	6	Use secure (TLS) socket (JES3 only)

Table 54. Columns on the NO Panel (continued)

Column name	Title (Displayed)	Width	Description
PWCNTL	PwCntl	8	Password encryption control (JES3 only)
XNAMEREQ	XNameReq	8	Specifies whether inbound SYSOUT can be held for processing by an external writer if no external writer name was supplied (JES3 only)
CONNECT	Connect	7	Automatically connect (JES2) or reconnect (JES3)
CTIME	Conn-int	8	Connection interval (minutes)
BUFSIZE	BufSz	5	Buffer size (JES3 only)
STREAM	Strm	4	Number of concurrent streams (JES3 only)
PRTDEF	PrtDef	8	Print class default for networking output received at the home node (JES3 only)
PRTTSO	PrtTSO	8	TSO data set default class for networking output received at the home node (JES3 only)
PRTXWTR	PrtXwtr	8	External writer data set default class for networking output received at the home node (JES3 only)
PUNDEF	PunDef	8	Punch class default for networking output received at the home node (JES3 only)
NETPR	NetPr	5	Number of logical network printers on the home node (JES3 only)
NETPU	NetPu	5	Number of logical network punches on the home node (JES3 only)
CTCNODE	CTC	5	Channel to channel node (JES3 only)

Output Descriptors panel (OD)

The OD panel allows the user to display JES output descriptors.

In a JES2 environment, columns can be overtyped only if you accessed the OD panel from the O or H panel, or from a JDS panel that was accessed from the O or H panel.

When you overtype a column on the OD panel, the change applies to all data sets for that group.

Table 55. Columns on the OD Panel

Column name	Title (Displayed)	Width	Description	Delay
DDNAME	DDNAME	8	DDname of the data set. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.	X
PAGEDEF	PageDef	6	Library member used by PSF to specify print characteristics such as page width	X
FORMDEF	FormDef	6	Library member used by PSF to specify print characteristics such as overlays	X
ODTITLE	Title	60	Report title to be printed on new separator pages	X
ODNAME	Name	60	Name to be printed on separator pages	X
ODBLDG	Building	60	Building location to be printed on separator pages	X

Table 55. Columns on the OD Panel (continued)

Column name	Title (Displayed)	Width	Description	Delay
ODDEPT	Department	60	Department to be printed on separator pages	X
ODROOM	Room	60	Room to be printed on separator pages	X
ODADDR	Address	60	Address to be printed on separator pages. There can be 1 to 4 lines, each with a maximum length of 60.	X
OUTBIN	OutBin	5	Output bin	X
COMSETUP	ComSetup	8	Setup options for microfiche printers	X
FORMLEN	FormLen	10	Form length	X
COLORMAP	ColorMap	8	AFP resource for the data set containing color translation information	X
INTRAY	InTray	3	Paper source	X
OVERLAYB	OverlayB	8	Overlay for the back of each sheet	X
OVERLAYF	OverlayF	8	Overlay for the front of each sheet	X
OFFSETXB	OffsetXB	13	Offset in the x direction from the page origin for the back of each page	X
OFFSETXF	OffsetXF	13	Offset in the x direction from the page origin for the front of each page	X
OFFSETYB	OffsetYB	13	Offset in the y direction from the page origin for the back of each page	X
OFFSETYF	OffsetYF	13	Offset in the y direction from the page origin for the front of each page	X
PORTNO	PortNo	6	Number of the TCP/IP port where the FSS connects to the printer	X
ODNOTIFY	Notify	17	Print complete notification message. There can be 1 to 4 user IDs, each with a maximum length of 17.	X
ODUSRLIB	UserLib	44	Libraries containing Advanced Function Printing (AFP) resources to be used by Print Services (PSF) when processing SYSOUT data sets. There can be 1 to 8 library names, each with a maximum length of 44.	X
RETAINS	RetainS	8	Retain time for successful transmissions	X
RETAINF	RetainF	8	Retain time for unsuccessful attempts	X
RETRYL	RetryL	3	Maximum number of retries	X
RETRYT	RetryT	8	Time between retries	X
PRINTO	PrtOptns	16	Entry in the PrintWay™ options data set	X
PRINTQ	PrtQueue	60	Print queue name. There can be 2 lines for this column, each with a maximum length of 60 characters.	X
IPDEST	IP Destination	60	IP address or TCP/IP name. There can be 2 lines for this column, each with a maximum length of 60 characters.	X
USERDATA	UserData	60	User data. There can be 16 lines, each with a maximum length of 60.	X

Table 55. Columns on the OD Panel (continued)

Column name	Title (Displayed)	Width	Description	Delay
AFPPARMS	AFPParms	54	Names a data set that contains the parameters to be used by the AFPPrint Distributor	X
OCOPYCNT	OCopyCnt	10	Copy count specified with COPYCNT. Used by InfoPrint printers.	X

Output Queue panel (O)

The Output Queue panel allows the user to display information about SYSOUT data sets for jobs, started tasks, and TSO users on any *nonheld* JES output queue.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 56. Columns on the O Panel

Column name	Title (Displayed)	Width	Description	Delay
JNAME	JOBNAME	8	Job name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.	
JNUM	JNum[1]	6	JES job number	
JOBID	JobID	8	JES job ID or work ID	
OWNERID	Owner	8	User ID of SYSIN/SYSOUT owner, or default values of ++++++++ or ????????, if user ID not defined to RACF	
DPRIO	Prty	4	JES output group priority	
OCLASS	C	1	JES output class	
FORMS	Forms	8	Output form number	
DESTN	Dest	18	JES print destination name	
RECCNT	Tot-Rec	9	Output total record count (lines). Blank for page-mode data.	
RECPRT	Prt-Rec	9	The number of lines printed. Blank for page-mode data. (JES2 only)	
PAGECNT	Tot-Page	9	Output page count. Blank if not for page-mode data.	
PAGEPRT	Prt-Page	9	Output pages printed. Blank if not for page-mode data. (JES2 only)	
DEVID	Device	18	Output device name (only if it is printing)	
STATUS	Status	11	JES job status	
SECLABEL	SecLabel	8	Security label of output group	
DSYSID	SysID	5	System on which the output is printing (only if it is printing) (JES2 only)	
DEST	Rmt	5	JES2 print routing. Remote number if routing is not local. (JES2 only)	
NODE	Node	5	JES2 print node (JES2 only)	
OGNAME	O-Grp-N	8	Output group name (JES2 only)	
OGID	OGID1	5	Output group ID 1 (JES2 only)	
OGID2	OGID2	5	Output group ID 2 (JES2 only)	
JPRIO	JP	2	JES job priority	

Table 56. Columns on the O Panel (continued)

Column name	Title (Displayed)	Width	Description	Delay
FCBID	FCB	4	Output FCB ID	
UCSID	UCS	4	Output UCS ID (print train required)	
WTRID	Wtr	8	Output external writer name	
FLASHID	Flash	5	Output flash ID	
BURST	Burst	5	3800 burst indicator	
PRMODE	PrMode	8	Printer process mode	
OUTDISP	ODisp	5	JES2 output disposition (JES2 only)	
DSDATE	CrDate	10	Output creation date. Length can be changed to 19 to produce the date and time. (JES2 only)	
OHREASON	OHR	3	Output hold reason code	
OHRSNTXT	Output-Hold-Text	37	Output hold reason text	
OFFDEVS	Offs	4	List of offload devices for a job or output that has been offloaded (JES2 only)	
RETCODE	Max-RC	10	Return code information for the job	
JTYPE	Type	4	Type of address space	
ROOMN	RNum	8	JES2 job room number	X
PNAME	Programmer-Name	20	JES programmer name field	X
ACCTN	Acct	4 (JES2) 8 (JES3)	JES account number	X
NOTIFY	Notify	8	TSO user ID from NOTIFY parameter on job card	X
ISYSID	ISys	4 (JES2) 8 (JES3)	JES input system ID	X
TIMER	Rd-Time	8	Time that the job was read in	X
DATER	Rd-Date	8	Date that the job was read in	X
ESYSID	ESys	4 (JES2) 8 (JES3)	JES execution system ID	X
TIMEE	St-Time	8	Time that execution began	X
DATEE	St-Date	8	Date that execution began	X
TIMEN	End-Time	8	Time that execution ended	X
DATEN	End-Date	8	Date that execution ended	X
ICARDS	Cards	5	Number of cards read for job	X
JCLASS	JC	1 or 8	JES input job class. Default width expands to 8 if there are long class names in the MAS.	
MCLASS	MC	2	Message class of job	X
SUBGROUP	SubGroup	8	Submittor group	X
JOBACCT1	JobAcct1[1]	20	Job accounting field 1	X
JOBACCT2	JobAcct2[1]	20	Job accounting field 2	X
JOBACCT3	JobAcct3[1]	20	Job accounting field 3	X
JOBACCT4	JobAcct4[1]	20	Job accounting field 4	X
JOBACCT5	JobAcct5[1]	20	Job accounting field 5	X
JOBCORR	JobCorrelator	32	User portion of the job correlator (JES2 only)	

Table 56. Columns on the O Panel (continued)

Column name	Title (Displayed)	Width	Description	Delay
DATETIMER	Rd-DateTime	19	Date and time that the job was read in. This column is displayed only with the SDSF task of z/OSMF. It combines the information in the Rd-Date and Rd-Time columns.	X
DATETIMEE	St-DateTime	19	Date and time that execution began. This column is displayed only with the SDSF task of z/OSMF. It combines the information in the St-Date and St-Time columns.	X
DATETIMEN	End-DateTime	19	Date and time that execution ended. This column is displayed only with the SDSF task of z/OSMF. It combines the information in the End-Date and End-Time columns.	X

Notes on the table:

1. This column is not included in the default field list.

Page panel (PAG)

The PAG panel shows the paging data sets in use for each system in the sysplex.

Note: RMF and the RMF Monitor 1 tasks must be active in order to see rows on the SDSF PAG display. When this requirement is not met, messages HSF0030E and HSF0028E are seen during SDSFAUX initialization.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 57. Columns on the PAG Panel

Column name	Title (Displayed)	Width	Description
DSNAME	DSNAME	13-44 (Varies based on longest name.)	Data set name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
TYPE	Type	6	Type of data set
SLOTS	Slots	8	Number of slots defined
USENUM	Used	8	Number of slots used
USEPCT	Use%	4	Percentage of total slots in use
VOLSER	VolSer	6	Volume serial
STATUS	Status	8	Data set status
VIO	VIO	3	VIO indicator. YES if data set eligible for VIO.
TOTERRS	IOError	7	Number of I/O errors
SYSNAME	SysName	8	System name
SYSLEVEL	SysLevel	25	Operating system level

PARMLIB panel (PARM)

The PARM panel shows the data sets in the PARMLIB concatenation for each system in the sysplex.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 58. Columns on the PARM Panel

Column name	Title (Displayed)	Width	Description
DSNAME	DSNAME	13-44 (Varies based on longest name.)	Data set name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
SEQ	Seq	3	Sequence number
VOLSER	VolSer	6	Volume serial
BLKSIZE	BlkSize	7	Data set block size
EXTENT	Extent	6	Number of extents
SMS	SMS	3	SMS indicator. YES if the data set is SMS managed. Otherwise, NO.
LRECL	LRecL	5	Logical record length
DSORG	DSOrg	5	Data set organization
RECFM	RecFm	5	Record format
CRDATE	CrDate	8	Data set creation date
REFDATE	RefDate	8	Data set last referenced date
SYSNAME	SysName	8	System name
SYSLEVEL	SysLevel	25	Operating system level

Printer panel (PR)

The Printer panel allows the user to display information about JES printers printing job, started task, and TSO user output.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 59. Columns on the PR Panel

Column name	Title (Displayed)	Width	Description	Delay
DEVNAME	PRINTER	10 [1]	Printer name. This is the fixed field. It is ignored in an FLD statement or ISFFLD macro.	
STATUS	Status	8	Printer status	
GROUP	Group	9	Device group (JES3 only)	
SFORMS	SForms	8	Printer selection form number	
SFORM2-8	SForm2-8	8	Printer selection form names (JES2 only)	
SCLASS	SClass	15	Printer output selection classes	
JNAME	JobName	8	Job name	X
JNUM	JNum[2]	6	JES job number	
JOBID	JobID	8	JES job ID or work ID	X

Table 59. Columns on the PR Panel (continued)

Column name	Title (Displayed)	Width	Description	Delay
OWNERID	Owner	8	User ID of job owner, or default values of ++++++++ or ????????, if user ID not defined to RACF	
RECCNT	Rec-Cnt	7	Number of line-mode records	
RECPRT	Rec-Prt	7	Number of line-mode records printed	
PAGECNT	Page-Cnt	8	Number of output pages	
PAGEPRT	Page-Prt	8	Number of output pages printed	
JPRIO	JP	2	JES job priority	
DPRIO	DP	3	Output data set priority	
OCLASS	C	1	JES output class	
SECLABEL	SecLabel	8	Security label of the output group	
FORMS	Forms	8	Output form number	
FCBID	FCB	4	Output FCB ID	
UCSID	UCS	4	Output UCS ID (print train required)	
WTRID	Writer	8	Output special writer ID or data set ID (JES2 only)	
FLASHID	Flash	5	Output flash ID	
DESTN	Dest	8	JES print destination name (JES2 only)	
BURST	Burst	5	3800 burst indicator	
SEP	Sep	3	Separator page between output groups (JES2 only)	
SEPDS	SepDS	5	Separator page between data sets	
PRMODE	PrMode	8	Printer process mode	
SFCBID	SFCB	5	Printer selection FCB ID	
SUCSID	SUCS	4	Printer selection UCS ID	
SWTRID	SWriter	8	Printer selection writer ID (JES2 only)	
SFLASHID	SFlh	5	3800 Printer selection flash ID	
PRTWS	Work-Selection	40	Printer work selection criteria	
SBURST	SBurst	6	3800 output selection burst mode	
SPRMODE1	SPrMode1	8	Output selection process mode 1	
SPRMODE2	SPrMode2	8	Output selection process mode 2	
SPRMODE3	SPrMode3	8	Output selection process mode 3	
SPRMODE4	SPrMode4	8	Output selection process mode 4	
SDESTN1	SDest1	8	Printer selection destination name 1 (JES2 only)	
SDESTN2	SDest2	8	Printer selection destination name 2 (JES2 only)	
SDESTN3	SDest3	8	Printer selection destination name 3 (JES2 only)	
SDESTN4	SDest4	8	Printer selection destination name 4 (JES2 only)	
SJOBNAME	SJobName	8	Printer selection job name (JES2 only)	
SOWNER	SOwner	8	Printer selection creator ID. Use with the CREATOR work selection criteria. (JES2 only)	
SRANGE	SRange	22	Printer selection job number range (JES2 only)	

Table 59. Columns on the PR Panel (continued)

Column name	Title (Displayed)	Width	Description	Delay
SEPMK	M	3	3800 mark forms control	
NPRO	NPro	4	Nonprocess run-out time in seconds (FSS only). This column is not overtypeable when the printer is active.	
MODE	Mode	4	Control mode of printer (FSS only)	
CKPTLINE	CkptLine	8	Number of lines per logical page (JES2 only)	
CKPTREC	CkptRec	7	Number of logical records per checkpoint (JES3 only)	
CKPTPAGE	CkptPage	8	Number of logical pages per checkpoint	
CKPTSEC	CkptSec	7	Default checkpoint interval (3800-FSS) in seconds	
CKPTMODE	CkptMode	8	Checkpoint mode indicator (take checkpoints based on pages or seconds)	
CPYMOD	CpyMod	7	Copy modification module ID for the 3800 printer	
UNIT	Unit	5	Printer unit name	
PSEL	PSel	4	Preselection option (JES2 only)	
OGNAME	O-Grp-N	8	Output group name for the active job on the printer (JES2 only)	
LINELIM	Line-Limit	21	Printer line limit, *m-n*. An * indicates maximum value. (JES2 only)	
PAGELIM	Page-Limit	21	Printer page limit, *m-n*. Not shown for remote printers. (JES2 only)	
DEVFCB	DFCB	5	Device default FCB name or RESET	
PSETUP	Seup	6	Printer setup mode	
COPYMARK	CopyMark	8	Copymark indicator. Shown only for non-impact or FSS controlled printers.	
PAUSE	Pau	3	Pause mode. Not shown for remote printers.	
PSPACE	K	1	Printer spacing. Not shown for remote printers. (JES2 only)	
PTRACE	Tr	3	Printer tracing	
SEPCHARS	SepChar	7	Separator character value. Not shown for remote printers. (JES2 only)	
UCSVERFY	UCSV	4	UCS verification option. Not shown for remote printers. (JES2 only)	
DEST	Rmt[2]	5	JES print routing (JES2 only)	
NODE	Node[2]	4	JES print node (JES2 only)	
FSSNAME	FSSName	8	FSS defined for the printer	
FSSPROC	FSSProc	8	Name of the proc used to start the FSS	
FSATRACE	FSATrace	8	Internal rolling trace for an FSS printer (JES2 only)	
SYSNAME	SysName	8	System name	
DSYSID	SysID	5	JES member name (JES2 only)	
JESNAME	JESN	4	JES subsystem name	

Table 59. Columns on the PR Panel (continued)

Column name	Title (Displayed)	Width	Description	Delay
JESLEVEL	JESLevel	8	JES level	
DEVSECLB	DSecLabel	9	Security label of the device (JES2 only)	
JTYPE	Type	4	Type of address space	
OGID1	OGID1	5	Output group ID1 for job on printer (JES2 only)	
OGID2	OGID2	5	Output group ID2 for job on printer (JES2 only)	
PTRANS	Trans	8	Data translation	
TRKCELL	TrkCell	7	De-spool the entire track cell (JES2 only)	
NEWPAGE	NewPage	7	Controls how a "skip to channel" is counted (JES2 only)	
HONORTRC	HonorTRC	8	Honor TRC (table reference character) keyword in JCL (JES2 only)	
SVOL	SVol1	6	Spool volumes for work selection (JES2 only)	
SVOL2	SVol2	6	Spool volume 2 for work selection (JES2 only)	
SVOL3	SVol3	6	Spool volume 3 for work selection (JES2 only)	
SVOL4	SVol4	6	Spool volume 4 for work selection (JES2 only)	
CHAR1	Char1	5	Character arrangement table 1	
CHAR2	Char2	5	Character arrangement table 2	
CHAR3	Char3	5	Character arrangement table 3	
CHAR4	Char4	5	Character arrangement table 4	
FSASYSNM	FSASysNm	8	MVS system where FSA is active	
DSPNAME	DSPName	7	Dynamic support program name (JES3 only)	
DEVTYPE	DevType	8	Device type name (JES3 only)	
SDEST1	SRout1[2]	6	Selection destination 1 (JES2 only)	
SDEST2	SRout2[2]	6	Selection destination 2 (JES2 only)	
SDEST3	SRout3[2]	6	Selection destination 3 (JES2 only)	
SDEST4	SRout14[2]	6	Selection destination 4 (JES2 only)	
SNODE1	SNode1[2]	6	Selection node (JES2 only)	
SNODE2	SNode2[2]	6	Selection node 2 (JES2 only)	
SNODE3	SNode3[2]	6	Selection node 3 (JES2 only)	
SNODE4	SNode4[2]	6	Selection node 4 (JES2 only)	
LINELIML	Line-Lim-Lo	12	Printer line limit, minimum	
LINELIMH	Line-Lim-Hi	12	Printer line limit, maximum	
PAGELIML	Page-Lim-Lo	12	Printer page limit, minimum	
PAGELIMH	Page-Lim-Hi	12	Printer page limit, maximum	
DGRPY	DGrpY	5	Device cannot process data sets that are destined for any local device (JES3 only)	
DYNAMIC	Dyn	3	Device can be started dynamically (JES3 only)	
OPACTLOG	OpLog	5	Operator command actions will be logged in the output of the modified device using message IAT7066 or IAT7067 (FSS devices, JES3 only)	
CGS	CGS	3	Character generation storage (JES3 only)	

Table 59. Columns on the PR Panel (continued)

Column name	Title (Displayed)	Width	Description	Delay
BURSTPAGE	B	1	Burst (JES3 only)	
PDEFAULT	PDefault	8	Defaults that should be applied, if not defined in the job's JCL (JES3 only)	
COPIES	Copies	6	Copy count (JES3 only)	
CLEAR	CB	2	Clear printer processing indicator (JES3 only)	
TRC	TRC	3	Table reference character (JES3 only)	
HFCB	HFCB	4	Use designated FCB until status is changed (JES3 only)	
HCHARS	HChars	6	Use designated CHARS until status is changed (JES3 only)	
HUCS	HUCS	4	Use designated UCS until status is changed (JES3 only)	
HCPYMOD	HCpyMod	7	Use designated Copy Mod until status is changed (JES3 only)	
HFLASH	HFlash	6	Use designated Flash until status is changed (JES3 only)	
HBURST	HBurst	6	Use designated Burst until status is changed (JES3 only)	
HFORMS	HForms	6	Use designated Forms until status is changed (JES3 only)	
ASIS	AsIs	4	Send print data as is (JES2 only)	
CCTL	CCtl	4	Data carriage control stream	
CMPCT	Cmpct	4	Compaction for SNA remote punches	
COMP	Comp	4	Compression	
COMPAC	Compact	8	Compaction table name for SNA remote punches	
FCBLOAD	FCBl	4	JES will load FCB	
LRECL	LRecL	5	Logical record length	
SUSPEND	Sus	3	Suspend/interrupt capability (JES2 only)	
SELECT	Select	8	Send output to device type and subaddress	

Note:

[1] The width of the PRINTER column is 7 if the shortened format of device names has been specified. See Table 31 on page 92.

[2] This column is not included in the default field list.

Processes panel (PS)

The PS panel displays information about z/OS UNIX System Services processes.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 60. Columns on the PS Panel

Column name	Title (Displayed)	Width	Description
JOBNAME	JOBNAME	8	Job name. This is the fixed field. It is ignored on an FLD statement or ISFFLD macro.
JOBID	JobID	8	Job ID of the process
STATUS	Status	32	Status of the process
OWNERID	Owner	8	User ID of owner
STATE	State	5	State of the process or of most recently created thread (corresponds to d omvs display)
CPU	CPU-Time	8	Compute time in hundredths of seconds
PID	PID	10	Process ID
PPID	PPID	10	Parent process ID
ASID	ASID	5	Address space id
ASIDX	ASIDX	5	Address space id in hexadecimal
LATCHPID	LatchWaitPID	12	PID on which this process is waiting
COMMAND	Command	40	Command that created process
SERVER	ServerName	32	Server name
TYPE	Type	4	Server type (only when the process is a server)
ACTFILES	ActFiles	8	Number of active files (only when the process is a server)
MAXFILES	MaxFiles	8	Maximum number of files (only when the process is a server)
TIMEE	St-Time	8	Time process was started
DATEE	St-Date	8	Date process was started
SYSLEVEL	SysLevel	25	Level of the operating system
SYSNAME	SysName	8	System name where process is executing
SECLABEL	SecLabel	8	Security label of the process

Punch panel (PUN)

The PUN panel allows the user to display information about punches.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 61. Columns on the PUN Panel

Column name	Title (Displayed)	Width	Description
DEVNAME	PUNCH	10	Device name. This is the fixed field. It is ignored on an FLD statement or ISFFLD macro.
STATUS	Status	8	Punch status
GROUP	Group	8	Device group name (JES3 only)

Table 61. Columns on the PUN Panel (continued)

Column name	Title (Displayed)	Width	Description
SFORMS	SForms	8	Selection form number
SFORM2	SForm2	8	Selection form number 2 (JES2 only)
SFORM3	SForm3	8	Selection form number 3 (JES2 only)
SFORM4	SForm4	8	Selection form number 4 (JES2 only)
SFORM5	SForm5	8	Selection form number 5 (JES2 only)
SFORM6	SForm6	8	Selection form number 6 (JES2 only)
SFORM7	SForm7	8	Selection form number 7 (JES2 only)
SFORM8	SForm8	8	Selection form number 8 (JES2 only)
JNAME	JobName	8	Active job name
JOBID	JobID	8	Active job ID
JTYPE	Type	5	Type of active address space
JNUM	JNum[1]	6	Active job number
OWNERID	Owner	8	User ID of owner
SCLASS	SClass	15	Output selection classes
RECCNT	Rec-Cnt	7	Number of line-mode records in the job
RECPRT	Rec-Prt	7	Number of line-mode records printed
PAGECNT	Page-Cnt	8	Output page count
PAGEPRT	Page-Prt	8	Output pages printed
SEP	Sep	3	Separator page between output groups (JES2 only)
SEPDS	SepDS	5	Separator page between data sets
CCTL	CCtl	4	Data carriage control stream
CMPCT	Cmpct	4	Compaction for SNA remote punches
COMP	Comp	4	Compression
COMPAC	Compact	8	Compaction table name for SNA remote punches
FLUSH	Fls	3	Blank card after each data set
SWTRID	SWriter	8	Punch selection writer ID (JES2 only)
PRTWS	Work-Selection	40	Punch work selection criteria
SPRMODE1	SPrMode1	8	Output selection process mode 1
SPRMODE2-4	SPrMode2-4	8	Output selection process modes 2-4
SDESTN1	SDest1	8	Punch selection destination name 1 (JES2 only)
SDESTN2-4	SDest2-4	8	Punch selection destination names 2-4 (JES2 only)
SJOBNAME	SJobName	8	Selection job name (JES2 only)
SOWNER	SOwner	8	Selection creator ID (JES2 only)
SVOL	SVol	6	Selection volume (JES2 only)
SELECT	Select	7	Send Output To (remote punches only)
CKPTLINE	CkptLine	8	Number of lines per logical page (JES2 only)
CKPTPAGE	CkptPage	8	Number of logical pages per checkpoint (JES2 only)
CKPTREC	CkptRec	3	Number of records per checkpoint (JES3 only)
UNIT	Unit	5	Punch unit name
LINELIM	Line-Limit	21	Punch line limit (JES2 only)

Table 61. Columns on the PUN Panel (continued)

Column name	Title (Displayed)	Width	Description
SRANGE	SRange	22	Selection job number range (JES2 only)
LRECL	LRecL	5	Logical record length of transmitted data (SNA only)
PSETUP	Setup	6	Setup option (JES2 only)
PAUSE	Pau	3	Pause mode
SUSPEND	Sus	3	Punch-interrupt feature option (BSC connection only, JES2 only)
PTRACE	Tr	3	Punch tracing
SYSNAME	SysName	8	System name
DSYSID	SysID	5	JES2 member name (JES2 only)
JESNAME	JESN	4	JES subsystem name
JESLEVEL	JESLevel	8	z/OS JES level
SECLABEL	Seclabel	8	Security label of the job on the device
DEVSECLB	DSecLabel	9	Security label of the device (JES2 only)
LINELIML	Line-Lim-LoMinimum Lines	11	Punch line limit, minimum
LINELIMH	Line-Lim-HiMaximum Lines	11	Punch line limit, maximum
SVOL2-4	Svol2-4	6	Selection volumes 2-4 (JES2 only)
OGNAME	O-Grp-N	8	Output group name (JES2 only)
OGID1	OGid1	5	Output group ID 1 (JES2 only)
OGID2	OGid2	5	Output group ID 2 (JES2 only)
FORMS	Forms	8	Output forms
PRMODE	Prmode	8	Output process mode
WTRID	Writer	8	Output writer name (JES2 only)
DESTN	Dest	8/18	Output destination (JES2 only)
DPRIO	DP	2	Output priority
JPRIO	JP	2	Job priority
OCLASS	C	1	Output class
DEVTYPE	DevType	8	Device type (JES3 only)
DSPNAME	DSPName	8	Dynamic support program name (JES3 only)
HFORMS	HForms	6	Use designated forms until status is changed (JES3 only)
COPIES	Copies	6	Copy count (JES3 only)
DYNAMIC	Dyn	3	Start device dynamically (JES3 only)
DGRPY	DGrpY	3	Device cannot process data sets that are destined for any local device (JES3 only)
BURSTPAGE	B	3	Punch burst page at end of job (JES3 only)

Note:

[1] This column is not included in the default field list.

Reader panel (RDR)

The RDR panel allows the user to display information about readers.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 62. Columns on the RDR Panel

Column name	Title (Displayed)	Width	Description
DEVNAME	READER	10	Device name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
STATUS	Status	8	Reader status
GROUP	Group	8	Device group name (JES3 only)
JNAME	JobName	8	Job name
JOBID	JobID	8	Active job ID (JES2 only)
JTYPE	Type[1]	5	Type of active address space
JNUM	JNum[1]	6	Active job number (JES2 only)
OWNERID	Owner	8	User ID of owner
RECCNT	Rec-Cnt	10	Number of records in the job (JES2 only)
RECPRT	Rec-Proc	10	Number of records processed
RCLASS	C	1 or 8	Default execution class. Default width expands to 8 if there are long class names in the MAS.
RHOLD	Hold	4	Job held after JCL conversion (JES2 only)
RMCLASS	MC	2	Message class (JES2 only)
RPRTDST	PrtDest	18	Default destination for print output (JES2 only)
RPUNDST	PunDest	18	Default destination for punch output (JES2 only)
RSYSAFF	SAff	5	System affinity (JES2 only)
RAUTH	Authority	13	Authority of the reader (JES2 only)
PRIOINC	PI	2	Increment to selection priority (JES2 only)
PRIOLIM	PL	2	Maximum priority level that can be assigned to jobs. Any job's priority that exceeds this level is reduced to it. (JES2 only)
RUNIT	Unit	5	Reader unit name
XEQDEST	XeqDest	18	Default execution node (JES2 only)
RTRACE	Tr	3	Reader tracing (JES2 only)
SYSNAME	SysName	8	System name
DSYSID	SysID	5	JES2 member name (JES2 only)
JESNAME	JESN	4	JES subsystem name
JESLEVEL	JESLevel	8	z/OS JES level
SECLABEL	SecLabel	8	Security label of the job on the reader (JES2 only)
DEVSECLB	DSecLabel	9	Security label of the device (JES2 only)
DEVTYPE	DevType	8	Device type name (JES3 only)
DSPNAME	DSPName	8	Dynamic support program name (JES3 only)
ACCTREQ	AReq	3	Account number required on job card (JES3 only)
PNAMEREQ	PReq	3	Programmer name required on job card (JES3 only)

Table 62. Columns on the RDR Panel (continued)

Column name	Title (Displayed)	Width	Description
SWA	SWA	5	SWA ABOVE or BELOW (JES3 only)
BLP	BLP	3	Bypass label processing label setting is respected (JES3 only)
RPRIO	DP	2	Default job priority (JES3 only)
RMLEVEL	ML	2	Default job message level (JES3 only)
RALEVEL	AL	2	Default allocation message level (JES3 only)
RTIME	Time	10	Default time limit (JES3 only)
RREGION	Region	10	Default region size (JES3 only)

Note:

[1] This column is not included in the default field list.

Resource panel (RES)

The RES panel allows users to display information about WLM resources in a scheduling environment, or in the sysplex.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 63. Columns on the RES Panel

Column name	Title (Displayed)	Width	Description
RESOURCE	RESOURCE	16	Resource name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
REQSTATE	ReqState	8	Required state of the resource for the scheduling environment. Displayed only if the panel is accessed with the R action character.
SYS1 to SYS32	Resolved from the actual names of the systems	8	Status of the resource on the system.

Note: Omit the column title when coding a field list for the RES panel. For example, you would code `SYS1,,8` for the first system column. Using statements, you would omit the TITLE keyword, for example:

```
FLDENT COLUMN(SYS1),WIDTH(*)
```

When there are more columns in the field list than are required for the panel, either because of the number of systems that are active or because the scope of the panel has been limited to systems in the MAS, SDSF displays only as many columns as are required.

Resource Monitor (RM) panel

The Resource Monitor panel shows information about JES2 resources. (JES2 only)

In REXX execs and Java programs, reference columns by name rather than by title.

Table 64. Columns on the RM Panel

Column name	Title (Displayed)	Width	Description	Delay
RESNAME	RESOURCE	8	JES2 resource name	
DSYSID	SysID	5	JES2 member name	
STATUS	Status	10	Resource status	X
LIMIT	Limit	6	Limit for the resource	X
USENUM	InUse	6	Number in use	X
USEPCT	InUse%	6	Percentage in use	X
WARNPCT	Warn%	5	Warning threshold (percentage)	X
INTAVG	IntAvg	6	Average amount in use for the interval	X
INTHIGH	IntHigh	7	Highest amount in use for the interval	X
INTLOW	IntLow	6	Lowest amount in use for the interval	X
OVERWARN	OverWarn%	9	Amount in use above the warning threshold (percentage)	X
TIMEE	Time	8	Time that the interval began	X
DATEE	Date	8	Date that the interval began	X
SYSNAME	SysName	8	System name	
JESNAME	JESN	4	JES2 subsystem name	
JESLEVEL	JESLevel	8	z/OS JES2 level	

Scheduling Environment panel (SE)

The SE panel allows the user to display information about scheduling environments.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 65. Columns on the SE Panel

Column Name	Title (Displayed)	Width	Description
SCHENV	SCHEDULING-ENV	16	Scheduling environment name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
DESCRIPT	Description	32	Description of scheduling environment
SYSTEMS	Systems	60	Systems with the scheduling environment available

Spool Offload panel (SO)

The Spool Offload panel allows the user to display information about JES2 spool offloaders (JES2 only).

In REXX execs and Java programs, reference columns by name rather than by title.

Table 66. Columns on the SO Panel

Column name	Title (Displayed)	Width	Description
DEVNAME	DEVICE	8	Device name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
STATUS	Status	9	Device status
TYPE	Type	8	Device type
JNAME	Jobname	8	Active jobname
JOBID	JobID	8	Active JES2 job ID
JTYPE	no default	4	Type of active address space
JNUM	JNum[1]	6	Active JES2 job number
OWNERID	Owner	8	User ID of owner
LINELIM	Line-Limit	21	Selection line limit
PAGELIM	Page-Limit	21	Selection page limit
RECPRT	Proc-Lines	10	Number of lines processed for the job.
RECCNT	Tot-Lines	10	Number of lines in the job.
SCLASS	SClass	15	Selection classes. Multi-character classes and groups shows as periods (.).
SOWNER	SOwner	8	Selection owner
SHOLD	SHold	5	Selection hold value
SJOBNAME	SJobName	8	Selection jobname
SRANGE	SRange	22	Selection job number range
SDESTN1	SDest1	18	Selection destination name
SSAFF	SSAff	5	Selection system affinity
SDISP	SDisp	6	Selection disposition
SVOL	SVol	6	Selection volume
SBURST	SBurst	6	Selection burst value
SFCBID	SFCB	4	Selection FCB
SFLASHID	SFlh	4	Selection flash
SFORMS	SForms	8	Selection forms name
SFORM2	SForm2	8	Selection forms name 2
SFORM3	SForm3	8	Selection forms name 3
SFORM4	SForm4	8	Selection forms name 4
SFORM5	SForm5	8	Selection forms name 5
SFORM6	SForm6	8	Selection forms name 6
SFORM7	SForm7	8	Selection forms name 7
SFORM8	SForm8	8	Selection forms name 8
SPRMODE1	SPrMode	8	Selection process mode

Table 66. Columns on the SO Panel (continued)

Column name	Title (Displayed)	Width	Description
SODISP	SODsp	5	Selection output disposition
SODISP2	SODsp2	5	Selection output disposition 2
SODISP3	SODsp3	5	Selection output disposition 3
SODISP4	SODsp4	5	Selection output disposition 4
SWTRID	SWriter	8	Selection writer name
SUCSID	SUCS	4	Selection UCS
PRTWS	Work-Selection	40	Work selection criteria
NOTIFY	Notify	6	Notification option
ODSNAME	DSName	44	Data set name
SSRVCLS	SSrvClass	9	Selection service class value for the job receiver or job transmitter
SSCHENV	SScheduling-Env	16	Selection scheduling environment value for the job receiver or job transmitter
MBURST	MBurst	6	Modification of the burst value, for post-execution jobs and output data sets that are selected for reloading, assigned during the reload process.
MDEST	MDest	18	Modification of the destination value, for post-execution jobs and output data sets that are selected for reloading, assigned during the reload process.
MFCB	MFCB	4	Modification of the FCB value, for post-execution jobs and output data sets that are selected for reloading, assigned during the reload process.
MFLASH	MFlh	4	Modification of the flash value, for post-execution jobs and output data sets that are selected for reloading, assigned during the reload process.
MFORMS	MForms	8	Modification of the forms value, for post-execution jobs and output data sets that are selected for reloading, assigned during the reload process.
MHOLD	MHold	5	Modification of the hold value, for post-execution jobs and output data sets that are selected for reloading, assigned during the reload process.
MSCLASS	MClass	8	Modification of the class value, for post-execution jobs and output data sets that are selected for reloading, assigned during the reload process.
MODISP	MODsp	5	Modification of the output disposition value, for post-execution jobs and output data sets that are selected for reloading, assigned during the reload process.
MPRMODE	MPrMode	8	Modification of the process mode value, for post-execution jobs and output data sets that are selected for reloading, assigned during the reload process.
MSAFF	MSAff	5	Modification of the system affinity value, for post-execution jobs and output data sets that are selected for reloading, assigned during the reload process.

Table 66. Columns on the SO Panel (continued)

Column name	Title (Displayed)	Width	Description
MUCS	MUCS	4	Modification of the universal character set (UCS) name value, for post-execution jobs and output data sets that are selected for reloading, assigned during the reload process.
MWRITER	MWriter	8	Modification of the writer name value, for post-execution jobs and output data sets that are selected for reloading, assigned during the reload process.
LABEL	Label	5	Label
PROTECT	Prot	4	Protect option
RETENT	RtPd	4	Retention
ARCHIVE	Archive	7	Archive option
VALIDAT	Validate	8	Validation option
UNIT	Unit	14	Unit
VOLS	Vols	4	Volume count (1-255) to be used for the offload data set
SYSNAME	SysName	8	System name
DSYSID	SysID	5	JES2 member name
JESNAME	JESN	4	JES2 subsystem name
JESLEVEL	JESLevel	8	JES2 level
DEVSECLB	DSecLabel	9	Security label of the device
CRTIME	CRTime	7	Indicates whether to restore or reset the original creation time of the output.
LINELIML	Line-Lim-Lo	11	Line limit, minimum
LINELIMH	Line-Lim-Hi	11	Line limit, maximum
PAGELIML	Page-Lim-Lo	11	Page limit, minimum
PAGELIMH	Page-Lim-Hi	11	Page limit, maximum
SCLASS1-8	SClass1-8	8	Selection classes 1-8, including multi-character classes and groups (job transmitters and receivers)

Note:

[1] JNUM is not included in the default field list.

Spool Volumes panel (SP)

The Spool Volumes panel lets you display and control JES2 spool volumes.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 67. Columns on the SP Panel

Column name	Title (Displayed)	Width	Description
DEVNAME	NAME	6 (JES2) 8 (JES3)	Spool volume name (JES2) or DDNAME (JES3). This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.

Table 67. Columns on the SP Panel (continued)

Column name	Title (Displayed)	Width	Description
STATUS	Status	8 (JES2) 12 (JES3)	Spool status (active, starting, halting, draining, inactive) or partition status
TGPCT	TGPct	5	Spool utilization
TGNUM	TGNum	5	Total track groups
TGUSE	TGUse	5	Track groups in use
COMMAND	Command	8	Command being processed (start, format, drain, halt) (JES2 only)
SPSYSAF	SAff	5	System affinity (JES2 only)
EXTENT	Ext	3	Extent number, in hexadecimal
CYLLO	LoCyl	8	Low cylinder
TRKLO	LoTrk	16	Absolute low track number, in hexadecimal
HEADLO	LoHead	8	Low head
CYLHI	HiCyl	8	High cylinder
TRKHI	HiTrk	16	Absolute high track number, in hexadecimal
HEADHI	HiHead	8	High head
TCYL	TrkPerCyl	9	Tracks per cylinder
TREC	RecPerTrk	9	Records per track
TGTRK	TrkPerTG	8	Tracks per track group
TYPE	Type	9	Spool type (PARTITION or EXTENT)
PARTNAME	PartName	8	Partition name (JES3 only)
OVFNAME	OverFNam	8	Overflow partition name (JES3 only)
OVALLOW	OverAllow	9	Indicates if overflow from this partition to another partition is allowed (JES3 only)
OVOCCUR	OverOccur	9	Indicates if overflow from this partition to another partition occurred (JES3 only)
OVINTO	OverInto	3	Indicates if overflow into this partition from another partition is allowed (JES3 only)
PTRACKS	PTracks	8	Total tracks in the partition
PTRACKU	PTrackU	8	Tracks in use in the partition
DTRACKS	DTracks	8	Total tracks in the data set
DTRACKU	DTrackU	8	Tracks in use in the data set
DEFAULT	Default	7	Default partition indicator (JES3 only)
STUNTED	Stunted	7	Extent is stunted (JES2 only)
STT	STT	3	Single track table indicator (JES3 only)
MARGPCT	MargPct	7	Marginal SLIM threshold percentage – shown only on the row for the partition (JES3 only)
MARGEXC	MargExc	7	Marginal threshold exceeded (JES3 only)
MINPCT	MinPct	6	Minimal SLIM threshold percentage (JES3 only)
MINEXC	MinExc	3	Marginal threshold exceeded (JES3 only)
DATASET	DataSetName	44	Data set name
VOLSER	VolSer	6	Actual volume serial upon which this spool extent resides (JES2 only)

Table 67. Columns on the SP Panel (continued)

Column name	Title (Displayed)	Width	Description
SELECT	Sel	3	Indicates if work is selectable on this volume (JES2 only)
RESERVED	Res	3	Indicates whether this volume is reserved (active but not allocatable) (JES2 only)
LGFREE	LgFree	6	Largest number of contiguous free tracks (JES2 only)
HIGHTRK	HiUsed	6	Highest used track on the volume (JES2 only)
COMPPCT	Comp%	5	Percentage complete of the current action against the volume (JES2 only)
PHASE	Phase	12	Migration phase (JES2 only)
MIGSYS	MigSys	6	JES2 member performing the spool migration (JES2 only)
TARGET	Target	8	Volume name in JES2 where this extent is migrating to or has migrated to (JES2 only)
MIGVOL	MigVol	6	
MIGDSN	MigDSName	44	Data set name to which this extent is migrating (JES2 only)

Search panel (SRCH)

The SRCH panel shows all data sets containing the specified member pattern. The resulting table shows all data sets containing that member pattern. You can use the SRCH command from the APF, LNK, LPA, or PARM panels.

Note: SRCH provides a different capability from the SEARCH command. SRCH implements a member search using a data set list, whereas SEARCH searches the SDSF help and tutorial.

The SRCH panel is not available through REXX or implemented in Java. You can use the SYSDSN function in REXX to implement this function, or implement it directly in Java.

Table 68. Columns on the SRCH Panel

Column name	Title (Displayed)	Width	Description
DSNAME	DSNAME	13-44 (Varies based on longest name.)	Data set name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
SEQ	Seq	3	Sequence number
VOLSER	VolSer	6	Volume serial
STATUS	Status	16	Data set or member status
DSORG	DSOrg	5	Data set organization
BLKSIZE	BlkSize	7	Data set block size
EXTENT	Extent	6	Number of extents
SMS	SMS	3	SMS indicator: YES if data set is SMS managed. Otherwise, NO.
LRECL	LRecL	5	Logical record length

Table 68. Columns on the SRCH Panel (continued)

Column name	Title (Displayed)	Width	Description
RECFM	RecFm	5	Record format
CRDATE	CrDate	8	Data set creation date
REFDATE	RefDate	8	Data set last referenced date
SYSNAME	Sysname	8	System name

System Symbols panel (SYM)

The System Symbols panel (SYM) allows authorized users to display the system dynamic and static symbols defined for each system in the sysplex. System symbols are elements that allow systems to share parmlib definitions while retaining unique values in those definitions. System symbols act like variables in a program; they can take on different values, based on the input to the program.

By default, the SYM panel is sorted by the system and symbol names. You can change the sort order with the **SORT** command.

The value of a static symbol is typically assigned through parmlib. In contrast, the value of a dynamic symbol is assigned by the system at the time the symbol is evaluated. For example, time and date symbols evaluate to the current time and date. The SYM panel shows the values of dynamic symbols at the time the panel is generated as an example of the value format. Jobs that reference a dynamic symbol may contain a different value when the symbol is evaluated.

Note: Action characters on the SYM panel generate commands to display the symbols in the syslog. Because dynamic symbols are not supported by operator commands, issuing an action against a dynamic symbol results in the message NOT VALID FOR TYPE.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 69. Columns on the System Symbols

Column name	Title (Displayed)	Width	Description
SYMBOL	SYMBOL	16	Symbol name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
VALUE	Value	44	Symbol value. For dynamic symbols, it is the current value.
TYPE	Type	8	Symbol type (static or dynamic)
SYSLEVEL	SysLevel	25	Operating system level
SYSNAME	SysName	8	System name

System panel (SYS)

The SYS panel shows information about systems in the sysplex.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 70. Columns on the SYS Panel

Column name	Title (Displayed)	Width	Description
SYSNAME	SYSNAME	8	System name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
SYSLEVEL	SysLevel	3	Operating system level
CPUPR	CPU%	4	CPU percent busy for the system
SIO	SIO	8	Start I/O rate EXCPs per second
AUXPCT	Aux%	4	Auxiliary storage percentage used
CSAPCT	CSA%	4	Common storage area percentage used
SQAPCT	SQA%	4	System queue area percentage used
ECSAPCT	ECSA%	5	Extended common area percentage used
ESQAPCT		5	Extended system queue area percentage used
UIC	UIC	5	High unreferenced interval count
SPOOLPCT	Spool%	6	Spool utilization for primary JES
CADSPCT	CADS%	5	Common Access Dataspace percentage used of maximum defined
PAGERATE	PageRate	8	Paging rate
REAL	Real	8	Number of real storage frames online
REALAFC	RealAFC	8	Real storage available frame count
REALAFCB	RealAFCB	8	Real storage available frame count below 16MB line
FIXPCT	Fix%	4	Percentage of real storage frames that are fixed
FIXBPCT	FixB%	5	Percentage of real storage frames that are fixed below the 16MB line
MAXASID	MaxASID	7	Maximum number of address spaces
FREEASID	FreeASID	8	Number of free address spaces
BADASID	BadASID	7	Number of non-reusable address spaces
STCNUM	STC	6	Number of active started tasks
TSUNUM	TSU	6	Number of active TSO users
JOBNUM	Job	6	Number of active batch jobs
WTORNUM	WTOR	4	Number of outstanding WTORs
SYSPLEX	Sysplex	8	Sysplex name
LPAR	LPAR	8	LPAR name
VMUSER	VMUser	8	VM user ID
JESNAME	JES	4	Job entry subsystem name
JESNODE	JESNode	8	JES node name
SMF	SMF	4	SMF system ID
IPLVOL	IPLVol	6	IPL volume serial
IPLUNIT	IPLUnit	7	IPL unit address

Table 70. Columns on the SYS Panel (continued)

Column name	Title (Displayed)	Width	Description
IPLDATE	IPLDate	19	IPL date
IPLTYPE	IPLType	7	IPL type
IPLDAYS	IPLDays	7	Number of days since last IPL
LOADPARM	LoadParm	8	Load parameter
CVTVERID	CVTVERID	16	CVT version ID associated with system
LOADDSN	LoadDSName	44	LOAD*xx* data set name
LOADUNIT	LoadUnit	8	LOAD*xx* unit address
IEASYS	IEASYS	16	IEASYS*xx* parameters for the system
IEASYM	IEASYM	16	IEASYMxx parameters for the system
GRS	GRS	4	GRS mode
HWNAME	HWName	8	Hardware name
CPC	CPC	30	Central Processor Complex node descriptor
MSU	MSU	8	MSU rating for processor
SYSMSU	SysMSU	8	MSU rating for image
AVGMSU	AvgMSU	8	Four hour rolling MSU for system
CPUNUM	#CPU	4	Number of online CPUs
ZAAPNUM	#ZAAP	5	Number of online zAAP processors
ZIIPNUM	#ZIIP	5	Number of online zIIP processors
OSCONFIG	OSConfig	8	Operating system configuration
EDT	EDT	3	Eligible device table ID
NUCLST	NUCLST	6	NUCLST*xx* member
IEANUC	IEANUC	6	IEANUC*xx* member
IODFDSN	IODFDSName	44	IODF data set name
IODFDATE	IODFDate	19	Date and time IODF last changed
CATDSN		44	Master catalog data set name
CATVOL	CatVol	6	Master catalog volume serial
MLA	MLA	3	Multi-level alias setting for system
CATTYPE	CatType	7	Master catalog type
NETID	NetID	8	VTAM network ID
SSCP	SSCP	17	VTAM SSCP name
STATDATE	StatDate	19	Date and time statistics collected

Status panel (ST)

The Status panel allows the user to display information about jobs, started tasks, and TSO users on the JES queues.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 71. Columns on the ST Panel

Column name	Title (Displayed)	Width	Description	Delay
JNAME	JOBNAME	8	Job name. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.	
JTYPE	Type	4	Type of address space	
JNUM	JNum[1]	6	JES job number	
JOBID	JobID	8	JES job ID	
OWNERID	Owner	8	User ID of job owner, or default values of ++++++++ or ????????, if user ID not defined to RACF	
JPRIO	Prty	4	JES job queue priority	
QUEUE	Queue	10	JES queue name for job	
JCLASS	C	8	JES input class	
POS	Pos	5	Position in JES queue	
SYSAFF	SAff	5 (JES2) 8 (JES3)	JES execution system affinity (if any)	
ACTSYS	ASys	4 (JES2) 8 (JES3)	JES active system ID (if job active)	
STATUS	Status	17	Status of job	
PRTDEST	PrtDest	18	JES print destination name	
SECLABEL	SecLabel	8	Security label of job	
TGNUM	TGNum	5	Track groups used by a job	
TGPCT	TGPct	6	Percentage of total track group usage	
ORIGNODE	OrigNode	8	Origin node name	
EXECNODE	ExecNode	8	Execution node name	
DEVID	Device	18	JES device name	
RETCODE	Max-RC	10	Return code information for the job	
SRVCLS	SrvClass	8	Service class	
WLMPOS	WPos	5	Position on the WLM queue	
SCHENV	Scheduling-Env	16	Scheduling environment for the job	
DELAY	Dly	3	Indicator that job processing is delayed	
SSMODE	Mode	4	Subsystem managing the job (JES or WLM)	
ROOMN	RNum	8	JES job room number	X
PNAME	Programmer-Name	20	JES programmer name	X
ACCTN	Acct	4 (JES2) 8 (JES3)	JES account number	X
NOTIFY	Notify	8	TSO user ID from NOTIFY parameter on job card	X

Table 71. Columns on the ST Panel (continued)

Column name	Title (Displayed)	Width	Description	Delay
ISYSID	ISys	4 (JES2) 8 (JES3)	JES input system ID	X
TIMER	Rd-Time	8	Time that the job was read in	X
DATER	Rd-Date	8	Date that the job was read in	X
ESYSID	ESys	4 (JES2) 8 (JES3)	JES execution system ID	X
TIMEE	St-Time	8	Time that execution began	X
DATEE	St-Date	8	Date that execution began	X
TIMEN	End-Time	8	Time that execution ended	X
DATEN	End-Date	8	Date that execution ended	X
ICARDS	Cards	5	Number of cards read for job	X
MCLASS	MC	2	MSGCLASS of job	X
TSREC	Tot-Lines	10	Total number of spool records for job	X
OFFDEVS	Offs	4	List of offload devices for a job or output that has been offloaded (JES2 only)	
SPIN	Spin	4	Indicator of whether the job is eligible to be spun	
SUBGROUP	SubGroup	8	Submittor group	X
PHASENAME	PhaseName	20	Name of the phase the job is in	
PHASE	Phase	8	Number of the phase the job is in	
JOBACCT1	JobAcct1[1]	20	Job accounting field 1	X
JOBACCT2	JobAcct2[1]	20	Job accounting field 2	X
JOBACCT3	JobAcct3[1]	20	Job accounting field 3	X
JOBACCT4	JobAcct4[1]	20	Job accounting field 4	X
JOBACCT5	JobAcct5[1]	20	Job accounting field 5	X
SUBUSER	SubUser	8	Submitting user ID	X
DELAYRSN	DelayRsn	32	Reason for the job delay (JES2 only). The width can be expanded to 127.	
JOBCORR	JobCorrelator	32	User portion of the job correlator (JES2 only)	

Note: [1] This column is not included in the default field list.

System Requests panel (SR)

The SR panel allows the user to display outstanding system requests.

In REXX execs and Java programs, reference columns by name rather than by title.

Table 72. Columns on the SR Panel

Column name	Title (Displayed)	Width	Description
REPLYID	REPLYID	7	Reply ID. This is the fixed field. It is ignored if coded on an FLD statement or ISFFLD macro.
SYSNAME	SysName	8	Originating system name
JNAME	JobName	8	Name of the issuing job

Table 72. Columns on the SR Panel (continued)

Column name	Title (Displayed)	Width	Description
MSGTEXT	Message-Text	127	Message text
JOBID	JobID	8	ID of the issuing job
DATEE	Date	8	Date the message was issued
TIMEE	Time	8	Time the message was issued
CONSOLE	Console	8	Target console
ROUTECD	RouteCd	7	First 28 routing codes
DESC	Desc	4	Descriptor codes
MSGTYPE	Type	6	Message type
QUEUE	Queue	5	Queue the message is on
AUTOREPLY	AutoReply	9	Automatic reply indicator
AUTODELAY	AutoRDelay	10	Message delay time until the automatic reply is done, in seconds
AUTOTIME	AutoReplyTime	19	Date and time when auto reply will be done
AUTOTEXT	AutoReplyText	16	Automatic reply text

Chapter 5. Using SAF for security

Use the Security Authorization Facility (SAF) interface, with an external security manager such as RACF, to provide security for SDSF. SAF is part of the z/OS environment and is always present. SDSF uses the SAF interface to route authorization requests to the external security manager.

The benefits of using SAF for SDSF security are:
- Dynamic change of security profiles
- Single image of security information
- Simple introduction of security philosophy
- Auditability
- Granular protection

Although in a JES2 environment you can also use ISFPARMS to provide SDSF security, some aspects of SDSF security can only be protected with SAF, such as the use of queues by WebSphere MQ. In a JES3 environment, only SAF can be used for SDSF security.

Relationship of SAF and ISFPARMS

You may choose to use SAF to protect some functions, while using ISFPARMS to protect others, in a JES2 environment.

Even if you use SAF for all of SDSF security, you need ISFPARMS to control:
- Global values (ISFPMAC macro or OPTIONS statement)
- Any values for groups that are not related to security (ISFGRP macro or GROUP statement). The relationship between the ISFGRP macros or GROUP statements of ISFPARMS and SAF is shown in Appendix B, "SAF equivalents for ISFPARMS," on page 561.
- Code page (ISFTR macro or TRTAB statement)

If you want to customize the columns on SDSF panels, you also need ISFFLD macros or FLD statements.

Using ISFPARMS as a backup to SAF

If you already use ISFPARMS for security, you should retain it as a backup to SAF. In the JES2 environment, ISFPARMS is used to determine authorization when SAF returns an *indeterminate* result (return code 04), that is, when SAF cannot make a security decision. SAF returns an indeterminate result when:
- The resource class is inactive
- The class is active but the profile to protect the resource is not defined. Note that this is not true for classes for which the default return code is 08, such as the JESSPOOL and XFACILIT classes. When a class with a default return code of 08 is active but the appropriate profile is not defined, SAF fails the request rather than returning an indeterminate result. This means that if the class is active, SDSF will never revert to ISFPARMS for the relevant security.

For information specific to RACF, see "RACF authorization checking and ISFPARMS security" on page 207.

© Copyright IBM Corp. 1997, 2016

For information on converting to SAF security, including a description of conversion samples, see Chapter 8, “Converting ISFPARMS to SAF security,” on page 311.

In a JES3 environment, when SAF returns an indeterminate result, the request fails. For information on controlling the messages issued in this case, see “Customized properties (PROPLIST)” on page 91.

Changing authorization dynamically

SAF security provides a dynamic means of authorizing SDSF users to issue commands and process job output. Once a user starts an SDSF session, SDSF checks user authorization for virtually every interaction with SDSF resources.

SAF authorization dynamically affects the next user interaction. You must end an SDSF session and restart it when changes are made to SAF authorization for destination names and for operator authority by destination.

Auditing access attempts

If you are using RACF as a security product, RACF logs access attempts to protected SDSF resources according to the audit setting in the RACF profile for the resource. Logging is performed for all access attempts except for the following resource names in the SDSF class:

- ISFOPER.DEST.*jesx*
- ISFAUTH.DEST.*destname*
- ISFAUTH.DEST.*destname*.DATASET.*dsname*
- ISFOPER.ANYDEST.*jesx*
- All resource names beginning with ISFATTR.

Logging is not performed for these access attempts because the user is not specifically trying to gain access to those resources.

For RACF auditing information, refer to *z/OS Security Server RACF Auditor's Guide*.

Diagnosing security

SDSF's security trace function helps you understand and diagnose SDSF security (SAF or ISFPARMS). In response to the actions that you take, such as issuing commands or overtyping columns, it issues messages that describe the associated SAF resources or ISFPARMS statements. You control security trace with commands, REXX variable or Java methods.

- With the SET SECTRACE command, you turn security tracing on and specify how the associated messages are handled.
  - SET SECTRACE ON causes the trace messages to be sent to the ULOG.
  - SET SECTRACE WTP causes the messages to be issued as write-to-programmer messages. Use this if security prevents you from accessing SDSF or the user log.
- With the SECTRACE option on the SDSF command, you can turn security tracing on as soon as you access SDSF.
- When SDSF SECTRACE is active, SDSFAUX SECTRACE is also activated. SDSFAUX uses SECTRACE to record the results of security calls for diagnosis.
- With the ISFSECTRACE REXX special variable, you can control security tracing from a REXX exec.

- With ISFRequestSettings methods addISFSecTrace and removeISFSecTrace, you can control security tracing from a Java program.

For more information about the commands, refer to the online help. You could use the SEARCH command, for example, `SEARCH SET SECTRACE`. For more information about the REXX special variable, refer to Chapter 13, "Using SDSF with the REXX programming language," on page 377. For more information about Java, refer to Chapter 14, "Using SDSF with the Java programming language," on page 451.

SAF concepts for SDSF resources

SDSF interacts with SAF to control access to the following resources:

- Membership in SDSF groups
- SDSF panels
- SDSF authorized commands
- Use of the / command to issue MVS and JES commands and receive responses
- Overtypeable fields
- Destination names
- Operator authority by destination
- Devices and system resources, such as initiators, printers, lines, nodes and scheduling environments
- Jobs affected by action characters and overtypeable fields
- Output groups affected by action characters and overtypeable fields
- SYSIN/SYSOUT data sets for browsing and viewing
- MVS and JES commands that are generated by action characters and overtypeable fields
- Reverting to ISFPARMS in assembler macro format when the server is not available or when no ISFPARMS have been defined with statements
- Use of the server MODIFY command
- Use of WebSphere MQ queues
- Access to the log stream and the JES logical log

The SDSF resources are grouped into classes, with each resource having a resource name. SDSF translates an asterisk (*) in resource names to a plus (+).

To accomplish security through SAF, you permit or deny users access to the SDSF resources by use of their classes and resource names. In addition, you can supplement SAF security with the SAF exit points and installation exit routines.Refer to Chapter 9, "Using installation exit routines," on page 329 for more information.

Protecting SDSF function

An SDSF function often requires access authority to more than one class and resource. In order to use the function, a user must have proper authority to all of the required resources.

For example, to overtype a field, a user must have access to the panel, to the overtypeable field, to the MVS or JES command that will be generated, and to the object (for example, the job, output group, initiator, or printer) being acted upon.

SDSF users must have authority to the resources at the correct access level (READ, CONTROL, UPDATE, or ALTER).

The classes used by SDSF must be defined to your security product. If you are using RACF you do not need to define the classes because they are already included in the IBM-supplied class descriptor table, ICHRRCDX.

The relationship of SDSF functions, classes and resources is shown in "Summary of SAF resources for SDSF function". For some resources, only the highest level qualifier is shown. Refer to Appendix C, "SDSF resource names for SAF security," on page 577 for a table of complete SDSF resource names.

You can use the CONSOLE class to restrict the use of resources in the OPERCMDS and WRITER classes to SDSF users only. The restriction is in effect for the duration of the SDSF session. Use of the CONSOLE class is described in "Using conditional access" on page 209.

Protecting SDSF function in a sysplex environment

Several of SDSF's panels can show data from all members in the MAS in a JES2 environment. In that environment, security is as follows:

- Access to the display is controlled by the profiles on the local system, that is, the system the user is logged on to.
- Access to the objects displayed on the panel (for example, printers on the PR panel) is controlled by SAF resources that include the name of the JES subsystem for the system the object is on. In this topic, the resources show a variable *jesx* which you replace with the subsystem name.
- Which systems are included on the panel is controlled by the SYSNAME command and the server group of the server the user is connected to.

Summary of SAF resources for SDSF function

This topic summarizes the SAF resources required to protect SDSF function.

Table 73. SDSF Functions and the Classes and Resources Required to Protect Them

Function	Specific Function	Classes and Resources to Protect	Refer to
Jobs and Output	Display job and output queues	SDSF – DA, H, I, O, and ST authorized commands	"Authorized SDSF commands" on page 235
	Issue action characters	JESSPOOL – Job or output group	"Jobs, output groups, and SYSIN/SYSOUT data sets" on page 245
		OPERCMDS – Generated MVS or JES command	"Action characters" on page 211
	Overtype fields	SDSF – Overtypeable field	"Overtypeable fields" on page 255
		JESSPOOL – Job or output group	"Jobs, output groups, and SYSIN/SYSOUT data sets" on page 245
		OPERCMDS – Generated MVS or JES command	"Overtypeable fields" on page 255
	Browse output	JESSPOOL – SYSIN/SYSOUT datasets	"Jobs, output groups, and SYSIN/SYSOUT data sets" on page 245

Table 73. SDSF Functions and the Classes and Resources Required to Protect Them (continued)

Function	Specific Function	Classes and Resources to Protect	Refer to
Printers	Display printers	SDSF – PR authorized command	"Authorized SDSF commands" on page 235
	Issue action characters	WRITER – Printer	"Printers" on page 295
		OPERCMDS – Generated MVS or JES command	"Action characters" on page 211
	Overtype fields	SDSF – Overtypeable field	"Overtypeable fields" on page 255
		WRITER – Printer	"Printers" on page 295
		OPERCMDS – Generated MVS or JES command	"Overtypeable fields" on page 255
Initiators	Display initiators	SDSF – INIT authorized command	"Authorized SDSF commands" on page 235
	Issue action characters	SDSF – Initiator	"Initiators" on page 243
		OPERCMDS – Generated MVS or JES command	"Action characters" on page 211
	Overtype fields	SDSF – Overtypeable field	"Overtypeable fields" on page 255
		SDSF – Initiator	"Initiators" on page 243
		OPERCMDS – Generated MVS or JES command	"Overtypeable fields" on page 255
	Display sysplex data (if JES2 on remote systems is prior to z/OS V1R12)	MQQUEUE, MQADMIN, MQCMDS, MQCONN	"WebSphere MQ" on page 306
Lines	Display lines	SDSF – LI authorized command	"Authorized SDSF commands" on page 235
	Issue action characters	SDSF – Line	"Lines" on page 250
		OPERCMDS – Generated MVS or JES command	"Action characters" on page 211
	Overtype fields	SDSF – Overtypeable field	"Overtypeable fields" on page 255
		SDSF – Line	"Lines" on page 250
		OPERCMDS – Generated MVS or JES command	"Overtypeable fields" on page 255
	Display sysplex data (if JES2 on remote systems is prior to z/OS V1R13)	MQQUEUE, MQADMIN, MQCMDS, MQCONN	"WebSphere MQ" on page 306

Table 73. SDSF Functions and the Classes and Resources Required to Protect Them (continued)

Function	Specific Function	Classes and Resources to Protect	Refer to
Nodes	Display nodes	SDSF – NO authorized command	"Authorized SDSF commands" on page 235
	Issue action characters	SDSF – Node	"Nodes" on page 254
		OPERCMDS – Generated MVS or JES command	"Action characters" on page 211
	Overtype fields	SDSF – Overtypeable field	"Overtypeable fields" on page 255
		SDSF – Node	"Nodes" on page 254
		OPERCMDS – Generated MVS or JES command	"Overtypeable fields" on page 255
	Display sysplex data (if JES2 on remote systems is prior to z/OS V1R13)	MQQUEUE, MQADMIN, MQCMDS, MQCONN	"WebSphere MQ" on page 306
Spool Offloaders (JES2 only)	Display spool offloaders	SDSF – SO authorized command	"Authorized SDSF commands" on page 235
	Issue action characters	SDSF – Offloader	"Spool offloaders" on page 301
		OPERCMDS – Generated MVS or JES2 command	"Action characters" on page 211
	Overtype fields	SDSF – Overtypeable field	"Overtypeable fields" on page 255
		SDSF – Offloader	"Spool offloaders" on page 301
		OPERCMDS – Generated MVS or JES2 command	"Overtypeable fields" on page 255
	Display sysplex data (if JES2 on remote systems is prior to z/OS V1R13)	MQQUEUE, MQADMIN, MQCMDS, MQCONN	"WebSphere MQ" on page 306
MAS and JESPLEX Members	Display the MAS or JESPLEX members	SDSF – MAS or JP authorized command	"Authorized SDSF commands" on page 235
	Issue action characters	SDSF – MAS or JESPLEX members	"MAS and JESPLEX members" on page 251
		OPERCMDS – Generated MVS or JES command	"Action characters" on page 211
	Overtype fields	SDSF – Overtypeable field	"Overtypeable fields" on page 255
		SDSF – MAS or JESPLEX members	"Printers" on page 295
		OPERCMDS – Generated MVS or JES command	"Overtypeable fields" on page 255
Network Connections	Display network connections	SDSF – NC authorized command	"Authorized SDSF commands" on page 235
	Issue action characters	SDSF – Network connection	"Network connections" on page 253
		OPERCMDS – Generated JES command	"Action characters" on page 211

Table 73. SDSF Functions and the Classes and Resources Required to Protect Them (continued)

Function	Specific Function	Classes and Resources to Protect	Refer to
Network Servers	Display network servers	SDSF – NS authorized command	"Authorized SDSF commands" on page 235
	Issue action characters	SDSF – Network server	"Network servers" on page 254
		OPERCMDS – Generated MVS or JES command	"Action characters" on page 211
	Overtype fields	SDSF – Overtypeable field	"Overtypeable fields" on page 255
		SDSF – Network server	"Network servers" on page 254
		OPERCMDS – Generated MVS or JES command	"Overtypeable fields" on page 255
Punches	Display punches	SDSF – PUN authorized command	"Authorized SDSF commands" on page 235
	Issue action characters	SDSF – Punch	"Punches" on page 297
		OPERCMDS – Generated MVS or JES command	"Action characters" on page 211
	Overtype fields	SDSF – Overtypeable field	"Overtypeable fields" on page 255
		SDSF – Punch	"Punches" on page 297
		OPERCMDS – Generated MVS or JES command	"Overtypeable fields" on page 255
	Display sysplex data (if JES2 on remote systems is prior to z/OS V1R13)	MQQUEUE, MQADMIN, MQCMDS, MQCONN	"WebSphere MQ" on page 306
Readers	Display readers	SDSF – RDR authorized command	"Authorized SDSF commands" on page 235
	Issue action characters	SDSF – Reader	"Readers" on page 298
		OPERCMDS – Generated MVS or JES command	"Action characters" on page 211
	Overtype fields	SDSF – Overtypeable field	"Overtypeable fields" on page 255
		SDSF – Reader	"Readers" on page 298
		OPERCMDS – Generated MVS or JES command	"Overtypeable fields" on page 255
	Display sysplex data (if JES2 on remote systems is prior to z/OS V1R13)	MQQUEUE, MQADMIN, MQCMDS, MQCONN	"WebSphere MQ" on page 306

Table 73. SDSF Functions and the Classes and Resources Required to Protect Them (continued)

Function	Specific Function	Classes and Resources to Protect	Refer to
Checks	Display checks	SDSF – CK authorized command	"Authorized SDSF commands" on page 235
	Display check history	LOGSTRM – Log stream	"Checks on the CK and CKH panels" on page 238
	Issue action characters	XFACILIT – Check	"Checks on the CK and CKH panels" on page 238
		OPERCMDS – Generated MVS command	"Action characters" on page 211
	Overtype fields	SDSF – Overtypeable field	"Overtypeable fields" on page 255
		XFACILIT – Check	"Checks on the CK and CKH panels" on page 238
		OPERCMDS – Generated MVS command	"Overtypeable fields" on page 255
	Display sysplex data (if any system is prior to z/OS V1R13)	MQQUEUE, MQADMIN, MQCMDS, MQCONN	"WebSphere MQ" on page 306
Enclaves	Display enclaves	SDSF – ENC authorized command	"Authorized SDSF commands" on page 235
	Issue action characters	SDSF – Enclave	"Enclaves" on page 242
	Overtype fields	SDSF – Overtypeable field	"Overtypeable fields" on page 255
		SDSF – Enclave	"Enclaves" on page 242
	Display sysplex data (if any system is prior to z/OS V1R13)	MQQUEUE, MQADMIN, MQCMDS, MQCONN	"WebSphere MQ" on page 306
JES2 resources (JES2 only)	Display JES2 resources	SDSF – RM authorized command	"Authorized SDSF commands" on page 235
	Issue action characters	SDSF – Resource	"JES2 resources on the RM panel" on page 244
		OPERCMDS – Generated MVS or JES2 command	"Action characters" on page 211
	Overtype fields	SDSF – Overtypeable field	"Overtypeable fields" on page 255
		SDSF – Resource	"JES2 resources on the RM panel" on page 244
		OPERCMDS – Generated MVS or JES2 command	"Overtypeable fields" on page 255
	Display sysplex data (if any system is prior to z/OS V1R13)	MQQUEUE, MQADMIN, MQCMDS, MQCONN	"WebSphere MQ" on page 306

Table 73. SDSF Functions and the Classes and Resources Required to Protect Them (continued)

Function	Specific Function	Classes and Resources to Protect	Refer to
Job Classes	Display job classes	SDSF – JC authorized command	"Authorized SDSF commands" on page 235
	Issue action characters	SDSF – Job class	"Job classes" on page 245
		OPERCMDS – Generated MVS or JES command	"Action characters" on page 211
	Overtype fields	SDSF – Overtypeable field	"Overtypeable fields" on page 255
		SDSF – Job class	"Job classes" on page 245
		OPERCMDS – Generated MVS or JES command	"Overtypeable fields" on page 255
Spool Volumes	Display spool volumes	SDSF – SP authorized command	"Authorized SDSF commands" on page 235
	Issue action characters	SDSF – Spool volume	"Spool volumes" on page 301
		OPERCMDS – Generated MVS or JES command	"Action characters" on page 211
	Overtype fields	SDSF – Overtypeable field	"Overtypeable fields" on page 255
		SDSF – Spool volume	"Spool volumes" on page 301
		OPERCMDS – Generated MVS or JES command	"Overtypeable fields" on page 255
WLM Resources	Display WLM resources	SDSF – RES authorized command	"Authorized SDSF commands" on page 235
	Issue action characters	SDSF – WLM resource	"Resources defined to WLM" on page 298
		OPERCMDS – Generated MVS command	"Action characters" on page 211
	Overtype fields	SDSF – Overtypeable field	"Overtypeable fields" on page 255
		SDSF – WLM resource	"Resources defined to WLM" on page 298
		OPERCMDS – Generated MVS command	"Overtypeable fields" on page 255
Scheduling Environments	Display scheduling environments	SDSF – SE authorized command	"Authorized SDSF commands" on page 235
	Issue action characters	SDSF – scheduling environment OPERCMDS – Generated MVS command	"Scheduling environments" on page 299 "Action characters" on page 211

Table 73. SDSF Functions and the Classes and Resources Required to Protect Them (continued)

Function	Specific Function	Classes and Resources to Protect	Refer to
Enqueues	Display enqueues	SDSF — ENQ authorized command	"Authorized SDSF commands" on page 235
	Issue action characters	SDSF – enqueue OPERCMDS — Generated MVS command	"System requests" on page 304 "Action characters" on page 211
System symbols	Display system symbols	SDSF — SYM authorized command	"Authorized SDSF commands" on page 235
	Issue action characters	SDSF – symbol OPERCMDS — Generated MVS command	"System requests" on page 304 "Action characters" on page 211
System Requests	Display system requests	SDSF — SR authorized command	"Authorized SDSF commands" on page 235
	Issue action characters	SDSF – system request OPERCMDS — Generated MVS command	"System requests" on page 304 "Action characters" on page 211
z/OS UNIX processes	Display processes	SDSF – PS authorized command	"Authorized SDSF commands" on page 235
	Issue action characters	SDSF –process OPERCMDS – Generated MVS command	"System requests" on page 304 "Action characters" on page 211
	Display sysplex data (if any system is prior to z/OS V1R13)	MQQUEUE, MQADMIN, MQCMDS, MQCONN	"WebSphere MQ" on page 306
Display the system log	Display the LOG panel	SDSF – LOG authorized command	"Authorized SDSF commands" on page 235
	Access the logical log (SYSLOG)	JESSPOOL	"SYSLOG" on page 302
	Access the log stream (OPERLOG)	LOGSTRM	"OPERLOG" on page 255

Table 73. SDSF Functions and the Classes and Resources Required to Protect Them (continued)

Function	Specific Function	Classes and Resources to Protect	Refer to
Destination Operator Authority	Issue action characters	SDSF – Operator authority	"Destination operator authority" on page 240
		SDSF – Jobs or output based on destination name	"Destination operator authority" on page 240
		OPERCMDS – Generated MVS or JES command	"Action characters" on page 211
	Overtype fields	SDSF – Operator authority	"Destination operator authority" on page 240
		SDSF – Overtypeable field	"Overtypeable fields" on page 255
		SDSF – Jobs or output based on destination name	"Destination operator authority" on page 240
		OPERCMDS – Generated MVS or JES command	"Overtypeable fields" on page 255
	Browse output	SDSF – Operator authority	"Destination operator authority" on page 240
		SDSF – Datasets based on job or output group destination	"Destination operator authority" on page 240
System Commands and responses	Use / command	SDSF – ULOG authorized command SDSF – / command	"Authorized SDSF commands" on page 235
		MVS and JES require authorization to OPERCMDS resources for MVS and JES commands issued.	"MVS and JES commands on the command line" on page 252
SDSF Commands	Use DEST command	SDSF – DEST authorized command	"Authorized SDSF commands" on page 235
		SDSF – Destination names	"Destination names" on page 239
	Use authorized SDSF commands	SDSF authorized commands	"Authorized SDSF commands" on page 235
SDSF Server	Specify a server on the SDSF command	SDSF – SERVER parameter on the SDSF command	"SDSF server" on page 300
	Refresh ISFPARMS or change server options, start and stop the server and server communications	OPERCMDS – START, MODIFY, and STOP commands	"SDSF server" on page 300

Class	SDSF Resource	Resource Name
JESSPOOL	Jobs, output groups, and SYSIN/SYSOUT data sets	*nodeid.userid.jobname.jobid* *nodeid.userid.jobname.jobid*.GROUP.*ogroupid* *nodeid.userid.jobname.jobid*.D*dsid.dsname*
JESSPOOL	Access to the JES logical log, to display the SYSLOG	*nodeid*.+MASTER+.SYSLOG.SYSTEM.sysname

Class	SDSF Resource	Resource Name
LOGSTRM	Access to the log stream, to display the OPERLOG	SYSPLEX.OPERLOG
	Access to the log stream, to display check history	*log-stream-name*
MQQUEUE	Queues for sysplex data	*ssid.queue-prefix* (High-level qualifiers)
MQCMDS	Server definition of queues	*ssid*.DEFINE (High-level qualifiers)
MQADMIN	Server definition of queues	*ssid*.QUEUE.*queue-prefix*
	WebSphere MQ context security	*ssid*.CONTEXT
MQCONN	WebSphere MQ connection security	*ssid*.BATCH (High-level qualifiers)
OPERCMDS	Generated MVS and JES commands	Resource name is dependent on command generated
	Server MODIFY command	Resource name is dependent on command parameters
SDSF	Membership in groups	GROUP.*groupname.servername*
	Connection to SDSFAUX	ISF.CONNECT.*sysname*
	APF data sets	ISFAPF.*datasetname*
	DYNX data sets	ISFDYNX.*exitname*
	Enqueues	ISFENQ.*majorname.sysname*
	LnkLst data sets	ISFLNK.*datasetname*
	LPA data sets	ISFLPA.*datasetname*
	Parmlib data sets	ISFPARM.*datasetname*
	Page data sets	ISFPAG.*datasetname*
	System symbols	ISFSYM.*symbolname.sysname*
	Systems	ISFSYS.*sysplexname.systemname*
	SDSF panels and authorized commands	ISFCMD (High-level qualifier)
	MVS/JES command line commands (/)	ISFOPER.SYSTEM
	Overtypeable fields	ISFATTR (High-level qualifier)
	Destination names	ISFOPER.ANYDEST.*jesx* (all destinations) ISFAUTH.DEST.*destname*
	Operator authority by destination	ISFOPER.DEST ISFAUTH.DEST (High-level qualifiers)
	Enclaves	ISFENC.*subsys-type.subsys-name*
	Initiators	ISFINIT.I(*xx*).*jesx*
	Job classes	ISFJOBCL.*class.jesx*
	MAS or JESPLEX members	ISFMEMB.*membername.jesx*
	Lines	ISFLINE.*devicename.jesx*
Network connections	ISFAPPL.*devicename.jesx* ISFLINE.*devicename.jesx* ISFSOCK.*devicename.jesx*	
Network servers	ISFNS.*devicename.jesx*	
Nodes	ISFNODE.*nodename.jesx*	

Class	SDSF Resource	Resource Name
SDSF (continued)	Spool offloaders (JES2 only)	ISFSO.*devicename.jesx*
	Readers	ISFRDR.*devicename.jesx*
	JES resources (JES2 only)	ISFRM.*resource.jesx*
	Spool volumes	ISFSP.*volser.jesx*
	Spool partitions	ISFSP.*partname.jesx*
	WLM resources	ISFRES.*resource.system*
	Scheduling environments	ISFSE.*scheduling-env.system*
	z/OS UNIX processes	ISFPROC.*owner.jobname*
	System requests	ISFSR.*type.system.jobname*
	Server name on SDSF command	ISFCMD.OPT.SERVER
	Reverting to ISFPARMS in assembler macro format	SERVER.NOPARM
WRITER	Printers and punches	*jesx*.LOCAL.*devicename* *jesx*.RJE.*devicename*
XFACILIT	Checks from IBM Health Checker for z/OS	HZS.*sysname.checkowner.checkname.action*

Chapter 6. SDSF and RACF

This topic provides general information about RACF security. It also demonstrates how to establish SAF security for SDSF tasks and resources using classes, resource names, and access levels.

For specific information about how to protect SDSF tasks and resources, see Chapter 7, "Protecting SDSF functions," on page 211.

Security administration

A key feature of RACF is its hierarchical management structure. The RACF security administrator is defined at the top of the hierarchy, with authority to control security for the whole system. The RACF security administrator has the authority to work with RACF profiles and system-wide settings. The RACF auditor produces reports of security-relevant activity based on auditing records generated by RACF.

RACF security administrators generally have system-SPECIAL authority. This allows them to issue any RACF command and change any RACF profile (except for some auditing specific operands).

For complete information about the authorities required to issue RACF commands, and for information on delegating authority and the scope of a RACF group, refer to *z/OS Security Server RACF Security Administrator's Guide*.

For information on the RACF requirements for issuing RACF commands, see the description of the specific command in *z/OS Security Server RACF Command Language Reference*.

Brief summary of RACF commands

Much of the RACF activity dealing with protected SDSF resources involves creating, changing, and deleting *general resource profiles*.

- To create a resource profile, use the RDEFINE command. Generally, once you have created a profile, you then create an access list for the profile using the PERMIT command. For example:

  ```
  RDEFINE  class_name  profile_name  UACC(NONE)
  PERMIT  profile_name  CLASS(class_name)  ID(user or group)
          ACCESS(access_authority)
  ```

  This document provides examples of how to do this for SDSF-related classes.
- To remove the entry for a user or group from an access list, issue the PERMIT command with the DELETE operand instead of the ACCESS operand.

  ```
  PERMIT  profile_name  CLASS(class_name)  ID(user or group)  DELETE
  ```
- If you want to change a profile, for example, changing UACC from NONE to READ, use the RALTER command:

  ```
  RALTER  class_name  profile_name  UACC(READ)
  ```
- To delete a resource profile, use the RDELETE command. For example:

  ```
  RDELETE  class_name  profile_name
  ```

© Copyright IBM Corp. 1997, 2016

- You can copy an access list from one profile to another. To do so, specify the FROM operand on the PERMIT command:

```
PERMIT  profile_name  CLASS(class_name)
        FROM(existing-profile_name)  FCLASS(class_name)
```

- You can copy information from one profile to another. To do so, specify the FROM operand on the RDEFINE or RALTER command:

```
RDEFINE  class_name  profile__name
        FROM(existing-profile_name)  FCLASS(class_name)
```

  Note: Do not plan to do this if you are using resource group names.

- To list the names of profiles in a particular class, use the SEARCH command. The following command lists the profiles in the SDSF class:

```
SEARCH CLASS(SDSF)
```

Delegation of RACF administrative authority

Your installation's security plan should indicate who is responsible for providing security for SDSF.

If you do not have the system-SPECIAL attribute, you need to be given the authority to do the following RACF-related tasks:

- Define and maintain profiles in SDSF-related general resource classes. In general, this authority is granted by assigning a user the CLAUTH (class authority) attribute in the specified classes. For example, the security administrator could issue the following command:

  ```
  ALTUSER your_userid CLAUTH(SDSF)
  ```

  Some of the general resource classes mentioned in this document (such as OPERCMDS and JESSPOOL) affect the operation of products other than SDSF. If you are not the RACF security administrator, you may need to ask that person to define profiles at your request.

- Add RACF user profiles to the system. In general, this authority is granted by assigning an administrator the CLAUTH (class authority) attribute in the user's profile. For example, the security administrator could issue the following command:

  ```
  ALTUSER your_userid CLAUTH(USER)
  ```

  Whenever you add a user to the system, you must assign that user a default connect group. Assigning that user a default connect group changes the membership of the group (by adding the user as a member of the group).

For more information about RACF general resource profiles, see *z/OS Security Server RACF Security Administrator's Guide*. For information about the resource names used by JES2, see *z/OS JES2 Initialization and Tuning Guide*. For information about the resource names used by JES3, see *z/OS JES3 Initialization and Tuning Guide*.

SDSF resource group class

The IBM-supplied class descriptor table provides a resource *group* class (GSDSF) and a resource *member* class (SDSF). For a resource group class, each user or group of users permitted access to that resource group is permitted access to all members of the resource group. For each GSDSF class created, a second class representing the members must also be created.

Creating a resource group profile

Resource group profiles enable you to protect multiple resources with one profile. However, the resources do not have to have similar names.

A resource group profile is a general resource profile with the following special characteristics:

- Its name does not match the resource it protects.
- The ADDMEM operand of the RDEFINE command specifies the resources it protects (not the profile name itself).
- The related member class (not the resource class itself) must be RACLISTed. For example, the SDSF class must be RACLISTed, not the GSDSF class. Use the SETROPTS command with the RACLIST operand for this task.

For more information on RACF group profiles, see *z/OS Security Server RACF Security Administrator's Guide*.

Establishing SAF security with RACF

To accomplish security through SAF with RACF, you:

1. Activate generic processing before defining profiles, using the SETROPTS command.
2. Define profiles to protect the resources in the appropriate classes, using the RDEFINE command. (Classes are already defined for RACF. You must define them for other security products.)

   Begin with generic profiles for broad access to resources and then define generic or discrete profiles that are more restrictive.
3. Permit users to access appropriate profiles in each class with the necessary access levels, using the PERMIT command.
4. Activate the classes, using the SETROPTS command.

You should also review installation exit routines for SAF control points. Refer to Chapter 9, "Using installation exit routines," on page 329 for more information.

RACF authorization checking and ISFPARMS security

When the class a resource is in is inactive, or the profile to protect the resource is not defined:

- In a JES2 environment, the result varies with the default return code for the class:
  - The SDSF and OPERCMDS classes, as defined by RACF, have a default return code of 04, and return an indeterminate result. Authorization is decided by ISFPARMS or an installation exit.
  - The JESSPOOL and WRITER classes, as defined by RACF, have a default return code of 08. The request fails.
- In a JES3 environment, the request fails.
- For requests processed by the SDSFAUX address space, the request fails unless CONNECT(NOFAILRC4) is specified in ISFPRM*xx*.

Considerations for broad access

The examples in this information typically show generic profiles that allow the user broad access to resources. The universal access authority (UACC) function of NONE is used to protect resources for all users on the system. Users of the system

who are not SDSF users may be affected when trying to access those resources. The examples of WRITER class profiles have UACC(READ) so that printers can select work for all users.

If you begin by defining broad generic profiles, you can then define more restrictive generic or discrete profiles. Users permitted to access the broad profiles must also be permitted to access the more restrictive profiles if they are to retain access to all the resources.

Using RACLIST and REFRESH

The SETROPTS RACLIST command copies the base segments of generic and discrete profiles into virtual storage. The profile copies are put in their own data space. RACF uses these profile copies to check the authorization of any user who wants to access a resource protected by them. Using RACLIST for the security classes improves performance.

Once a class is RACLISTed, any changes to the profiles in the class require that the class be RACLIST REFRESHed.

See the discussions of generic profiles and the RACLIST option in *z/OS Security Server RACF Command Language Reference*.

Using RACLIST and REFRESH with the SDSF class

When running RACF, SDSFAUX requires that the SDSF class be RACLISTed.

By default, SDSFAUX fails all authorization requests that result in return code 04 (indeterminate) from SAF. You can change this by specifying AUXSAF(NOFAILRC4) on the CONNECT statement of ISFPRMxx.

If you have not already done so, you must use the SETROPTS RACLIST command for the SDSF class.

For example, assume that you issue the following command to RACLIST the SDSF class:

```
SETROPTS RACLIST(SDSF)
```

If you then change profiles in the SDSF class, you must issue a RACLIST REFRESH command for those changes to take effect:

```
SETROPTS RACLIST(SDSF) REFRESH
```

See the discussions of generic profiles and the RACLIST option in *z/OS Security Server RACF Command Language Reference*.

Using RACLIST and REFRESH with the OPERCMDS class

When using RACF, you must use the SETROPTS RACLIST command for the OPERCMDS class. If you then make changes to these OPERCMDS profiles, you must issue a SETROPTS RACLIST REFRESH command for those changes to take effect.

For example, if you issue the following command to permit GROUP1 to resources in the OPERCMDS class:

```
PERMIT jesx.** CLASS(OPERCMDS) ID(GROUP1) ACCESS(CONTROL)
```

you must then use the REFRESH operand for the change to be effective:

```
SETROPTS RACLIST(OPERCMDS) REFRESH
```

See the discussions of generic profiles and the RACLIST option in *z/OS Security Server RACF Command Language Reference*.

Using conditional access

If you use generic profiles (as in the preceding examples) to give the user access to all JES and MVS commands, the profiles not only include protection for generated MVS and JES commands within SDSF, but also for those commands used outside of SDSF.

Because of this, you may want to make the user's access conditional, only in effect when he or she is using SDSF. You can provide this conditional access for the WRITER and OPERCMDS classes. With RACF, this is done with the clause `WHEN(CONSOLE(SDSF))`.

To use this conditional access checking, you must have the CONSOLE class active and the SDSF console defined in the CONSOLE class.

For example, you would issue the following RACF commands:

```
SETROPTS CLASSACT(CONSOLE
RDEFINE CONSOLE SDSF UACC(NONE)
```

Then, to give conditional access (to permit users to issue JES2 commands only while running SDSF):

```
RDEFINE OPERCMDS JES2.** UACC NONE
PERMIT JES2.** CLASS(OPERCMDS)ID(userid or groupid) ACCESS(CONTROL)
WHEN(CONSOLE(SDSF))
```

To permit users unconditionally to issue all JES2 commands:

```
PERMIT JES2.** CLASS(OPERCMDS)ID(userid or groupid) ACCESS(CONTROL)
```

See also the discussions of “Action characters” on page 211, “Overtypeable fields” on page 255, “Printers” on page 295, and “Punches” on page 297.

Sample RACF commands

SDSF provides sample RACF commands for SDSF security in member ISFRAC of ISF.SISFEXEC.

Multilevel Security

SDSF supports the multilevel security in z/OS V1R5. For information on implementing multilevel security, including the resources used with SDSF, see *z/OS Introduction and Release Guide*.

Chapter 7. Protecting SDSF functions

This topic describes how to protect each of the SDSF functions, which are presented in alphabetical order. It includes discussions and RACF examples.

Action characters

Most action characters cause an interaction with two resources:

- The object of the action character, such as an initiator, printer, MAS member, job, or data set
- The MVS command that is generated by the action

When these resources are protected, a user must have authority to both resources to use the action characters. For ISPF-only actions such as browse and edit, the user must be permitted to open the data set.

Protecting the objects of action characters

The objects of action characters are such things as initiators in the SDSF class, printers and punches in the WRITER class, and jobs, output groups, and SYSIN/SYSOUT data sets in the JESSPOOL class.

The resource name that protects the object and the access level required varies from panel to panel. For information about protecting the objects of action characters, see

- "Authorized program facility data sets" on page 235
- "Checks on the CK and CKH panels" on page 238
- "Destination operator authority" on page 240
- "Dynamic exit information" on page 242
- "Enqueue information" on page 243
- "Enclaves" on page 242
- "Initiators" on page 243
- "JES2 resources on the RM panel" on page 244
- "Job classes" on page 245
- "Jobs, output groups, and SYSIN/SYSOUT data sets" on page 245
- "Lines" on page 250
- "Link list data sets" on page 250
- "MAS and JESPLEX members" on page 251
- "Network connections" on page 253
- "Network servers" on page 254
- "Nodes" on page 254
- "Page data sets" on page 294
- "PARMLIB data sets" on page 295
- "Printers" on page 295
- "Processes (z/OS UNIX System Services)" on page 296
- "Punches" on page 297
- "Readers" on page 298

© Copyright IBM Corp. 1997, 2016

- "Resources defined to WLM" on page 298
- "Scheduling environments" on page 299
- "Spool offloaders" on page 301
- "Spool volumes" on page 301
- "System information" on page 303
- "System Symbol information" on page 303
- "System requests" on page 304

Protecting the generated MVS and JES commands

Most action characters generate MVS or JES commands. The resource names that protect these commands are in the OPERCMDS class. "Tables of action characters" on page 213 shows all the action characters and their resource names.

Controlling access authority

Access to the OPERCMDS resources can be controlled by which resources a user is authorized to access and also by which access level is given to the user. For example, an installation may create just one profile to protect all commands in the OPERCMDS class, but control a user's ability to issue commands by granting the user READ, UPDATE, CONTROL, or ALTER authority. Each authority level gives the user access to a different set of commands. Other installations may choose to define several OPERCMDS resources, and authorize users to access individual resources with the appropriate levels of access.

To see how this information relates to the command levels for the action characters and resource names, refer to the CMDLEV parameter in "Group function parameters reference" on page 40 and "Action characters and overtypeable fields for each command level" on page 73.

Permitting access only while using SDSF

Users can be conditionally permitted to access OPERCMDS resources so they are authorized to use MVS and JES commands only while they are using SDSF. See "Using conditional access" on page 209 for more information.

Setting up generic profiles

You can set up two generic profiles to allow use of all action characters, as shown in Table 74.

Table 74. Generic Profiles for Commands Generated by Actions Characters

Generated Commands	Resource Name	Class	Access
JES Commands	*jesx*.**	OPERCMDS	CONTROL
MVS Commands	MVS.**	OPERCMDS	CONTROL

To protect resources individually in the OPERCMDS class with more restrictive profiles, you would use the specific resource name for the command generated by the action character. See "Tables of action characters" on page 213.

Note: In cases where JES issues an MVS command for processing, the user ID running JES must be authorized to access the OPERCMDS profiles protecting MVS commands, or the JES task must be running in a "trusted" state.

Examples of protecting action characters

1. To allow use of all action characters on all panels, define the following profiles:

   ```
   RDEFINE OPERCMDS jesx.** UACC(NONE)
   RDEFINE OPERCMDS MVS.** UACC(NONE)
   ```

   Give users CONTROL access with these commands:

   ```
   PERMIT jesx.** CLASS(OPERCMDS) ID(userid or groupid) ACCESS(CONTROL)
   PERMIT MVS.** CLASS(OPERCMDS) ID(userid or groupid) ACCESS(CONTROL)
   ```

2. To restrict the use of the C, CD, P, and PP action characters on the Display Active Users panel, define the restrictive profiles:

   ```
   RDEFINE OPERCMDS jesx.CANCEL.** UACC(NONE)
   RDEFINE OPERCMDS MVS.CANCEL.TSU.** UACC(NONE)
   ```

   To restrict the canceling of active APPC transaction programs define the profile:

   ```
   RDEFINE OPERCMDS MVS.CANCEL.ATX.** UACC(NONE)
   ```

   Giving UPDATE authority to only these three profiles will limit action character use to C, CD, P and PP on the Display Active Users panel.

Tables of action characters

SDSF action characters, the MVS and JES commands that they generate, the necessary access authorities, and the OPERCMDS class resource names are shown in Table 75. The table shows the command that is issued, and the associated OPERCMDS resource, for the JES2 environment for each action character; if the action is available in the JES3 environment, the JES3 command and associated OPERCMDS resource are shown beneath the JES2 values.

This information is shown sorted by OPERCMDS resource names in Table 76 on page 228.

Table 75. Action Characters.

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an action character does not apply in a particular environment, the command and OPERCMDS resource are shown as a hyphen (-).

Action Character	SDSF Panel	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	OPERCMDS Required Access
A	H O	$TO	*jesx*.MODIFY.*type*OUT	UPDATE
		-	-	-
A	DA I ST	$A	*jesx*.MODIFYRELEASE.*type*	UPDATE
		*F	*jesx*.MODIFY.JOB	
A	CK	F *hcstcid*,ACTIVATE	MVS.MODIFY.STC.*hcproc.hcstcid*	UPDATE
A	NO	-	-	-
		*F	*jesx*.MODIFY.NJE	UPDATE
A	SP	-	-	-
		*F Q	*jesx*.MODIFY.Q	UPDATE

Table 75. Action Characters (continued).

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an action character does not apply in a particular environment, the command and OPERCMDS resource are shown as a hyphen (-).

Action Character	SDSF Panel	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	OPERCMDS Required Access
AI	SR	SETAUTOR	MVS.SETAUTOR.AUTOR	READ
		SETAUTOR	MVS.SETAUTOR.AUTOR	READ
B	PR PUN	$B	*jesx*.BACKSP.DEV	UPDATE
		-	-	-
B*number*	PR PUN	$B	*jesx*.BACKSP.DEV	UPDATE
		-	-	-
BC	PR PUN	$B	*jesx*.BACKSP.DEV	UPDATE
		*R,*device*,C	*jesx*.RESTART.DEV.*device*	
BC*number*	PR PUN	$B	*jesx*.BACKSP.DEV	UPDATE
		*R,*device*,C	*jesx*.RESTART.DEV.*device*	UPDATE
BC*number*P	PR PUN	-	-	-
		*R,*device*,C	*jesx*.RESTART.DEV.*device*	UPDATE
BD	PR PUN	$B	*jesx*.BACKSP.DEV	UPDATE
		*R,*device*,G	*jesx*.RESTART.DEV.*device*	
BN	PR PUN	-	-	-
		*R,*device*,N	*jesx*.RESTART.DEV.*device*	UPDATE
BN*number*	PR PUN	-	-	-
		*R,*device*,N	*jesx*.RESTART.DEV.*device*	UPDATE
BN*number*P	PR PUN	-	-	-
		*R,*device*,N	*jesx*.RESTART.DEV.*device*	UPDATE
C (TSU jobs)	DA I ST	C U=	MVS.CANCEL.*type.jobname*	UPDATE
		*F J=,C	*jesx*.MODIFY.JOB	
C (APPC transactions)	DA	C *jobname*,A=	MVS.CANCEL.*type.jobname*	UPDATE
C	DA I ST	$C	*jesx*.CANCEL.*type*	UPDATE
		*F J=,C	*jesx*.MODIFY.JOB	
C	H O	$C $CO	*jesx*.CANCEL.*type* *jesx*.CANCEL.*type*OUT	UPDATE
		-	-	-
C	PR PUN RDR	$C	*jesx*.CANCEL.DEV	UPDATE
		*CANCEL	*jesx*.CANCEL.DEV.*device*	
C	H (secondary JES2)	$O,CANCEL	*jesx*.RELEASE.*type*OUT	UPDATE
		-	-	-
C (held data set)	JDS	SSI [1]		
		*F U	*jesx*.MODIFY.U	UPDATE
C	JP	*S	*jesx*.START.DEV.*main*	UPDATE

Table 75. Action Characters (continued).

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an action character does not apply in a particular environment, the command and OPERCMDS resource are shown as a hyphen (-).

Action Character	SDSF Panel	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	OPERCMDS Required Access
C	J0	-	-	-
		*F U	*jesx*.MODIFY.U	UPDATE
C (transmitters, receivers)	LI	$C	*jesx*.CANCEL.DEV	UPDATE
		-	-	-
C (lines)	LI	-	-	-
		*C	*jesx*.CANCEL.*name* *jesx*.CANCEL.DEV.*name*	UPDATE
C	NC	-	-	-
		*C	*jesx*.CANCEL.TCP *jesx*.CANCEL.*devname*	UPDATE
C	NS	-	-	-
		*C	*jesx*.CANCEL.*devname*	UPDATE
C (transmitters, receivers)	SO	$C	*jesx*.CANCEL.DEV	UPDATE
C (processes)	PS	C *jobname*,A= C U=	MVS.CANCEL.*type*.*jobname*	UPDATE
C	SR	K C	MVS.CONTROL.C	READ
CA	DA I ST	$C,ARMRESTART	*jesx*.CANCEL.*type*	UPDATE
		*F J=,C,ARMR	*jesx*.MODIFY.JOB	
CD (TSU jobs)	DA	C U=,DUMP	MVS.CANCEL.*type*.*jobname*	UPDATE
		*F J=,C,D	*jesx*.MODIFY.JOB	
CD (APPC transactions)	DA	C *jobname*, DUMP,A=	MVS.CANCEL.*type*.*jobname*	UPDATE
CD	DA I ST	$C,D	*jesx*.CANCEL.*type*	UPDATE
		*F J=,C,D	*jesx*.MODIFY.JOB	
CDA	DA I ST	$C,D,ARMRESTART	*jesx*.CANCEL.*type*	UPDATE
		*F J=,C,D,ARMR	*jesx*.MODIFY.JOB	
CDP	DA I ST	-	-	-
		*F J=,CO,D	*jesx*.MODIFY.JOB	UPDATE
CG	PR PUN	-	-	-
		*C,*device*,G	*jesx*.CANCEL.DEV.*device*	UPDATE
CJ	PR PUN	-	-	-
		*C,*device*,J	*jesx*.CANCEL.DEV.*device*	UPDATE
CP	PR PUN	-	-	-
		*C,*device*,P	*jesx*.CANCEL.DEV.*device*	UPDATE

Table 75. Action Characters (continued).

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an action character does not apply in a particular environment, the command and OPERCMDS resource are shown as a hyphen (-).

Action Character	SDSF Panel	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	OPERCMDS Required Access
CT	PR PUN	-	-	-
		*C,*device*,T	*jesx*.CANCEL.DEV.*device*	UPDATE
C*options*	RDR	-	-	-
		*C,*device,options*	*jesx*.CANCEL.DEV.*device*	UPDATE
D	DA I ST	$D	*jesx*.DISPLAY.*type*	READ
		*I J=	*jesx*.MODIFY.JOB	
D	APF	D PROG,APF, DSNAME=	MVS.DISPLAY.PROG.	READ
D	CK	F *hcstcid*,DISPLAY	MVS.MODIFY.STC.*hcproc.hcstcid*	UPDATE
D	DYNX	D PROG,EXIT,EX=	MVS.DISPLAY.PROG	READ
D	ENQ	D GRS,HEX,RES=	MVS.DISPLAY.GRS	READ
D	JC	$D	*jesx*.DISPLAY.JOBCLASS	READ
		*I C=	*jesx*.DISPLAY.CLASS	READ
D	JP	*I	*jesx*.DISPLAY.MAIN	READ
D	J0	*I	*jesx*.DISPLAY.U	READ
D	INIT	$D	*jesx*.DISPLAY.INITIATOR	READ
		*I	*jesx*.DISPLAY.G	
D	LI	$D	*jesx*.DISPLAY.L *jesx*.DISPLAY.LINE	READ
		*I	*jesx*.DISPLAY.D	
D	LNK	D PROG,LNKLST, NAME=	MVS.DISPLAY.PROG.	READ
D	MAS SO	$D	*jesx*.DISPLAY.MEMBER *jesx*.DISPLAY.DEV	READ
D	NC	$D	*jesx*.DISPLAY.APPL *jesx*.DISPLAY.L *jesx*.DISPLAY.LINE *jesx*.DISPLAY.SOCKET	READ
		*I	*jesx*.DISPLAY.SOCKET	READ
D	NO	$D	*jesx*.DISPLAY.NODE	READ
		*I	*jesx*.DISPLAY.NJE	
D	NS	$D	*jesx*.DISPLAY.NETSRV *jesx*.DISPLAY.LOGON	READ
		*I	*jesx*.DISPLAY.NETSRV	
D	PAG	D ASM,PAGE=	MVS.DISPLAY.ASM	READ
D	PARM	D PARMLIB	MVS.DISPLAY.PARMLIB	READ
D	PR PUN	$D	*jesx*.DISPLAY.DEV	READ
		*I	*jesx*.DISPLAY.D	
D	PS	D	MVS.DISPLAY.OMVS	READ

Table 75. Action Characters (continued).

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an action character does not apply in a particular environment, the command and OPERCMDS resource are shown as a hyphen (-).

Action Character	SDSF Panel	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	OPERCMDS Required Access
D	RDR	$D	*jesx*.DISPLAY.DEV	READ
		*I		
D	RES	D	MVS.DISPLAY.WLM	READ
D	RM	$D	*jesx*.DISPLAY.*resource*[3]	READ
D	SE	D	MVS.DISPLAY.WLM	READ
D	SP	$DSPL	*jesx*.DISPLAY.SPOOL	READ
		*I Q	*jesx*.DISPLAY.Q	
D	SR	D	MVS.DISPLAY.R	READ
D	SYS	D IPLINFO	MVS.DISPLAY.IPLINFO	READ
DA	APF	D PROG,APF,ALL	MVS.DISPLAY.PROG	READ
DA	DYNX	D PROG,EXIT,ALL	MVS.DISPLAY.PROG	READ
DA	NS	$D	*jesx*.DISPLAY.APPL	READ
		-	-	-
DAA	SYS	D A,ALL	MVS.DISPLAY.JOB	READ
DAI	DYNX	D PROG,EXIT,ALL,IMPLICIT	MVS.DISPLAY.PROG	READ
DAL	SYS	D A,L	MVS.DISPLAY.JOB	READ
DALO	SYS	D ALLOC,OPTIONS	MVS.DISPLAY.ALLOC	READ
DC	NO	$D	*jesx*.DISPLAY.NODE	READ
		-	-	-
DC	PAG	D ASM,COMMON	MVS.DISPLAY.ASM	READ
DC	JC	-	-	-
		*I G,*main*,C,*class*	*jesx*.DISPLAY.G	READ
DC	SYS	D C	MVS.DISPLAY.CONSOLES	READ
DCEE	SYS	D CEE,ALL	MVS.DISPLAY.CEE	READ
DD	DYNX	D PROG,EXIT,EX=,DIAG	MVS.DISPLAY.PROG	READ
DD	PAG	D ASM,PAGEDEL	MVS.DISPLAY.ASM	READ
DD	SYS	D D,E	MVS.DISPLAY.DUMP	READ
DE	DA I ST	-	-	-
		*I J=,E	*jesx*.DISPLAY.JOBE	READ
DE	LI	-	-	-
		*I	*jesx*.DISPLAY.T	READ
DE	PARM	D PARMLIB,ERRORS	MVS.DISPLAY.PARMLIB	READ
DEM	SYS	D EMCS	MVS.DISPLAY.EMCS	READ
DG	JC	-	-	-
		*I G,*main*,G,*group*	*jesx*.DISPLAY.G	READ

Table 75. Action Characters (continued).

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an action character does not apply in a particular environment, the command and OPERCMDS resource are shown as a hyphen (-).

Action Character	SDSF Panel	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	OPERCMDS Required Access
DG	SYS	D GRS,SYSTEM	MVS.DISPLAY.GRS	READ
DI	DYNX	D PROG,EXIT,INSTALLATION	MVS.DISPLAY.PROG	READ
DI	SYS	D IOS,CONFIG	MVS.DISPLAY.IOS	READ
DIQP	SYS	D IQP	MVS.DISPLAY.IQP	READ
DL	CK	F *hcstcid*,DISPLAY	MVS.MODIFY.STC.*hcproc.hcstcid*	UPDATE
DL	DA	$D	*jesx*.DISPLAY.*type*	READ
		*I A,J=	*jesx*.DISPLAY.A	
DL	I ST	$D	*jesx*.DISPLAY.*type*	READ
DL	INIT	$D	*jesx*.DISPLAY.INITIATOR	READ
		*I	*jesx*.DISPLAY.G	
DL	JP	*I	*jesx*.DISPLAY.MAINX	READ
DL	LI	$D	*jesx*.DISPLAY.L *jesx*.DISPLAY.LINE	READ
		*I	*jesx*.DISPLAY.D	
DL	MAS SO	$D	*jesx*.DISPLAY.MEMBER *jesx*.DISPLAY.DEV	READ
DL	NC	$D	*jesx*.DISPLAY.LINE	READ
		-	-	-
DL	NS	$D	*jesx*.DISPLAY.NETSRV *jesx*.DISPLAY.LOGON	READ
		*I	*jesx*.DISPLAY.NETSRV	
DL	NO	$D	*jesx*.DISPLAY.NODE	READ
		*I	*jesx*.DISPLAY.NJE	READ
DL	RDR	$D	*jesx*.DISPLAY.DEV	READ
DL	PAG	D ASM,LOCAL	MVS.DISPLAY.ASM	READ
DL	PR PUN	$D	*jesx*.DISPLAY.DEV	READ
		*I	*jesx*.DISPLAY.D	
DL	SP	$DSPL,L	*jesx*.DISPLAY.SPOOL	READ
		*I Q	*jesx*.DISPLAY.Q	
DL	SYM	D SYMBOLS	MVS.DISPLAY.SYMBOLS	READ
DLL	SYS	D LLA	MVS.DISPLAY.LLA	READ
DLO	SYS	D LOGGER	MVS.DISPLAY.LOGGER	READ
DLR	SYS	D LOGREC	MVS.DISPLAY.LOGREC	READ
DM	JP	*START	*jesx*.START.MONITOR	UPDATE
DMA	I ST	-	-	-
		*I S,A,J=	*jesx*.DISPLAY.S	READ

Table 75. Action Characters (continued).

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an action character does not apply in a particular environment, the command and OPERCMDS resource are shown as a hyphen (-).

Action Character	SDSF Panel	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	OPERCMDS Required Access
DM	SYS	D M	MVS.DISPLAY.M	READ
DMP	SYS	D MPF	MVS.DISPLAY.MPF	READ
DMR	I ST	-	-	-
		*I S,A,J=	*jesx*.DISPLAY.S	READ
DMSS	I ST	-	-	-
		*I S,SS,J=	*jesx*.DISPLAY.S	READ
DMSV	I ST	-	-	-
		*I S,SV,J=	*jesx*.DISPLAY.S	READ
DMU	I ST	-	-	-
		*I S,U,J=	*jesx*.DISPLAY.S	READ
DN	LNK	D PROG,LNKST, NAMES	MVS.DISPLAY.PROG	READ
DNP	DYNX	D PROG,EXIT,NOTPROGRAM	MVS.DISPLAY.PROG	READ
DO	SYS	D OMVS,O	MVS.DISPLAY.OMVS	READ
DP	CK	F *hcstcid*,DISPLAY	MVS.MODIFY.STC.*hcproc.hcstcid*	UPDATE
DP	DYNX	D PROG,EXIT,PROGRAM	MVS.DISPLAY.PROG	READ
DP	NO	$D,PATH	*jesx*.DISPLAY.NODE	READ
		-	-	-
DP	PAG	D ASM,PLPA	MVS.DISPLAY.ASM	READ
DP	SYS	D PROD,REG	MVS.DISPLAY.PROD	READ
DPCD	SYS	D PCIE,DD	MVS.DISPLAY.PCIE	READ
DPCI	SYS	D PCIE	MVS.DISPLAY.PCIE	READ
DPO	CK	F *hcstcid*,DISPLAY	MVS.MODIFY.STC.*hcproc.hcstcid*	UPDATE
DS	CK	F *hcstcid*,DISPLAY	MVS.MODIFY.STC.*hcproc.hcstcid*	UPDATE
DS	NS	$D	*jesx*.DISPLAY.SOCKET	READ
		-	-	-
DS	PAG	D ASM,SCM	MVS.DISPLAY.ASM	READ
DSF	SYS	D SMF,O	MVS.DISPLAY.SMF	READ
DSL	SYS	D SLIP	MVS.DISPLAY.SLIP	READ
DSM	SYS	D SMS	MVS.DISPLAY.SMS	READ
DSY	SYS	D SYMBOLS	MVS.DISPLAY.SYMBOLS	READ
DT	SYS	D T	MVS.DISPLAY.TIMEDATE	READ
DTO	SYS	D IKJTSO	MVS.DISPLAY.IKJTSO	READ
DTR	SYS	D TRACE	MVS.DISPLAY.TRACE	READ
DTS	SYS	D TS,L	MVS.DISPLAY.JOB	READ
DW	SYS	D WLM	MVS.DISPLAY.WLM	READ

Table 75. Action Characters (continued).

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an action character does not apply in a particular environment, the command and OPERCMDS resource are shown as a hyphen (-).

Action Character	SDSF Panel	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	OPERCMDS Required Access
DX	SYS	D XCF	MVS.DISPLAY.XCF	READ
E	CK	F *hcstcid*,REFRESH	MVS.MODIFY.STC.*hcproc.hcstcid*	UPDATE
E	DA I ST	$E	*jesx*.RESTART.BAT	CONTROL
		*R MAIN,J=	*jesx*.RESTART.DEV.*main*	
E (lines)	LI	$E	*jesx*.RESTART.LINE	CONTROL
		*R	*jesx*.RESTART.RJP	UPDATE
E (transmitters, receivers)	LI	$E	*jesx*.RESTART.DEV	UPDATE
		-	-	-
E	NC	$E	*jesx*.RESTART.DEV	UPDATE
		-	-	-
E (subdevice)	NC	$E	*jesx*.RESTART.LINE	CONTROL
		-	-	-
E (connection)	NS	$E	*jesx*.RESTART.DEV	UPDATE
		*R	*jesx*.RESTART.DEV.*devname*	
E	OD			
E (transmitters)	SO	$E	*jesx*.RESTART.DEV	UPDATE
E	MAS	$E	*jesx*.RESTART.SYS	CONTROL
E	PR PUN	$E	*jesx*.RESTART.DEV	UPDATE CONTROL
		*R	*jesx*.RESTART.DEV.*device*	UPDATE
E*options*	PR PUN	-	-	-
		*R,*device,options*	*jesx*.RESTART.DEV.*device*	UPDATE
EC	DA I ST	$E,C	*jesx*.RESTART.BAT	CONTROL
EL	NO	-	-	-
		*F	*jesx*.MODIFY.NJE	UPDATE
ES	DA I ST	$E	*jesx*.RESTART.BAT	CONTROL
		-	-	-
ESH	DA I ST	$E	*jesx*.RESTART.BAT	CONTROL
		-	-	-
F	JP	*S	*jesx*.START.DEV.*main*	UPDATE
F	PR PUN	$F	*jesx*.FORWARD.DEV	UPDATE
		-	-	-
F*number*	PR PUN	$F	*jesx*.FORWARD.DEV	UPDATE
		*R,*device*,R	*jesx*.RESTART.DEV.*device*	
FC	PR PUN	$F	*jesx*.FORWARD.DEV	UPDATE
		*R,*device*,R,C	*jesx*.RESTART.DEV.*device*	

Table 75. Action Characters (continued).

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an action character does not apply in a particular environment, the command and OPERCMDS resource are shown as a hyphen (-).

Action Character	SDSF Panel	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	OPERCMDS Required Access
FC*number*	PR PUN	$F	*jesx*.FORWARD.DEV	UPDATE
		*R,*device*,R,C	*jesx*.RESTART.DEV.*device*	
FC*number*P	PR PUN	-	-	-
		*R,*device*,R,C	*jesx*.RESTART.DEV.*device*	UPDATE
FD	PR	$F	*jesx*.FORWARD.DEV	UPDATE
		-	-	-
FN	PR PUN	-	-	-
		*R,*device*,R,N	*jesx*.RESTART.DEV.*device*	UPDATE
FN*number*	PR PUN	-	-	-
		*R,*device*,R,N	*jesx*.RESTART.DEV.*device*	UPDATE
FN*number*P	PR PUN	-	-	-
		*R,*device*,R,N	*jesx*.RESTART.DEV.*device*	UPDATE
H	CK	F *hcstcid*, DEACTIVATE	MVS.MODIFY.STC.*hcproc.hcstcid*	UPDATE
H	DA I ST	$H	*jesx*.MODIFYHOLD.*type*	UPDATE
		*F J=,H	*jesx*.MODIFY.JOB	
H	O	$TO	*jesx*.MODIFY.*type*OUT	UPDATE
		-	-	-
H	JDS	SSI[1]	None	
		*F U,J=	*jesx*.MODIFY.U	UPDATE
H	J0	-	-	-
		*F U,J	*jesx*.MODIFY.U	UPDATE
H	NO	-	-	-
		*F	*jesx*.MODIFY.NJE	UPDATE
H	SP	-	-	-
		*F Q	*jesx*.MODIFY.Q	UPDATE
HC	SP	-	-	-
		*F Q	*jesx*.MODIFY.Q	UPDATE
HP	SP	-	-	-
		*F Q	*jesx*.MODIFY.Q	UPDATE
I	LI	$TLINE,D=	*jesx*.MODIFY.LINE	CONTROL
		*C	*jesx*.CANCEL.*device*	UPDATE
I	PR PUN	$I	*jesx*.INTERRUPT.DEV	UPDATE
		-	-	-
I	ENC I ST			

Table 75. Action Characters (continued).

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an action character does not apply in a particular environment, the command and OPERCMDS resource are shown as a hyphen (-).

Action Character	SDSF Panel	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	OPERCMDS Required Access
J	I ST	$SJ	*jesx*.START.BAT	UPDATE
		*F J=,RUN	*jesx*.MODIFY.JOB	
J	SP	$DJOBQ,SPL=	*jesx*.DISPLAY.JST	READ
		*I Q	*jesx*.DISPLAY.Q	
J (members)	MAS	$J	*jesx*MON.DISPLAY.MONITOR	READ
JD (members)	MAS	$J	*jesx*MON.DISPLAY.DETAIL	READ
JH (members)	MAS	$J	*jesx*MON.DISPLAY.HISTORY	READ
JJ (members)	MAS	$J	*jesx*MON.DISPLAY.JES	READ
JS (members)	MAS	$J	*jesx*MON.DISPLAY.STATUS	READ
K	DA	C *jobname*,A=	MVS.CANCEL.*type*.*jobname*	UPDATE
		C *jobname*,A=	MVS.CANCEL.*type*.*jobname*	
K	NS	C	MVS.CANCEL.STC.*servername*	CONTROL
K	PR	F	MVS.MODIFY.STC.*fssproc*.*fssname*	UPDATE
			MVS.MODIFY.STC.*fssproc*.*fssname*	
K	PS	F	MVS.MODIFY.STC.BPXOINIT.BPXOINIT	UPDATE
KD	DA	C *jobname*,A=	MVS.CANCEL.*type*.*jobname*	UPDATE
		C *jobname*,A=	MVS.CANCEL.*type*.*jobname*	
KD	NS	C	MVS.CANCEL.STC.*servername*	CONTROL
L	CK			
L	DA I ST	$L	*jesx*.DISPLAY.*type*OUT	READ
		*I J=	*jesx*.DISPLAY.JOB	
LB, LH, LT	DA I ST	-	-	-
		*I J=	*jesx*.DISPLAY.JOB	READ
L	H O	$DO	*jesx*.DISPLAY.*type*OUT	READ
		-	-	-
L	LI	-	-	-
		*FAIL	*jesx*.FAIL.*device*	CONTROL
L	NS	-	-	-
		*FAIL	*jesx*.FAIL.DEV*devname*	CONTROL
L	PR RDR	-	-	-
		*FAIL	*jesx*.FAIL.DEV.*device*	CONTROL
L	PUN	-	-	-
		*FAIL	*jesx*.FAIL.*dspname*	CONTROL
LD	LI	-	-	-
		*FAIL	*jesx*.FAIL.*device*	CONTROL

Table 75. Action Characters (continued).

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an action character does not apply in a particular environment, the command and OPERCMDS resource are shown as a hyphen (-).

Action Character	SDSF Panel	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	OPERCMDS Required Access
LD	PR RDR	-	-	-
		*FAIL	*jesx*.FAIL.DEV.*device*	CONTROL
LD	PUN	-	-	-
		*FAIL	*jesx*.FAIL.*dspname*	CONTROL
LL	DA	$L	*jesx*.DISPLAY.*type*OUT	READ
LL	H O ST	$DO	*jesx*.DISPLAY.*type*OUT	READ
		-	-	-
M	ENC			
N	DA PR PUN	$N	*jesx*.REPEAT.DEV	UPDATE
		-	-	-
O	ST H	$O $TO	*jesx*.RELEASE.*type*OUT *jesx*.MODIFY.*type*OUT	UPDATE
		-	-	-
O (Held data set)	JDS	SSI[1]		
		*F U,J=	*jesx*.MODIFY.U	UPDATE
O	J0	-	-	-
		*F U,J	*jesx*.MODIFY.U	UPDATE
OK	H	$TO	*jesx*.MODIFY.*typeout*	UPDATE
		-	-	-
P	CK	F *hcstcid*,DELETE	MVS.MODIFY.STC.*hcproc.hcstcid*	UPDATE
P (TSU jobs)	DA I ST	C U= $CT	MVS.CANCEL.*type.jobname* *jesx*.CANCEL.*type*	UPDATE
		F J=,C	*jesx*.MODIFY.JOB	
P (APPC transactions)	DA	C *jobname*,A=	MVS.CANCEL.*type.jobname*	UPDATE
		C *jobname*,A=	MVS.CANCEL.*type.jobname*	
P	DA I ST	$C	*jesx*.CANCEL.*type*	UPDATE
		*F J=,C	*jesx*.MODIFY.JOB	
P	H O H O	$C $CO	*jesx*.CANCEL.*type* *jesx*.CANCEL.*type*OUT	UPDATE
		-	-	-
P (Held data set)	JDS	SSI [1]		
		*F U,J=	*jesx*.MODIFY.U	UPDATE
P	JP	-	-	-
		*RETURN	*jesx*.STOP.RETURN	CONTROL
P	J0	-	-	-
		*F U,J=	*jesx*.MODIFY.U	UPDATE

Table 75. Action Characters (continued).

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an action character does not apply in a particular environment, the command and OPERCMDS resource are shown as a hyphen (-).

Action Character	SDSF Panel	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	OPERCMDS Required Access
P	SO MAS	$P	*jesx*.STOP.DEV *jesx*.STOP.SYS	UPDATE CONTROL
P	INIT	$P	*jesx*.STOP.INITIATOR	CONTROL
		*F	*jesx*.MODIFY.G	UPDATE
P (lines)	LI	$P	*jesx*.STOP.LINE	CONTROL
		-	-	-
P (transmitters, receivers)	LI	$P	*jesx*.STOP.DEV	UPDATE
		-	-	-
P	NC	$P	*jesx*.STOP.DEV	UPDATE
		-	-	-
P	NS	$P	*jesx*.STOP.DEV	UPDATE
		-	-	-
P	PR PUN RDR	$P	*jesx*.STOP.DEV	UPDATE
		-	-	-
P (spool volumes)	SP	$PSPL	*jesx*.STOP.SPOOL	CONTROL
		*F Q	*jesx*.MODIFY.Q	UPDATE
PC (spool volumes)	SP	$PSPL	*jesx*.STOP.SPOOL	CONTROL
		-	-	-
PP (TSU jobs)	DA I ST	C U= $C	MVS.CANCEL.*type.jobname* *jesx*.CANCEL.*type*	UPDATE
PF	CK	F *hcstcid*,DELETE	MVS.MODIFY.STC.*hcproc.hcstcid*	UPDATE
PP (APPC transactions)	DA	C *jobname*,A=	MVS.CANCEL.*type.jobname*	UPDATE
PP	DA I ST	$C	*jesx*.CANCEL.*type*	UPDATE
PX	MAS	$P	*jesx*.STOP.SYS	CONTROL
	JP	*F	*jesx*.MODIFY.V	UPDATE
Q	DA H I JDS J0 O ST			-
Q	LI	$TLINE,D=	*jesx*.MODIFY.LINE	CONTROL
		-	-	-
R	CK	F *hcstcid*,RUN	MVS.MODIFY.STC.*hcproc.hcstcid*	UPDATE
R[RMF]	DA	RESET	MVS.RESET	UPDATE
		RESET	MVS.RESET	
R	ENC			
R	SR	R	MVS.REPLY	READ
R	SE			

Table 75. Action Characters (continued).

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an action character does not apply in a particular environment, the command and OPERCMDS resource are shown as a hyphen (-).

Action Character	SDSF Panel	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	OPERCMDS Required Access
RQ[RMF]	DA	RESET	MVS.RESET	UPDATE
		RESET	MVS.RESET	
RQ	ENC			
S	INIT	$S	*jesx*.START.INITIATOR	CONTROL
		*F	*jesx*.MODIFY.G	UPDATE
S	SO INIT MAS	$S	*jesx*.START.DEV *jesx*.START.SYS	UPDATE CONTROL
S (members)	JP	*S	*jesx*.START.JSS	UPDATE
S (lines)	LI	$S	*jesx*.START.LINE	CONTROL
		*S	*jesx*.START.DEV.*device*	
S (transmitters, receivers)	LI	$S	*jesx*.START.DEV	UPDATE
		-	-	-
S	NC	$S	*jesx*.START.DEV	UPDATE
		-	-	-
S	NS	$S	*jesx*.START.DEV	UPDATE
		-	-	-
S	PR PUN RDR	$S	*jesx*.START.DEV	UPDATE
		*S*device*	*jesx*.START.DEV.*device*	
S (spool volumes)	SP	$SSPL	*jesx*.START.SPOOL	CONTROL
S*options*	PR PUN RDR	-	-	-
		*S,*device*	*jesx*.START.DEV.*device*	UPDATE
S, SB, SE	CK CKH DA H I JDS J0 O OD ST			
SBI, SBO, SEI, SEO	CK			
SJ	DA H I JDS O OD ST			
SL	LI	-	-	-
		*S	*jesx*.START.DEV.*device*	CONTROL
SM	JP	*CALL	*jesx*.CALL.MONITOR	UPDATE
SN	LI	$SN	*jesx*.START.NET	CONTROL
		-	-	-
SN	NO	$SN	*jesx*.START.NET	CONTROL
		*S *X	*jesx*.START.TCP *jesx*.CALL.NJE	UPDATE

Table 75. Action Characters (continued).

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an action character does not apply in a particular environment, the command and OPERCMDS resource are shown as a hyphen (-).

Action Character	SDSF Panel	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	OPERCMDS Required Access
SN	NC	$SN	*jesx*.START.NET	CONTROL
		*S *X	*jesx*.START.TCP *jesx*.CALL.NJE	UPDATE
SNL, SNR	LI	-	-	-
		*S	*jesx*.START.DEV.*device*	CONTROL
SR	LI	-	-	-
		*S	*jesx*.START.DEV.*device*	CONTROL
SR	SO	$S	*jesx*.START.DEV	UPDATE
SRJP	LI	-	-	-
		*S	*jesx*.START.RJP	UPDATE
ST	SO	$S	*jesx*.START.DEV	UPDATE
ST	JC SE			
SX	MAS	$S	*jesx*.START.SYS	CONTROL
T	PS	F	MVS.MODIFY.STC.BPXOINIT.BPXOINIT	UPDATE
U	CK	F*hcstcid*,UPDATE	MVS.MODIFY.STC.*hcproc.hcstcid*	UPDATE
U	SP	-	-	-
		*F Q	*jesx*.MODIFY.Q	UPDATE
V	JDS OD			
V	JP LI PR PUN RDR	-	-	-
		*F VARY	*jesx*.MODIFY.V	UPDATE
VF	JP LI PR PUN RDR	-	-	-
		*F VARY	*jesx*.MODIFY.V	UPDATE
W[RMF]	DA I ST	$T	*jesx*.MODIFY.BAT *jesx*.MODIFY.TSU *jesx*.MODIFY.STC	UPDATE
		*F J=,SPIN	*jesx*.MODIFY.JOB	
W	JDS	$T	*jesx*.MODIFY.BAT *jesx*.MODIFY.TSU *jesx*.MODIFY.STC	UPDATE
		-	-	-
X	CK CKH DA H I JDS J0 O OD ST			
X	NS	-	-	-
		*C	*jesx*.CALL.TCP	UPDATE
X	PR PUN	-	-	-
		*X,WTR,OUT=	*jesx*.CALL.*dspname*	UPDATE

Table 75. Action Characters (continued).

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an action character does not apply in a particular environment, the command and OPERCMDS resource are shown as a hyphen (-).

Action Character	SDSF Panel	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	OPERCMDS Required Access
X*options*	PR PUN	-	-	-
		*X,WTR,OUT=	*jesx*.CALL.*dspname*	UPDATE
X	RDR	-	-	-
		*X CR,IN=*device*	*jesx*.CALL.CR	UPDATE
X*options*	RDR	-	-	-
		*X CR,IN=*device*	*jesx*.CALL.CR	UPDATE
Y[RMF]	DA	STOP	MVS.STOP.*type.jobname* MVS.STOP.*type.jobname.id*	UPDATE
		STOP	MVS.STOP.*type.jobname* MVS.STOP.*type.jobname.id*	
Z	DA	FORCE	MVS.FORCE.*type.jobname* MVS.FORCE.*type.jobname.id*	CONTROL
		FORCE	MVS.FORCE.*type.jobname* MVS.FORCE.*type.jobname.id*	
Z	INIT	$Z	*jesx*.HALT.INITIATOR	CONTROL
Z	NS	FORCE	MVS.FORCE.STC.*servername*	CONTROL
Z	PR PUN RDR	$Z	*jesx*.HALT.DEV	UPDATE
		-	-	
Z	SP	$ZSPL	*jesx*.HALT.SPOOL	CONTROL
ZM	MAS	$J	*jesx*MON.STOP.MONITOR	CONTROL
	JP	*CANCEL	*jesx*.CANCEL.MONITOR	UPDATE
?	DA H I J0 O OD ST			
//	all tabular panels			
=	all tabular panels			
+	all tabular panels			
Any	Sysplex-wide panels[2]	ROUTE	MVS.ROUTE	READ

Notes:

[1] SDSF uses the subsystem interface (SSI) when you enter a C, H, O, or P action character on the JDS panel. When all data sets are deleted by use of the C and P action characters on the H panel, SDSF issues $O.

[2] SDSF uses the MVS ROUTE command to route commands to a system in a sysplex other than the one the user is logged on to, for these panels, when they are using SDSF's sysplex support: CK, ENC, INIT, LI, NO, PR, PS, PUN, RDR, RM and SO.

[3] The SAF resource varies with the JES2 resource. See "JES2 resources" on page 234.

[RMF] The DA panel must be using RMF as the source of its data.

In Table 76, many action characters have more than one OPERCMDS resource name associated with them. The names vary according to the panel. Choose the OPERCMDS resource name that is related to the panel for which action character access is being given.

Table 76. Action Characters by OPERCMDS Resource Name.

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the JES/MVS Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	Action Character	JES/MVS Command	SDSF Panel
jesx.BACKSP.DEV	UPDATE	B*number*	$B	PR PUN
jesx.CALL.CR	UPDATE	X	*X CR	RDR
jesx.CALL.*dspname*	UPDATE	X	*X	PR PUN
jesx.CALL.MONITOR	UPDATE	SM	*X	JP
jesx.CALL.NJE	UPDATE	SN	*X	NC NO
jesx.CANCEL.DEV	UPDATE	C	$C	PR PUN LI SO RDR
jesx.CANCEL.DEV.*device*	UPDATE	C	*CANCEL	LI NC NS PR PUN RDR
jesx.CANCEL.*device*	UPDATE	C, I	*CANCEL	LI
jesx.CANCEL.*type*	UPDATE	C C CA CD CDA P P PP	$C $CO $C,ARMRESTART $C,D $C,D,ARMRESTART $C $CO $C	DA I O ST H H O DA I ST [1] DA I ST DA I ST DA I H O ST H O DA I ST
jesx.CANCEL.*type*	UPDATE	P	SSI	H
jesx.CANCEL.*type*	UPDATE	PP (TSU jobs)	$C	DA
jesx.CANCEL.MONITOR	UPDATE	ZM	*CANCEL	JP
jesx.CANCEL.TCP	UPDATE	C	*CANCEL	NC
jesx.DISPLAY.*resource*[4]	READ	D	$T	RM
jesx.DISPLAY.*type*OUT	READ	L, LL	$DO $L	H O ST DA I
jesx.DISPLAY.*type*	READ	D, DL	$D	ST I DA
jesx.DISPLAY.A	READ	DL	*I	DA
jesx.DISPLAY.APPL	READ	DA	$D	NC NS
jesx.DISPLAY.CLASS	READ	D	*I	JC
jesx.DISPLAY.D	READ	D, DL	*I	LI
jesx.DISPLAY.D	READ	D, DL	*I	PR PUN
jesx.DISPLAY.DEV	READ	D, DL	$D	PR PUN SO RDR
jesx.DISPLAY.G	READ	D, DL	*I	INIT
jesx.DISPLAY.G	READ	DC, DG	*I	JC
jesx.DISPLAY.INITIATOR	READ	D, DL	$D	INIT

Table 76. Action Characters by OPERCMDS Resource Name (continued).

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the JES/MVS Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	Action Character	JES/MVS Command	SDSF Panel
jesx.DISPLAY.JOB	READ	D, DL	$D	JC
jesx.DISPLAY.JOB	READ	D, L, LB, LH, LT	*I	DA I ST
jesx.DISPLAY.JOBE	READ	DE	*I	DA I ST
jesx.DISPLAY.JST	READ	J	$D	SP
jesx.DISPLAY.L	READ	D	$D	LI NC
jesx.DISPLAY.LINE	READ	D	$D	LI NC
jesx.DISPLAY.LINE	READ	DL	$D	NC
jesx.DISPLAY.LOGON	READ	D	$D	NS
jesx.DISPLAY.MAIN	READ	D	*I	JP
jesx.DISPLAY.MAINX	READ	DL	*I	JP
jesx.DISPLAY.MEMBER	READ	D	$D	MAS
jesx.DISPLAY.NETSRV	READ	D, DL	$D *I	NS
jesx.DISPLAY.NJE	READ	D, DL	*I	NO
jesx.DISPLAY.NODE	READ	D, DC, DL, DP	$D	NO
jesx.DISPLAY.Q	READ	D, DL, J	*I Q	SP
jesx.DISPLAY.S	READ	DMA, DME, DMR, DMSS, DMSV, DMU	*I	I ST
jesx.DISPLAY.SOCKET	READ	D	$D *I	NC
jesx.DISPLAY.SOCKET	READ	DS	$D	NS
jesx.DISPLAY.SPOOL	READ	D, DL	$D	SP
jesx.DISPLAY.T	READ	DE	*I	LI
jesx.DISPLAY.U	READ	D	*I U J	J0
jesx.FAIL.DEV.*device*	CONTROL	L	*FAIL	NS
jesx.FAIL.DEV.*device*	CONTROL	L, LD	*FAIL	PR RDR
jesx.FAIL.*device*	CONTROL	L, LD	*FAIL	LI
jesx.FAIL.*dspname*	CONTROL	L, LD	*FAIL	PUN
jesx.FORWARD.DEV	UPDATE	F*number*	$F	PR PUN
jesx.HALT.DEV *jesx*.HALT.INITIATOR *jesx*.HALT.SPOOL	UPDATE CONTROL CONTORL	Z	$Z	PR PUN RDR INIT SP
jesx.INTERRUPT.DEV	UPDATE	I	$I	PR PUN
jesx.MODIFY.G	UPDATE	P, S	*F	INIT

Table 76. Action Characters by OPERCMDS Resource Name (continued).

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the JES/MVS Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	Action Character	JES/MVS Command	SDSF Panel
jesx.MODIFY.JOB	UPDATE	A, C, CA, CD, CDA, CDP, H, P, W	*F	DA I ST
jesx.MODIFY.JOB	UPDATE	J	*F	I ST
jesx.MODIFY.*type*	UPDATE	W	$T	DA I JDS ST
jesx.MODIFY.*type*OUT	UPDATE	A H OK	$TO	O H O H H
jesx.MODIFY.LINE	CONTROL	I Q	$TLINE $TLINE	LI
jesx.MODIFY.NJE	UPDATE	A, EL, H	*F	NO
jesx.MODIFY.Q	UPDATE	A, H, HC, HP, P, U	*F Q	SP
jesx.MODIFY.U	UPDATE	C, H, O, P	*F	JDS J0
jesx.MODIFY.V	UPDATE	PX, V, VF V, VF	*F	JP LI PR PUN RDR
jesx.MODIFYHOLD.*type*	UPDATE	H	$H	DA I ST
jesx.MODIFYRELEASE.*type*	UPDATE	A	$A	DA I ST
*jesx*MON.DISPLAY.DETAIL	READ	JD	$J	MAS
*jesx*MON.DISPLAY.HISTORY	READ	JH	$J	MAS
*jesx*MON.DISPLAY.JES	READ	JJ	$J	MAS
*jesx*MON.DISPLAY.MONITOR	READ	J	$J	MAS
*jesx*MON.DISPLAY.STATUS	READ	JS	$J	MAS
*jesx*MON.STOP.MONITOR	CONTROL	ZM	$J	MAS
jesx.MSEND.CMD	READ	Any	$M	I ST
jesx.RELEASE.*type*OUT	UPDATE	C O P	$O $O $O	H[1] ST H[1]
jesx.REPEAT.DEV	UPDATE	N	$N	PR PUN
jesx.RESTART.BAT	CONTROL	E (all forms)	$E	DA I ST
jesx.RESTART.DEV *jesx*.RESTART.LINE *jesx*.RESTART.SYS	UPDATE CONTROL CONTROL	E, EC E	$E	PR PUN LI SO LI MAS
jesx.RESTART.DEV	UPDATE	E	$E	NC NS
jesx.RESTART.DEV.*device*	UPDATE	E	*R	NS
jesx.RESTART.DEV.*device*	UPDATE	B, E, F	*R	PR PUN
jesx.RESTART.DEV.*main*	CONTROL	E	*R	DA I ST
jesx.RESTART.LINE	CONTROL	E	$E	NC
jesx.RESTART.RJP	UPDATE	E	*R	LI
jesx.START.BAT	UPDATE	J	$SJ	I ST

Table 76. Action Characters by OPERCMDS Resource Name (continued).

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the JES/MVS Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	Action Character	JES/MVS Command	SDSF Panel
jesx.START.DEV	UPDATE	S	$S	NC NS PR PUN LI SO RDR
jesx.START.DEV	UPDATE	SR, ST	$S	SO
jesx.START.DEV.*device*	CONTROL	S	*START	LI
jesx.START.DEV.*device*	UPDATE	S	*START	PR PUN RDR
jesx.START.DEV.*main*	UPDATE	C, F	*S	JP
jesx.START.INITIATOR	CONTROL	S	$S	INIT
jesx.START.JSS	UPDATE	S	*S	JP
jesx.START.LINE	CONTROL	S	$S	LI
jesx.START.MONITOR	UPDATE	DM	*S	JP
jesx.START.NET	CONTROL	SN	$S	LI NC NO
jesx.START.SPOOL	CONTROL	SP	$S	SP
jesx.START.SYS	CONTROL	S	$S	MAS
jesx.START.TCP	UPDATE	SN	*S	NC
jesx.START.TCP	UPDATE	SN	*S	NO
jesx.STOP.DEV	UPDATE	P	$P	NC NS PR PUN LI SO RDR
jesx.STOP.INITIATOR	CONTROL	P	$P	INIT
jesx.STOP.LINE	CONTROL	P	$P	LI
jesx.STOP.SPOOL	CONTROL	P, PC	$P	SP
jesx.STOP.SYS	CONTROL	P	$P	MAS
jesx.STOP.RETURN	CONTROL	P	*RETURN	JP
MVS.CANCEL.*type.jobname*	UPDATE	C CD K, KD P PP	C U=*userid* C U=, DUMP C *jobname*,A=*asid* C U=*userid* C U=*userid* C *jobname*,A=*asid* [2]	DA
MVS.CANCEL.*type.jobname*	UPDATE	C	C U=*userid* C *jobname*,A=*asid*	PS
MVS.CANCEL.STC.*servername*	CONTROL	K, KD	C	NS
MVS.CONTROL.C	READ	C	K C	SR
MVS.DISPLAY.ALLOC	READ	DALO	D ALLOC,OPTIONS	SYS
MVS.DISPLAY.ASM	READ	D	D ASM,PAGE=	PAG
	READ	DC	D ASM,COMMON	PAG
	READ	DD	D ASM,PAGEDEL	PAG
	READ	DL	D ASM,LOCAL	PAG
	READ	DP	D ASM,PLPA	PAG
	READ	DS	D ASM,SCM	PAG

Table 76. Action Characters by OPERCMDS Resource Name (continued).

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the JES/MVS Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	Action Character	JES/MVS Command	SDSF Panel
MVS.DISPLAY.CEE	READ	DCEE	D CEE,ALL	SYS
MVS.DISPLAY.CONSOLES	READ	DC	D C	SYS
MVS.DISPLAY.DUMP	READ	DD	D D,E	SYS
MVS.DISPLAY.EMCS	READ	DEM	D EMCS	SYS
MVS.DISPLAY.GRS	READ	D	D GRS,HEX,RES=	ENQ
	READ	DG	D GRS,SYSTEM	SYS
MVS.DISPLAY.IKJTSO	READ	DTO	D IKJTSO	SYS
MVS.DISPLAY.IOS	READ	DI	D IOS,CONFIG	SYS
MVS.DISPLAY.IQP	READ	DIQP	D IQP	SYS
MVS.DISPLAY.JOB	READ	DTS	D TS,L	SYS
	READ	DAA	D A,ALL	SYS
	READ	DAL	D A,L	SYS
MVS.DISPLAY.LLA	READ	DLL	D LLA	SYS
MVS.DISPLAY.LOGGER	READ	DLO	D LOGGER	SYS
MVS.DISPLAY.LOGREC	READ	DLR	D LOGREC	SYS
MVS.DISPLAY.M	READ	DM	D M	SYS
MVS.DISPLAY.MPF	READ	DMP	D MPF	SYS
MVS.DISPLAY.OMVS	READ	D	D	SP
	READ	DO	OMVS,O	SYS
MVS.DISPLAY.PARMLIB	READ	DE	D PARMLIB,ERRORS	PARM
	READ	D	D PARMLIB	PARM
MVS.DISPLAY.PCIE	READ	DPCD	D PCIE,DD	SYS
	READ	DPCI	D PCIE	SYS
MVS.DISPLAY.PROD	READ	DP	D PROD,REG	SYS

Table 76. Action Characters by OPERCMDS Resource Name (continued).

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the JES/MVS Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	Action Character	JES/MVS Command	SDSF Panel
MVS.DISPLAY.PROG	READ	D	D PROG,APF, DSNAME=	APF
	READ	D	D PROG,EXIT,EX=	DYNX
	READ	D	D PROG,LNKLST, NAME=	LNK
	READ	DA	D PROG,APF,ALL	APF
	READ	DA	D PROG,EXIT,ALL	DYNX
	READ	DAI	D PROG,EXIT,ALL,IMPLICIT	DYNX
	READ	DD	D PROG,EXIT,EX=,DIAG	DYNX
	READ	DI	D PROG,EXIT,INSTALLATION	DYNX
	READ	DN	D PROG,LNKST, NAMES	LNK
	READ	DNP	D PROG,EXIT,NOTPROGRAM	DYNX
	READ	DP	D PROG,EXIT,PROGRAM	DYNX
MVS.DISPLAY.OMVS	READ	D	D	SP
MVS.DISPLAY.R	READ	D	D	SR
MVS.DISPLAY.SLIP	READ	DSL	D SLIP	SYS
MVS.DISPLAY.SMF	READ	DSF	D SMF,O	SYS
MVS.DISPLAY.SMS	READ	DSM	D SMS	SYS
MVS.DISPLAY.SYMBOLS	READ	DSY	D SYMBOLS	SYS
	READ	D	D SYMBOLS,S=	SYM
MVS.DISPLAY.TIMEDATE	READ	DT	D T	SYS
MVS.DISPLAY.TRACE	READ	DTR	D TRACE	SYS
MVS.DISPLAY.XCF	READ	D	D	JD
	READ	DX	D XCF	SYM
MVS.FORCE.*type.jobname* MVS.FORCE.*type.jobname.id*	CONTROL	Z	FORCE	DA
MVS.FORCE.STC.*servername*	CONTROL	Z	FORCE	NS
MVS.MODIFY.STC.*fssproc.fssname*	UPDATE	K	F	PR
MVS.MODIFY.STC.*hcproc.hcstcid*	UPDATE	A, D, DL, DP, DPO, DS, E, H, P, PF, R, U	F	CK
MVS.MODIFY.STC.BPXOINIT.BPXOINIT	UPDATE	K, T	F	PS
MVS.DISPLAY.WLM	READ	D	D	SE RES
	READ	DW	D WLM	SYS
MVS.RESET	UPDATE	R	RESET	DA[RMF]
MVS.RESET	UPDATE	RQ	RESET	DA[RMF]
MVS.REPLY	READ	R	R	SR

Table 76. Action Characters by OPERCMDS Resource Name (continued).

Replace ***jesx*** with the name of the targeted JES subsystem, for example, JES2.

Replace ***type*** with BAT (batch jobs), STC (started tasks), or TSU (TSO users). For APPC transactions, replace ***type*** with STC for transaction SYSOUT on the H and O panels, or ATX for transactions on the DA, I, and ST panels.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the JES/MVS Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	Action Character	JES/MVS Command	SDSF Panel
MVS.ROUTE	READ	Any	ROUTE	DA ENC INIT LI NO MAS PR PS PUN RDR SO [3]
MVS.SETAUTOR.AUTOR	READ	AI	SETAUTOR	SR
MVS.STOP.*type.jobname* MVS.STOP.*type.jobname.id*	UPDATE	Y	STOP	DA[RMF]

Notes:

1 This occurs only on a secondary JES system.

2 This form of the CANCEL command is issued against APPC transaction programs

3 SDSF uses the MVS ROUTE command to route commands to a system in a sysplex other than the one the user is logged on to, for these panels, when they are showing sysplex-wide data: CK, ENC, INIT, LI, NO, PR, PS, PUN, RDR, RM and SO.

4 The SAF resource varies with the JES2 resource. See "JES2 resources."

RMF The DA panel must be using RMF as the source of its data.

JES2 resources

The following table shows the SAF resources in the OPERCMDS class for the JES2 resources displayed on the RM panel.

Table 77. OPERCMDS Resources That Protect Issuing Action Characters for JES2 Resources

JES2 Resource	OPERCMDS Resource	Required Access
BERT	*jesx*.DISPLAY.CKPTSPACE	READ
BSCB	*jesx*.DISPLAY.TPDEF	READ
BUFX	*jesx*.DISPLAY.BUFDEF	READ
CKVR	*jesx*.DISPLAY.CKPTDEF	READ
CMBS	*jesx*.DISPLAY.CONDEF	READ
CMDS	*jesx*.DISPLAY.CONDEF	READ
ICES	*jesx*.DISPLAY.TPDEF	READ
JNUM	*jesx*.DISPLAY.JOBDEF	READ
JOES	*jesx*.DISPLAY.OUTDEF	READ
JQES	*jesx*.DISPLAY.JOBDEF	READ
LBUF	*jesx*.DISPLAY.BUFDEF	READ
NHBS	*jesx*.DISPLAY.NJEDEF	READ
SMFB	*jesx*.DISPLAY.SMFDEF	READ
TBUF	Not applicable	

Table 77. OPERCMDS Resources That Protect Issuing Action Characters for JES2 Resources (continued)

JES2 Resource	OPERCMDS Resource	Required Access
TGS	*jesx*.DISPLAY.SPOOLDEF	READ
TTAB	*jesx*.DISPLAY.TRACEDEF	READ
VTMB	*jesx*.DISPLAY.TPDEF	READ

Authorized program facility data sets

Protecting authorized program facility data sets

Protect authorized program facility data sets by defining resource names in the SDSF class. The resources are shown in Table 78.

Table 78. SAF Resources for Authorized Program Facility Data Sets

Action Characters and Overtypes	Resource Name	Class	Access Required
D	ISFAPF.*datasetname*	SDSF	READ
DA	ISFAPF.*datasetname*	SDSF	READ

To control access to the APF panel, protect the APF command. This is described in "Authorized SDSF commands."

Example of protecting authorized program facility data sets

To protect all authorized program facility data sets and permit a user to control them, define a generic profile as follows:

```
REDEFINE SDSF ISFAPF.** UACC(NONE)
PERMIT ISFAPF.** CLASS(SDSF) ID(userid) ACCESS(READ)
```

Authorized SDSF commands

The authorized SDSF commands are the SERVER parameter on the SDSF command, and the SDSF commands that can be on the AUTH parameter in ISFPARMS, with the addition of OWNER, which can only be protected through SAF. If no SAF protection exists for the OWNER command, then all users can issue the OWNER command.

Only those SDSF panel commands (such as DA, I, and O) for which the user is authorized are displayed on the SDSF Primary Option Menu.

Protecting SDSF commands

Protect authorized SDSF commands by defining resource names in the SDSF class.

SDSF authorized commands and their resource names are listed in Table 79 on page 236. All commands are valid in the JES2 environment; only those commands indicated in the table are valid in the JES3 environment.

Some commands (APF, AS, DYNX, ENQ, LNK, LPA, PAG, PARM, SYM, and SYS) require use of the SDSFAUX address space. Access to SDSFAUX is controlled through access to the ISF.CONNECT.sysname resource. The user must be permitted to this resource in addition to the resources that protect the individual commands.

Table 79. SDSF Class Resource Names and SDSF Commands

Command	SDSF Class Resource Name	Class	Required Access
ABEND	ISFCMD.MAINT.ABEND	SDSF	READ
APF	ISFCMD.ODSP.APF.*system*	SDSF	READ
	ISF.CONNECT.*sysname*	SDSF	READ
AS	ISFCMD.ODSP.AS.*system*	SDSF	READ
	ISF.CONNECT.*sysname*	SDSF	READ
ACTION	ISFCMD.FILTER.ACTION	SDSF	READ
CK	ISFCMD.ODSP.HCHECKER.*system*	SDSF	READ
DA	ISFCMD.DSP.ACTIVE.*jesx*	SDSF	READ
DEST	ISFCMD.FILTER.DEST	SDSF	READ
DYNX	ISFCMD.ODSP.DYNX.*system*	SDSF	READ
	ISF.CONNECT.*sysname*	SDSF	READ
ENC	ISFCMD.ODSP.ENCLAVE.*system*	SDSF	READ
ENQ	ISFCMD.ODSP.ENQUEUE.*system*	SDSF	READ
	ISF.CONNECT.*sysname*	SDSF	READ
FINDLIM	ISFCMD.FILTER.FINDLIM	SDSF	READ
H	ISFCMD.DSP.HELD.*jesx*	SDSF	READ
I	ISFCMD.DSP.INPUT.*jesx*	SDSF	READ
INIT	ISFCMD.ODSP.INITIATOR.*jesx*	SDSF	READ
INPUT	ISFCMD.FILTER.INPUT	SDSF	READ
JC	ISFCMD.ODSP.JOBCLASS.*jesx*	SDSF	READ
JESNAME parameter on SDSF command	ISFCMD.OPT.JESNAME	SDSF	READ
JES3NAME parameter on SDSF command	ISFCMD.OPT.JES3NAME	SDSF	READ
JP and MAS	ISFCMD.ODSP.MAS.*jesx*	SDSF	READ
J0 (JES3 only)	ISFCMD.ODSP.JOB0.*jesx*	SDSF	READ
LI	ISFCMD.ODSP.LINE.*jesx*	SDSF	READ
LNK	ISFCMD.ODSP.LNK.*system*	SDSF	READ
	ISF.CONNECT.*sysname*	SDSF	READ
LOG	ISFCMD.ODSP.SYSLOG.*jesx*	SDSF	READ
LPA	ISFCMD.ODSP.LPA.*system*	SDSF	READ
	ISF.CONNECT.*sysname*	SDSF	READ
MAS and JP	ISFCMD.ODSP.MAS.*jesx*	SDSF	READ
NC	ISFCMD.ODSP.NC.*jesx*	SDSF	READ
NO	ISFCMD.ODSP.NODE.*jesx*	SDSF	READ
NS	ISFCMD.ODSP.NS.*jesx*	SDSF	READ
O	ISFCMD.DSP.OUTPUT.*jesx*	SDSF	READ
OWNER	ISFCMD.FILTER.OWNER	SDSF	READ
PAG	ISFCMD.ODSP.PAGE.*system*	SDSF	READ
	ISF.CONNECT.*sysname*	SDSF	READ

Table 79. SDSF Class Resource Names and SDSF Commands (continued)

Command	SDSF Class Resource Name	Class	Required Access
PARM	ISFCMD.ODSP.PARMLIB.*system*	SDSF	READ
	ISF.CONNECT.*sysname*	SDSF	READ
PR	ISFCMD.ODSP.PRINTER.*jesx*	SDSF	READ
PREFIX	ISFCMD.FILTER.PREFIX	SDSF	READ
PS	ISFCMD.ODSP.PROCESS.*system*	SDSF	READ
PUN	ISFCMD.ODSP.PUNCH.*jesx*	SDSF	READ
RDR	ISFCMD.ODSP.READER.*jesx*	SDSF	READ
RES	ISFCMD.ODSP.RESOURCE.*system*	SDSF	READ
RM (JES2 only)	ISFCMD.ODSP.RESMON.*jesx*	SDSF	READ
RSYS	ISFCMD.FILTER.RSYS	SDSF	READ
SE	ISFCMD.DSP.SCHENV.*system*	SDSF	READ
SERVER parameter on SDSF command	ISFCMD.OPT.SERVER	SDSF	READ
SO (JES2 only)	ISFCMD.ODSP.SO.*jesx*	SDSF	READ
SP	ISFCMD.ODSP.SPOOL.*jesx*	SDSF	READ
SR	ISFCMD.ODSP.SR.*system*	SDSF	READ
ST	ISFCMD.DSP.STATUS.*jesx*	SDSF	READ
SYM	ISFCMD.DSP.SYMBOL.*system*	SDSF	READ
	ISF.CONNECT.*sysname*	SDSF	READ
SYS	ISFCMD.ODSP.SYSTEM.*system*	SDSF	READ
	ISF.CONNECT.*sysname*	SDSF	READ
SYSID	ISFCMD.FILTER.SYSID	SDSF	READ
SYSNAME	ISFCMD.FILTER.SYSNAME	SDSF	READ
TRACE	ISFCMD.MAINT.TRACE	SDSF	READ
ULOG	ISFCMD.ODSP.ULOG.*jesx*	SDSF	READ

The DEST command is protected like any other SDSF authorized command, but you can also protect the destination names used with the DEST command. What is actually shown on the tabular SDSF panels can be affected by your destination authority, as explained in "Destination names" on page 239.

Setting up generic profiles

You can set up different levels of generic profiles to allow use of different kinds of SDSF commands:

Generic Profile	Type of Command	Protects
ISFCMD.**	All	All SDSF authorized commands
ISFCMD.MAINT.*	Maintenance commands	ABEND, TRACE
ISFCMD.DSP.*	End user displays	DA, H, I, O, ST, SE, SYM

Generic Profile	Type of Command	Protects
ISFCMD.ODSP.*	Operator displays	APF, AS, CK, DYNX, ENC, ENQ, INIT, JC, JP, J0, LI, LNK, LOG, LPA, MAS, NC, NO, NS, PAG, PARM, PR, PS, PUN, RDR, RES, RM, SO, SP, SR, SYS, ULOG
ISFCMD.FILTER.*	Filtering commands	ACTION, DEST, FINDLIM, INPUT, OWNER, PREFIX, RSYS, SYSID, SYSNAME
ISFCMD.OPT.**	Parameters on the SDSF command	SERVER

Examples of protecting commands

1. To protect all commands and grant access to user SHERRYF, issue the following:

```
RDEFINE SDSF ISFCMD.** UACC(NONE)
PERMIT ISFCMD.** CLASS(SDSF) ID(SHERRYF) ACCESS(READ)
```

2. To allow access only to the DA, H, I, O, SE and ST panels, issue the following:

```
RDEFINE SDSF ISFCMD.DSP.** UACC(NONE)
PERMIT ISFCMD.DSP.** CLASS(SDSF) ID(SHERRYF) ACCESS(READ)
```

3. To protect the DA command, issue the following:

```
RDEFINE SDSF ISFCMD.DSP.ACTIVE.jesx UACC(NONE)
PERMIT ISFCMD.DSP.ACTIVE.jesx CLASS(SDSF) ID(SHERRYF) ACCESS(READ)
```

Checks on the CK and CKH panels

You can protect the checks from IBM Health Checker for z/OS that are displayed on the CK and CKH panels.

Protecting checks

Protect checks by defining resource names in the XFACILIT class. The resources are shown in Table 80.

Table 80. Authority Required to Checks for Actions and Overtypes

Action Character or Overtypeable Field	Panel	Resource Name	Class	Access
A action character	CK	HZS.*sysname.checkowner.checkname.*ACTIVATE	XFACILIT	UPDATE
D action character	CK	HZS.*sysname.checkowner.checkname.*QUERY	XFACILIT	READ
E action character	CK	HZS.*sysname.checkowner.checkname.*REFRESH	XFACILIT	CONTROL
H action character	CK	HZS.*sysname.checkowner.checkname.*DEACTIVATE	XFACILIT	UPDATE
P action character	CK	HZS.*sysname.checkowner.checkname.*DELETE	XFACILIT	CONTROL
R action character	CK	HZS.*sysname.checkowner.checkname.*RUN	XFACILIT	UPDATE
S and X action characters	CK, CKH	HZS.*sysname.checkowner.checkname.*MESSAGES	XFACILIT	READ
U action character and all overtypeable fields	CK	HZS.*sysname.checkowner.checkname.*UPDATE	XFACILIT	UPDATE

Protect access to the log stream that is used to record check history by defining a resource in the LOGSTRM class.

Table 81. Authority Required to the Log Stream Used for Check History

Action Character or Overtypeable Field	Resource Name	Class	Access
L action character on the CK panel	*log-stream-name*	LOGSTRM	READ

To protect the MVS commands generated by action characters and overtypeable fields on the CK panel, see "Tables of action characters" on page 213 and "Tables of overtypeable fields" on page 258.

To control access to the CK and CKH panels, protect the CK command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting checks

To protect all checks and permit a user to control the checks, you can define generic profiles as follows:

```
RDEFINE XFACILIT HZS.** UACC(NONE)
PERMIT HZS.** CLASS(XFACILIT) ID(userid or groupid) ACCESS(CONTROL)
```

Destination names

You can protect destination names that are used on the DEST command and the IDEST parameter of ISFPARMS.

You can also give users operator authority by destination to jobs, output groups, and SYSIN/SYSOUT data sets without explicitly authorizing the users to the JESSPOOL resources. For more information see "Destination operator authority" on page 240.

The DEST command is protected like any other SDSF authorized command; see "Authorized SDSF commands" on page 235.

Protecting destination names

You use two resources:

- ISFOPER.ANYDEST.*jesx*
- ISFAUTH.DEST.*destname*

You must define the ISFOPER.ANYDEST.*jesx* resource before defining any ISFAUTH.DEST.*destname* resources. Otherwise, unexpected authorization results may occur.

The resources are described in Table 82.

Table 82. Authority Required for Destination Names

Object	Resource Names	Class	Access
Any destination name on the DEST command or IDEST list	ISFOPER.ANYDEST.*jesx*	SDSF	READ
Specific destination names on the DEST command or IDEST list	ISFAUTH.DEST.*destname*	SDSF	READ

In the table,

jesx
is the name of the JES subsystem. For example, it might be *JES2*, *JESA*, or, to protect all JES2 subsystems, *JES%*.

destname
is a destination name in the standard form: ISFAUTH.DEST.N*x*.R*x*

Initializing destinations

Each SDSF user should have a set of default destinations. SDSF uses these default destinations to:

- Initialize the SDSF panels
- Respond to a DEST command that is entered with no parameters

When no default destinations are defined, the user's destination filter is set to blanks or the character string ????????, and no jobs appear on the tabular SDSF panels. To establish the default destinations you can:

- Use the IDEST parameter in ISFPARMS. Refer to "Group function parameters reference" on page 40 for more information.
- Give the user access to all destinations with the ISFOPER.ANYDEST.*jesx* resource.
- Give the user access to specific destinations with the ISFAUTH.DEST.*destname* resource.

If you don't define default destinations with the IDEST parameter, give the user authority to issue the DEST command. DEST allows the user to define a default set of destinations. The command only has to be entered once, as SDSF saves the values across sessions.

Example of protecting destination names

To allow USER1 unlimited use of all destination names, define the following profile and give the user READ authority:

```
RDEFINE SDSF ISFOPER.ANYDEST.jesx UACC(NONE)
PERMIT ISFOPER.ANYDEST.jesx CLASS(SDSF) ID(USER1) ACCESS(READ)
```

Then, to restrict the use of the destination names for USER2, define profiles for a specific destination name and give that user READ authority to only that resource:

```
RDEFINE SDSF ISFAUTH.DEST.RMT1 UACC(NONE)
PERMIT ISFAUTH.DEST.RMT1 CLASS(SDSF) ID(USER2) ACCESS(READ)
```

Destination operator authority

You can give operators access to jobs, output groups, or SYSIN/SYSOUT data sets for a particular destination, without authorizing the operators to those jobs, output groups, or SYSIN/SYSOUT data sets through the JESSPOOL class.

This *destination operator authority* is the equivalent of specifying DEST for CMDAUTH and ADEST for DSPAUTH in ISFPARMS. This is also used for authorizing destinations as described in "Destination names" on page 239.

To provide destination operator authority you:

1. Give the user READ authority to the ISFOPER.DEST.*jesx* profile in the SDSF class. This identifies a user as a destination operator for the SDSF session.
2. Give the user authorization for the profiles that protect destinations for jobs, output groups, and data sets.

The ability to modify output descriptors (Address, Building and so on) on the JDS and OD panels in a JES3 environment cannot be granted using destination operator authority. You must use the resources in the JESSPOOL class, as described in "Jobs, output groups, and SYSIN/SYSOUT data sets" on page 245.

Protecting operator authority by destination

The resources are shown in Table 83.

Table 83. Authority Required for Destination Operator Authority

Action Characters and Overtypeable Fields	Resource Name	Class	Access
//, =, +, ? or Q action characters on the DA, H, I, JDS, J0, O, OD, and ST panels	No security checking is done.	N/A	N/A
S, X, or V action characters on the H, I, JDS, J0, O, OD, and ST panels	ISFOPER.DEST.*jesx* ISFAUTH.DEST.*destname*.DATASET.*dsname*	SDSF	READ READ
S, X, or V action characters on the DA panel	ISFOPER.DEST.*jesx* ISFAUTH.DEST..DATASET.*dsname*	SDSF	READ READ
D or L action characters on the H, I, O, and ST panels	ISFOPER.DEST.*jesx* ISFAUTH.DEST.*destname*	SDSF	READ READ
D or L action characters on the DA panel	ISFOPER.DEST.*jesx* ISFAUTH.DEST.	SDSF	READ READ
All others on the H, I, JDS, J0, O, OD, and ST panels	ISFOPER.DEST.*jesx* ISFAUTH.DEST.*destname*	SDSF	READ ALTER
All others on the DA panel	ISFOPER.DEST.*jesx* ISFAUTH.DEST.	SDSF	READ ALTER

If the user does not have authority to both of the required resources, then the user must have access to the individual job or data set defined in the JESSPOOL class.

If your installation is performing SECLABEL checking, a user must be logged on with the appropriate SECLABEL in order to access the JESSPOOL resources even if the user has operator authorization. For more information about SECLABEL checking, see *z/OS Security Server RACF Security Administrator's Guide* .

The authority level (READ or ALTER) must be the same as the authority for the JESSPOOL resources, as described in "Jobs, output groups, and SYSIN/SYSOUT data sets" on page 245.

Reverting to ISFPARMS from SAF

In most cases, when SAF cannot make a decision, SDSF reverts to ISFPARMS to determine authorization. However, ISFPARMS corresponding to destination operator authority (such as CMDAUTH=DEST and DSPAUTH=ADEST in ISFPARMS) will be bypassed when SDSF reverts to ISFPARMS if:

- The user is authorized to access the ISFOPER.DEST.*jesx* resource but is not authorized to access the ISFAUTH.DEST.*destname*, ISFAUTH.DEST..DATASET.*dsname* or ISFAUTH.DEST.*destname*.DATASET.*dsname* resource
- The SDSF resource ISFOPER.DEST.*jesx* is defined, but the user is not authorized to access the resource. If SAF denies operator authority to a user, ISFPARMS does not override that decision.

Dynamic exit information

Protecting dynamic exits

Protect dynamic exits by defining resource names in the SDSF class. The resources are shown in Table 84.

Table 84. SAF Resources for Dynamic Exits

Action Characters and Overtypes	Resource Name	Class	Access Required
D	ISFDYNX.*exitname*	SDSF	READ
DA	ISFDYNX.*exitname*	SDSF	READ
DAI	ISFDYNX.*exitname*	SDSF	READ
DD	ISFDYNX.*exitname*	SDSF	READ
DI	ISFDYNX.*exitname*	SDSF	READ
DNP	ISFDYNX.*exitname*	SDSF	READ
DP	ISFDYNX.*exitname*	SDSF	READ

To control access to the DYNX panel, protect the DYNX command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting dynamic exits

To protect dynamic exits and permit a user to control it, define a generic profile as follows:

```
REDEFINE SDSF ISFDYNX.** UACC(NONE)
PERMIT ISFDYNX.** CLASS(SDSF) ID(userid) ACCESS(READ)
```

Enclaves

Protecting enclaves

Protect enclaves by defining resource names in the SDSF class. The resources are shown in Table 85.

Table 85. SAF Resources for Enclaves

Action Characters and Overtypes	Resource Name	Class	Access Required
R and RQ action characters and SrvClass overtype	ISFENC.*subsystem-type.subsystem-name*	SDSF	ALTER

To control access to the ENC panel, protect the ENC command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting enclaves

To protect all enclaves and permit a user to control them, define a generic profile as follows:

```
RDEFINE SDSF ISFENC.** UACC(NONE)
PERMIT ISFENC.** CLASS(SDSF) ID(userid) ACCESS(ALTER)
```

Enqueue information

Protecting Enqueue Information

Protect enqueue information by defining resource names in the SDSF class. The resources are shown in Table 86.

Table 86. SAF Resources for Enqueue Information

Action Characters and Overtypes	Resource Name	Class	Access Required
D	ISFENQ.*majorname.sysname*	SDSF	READ

To control access to the ENQ panel, protect the ENQ command. This is described in "Authorized SDSF commands" on page 235.

To protect the N action character to display enqueues from the DA panel, protect the ENQ command. This is described in "Authorized SDSF commands" on page 235. The N action is valid only in the interactive environment. It is not supported in REXX, Java, or the z/OSMF. You can obtain this information by invoking the ENQ panel directly and implementing logic to filter by ASID.

Example of protecting enqueue information

To protect enqueue information and permit a user to control it, define a generic profile as follows:

```
REDEFINE SDSF ISFENQ.** UACC(NONE)
PERMIT ISFENQ.** CLASS(SDSF) ID(userid) ACCESS(READ)
```

Initiators

You can protect the initiators that are displayed on the INIT panel.

Protecting initiators

Protect initiators by defining resource names in the SDSF class. The resources are shown in Table 87.

Table 87. Authority Required to Initiator Resource for Actions and Overtypes

Action Character or Overtypeable Field	Resource Name	Class	Access
D action character	ISFINIT.I(*xx*).*jesx*	SDSF	READ
All others	ISFINIT.I(*xx*).*jesx*	SDSF	CONTROL

In the table, *jesx* is the name of the JES subsystem the initiator is on.

To protect the MVS or JES commands generated by action characters or overtypeable fields, see "Tables of action characters" on page 213 and "Tables of overtypeable fields" on page 258.

To control access to the INIT panel, protect the INIT command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting initiators

To protect all initiators and permit a user to control the initiators, define a generic profile as follows:

```
RDEFINE SDSF ISFINIT.** UACC(NONE)
PERMIT ISFINIT.** CLASS(SDSF) ID(userid) ACCESS(CONTROL)
```

JES2 resources on the RM panel

You can protect the JES2 resources that are displayed on the RM panel (JES2 only).

Protecting JES2 resources

Protect the JES2 resources by defining resource names in the SDSF class. The resources are shown in Table 88.

Table 88. Authority Required to JES2 Resources for Actions and Overtypes

Action Character or Overtypeable Field	Resource Name	Class	Access
D action characters	ISFRM.*resource.jesx*	SDSF	READ
All others	ISFRM.*resource.jesx*	SDSF	CONTROL

The values for *resource* are:

BERT block extension reuse table

BSCB bisynchronous buffers

BUFX extended logical buffers

CKVR checkpoint versions

CMBS console message buffers

CMDS
console message buffers for command processing

ICES SNA interface control elements

LBUF logical buffers

JNUM job numbers

JQES job queue elements

JOES job output elements

NHBS NJE header/trailer buffers

SMFB system management facilities buffers

TGS spool space/track groups

TTAB trace tables

VTMB
VTAM® buffers

To protect the MVS commands generated, see "Tables of action characters" on page 213 and "Tables of overtypeable fields" on page 258.

To control access to the RM panel, protect the RM command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting JES2 resources

To protect all JES2 resources and permit a user to control them, you can define generic profiles as follows:

```
RDEFINE SDSF ISFRM.** UACC(NONE)
PERMIT ISFRM.** CLASS(SDSF) ID(userid or groupid) ACCESS(CONTROL)
```

Job classes

You can protect the job classes that are displayed on the JC panel.

Protecting job classes

Protect job classes by defining resource names in the SDSF class. The resources are shown in Table 89.

Table 89. Authority Required to Job Class Resource for Actions and Overtypes

Action Character or Overtypeable Field	Resource Name	Class	Access
D and ST action characters	ISFJOBCL.*class.jesx*	SDSF	READ
Overtypes	ISFJOBCL.*class.jesx*	SDSF	CONTROL

To protect the MVS or JES commands generated by action characters or overtypeable fields, see "Tables of action characters" on page 213 and "Tables of overtypeable fields" on page 258.

To protect the ST action character, protect the ST command. To control access to the JC panel, protect the JC command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting job classes

To protect all job classes and permit a user to control them, define a generic profile as follows:

```
RDEFINE SDSF ISFJOBCL.** UACC(NONE)
PERMIT ISFJOBCL.** CLASS(SDSF) ID(userid or groupid) ACCESS(CONTROL)
```

Jobs, output groups, and SYSIN/SYSOUT data sets

JES uses the JESSPOOL class to protect SYSIN/SYSOUT data sets. SDSF extends the use of the JESSPOOL class to protect SDSF job and output group resources as well.

SDSF checks a user's SAF authorization to:

- Job resources on the Display Active Users, Input Queue, and Status panels.
- Output groups on the Held Output Queue, Job Data Set, Output Queue, and Output Descriptors panels.
- SYSIN/SYSOUT data sets on the Job Data Set panel, Job 0 panel, and any other panel used for browsing with the S or V action characters and printing with the X action character.

Controlling access to the commands that display jobs and output is described in "Authorized SDSF commands" on page 235.

Protection for each type of resource can be defined separately, so that, for example, a user may be authorized to issue action characters for a job, but not be authorized to browse that job's data sets. Users can always access the JESSPOOL resources they own; they do not need additional authority to work with their own jobs and output.

Protecting jobs, output groups, and SYSIN/SYSOUT data sets

The JESSPOOL resources are described in Table 90.

Table 90. Authority Required to JESSPOOL Resources

Action Character or Overtypeable Field	Resource Name	Class	Access
//, =, +, ? or Q action characters on the DA, H, I, JDS, J0, O, OD and ST panels	No security checking is done.		
S or X action characters on the DA, H, I, J0, O, OD, and ST panels, and S, X or V action characters on the JDS panel	*nodeid.userid.jobname.jobid*.D*dsid.dsname*	JESSPOOL	READ
SJ action character on the DA, H, I, JDS, O, OD, and ST panels	*nodeid.userid.jobname.jobid*.JCL	JESSPOOL	READ
SB and SE action characters on the DA, H, I, JDS, J0, O, OD, and ST panels	*nodeid.userid.jobname.jobid*.JESMSGLG *nodeid.userid.jobname.jobid*.JESYSMSG	JESSPOOL	READ
D or L action characters on the DA, I, and ST panels	*nodeid.userid.jobname.jobid*	JESSPOOL	READ
D or L action characters on the H and O panels (JES2 only)	*nodeid.userid.jobname.jobid*.GROUP.*ogroupid*	JESSPOOL	READ
Overtype output descriptors on the JDS, J0 (JES3) and OD panels	*nodeid.userid.jobname.jobid*.D*dsid.dsname*	JESSPOOL	ALTER
All others on the DA, I, and ST panels	*nodeid.userid.jobname.jobid*	JESSPOOL	ALTER
All others on the JDS, J0, and OD panels, and on the H and O panels (JES2 only)	*nodeid.userid.jobname.jobid*.GROUP.*ogroupid*	JESSPOOL	ALTER

In the table,

nodeid
: is the NJE node ID of the target JES subsystem.

userid
: is the local user ID of the job owner.

jobname
: is the name of the job.

jobid
: is the JES job ID of the:
- job (for jobs on DA, I, and ST)
- job with which the output group is associated (for output groups on H, O, JDS, and OD)

- job with which the data set is associated (for SYSIN or SYSOUT data sets) This contains the type of object that the job is (TSU, JOB, or STC), as well as the job number.

GROUP
is the character string GROUP.

ogroupid
s the output group name as specified through the GRPID=keyword on the MVS //OUTPUT statement describing the group.

D*dsid*
is the data set ID number that identifies the job data set prefixed by the required letter D.

dsname
is the user-specified or system-assigned data set name.

If you don't want to make a distinction between types of resources, you can allow users access to everything with the following profile for USER1 on node N1:

```
RDEFINE JESSPOOL N1.USER1.** UACC(NONE)
```

You may also want to allow users to access all JESSPOOL resources by giving them operator authority, as described in "Destination operator authority" on page 240. Operators do not need explicit authorization to access JESSPOOL resources if they are given operator authority.

In addition, you can use the JESSPOOL class to permit users to select other user's jobs, output, and SYSIN/SYSOUT data sets for browsing, viewing and printing, as described in "Permitting other users to view your data." Also, the JESSPOOL class can be used to provide function comparable to the notify authority provided by ISFPARMS (by specifying NOTIFY for CMDAUTH and DSPAUTH) as described in "Providing function comparable to NOTIFY authority" on page 248.

Typically, when you define SAF authority for JESSPOOL resources, you will also need to define other authority for action characters and overtypeable fields. See Table 75 on page 213 and Table 101 on page 259 for the resources to define them. For most action characters, a user must be authorized for jobs or output groups. However, the S, V, and X action characters require authorization only for SYSIN/SYSOUT data sets. No security checking is made for the object when the ? or Q action character is used.

If your installation is performing security label (SECLABEL) checking, a user must be logged on with the appropriate SECLABEL to access JESSPOOL resources. For more information about SECLABEL checking, see *z/OS Security Server RACF Security Administrator's Guide*.

Permitting other users to view your data

Users can permit others to select their jobs, output groups, and SYSIN/SYSOUT data sets using the S (browse), V (view page mode), and X (print) action characters.

When using the S, V, and X action characters, the user is not automatically authorized to access all SYSIN/SYSOUT data sets within a job or output group when the user is authorized to access the job or output group itself. Security checks are made for each data set within the job or output group to verify the

user's authority to access each data set, and only those SYSIN/SYSOUT data sets to which the user has at least READ authority are displayed.

To protect all of the user's jobs, output groups, and SYSIN/SYSOUT data in the same way, use the following profile to protect resources for USER1 on node N1:

```
RDEFINE JESSPOOL N1.USER1.** UACC(NONE)
```

To just permit USER2 to browse USER1's output:

1. Define the profile:

   ```
   RDEFINE JESSPOOL N1.USER1.*.*.D*.* UACC(NONE)
   ```

2. Permit USER2 to read USER1's output:

   ```
   PERMIT N1.USER1.*.*.D*.* CLASS(JESSPOOL) ID(USER2) ACCESS(READ)
   ```

To provide short-term authorization, a user can overtype the DEST field with another user's user ID. This can be done on either the O or H panels.

Providing function comparable to NOTIFY authority

By specifying a value of NOTIFY for the DSPAUTH and CMDAUTH parameters in the ISFGRP macros or GROUP statements, you can allow a group member to display output and issue commands, respectively, for any job that has the NOTIFY parameter on its job card set to the member's user or group ID. There is no one-to-one SAF equivalent for this authorization.

However, when using RACF, the security administrator and job owner can give a user comparable authority, under the scope of the GENERICOWNER option of the SETROPTS command, through profiles that use the JESSPOOL class, and for CMDAUTH, the OPERCMDS class.

With RACF, when GENERICOWNER processing is in effect, a security administrator can assign ownership to profiles in a general resource class, so that end users can create and/or manipulate those general resource class profiles they own, while ensuring that the end users cannot interfere with profiles created by another user. (For the impact of GENERICOWNER on the CLAUTH user attribute and on the system as a whole, see *z/OS Security Server RACF Security Administrator's Guide*).

For an example of providing NOTIFY authority, see "Examples of protecting jobs and output groups."

Examples of protecting jobs and output groups

1. To protect all jobs for user ID USER1 on node N1, issue the following command:

   ```
   RDEFINE JESSPOOL N1.USER1.*.* UACC(NONE)
   ```

   To permit USER2 to access the resource, issue the following command:

   ```
   PERMIT N1.USER1.*.* CLASS(JESSPOOL) ID(USER2) ACCESS(ALTER)
   ```

2. To protect all output groups for user ID USER1 on node N1, issue the following command:

   ```
   RDEFINE JESSPOOL N1.USER1.*.*.GROUP.* UACC(NONE)
   ```

   Then, to permit USER2 to access this resource, issue the following command:

   ```
   PERMIT N1.USER1.*.*.GROUP.* CLASS(JESSPOOL) ID(USER2) ACCESS(ALTER)
   ```

   The use of the GROUP character string in the fifth qualifier of the profile name distinguishes the output group's profile from other JESSPOOL profiles.

3. To protect all SYSIN/SYSOUT data sets for jobs beginning with DPT on node N1, use the following:

```
RDEFINE JESSPOOL N1.*.DPT*.*.D*.* UACC(NONE)
PERMIT N1.*.DPT*.*.D*.* CLASS(JESSPOOL) ID(USER2) ACCESS(READ)
```

   The use of the D character string in the fifth qualifier of the profile name distinguishes the data set's profile from other JESSPOOL profiles.

4. The following example shows how a security administrator can give USER1 at node N1 authority to control access to his own output via the JESSPOOL class. USER1 can then give authority to USER2 to some or all of that output. A generic refresh for USER2 on the JESSPOOL class generic profiles is required for this support to take effect.

   The security administrator does the following:

   - Activates the GENERICOWNER option:
     ```
     SETROPTS GENERICOWNER
     ```
   - Owns the least specific JESSPOOL profile:
     –
     ```
     RDEFINE JESSPOOL N1.** UACC(NONE) OWNER(SECADM)
     RDEFINE JESSPOOL **  UACC(NONE) OWNER(SECADM)
     ```
   - Gives USER1 the ability to create JESSPOOL profiles more specific than N1.USER1.** and to control access to the jobs, output groups, and SYSIN/SYSOUT data sets governed by those profiles:
     ```
     RDEFINE JESSPOOL N1.USER1.** UACC(NONE) OWNER(USER1)
     ```
     The above profile, along with a generic refresh, restricts a user with JESSPOOL class authorization to create and manipulate only a small subset of profiles within the JESSPOOL class (such as N1.USER1.** and any that are more specific).

     The security administrator should caution the user not to delete the *nodeid.userid*.** profile. If deleted, the user may lose control over any more specific profiles created and the access to them.
   - Gives USER1 class authorization to the JESSPOOL class:
     ```
     ALTUSER USER1 CLAUTH(JESSPOOL)
     ```
   - Effects a generic refresh so this support will take effect for newly created profiles, by either:
     Creating an STC (started task) that will automatically refresh a specific general resource class at specified intervals of time, or
     Instructing USER2, after being permitted by USER1, to log off and logon to effect the refresh. (This method will not work when the JESSPOOL class has SETROPTS RACLIST or GENLIST processing activated.)

   With GENERICOWNER support in effect, USER1 can create and manipulate JESSPOOL profiles to control another user's access to his output. USER1 does this as follows:

   - The profile N1.USER1.** is defined by the security administrator and USER1 has the following output groups on the Held Output Queue panel:

JOBNAME	JOBID	OWNER
JOBA	JOB123	USER1
JOBB	JOB345	USER1
JOBC	JOB678	USER1

   - To permit USER2 to browse only JOB123, USER1 issues the following commands:

```
RDEFINE JESSPOOL N1.USER1.JOBA.JOB123.**
PERMIT N1.USER1.JOBA.JOB123.** CLASS(JESSPOOL) ID(USER2) ACCESS(READ)
```

- To permit USER2 to issue action characters and overtypes against JOB123, USER1 gives USER2 access of ALTER. Also, USER2 must have authority to the OPERCMDS resources for the MVS and JES commands generated, as described in "Action characters" on page 211 and "Overtypeable fields" on page 255.
- For USER2's authorization to take effect, a generic refresh is required. This will be automatic if there is an STC in effect, or USER2 can log off and logon when RACLIST or GENLIST processing for the JESSPOOL class is not in effect.

Lines

You can protect the lines displayed on the LI panel.

Protecting lines

Protect lines by defining resource names in the SDSF class. The resources are shown in Table 91.

Table 91. Authority Required to Lines Resources for Actions and Overtypes

Action Character or Overtypeable Field	Resource Name	Class	Access
D action character	ISFLINE.*device-name.jesx*	SDSF	READ
C action character	ISFLINE.*device-name.jesx*	SDSF	ALTER
All others	ISFLINE.*device-name.jesx*	SDSF	CONTROL

In the table,

device-name
is the name of the line, transmitter, or receiver.

jesx
is the name of the JES subsystem.

To protect the MVS and JES commands generated, see "Tables of action characters" on page 213 and "Tables of overtypeable fields" on page 258.

To control access to the LI panel, protect the LI command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting lines

To protect all lines, issue the following commands:

```
RDEFINE SDSF ISFLINE.** UACC(NONE)
PERMIT ISFLINE.** CLASS(SDSF) ID(userid or groupid) ACCESS(CONTROL)
```

Link list data sets

Protecting link list data sets

Protect link list data sets by defining resource names in the SDSF class. The resources are shown in Table 92 on page 251.

Table 92. SAF Resources for Link List Data Sets

Action Characters and Overtypes	Resource Name	Class	Access Required
D	ISFLNK.*datasetname*	SDSF	READ
DN	ISFLNK.*datasetname*	SDSF	READ

To control access to the LNK panel, protect the LNK command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting link list data sets

To protect all link list data sets and permit a user to control them, define a generic profile as follows:

```
REDEFINE SDSF ISFLNK.** UACC(NONE)
PERMIT ISFLNK.** CLASS(SDSF) ID(userid) ACCESS(READ)
```

MAS and JESPLEX members

You can protect the members of a JES2 MAS, displayed on the MAS panel, and the members of a JES3 JESPLEX, displayed on the JP panel.

Protecting MAS and JESPLEX members

Protect members of a MAS or JESPLEX by defining resource names in the SDSF class. The resources are shown in Table 93.

Table 93. Authority Required to MAS or JESPLEX Members for Actions and Overtypes

Action Character or Overtypeable Field	Resource Name	Class	Access
D, DL (JP only) and J action characters	ISFMEMB.*member-name.jesx*	SDSF	READ
E action character (MAS only)	ISFMEMB.*member-name.jesx*	SDSF	ALTER
P action character (MAS only)	ISFMEMB.*member-name.jesx*	SDSF	ALTER
All others	ISFMEMB.*member-name.jesx*	SDSF	CONTROL

where *member-name* is a member name in a JES2 environment and main name in a JES3 environment.

Commands sent to target systems are routed using the MVS ROUTE command. This occurs when the generated command is for a system other than the one to which the user is logged on to.

To protect the MVS or JES commands generated, see "Tables of action characters" on page 213 and "Tables of overtypeable fields" on page 258.

To control access to the MAS and JP panels, protect the MAS and JP commands. This is described in "Authorized SDSF commands" on page 235.

Example of protecting MAS members

To protect all MAS members and permit a user to control the members, you can define generic profiles as follows:

```
RDEFINE SDSF ISFMEMB.** UACC(NONE)
PERMIT ISFMEMB.** CLASS(SDSF) ID(userid or groupid) ACCESS(ALTER)
```

Membership in groups

You can control membership in groups defined by ISFPARMS using SAF. This is an alternative to using ISFPARMS to control membership in the groups.

Controlling membership in groups

Define a resource in the SDSF class. The resource is shown in Table 94.

Table 94. Authority Required for membership in an ISFPARMS group

Function	Resource Name	Class	Access
Membership in group	GROUP.*group-name.server-name*	SDSF	READ

If the SDSF client is not connected to the SDSF server, *server-name* is blank

For more information, see "Using SAF to control group membership" on page 36.

Example of protecting membership in a group in ISFPARMS

To authorize membership in a group in ISFPARMS, issue the following commands:

```
RDEFINE SDSF GROUP.group-name.server-name UACC(NONE)
PERMIT GROUP.group-name.server-name CLASS(SDSF) ID(userid or groupid)
 ACCESS(READ)
```

MVS and JES commands on the command line

You can control a user's authority to use the SDSF slash (/) command to issue MVS or JES commands from SDSF. SAF checks the user's authority to use the slash command, but does not check the MVS or JES command or the object of the command. MVS and JES command authorization to the OPERCMDS class is done by MVS and JES only after SDSF authorizes use of the slash command.

You should control use of the slash command as you would a console with master authority.

The character for the slash command can be changed from the default, /, to some other character with a custom property in ISFPARMS. For more information, refer to "Customized properties (PROPLIST)" on page 91.

For more information on the console used by SDSF to issue the command, see "Issuing MVS and JES commands" on page 342. For more information on protecting the console, see *z/OS MVS Planning: Operations*.

Protecting the slash command

Protect the slash command by defining a resource name in the SDSF class. The resource is shown in Table 95.

Table 95. Authority Required for the Slash Command

Command	Resource Name	Class	Access
Slash (/)	ISFOPER.SYSTEM	SDSF	READ

Note: The `WHEN(CONSOLE(SDSF))` clause for conditional access checking does not apply to commands issued from the command line.

The character for the slash command can be changed from the default, /, to some other character with a custom property in ISFPARMS. For more information, refer to "Customized properties (PROPLIST)" on page 91.

For more information on the console used by SDSF to issue the command, see "Issuing MVS and JES commands" on page 342. For more information on protecting the console, see *z/OS MVS Planning: Operations*.

Slash command and User Log

The slash command can return a response to the user terminal and write a response to the User Log (ULOG). To have the response sent back to the user's terminal, the user needs authorization to the ULOG command and to the extended console. See "User log (ULOG)" on page 305 for information.

Example of protecting the slash command

To authorize use of the slash command, issue the following commands:

```
RDEFINE SDSF ISFOPER.SYSTEM UACC(NONE)
PERMIT ISFOPER.SYSTEM CLASS(SDSF) ID(userid or groupid) ACCESS(READ)
```

Network connections

You can protect the network connections displayed on the NC panel.

Protecting network connections

Protect network connections by defining resource names in the SDSF class. The resources are shown in Table 96.

Table 96. Authority Required to Network Connection Resources for Actions and Overtypes

Action Character or Overtypeable Field	Resource Name	Class	Access
D action character	ISFAPPL.*device-name.jesx* (APPLs) ISFSOCK.*device-name.jesx* (sockets) ISFLINE.*device-name.jesx* (lines, transmitters or receivers)	SDSF	READ
All others	ISFAPPL.*device-name.jesx* ISFSOCK.*device-name.jesx* ISFLINE.*device-name.jesx*	SDSF	CONTROL

In the table,

device-name
: is the name of the device.

jesx
: is the name of the JES subsystem.

To protect the JES commands generated, see "Tables of action characters" on page 213 and "Tables of overtypeable fields" on page 258.

To control access to the NC panel, protect the NC command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting network connections

To protect all network connections, issue the following commands:

```
RDEFINE SDSF ISFNC.** UACC(NONE)
PERMIT ISFAPPL.** CLASS(SDSF) ID(userid or groupid) ACCESS(CONTROL)
PERMIT ISFSOCK.** CLASS(SDSF) ID(userid or groupid) ACCESS(CONTROL)
PERMIT ISFLIINE.** CLASS(SDSF) ID(userid or groupid) ACCESS(CONTROL)
```

Network servers

You can protect the network servers displayed on the NS panel.

Protecting network servers

Protect network servers by defining resource names in the SDSF class. The resources are shown in Table 97.

Table 97. Authority Required to Network Servers Resources for Actions and Overtypes

Action Character or Overtypeable Field	Resource Name	Class	Access
D action character	ISFNS.*device-name.jesx*	SDSF	READ
All others	ISFNS.*device-name.jesx*	SDSF	CONTROL

In the table,

device-name
is the name of the device.

jesx
is the name of the JES subsystem.

To protect the MVS and JES commands generated, see "Tables of action characters" on page 213 and "Tables of overtypeable fields" on page 258.

To control access to the NS panel, protect the NS command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting network servers

To protect all network servers, issue the following commands:

```
RDEFINE SDSF ISFNS.** UACC(NONE)
PERMIT ISFNS.** CLASS(SDSF) ID(userid or groupid) ACCESS(CONTROL)
```

Nodes

You can protect the nodes displayed on the NO panel.

Protecting nodes

Protect nodes by defining resource names in the SDSF class. The resources are shown in Table 98.

Table 98. Authority Required to Nodes Resources for Actions and Overtypes

Action Character or Overtypeable Field	Resource Name	Class	Access
D action character	ISFNODE.*node-name.jesx*	SDSF	READ
All others	ISFNODE.*node-name.jesx*	SDSF	CONTROL

In the table,

node-name
is the name of the node.

jesx
is the name of the JES subsystem.

To protect the MVS and JES commands generated, see "Tables of action characters" on page 213 and "Tables of overtypeable fields" on page 258.

To control access to the NO panel, protect the NO command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting nodes

To protect all nodes, issue the following commands:

```
RDEFINE SDSF ISFNODE.** UACC(NONE)
PERMIT ISFNODE.** CLASS(SDSF) ID(userid or groupid) ACCESS(CONTROL)
```

OPERLOG

The OPERLOG is a merged, sysplex-wide system message log. It is provided by a log stream, which is a collection of log data used by the MVS System Logger.

You protect the OPERLOG panel by controlling:

- Access to the LOG command, which displays the log panel. This is explained in "Authorized SDSF commands" on page 235.
- Authorization to the log stream used for OPERLOG. The system logger, rather than SDSF, issues a SAF call to ensure the authorization.

Parameters of the LOG command allow users to choose the SYSLOG rather than the OPERLOG. For information on protecting the SYSLOG, see "SYSLOG" on page 302.

Protecting the log stream

Protect the log stream user for OPERLOG by defining a resource name in the LOGSTRM class. The resource is shown in Table 99.

Table 99. Authority Required for Accessing the Log Stream

Function	Resource Name	Class	Access
Access to the log stream	SYSPLEX.OPERLOG	LOGSTRM	READ

Overtypeable fields

Use of an overtypeable field causes an interaction with three resources, all of which must be protected:

- The overtypeable field
- The object of the overtypeable field, such as an initiator, printer, MAS member, or job
- The MVS or JES command generated by overtyping the field

Protecting the overtypeable field

The resource names for the overtypeable fields are in the SDSF class or GSDSF class and have a high level qualifier of ISFATTR. A user must have UPDATE authority to the ISFATTR resource to overtype a field. The fields and their resource names are shown in "Tables of overtypeable fields" on page 258.

If the user is not authorized to overtype the field, the field is displayed on the panel but is not overtypeable. (The ISFFLD macros or the FLD statements of ISFPARMS can be used to control whether a field is displayed.)

Protecting the objects of overtypeable fields

The objects of the overtypeable fields are such things as jobs, output groups, initiators, MAS members, nodes, printers, and so on. For information about protecting the objects see:

- "Checks on the CK and CKH panels" on page 238
- "Destination operator authority" on page 240
- "Enclaves" on page 242
- "Initiators" on page 243
- "JES2 resources on the RM panel" on page 244
- "Job classes" on page 245
- "Jobs, output groups, and SYSIN/SYSOUT data sets" on page 245
- "Lines" on page 250
- "MAS and JESPLEX members" on page 251
- "Network connections" on page 253
- "Network servers" on page 254
- "Nodes" on page 254
- "Printers" on page 295
- "Processes (z/OS UNIX System Services)" on page 296
- "Punches" on page 297
- "Readers" on page 298
- "Resources defined to WLM" on page 298
- "Scheduling environments" on page 299
- "Spool offloaders" on page 301
- "Spool volumes" on page 301
- "System requests" on page 304

Protecting the generated MVS and JES commands

Overtyping fields generates MVS and JES commands. The resource names that protect these commands are in the OPERCMDS class and are shown in "Tables of overtypeable fields" on page 258. The tables also contain the access levels required.

Permitting access only while using SDSF

Users can be conditionally permitted to access OPERCMDS resources so they are authorized to use MVS and JES commands only while they are using SDSF. See "Using conditional access" on page 209 for more information.

Generic profiles

You can set up a generic profile in the SDSF class to allow access to all overtypeable fields. To protect resources individually in the SDSF class with more

restrictive profiles, use the specific resource name associated with the overtypeable field. Table 101 on page 259 contains these resource names.

Generic profiles in the SDSF class that protect different types of overtypeable fields are shown in Table 100. For the generic profiles in the OPERCMDS class, use Table 103 on page 279.

Table 100. Generic Profiles for Overtypeable Fields

Generic Profile	Protects
ISFATTR.**	All
ISFATTR.JOB.**	DA, I, ST (jobs)
ISFATTR.OUTPUT.**	JDS (job data sets), J0 (JES3 job 0), H and O (output groups)
ISFATTR.OUTDESC.**	JDS (job data sets), J0 (JES3 job 0), OD (output descriptors)
ISFATTR.CHECK.**	CK (checks)
ISFATTR.ENCLAVE.**	ENC (enclaves)
ISFATTR.JOBCL.**	JC (job classes)
ISFATTR.LINE.**	LI (lines), NC (network connections)
ISFATTR.LOGON.**	NS (network servers)
ISFATTR.MEMBER.**	MAS (members of the MAS), JP (members of the JESPLEX)
ISFATTR.MODIFY.**	SO (spool offloaders)
ISFATTR.NETOPTS.**	NC, NS
ISFATTR.NODE.**	NO (nodes), NC
ISFATTR.OFFLOAD.**	SO (spool offloaders)
ISFATTR.PROPTS.**	LI, NC, NS, PR (printers), PUN (punches)
ISFATTR.RDR.**	RDR (readers)
ISFATTR.RESMON.**	RM (JES2 resources)
ISFATTR.RESOURCE.**	RES (WLM resources)
ISFATTR.SELECT.**	INIT, LI, NC, NS, PR, PUN, SO (selection criteria for devices)
ISFATTR.SPOOL.**	SP (spool volumes)

Examples of protecting overtypeable fields

In the following examples, *jesx* is the name of the JES2 or JES3 subsystem. For example, it might be *JES2*, *JESA*, or to protect all JES2 subsystems, *JES%*.

1. To protect all overtypeable fields, the objects of the overtypeable fields, and the commands they generate, define the following profiles:

```
RDEFINE SDSF ISFAPPL.** UACC(NONE)
RDEFINE SDSF ISFATTR.** UACC(NONE)
RDEFINE SDSF ISFINIT.** UACC(NONE)
RDEFINE SDSF ISFENC.** UACC(NONE)
RDEFINE SDSF ISFJOBCL.** UACC(NONE)
RDEFINE SDSF ISFLINE.** UACC(NONE)
RDEFINE SDSF ISFNS.** UACC(NONE)
RDEFINE SDSF ISFNODE.** UACC(NONE)
RDEFINE SDSF ISFMEMB.** UACC(NONE)
RDEFINE SDSF ISFRDR.** UACC(NONE)
RDEFINE SDSF ISFRM.** UACC(NONE)
RDEFINE SDSF ISFRES.** UACC(NONE)
```

```
RDEFINE SDSF ISFSO.** UACC(NONE)
RDEFINE SDSF ISFSOCK.** UACC(NONE)
RDEFINE SDSF ISFSP.** UACC(NONE)
RDEFINE WRITER jesx.** UACC(NONE)
RDEFINE JESSPOOL ** UACC(NONE)
RDEFINE OPERCMDS jesx.CALL.** UACC(NONE)
RDEFINE OPERCMDS jesx.MODIFY.** UACC(NONE)
RDEFINE OPERCMDS jesx.RESTART.** UACC(NONE)
RDEFINE OPERCMDS jesx.ROUTE.** UACC(NONE)
RDEFINE OPERCMDS jesx.START.** UACC(NONE)
RDEFINE OPERCMDS MVS.MODIFY.** UACC(NONE)
RDEFINE OPERCMDS MVS.RESET UACC(NONE)
RDEFINE XFACILIT HZS.** UACC(NONE)
```

2. To restrict the use of the overtypeable fields for all output groups on the Held Output Queue and Output Queue panels, define the more restrictive profiles:

```
RDEFINE SDSF ISFATTR.OUTPUT.** UACC(NONE)
RDEFINE JESSPOOL *.*.*.*.GROUP.* UACC(NONE)
RDEFINE OPERCMDS jesx.MODIFY.BATOUT UACC(NONE)
RDEFINE OPERCMDS jesx.MODIFY.STCOUT UACC(NONE)
RDEFINE OPERCMDS jesx.MODIFY.TSUOUT UACC(NONE)
```

3. To further restrict the use to only the DEST field on the Held Output Queue and Output Queue panels, define the more restrictive profiles:

```
RDEFINE SDSF ISFATTR.OUTPUT.DEST UACC(NONE)
RDEFINE JESSPOOL *.*.*.*.GROUP.* UACC(NONE)
RDEFINE OPERCMDS jesx.MODIFY.BATOUT UACC(NONE)
RDEFINE OPERCMDS jesx.MODIFY.STCOUT UACC(NONE)
RDEFINE OPERCMDS jesx.MODIFY.TSUOUT UACC(NONE)
```

Tables of overtypeable fields

The following tables describe the SDSF classes and resource names for each overtypeable field and the panels on which they are valid. The table shows the command that is issued, and the associated OPERCMDS resource, for the JES2 environment for each overtypeable field; if the field is overtypeable in the JES3 environment, the JES3 command and associated OPERCMDS resource are shown beneath the JES2 values.

For an alphabetical list by field name, see Table 101 on page 259.

For an alphabetical list by OPERCMDS resource name, see Table 103 on page 279.

Appendix C, “SDSF resource names for SAF security,” on page 577 contains a table of all overtypeable fields.

Table 101. Overtypeable Fields.

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

When a set of related fields can be overtyped with the Overtype Extension pop-up, all of the fields in the set are protected by the same resource.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an overtypeable field does not apply in a particular JES environment, the command and OPERCMDS resource are shown as a hyphen (-).

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
System	RES	ISFATTR.RESOURCE.*system*	F	MVS.MODIFY.WLM	UPDATE
ACCT	JC	ISFATTR.JOBCL.ACCT	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
ACTIVE	JC	ISFATTR.JOBCL.ACTIVE	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
ADDRESS	JDS OD	ISFATTR.OUTDESC.ADDRESS	SSI		
			SSI		
ADISC	LI	ISFATTR.LINE.AUTODISC	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-
AFPPARMS	JDS OD	ISFATTR.OUTDESC.AFPPARMS	SSI		
			SSI		
ALLOC	INIT	ISFATTR.INIT.ALLOC	-	-	-
			*F	*jesx*.MODIFY.G	UPDATE
ANODE	NC	ISFATTR.NETOPTS.NODE	$T	*jesx*.MODIFY.APPL *jesx*.MODIFY.SOCKET *jesx*.MODIFY.LINE	CONTROL
			-	-	-
APPL	NS	ISFATTR.NETOPTS.APPL	$T	*jesx*.MODIFY.LOGON	CONTROL
			-	-	-
APPLID	LI	ISFATTR.LINE.APPLID	$S	*jesx*.START.NET	CONTROL
			-	-	-
ARCHIVE	SO	ISFATTR.OFFLOAD.ARCHIVE	$T	*jesx*.MODIFY.OFFLOAD	CONTROL
ASIS	PR	ISFATTR.PROPTS.ASIS	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
AUTH	JC	ISFATTR.JOBCL.AUTH	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
AUTHORITY	RDR	ISFATTR.RDR.AUTHORITY	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
AUTHORITY	NO	ISFATTR.NODE.AUTHORITY	$T	*jesx*.MODIFY.NODE	CONTROL
			-	-	-
B	PR PUN	ISFATTR.PROPTS.BPAGE	-	-	-
			See note 3.		
BARRIER	INIT	ISFATTR.INIT.BARRIER	-	-	-
			*F	*jesx*.MODIFY.G	UPDATE
BLP	JC	ISFATTR.JOBCL.BLP	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
BUILDING	JDS OD	ISFATTR.OUTDESC.BLDG	SSI		
			SSI		

Table 101. Overtypeable Fields (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

When a set of related fields can be overtyped with the Overtype Extension pop-up, all of the fields in the set are protected by the same resource.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an overtypeable field does not apply in a particular JES environment, the command and OPERCMDS resource are shown as a hyphen (-).

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
BURST	H O	ISFATTR.OUTPUT.BURST	$TO	*jesx*.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE
BURST	JDS J0	ISFATTR.OUTPUT.BURST	-	-	-
			*F	*jesx*.MODIFY.U	UPDATE
C	H O	ISFATTR.OUTPUT.CLASS	$TO[1]	*jesx*.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE
C	JDS J0	ISFATTR.OUTPUT.CLASS	SSI[1]		
			*F	*jesx*.MODIFY.U	UPDATE
C	I ST	ISFATTR.JOB.CLASS	$T	*jesx*.MODIFY.BAT *jesx*.MODIFY.STC *jesx*.MODIFY.TSU	UPDATE
			*F J	*jesx*.MODIFY.JOB	UPDATE
C	RDR	ISFATTR.RDR.CLASS	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
CC	JDS J0	ISFATTR.OUTPUT.COPYCNT	SSI		
			*F	*jesx*.MODIFY.U	UPDATE
CATEGORY	CK	ISFATTR.CHECK.CATEGORY	F	MVS.MODIFY.STC. *hcproc.hcstcid*	UPDATE
CB	PR	ISFATTR.PROPTS.CB	-	-	-
			*S, *X	See note 3.	
CCTL	PR PUN	ISFATTR.PROPTS.CCTL	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
CHARS	JDS J0	ISFATTR.OUTPUT.CHARS	-	-	-
			*F	*jesx*.MODIFY.U	UPDATE
CHAR1–4	PR	ISFATTR.PROPTS.CHAR	$T	*jesx*.MODIFY.DEV	UPDATE
CHAR1			See note 3.		
CKPTHOLD	MAS	ISFATTR.MEMBER.CKPTHOLD	$T	*jesx*.MODIFY.MASDEF	CONTROL
CKPTLINE	PR	ISFATTR.PROPTS.CKPTLINE	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
CKPTLINE	PUN	ISFATTR.PROPTS.CKPTLINE	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
CKPTMODE	PR	ISFATTR.PROPTS.CKPTMODE	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
CKPTPAGE	PR	ISFATTR.PROPTS.CKPTPAGE	$T	*jesx*.MODIFY.DEV	UPDATE
			See note 3.		

Table 101. Overtypeable Fields (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

When a set of related fields can be overtyped with the Overtype Extension pop-up, all of the fields in the set are protected by the same resource.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an overtypeable field does not apply in a particular JES environment, the command and OPERCMDS resource are shown as a hyphen (-).

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
CKPTPAGE	PUN	ISFATTR.PROPTS.CKPTPAGE	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
CKPTSEC	PR	ISFATTR.PROPTS.CKPTSEC	$T	*jesx*.MODIFY.DEV	UPDATE
			See note 3.		
CLASSES	INIT	ISFATTR.SELECT.JOBCLASS	$T	*jesx*.MODIFY.INITIATOR	CONTROL
			-	-	-
CLASS1-8	INIT	ISFATTR.SELECT.JOBCLASS	$T	*jesx*.MODIFY.INITIATOR	CONTROL
			-	-	-
CMPCT	PR PUN	ISFATTR.PROPTS.CMPCT	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
CODE	LI	ISFATTR.LINE.CODE	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-
COLORMAP	JDS OD	ISFATTR.OUTDESC. COLORMAP	SSI		
			SSI		
COMMAND	JC	ISFATTR.JOBCL.COMMAND	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
COMP	LI	ISFATTR.LINE.COMPRESS	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-
COMP	PR PUN	ISFATTR.PROPTS. COMPRESS	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
COMPACT	NC	ISFATTR.NODE.COMPACT	$T	*jesx*.MODIFY.APPL	CONTROL
			-	-	-
COMPACT	PR PUN	ISFATTR.PROPTS.COMPACT	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
COMSETUP	JDS OD	ISFATTR.OUTDESC.COMSETUP	SSI		
			SSI		
CONNECT	LI	ISFATTR.NETOPTS.CONNECT	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-
CONNECT	NC	ISFATTR.NETOPTS.CONNECT	$T	*jesx*.MODIFY.APPL *jesx*.MODIFY.SOCKET *jesx*.MODIFY.LINE	CONTROL
			-	-	-
CONNECT	NO	ISFATTR.NETOPTS.CONNECT	$T	*jesx*.MODIFY.NODE	CONTROL
			-	-	-
CONN-INT	LI	ISFATTR.NETOPTS.CTIME	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-

Table 101. Overtypeable Fields (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

When a set of related fields can be overtyped with the Overtype Extension pop-up, all of the fields in the set are protected by the same resource.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an overtypeable field does not apply in a particular JES environment, the command and OPERCMDS resource are shown as a hyphen (-).

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
CONN-INT	NC	ISFATTR.NETOPTS.CTIME	$T	*jesx*.MODIFY.APPL *jesx*.MODIFY.SOCKET *jesx*.MODIFY.LINE	CONTROL
			-	-	-
CONN-INT	NO	ISFATTR.NETOPTS.CTIME	$T	*jesx*.MODIFY.NODE	CONTROL
			-	-	-
COPIES	PR PUN	ISFATTR.PROPTS.COPIES	-	-	-
			See note 3.		
COPYMARK	PR	ISFATTR.PROPTS.COPYMARK	$T	*jesx*.MODIFY.DEV	UPDATE
			See note 3.		
CP	NO	ISFATTR.NODE.COMPACT	$T	*jesx*.MODIFY.NODE	CONTROL
			-	-	-
CPR	JC	ISFATTR.JOBCL.CONDPURG	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
CPY	JC	ISFATTR.JOBCL.COPY	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
CPYMOD	JDS	ISFATTR.OUTPUT.COPYMOD	-	-	-
			*F	*jesx*.MODIFY.U	UPDATE
CPYMOD	J0 PR	ISFATTR.PROPTS.COPYMOD	$T	*jesx*.MODIFY.DEV	UPDATE
			*S	*jesx*.START.DEV.*device*	
CRTIME	SO	ISFATTR.OFFLOAD.CRTIME	$T	*jesx*.MODIFY.OFFLOAD	CONTROL
CTR	LI	ISFATTR.PROPTS.CTRACE	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-
CTR	NC	ISFATTR.PROPTS.CTRACE	$T	*jesx*.MODIFY.LINE	CONTROL
			*F	*jesx*.MODIFY.SOCKET	UPDATE
CTR	NS	ISFATTR.PROPTS.CTRACE	$T	*jesx*.MODIFY.NETSRV	CONTROL
			*F	*jesx*.MODIFY.NETSERV	
DEBUG	CK	ISFATTR.CHECK.DEBUG	F	MVS.MODIFY.STC. *hcproc.hcstcid*	UPDATE
DEFCOUNT	INIT	ISFATTR.INIT.DEFCNT	-	-	-
			*F	*jesx*.MODIFY.G	UPDATE
DEPARTMENT	JDS OD	ISFATTR.OUTDESC.DEPT	SSI		
			SSI		
DEST	H O	ISFATTR.OUTPUT.DEST	$TOF[1]	*jesx*.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE
			-	-	-

Table 101. Overtypeable Fields (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

When a set of related fields can be overtyped with the Overtype Extension pop-up, all of the fields in the set are protected by the same resource.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an overtypeable field does not apply in a particular JES environment, the command and OPERCMDS resource are shown as a hyphen (-).

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
DEST	JDS J0	ISFATTR.OUTPUT.DEST	SSI[1]		
			*F	*jesx*.MODIFY.U	UPDATE
DEST (secondary JES2)	H	ISFATTR.OUTPUT.DEST	$O	*jesx*.RELEASE.BATOUT *jesx*.RELEASE.STCOUT *jesx*.RELEASE.TSUOUT	UPDATE
			-	-	-
DFCB	PR	ISFATTR.PROPTS.DEVFCB	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
DGRPY	PR PUN	ISFATTR.PROPTS.DGRPY	-	-	-
			*F	*jesx*.MODIFY.W	UPDATE
DIRECT	NO	SFATTR.NODE,DIRECT	$T	*jesx*.MODIFY.NODE	CONTROL
			-	-	-
DORMANCY	MAS	ISFATTR.MEMBER. DORMANCY	$T	*jesx*.MODIFY.MASDEF	CONTROL
DSENQSHR	JC	ISFATTR.JOBCL.DSENQSHR	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
			-	-	-
DSNAME	SO	ISFATTR.OFFLOAD.DATASET	$T	*jesx*.MODIFY.OFFLOAD	CONTROL
DUPLEX	LI	ISFATTR.LINE.DUPLEX	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-
DYN	PR PUN	ISFATTR.PROPTS.DYN	-	-	-
			*F	*jesx*.MODIFY.W	UPDATE
EINTERVAL	CK	ISFATTR.CHECK.EINTERVAL	F	MVS.MODIFY.STC. *hcproc.hcstcid*	UPDATE
END	NO	ISFATTR.NODE.ENDNODE	$T	*jesx*.MODIFY.NODE	CONTROL
			-	-	-
EXECNODE	I ST	ISFATTR.JOB.EXECNODE	$R	*jesx*.ROUTE.JOBOUT	UPDATE
			-	-	-
FCB	H O	ISFATTR.OUTPUT.FCB	$TO	*jesx*.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE
			-	-	-
FCB	JDS J0	ISFATTR.OUTPUT.FCB	-	-	-
			*F U	*jesx*.MODIFY.U	UPDATE
FCBL	PR	ISFATTR.PROPTS.FCBLOAD	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-

Table 101. Overtypeable Fields (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

When a set of related fields can be overtyped with the Overtype Extension pop-up, all of the fields in the set are protected by the same resource.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an overtypeable field does not apply in a particular JES environment, the command and OPERCMDS resource are shown as a hyphen (-).

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
FLASH	H O	ISFATTR.OUTPUT.FLASH	$TO	*jesx*.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE
			-	-	-
FLASH	JDS J0	ISFATTR.OUTPUT.FLASH	-	-	-
			*F	*jesx*.MODIFY.U	UPDATE
FLS	PUN	ISFATTR.PROPTS.FLUSH	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
FORMDEF	JDS OD	ISFATTR.OUTDESC.FORMDEF	SSI		
			SSI		
FORMLEN	JDS OD	ISFATTR.OUTDESC.FORMLEN	SSI		
			SSI		
FORMS	H O	ISFATTR.OUTPUT.FORMS	$TO	*jesx*.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE
			-	-	-
FORMS	JDS J0	ISFATTR.OUTPUT.FORMS	SSI		
			*F U	*jesx*.MODIFY.U	UPDATE
FSATRACE	PR	ISFATTR.PROPTS.FSATRACE	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
FSSNAME	PR	ISFATTR.PROPTS.FSSNAME	F	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
GROUP	INIT	ISFATTR.INIT.GROUP	-	-	-
			*F	*jesx*.MODIFY.C	UPDATE
GROUP	JC	ISFATTR.JOBCL.GROUP	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
HOLD	JC	ISFATTR.JOBCL.HOLD	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
HOLD	RDR	ISFATTR.RDR.HOLD	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
HONORTRC	PR	ISFATTR.PROPTS.HONORTRC	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
INTERVAL	CK	ISFATTR.CHECK.INTERVAL	F	MVS.MODIFY.STC. *hcproc.hcstcid*	UPDATE
INTF	LI	ISFATTR.LINE.INTERFACE	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-
INTRAY	OD	ISFATTR.OUTDESC.INTRAY	SSI		

Table 101. Overtypeable Fields (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

When a set of related fields can be overtyped with the Overtype Extension pop-up, all of the fields in the set are protected by the same resource.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an overtypeable field does not apply in a particular JES environment, the command and OPERCMDS resource are shown as a hyphen (-).

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
IP DESTINATION	OD	ISFATTR.OUTDESC.IPDEST	SSI		
IPNAME	NC	ISFATTR.NETOPTS.IPNAME	$T	*jesx*.MODIFY.SOCKET	CONTROL
			*F	*jesx*.MODIFY.SOCKET	UPDATE
IPNAME	NS	ISFATTR.NETOPTS.IPNAME	$T	*jesx*.MODIFY.SOCKET	CONTROL
			*F	*jesx*.MODIFY.NETSERV	UPDATE
ITY	JDS	ISFATTR.OUTDESC.INTRAY	SSI		
			SSI		
JCLIM	JC	ISFATTR.JOBCL.JCLIM	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
JESLOG	JC	ISFATTR.JOBCL.JESLOG	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
			*F	*jesx*.MODIFY.C	UPDATE
JOBRC	JC	ISFATTR.JOBCL.JOBRC	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
			-	-	-
JRNL	JC	ISFATTR.JOBCL.JOURNAL	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
JRNUM	LI	ISFATTR.LINE.JRNUM	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-
JRNUM	NO	ISFATTR.NODE.JRNUM	-	-	-
			*F	*jesx*.MODIFY.NJE	UPDATE
JTNUM	LI	ISFATTR.LINE.JTNUM	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-
JTNUM	NO	ISFATTR.NODE.JTNUM	-	-	-
			*F	*jesx*.MODIFY.NJE	UPDATE
JTR	LI	ISFATTR.PROPTS.JTRACE	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-
JTR	NC	ISFATTR.PROPTS.JTRACE	$T	*jesx*.MODIFY.LINE	CONTROL
			*F	*jesx*.MODIFY.SOCKET	UPDATE
JTR	NS	ISFATTR.PROPTS.JTRACE	$T	*jesx*.MODIFY.NETSRV	CONTROL
			*F	*jesx*.MODIFY.NETSERV	UPDATE
K	PR	ISFATTR.PROPTS.SPACE	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
LABEL	SO	ISFATTR.OFFLOAD.LABEL	$T	*jesx*.MODIFY.OFFLOAD	CONTROL
LIMIT	RM	ISFATTR.RESMON.LIMIT	$T	*jesx*.MODIFY.*resource*[2]	CONTROL
LINE	NC	ISFATTR.NODE.LINE	$T	*jesx*.MODIFY.APPL *jesx*.MODIFY.SOCKET	CONTROL
			-	-	-

Table 101. Overtypeable Fields (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

When a set of related fields can be overtyped with the Overtype Extension pop-up, all of the fields in the set are protected by the same resource.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an overtypeable field does not apply in a particular JES environment, the command and OPERCMDS resource are shown as a hyphen (-).

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
LINE	NO	ISFATTR.NODE.LINE	$T	*jesx*.MODIFY.NODE	CONTROL
			-	-	-
LINECCHR	LI	ISFATTR.LINE.LINECCHR	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-
LINE-LIMIT	LI NC	ISFATTR.SELECT.LIM	$T	*jesx*.MODIFY.L	CONTROL
			-	-	-
LINE-LIMIT	PR	ISFATTR.SELECT.LIM	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
LINE-LIMIT	PUN	ISFATTR.SELECT.LIM	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
LINE-LIMIT	SO	ISFATTR.SELECT.LIM	$T	*jesx*.MODIFY.OFF	CONTROL
LINE-LIM-HI	PR PUN	ISFATTR.SELECT.LIM	-	-	-
			See note 3.		
LINE-LIM-LOW	PR PUN	ISFATTR.SELECT.LIM	-	-	-
			See note 3.		
LOG	LI	ISFATTR.LINE.LOG	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-
LOG	JC	ISFATTR.JOBCL.JLOG	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
			*F	*jesx*.MODIFY.C	UPDATE
LOG	NS	ISFATTR.NETOPTS.LOG	$T	*jesx*.MODIFY.LOGON	CONTROL
			-	-	-
LOGMODE	NC	ISFATTR.NODE.LOGMODE	$T	*jesx*.MODIFY.APPL	CONTROL
			-	-	
LOGMODE	NO	ISFATTR.NODE.LOGMODE	$T	*jesx*.MODIFY.NODE	CONTROL
			-	-	-
LOGON	NC	ISFATTR.NETOPTS.LOGON	$T	*jesx*.MODIFY.APPL	CONTROL
			-	-	-
LOGON	NO	ISFATTR.NODE.LOGON	$T	*jesx*.MODIFY.NODE	CONTROL
			-	-	-
LRECL	PUN	ISFATTR.PROPTS.LRECL	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
M	PR	ISFATTR.PROPTS.MARK	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
MAX-TIME	JC	ISFATTR.JOBCL.TIME	$T	*jesx*.MODIFY.JOBCLASS	CONTROL

Table 101. Overtypeable Fields (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

When a set of related fields can be overtyped with the Overtype Extension pop-up, all of the fields in the set are protected by the same resource.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an overtypeable field does not apply in a particular JES environment, the command and OPERCMDS resource are shown as a hyphen (-).

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
MAXRETRIES	NO	ISFATTR.NODE.MAXRETR	-	-	-
			*F	*jesx*.MODIFY.NJE	UPDATE
MBURST	SO	ISFATTR.MODIFY.BURST	$T	*jesx*.MODIFY.OFF	CONTROL
MC	RDR	ISFATTR.RDR.MCLASS	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
MC	JC	ISFATTR.JOBCL.MSGCLASS	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
MCLASS	SO	ISFATTR.MODIFY.CLASS	$T	*jesx*.MODIFY.OFF	CONTROL
MDEST	SO	ISFATTR.MODIFY.DEST	$T	*jesx*.MODIFY.OFF	CONTROL
MFCB	SO	ISFATTR.MODIFY.FCB	$T	*jesx*.MODIFY.OFF	CONTROL
MFLH	SO	ISFATTR.MODIFY.FLASH	$T	*jesx*.MODIFY.OFF	CONTROL
MFORMS	SO	ISFATTR.MODIFY.FORMS	$T	*jesx*.MODIFY.OFF	CONTROL
MHOLD	SO	ISFATTR.MODIFY.HOLD	$T	*jesx*.MODIFY.OFF	CONTROL
MINPCT	SP	ISFATTR.SPOOL.MINPCT	-	-	-
			*F Q	*jesx*,MODIFY.Q	UPDATE
MODE	INIT	ISFATTR.INIT.MODE	-	-	-
			*F	*jesx*,MODIFY.G	UPDATE
MODE	JC	ISFATTR.JOBCL.MODE	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
MODE	PR	ISFATTR.PRPOPTS.MODE	$T	*jesx*.MODIFY.DEV	UPDATE
			*F	*jesx*.MODIFY.F	
MODSP	SO	ISFATTR.MODIFY.ODISP	$T	*jesx*.MODIFY.OFF	CONTROL
MPRMODE	SO	ISFATTR.MODIFY.PRMODE	$T	*jesx*.MODIFY.OFF	CONTROL
MSAFF	SO	ISFATTR.MODIFY.SYSAFF	$T	*jesx*.MODIFY.OFF	CONTROL
MSGLV	JC	ISFATTR.JOBCL.MSGLEVEL	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
MUCS	SO	ISFATTR.MODIFY.UCS	$T	*jesx*.MODIFY.OFF	CONTROL
MWRITER	SO	ISFATTR.MODIFY.WRITER	$T	*jesx*.MODIFY.OFF	CONTROL
NAME	JDS OD	ISFATTR.OUTDESC.NAME	SSI		
			SSI		
NETSRV	NC	ISFATTR.NETOPTS.NETSRV	$T	*jesx*.MODIFY.SOCKET	CONTROL
			-	-	-
NETSRV	NO	ISFATTR.NODE.NETSRV	$T	*jesx*.MODIFY.NODE	CONTROL
			-	-	-
NEWPAGE	PR	ISFATTR.PROPTS.NEWPAGE	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-

Table 101. Overtypeable Fields (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

When a set of related fields can be overtyped with the Overtype Extension pop-up, all of the fields in the set are protected by the same resource.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an overtypeable field does not apply in a particular JES environment, the command and OPERCMDS resource are shown as a hyphen (-).

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
NHOLD	NO	ISFATTR.NODE.NETHOLD	-	-	-
			*F	*jesx*.MODIFY.NJE	UPDATE
NODE	LI	ISFATTR.LINE.NODE	$SN	*jesx*.START.NET	CONTROL
			*X	*jesx*.CALL.NJE	UPDATE
NODENAME	NO	ISFATTR.NODE.NODENAME	$T	*jesx*.MODIFY.NODE	CONTROL
			-	-	-
NOTIFY	JDS OD	ISFATTR.OUTDESC.NOTIFY	SSI		
			SSI		
NOTIFY	SO	ISFATTR.OFFLOAD.NOTIFY	$T	*jesx*.MODIFY.OFF	CONTROL
NPRO	PR	ISFATTR.PROPTS.NPRO	$T	*jesx*.MODIFY.DEV	UPDATE
			See note 3.		
OCOPYCNT	JDS OD	SFATTR.OUTDESC.OCOPYCNT	SSI		
			SSI		
ODISP	JC	ISFATTR.JOBCL.ODISP	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
ODISP	H O	ISFATTR.OUTPUT.ODISP	$TO	*jesx*.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE
			-	-	-
OFFSETXB	JDS OD	ISFATTR.OUTDESC. OFFSETXB	SSI		
			SSI		
OFFSETXF	JDS OD	ISFATTR.OUTDESC. OFFSETXF	SSI		
			SSI		
OFFSETYB	JDS OD	ISFATTR.OUTDESC. OFFSETYB	SSI		
			SSI		
OFFSETYF	JDS OD	ISFATTR.OUTDESC. OFFSETYF	SSI		
			SSI		
OPLOG	PR	ISFATTR.PROPTS.OPACTLOG	-	-	-
			*F	*jesx*.MODIFY.W	UPDATE
OUT	JC	ISFATTR.JOBCL.OUTPUT	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
OUTBIN	OD	ISFATTR.OUTDESC.OUTBIN	SSI		
			SSI		
OUTBN	JDS	ISFATTR.OUTDESC.OUTBIN	SSI		
			SSI		
OVERFNAM	SP	ISFATTR.SPOOL.OVFNAME	-	-	-
			*F Q	*jesx*.MODIFY.Q	UPDATE

Table 101. Overtypeable Fields (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

When a set of related fields can be overtyped with the Overtype Extension pop-up, all of the fields in the set are protected by the same resource.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an overtypeable field does not apply in a particular JES environment, the command and OPERCMDS resource are shown as a hyphen (-).

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
OVERLAYB	JDS OD	ISFATTR.OUTDESC. OVERLAYB	SSI		
			SSI		
OVERLAYF	JDS OD	ISFATTR.OUTDESC. OVERLAYF	SSI		
			SSI		
PAGEDEF	JDS OD	ISFATTR.OUTDESC.PAGEDEF	SSI		
			SSI		
PAGE-LIMIT	LI NC	ISFATTR.SELECT.PLIM	$T	*jesx*.MODIFY.L	CONTROL
			-	-	-
PAGE-LIMIT	PR	ISFATTR.SELECT.PLIM	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
PAGE-LIMIT	SO	ISFATTR.SELECT.PLIM	$T	*jesx*.MODIFY.OFF	CONTROL
PAGE-LIM-HI	PR	ISFATTR.SELECT.PLIM	-	-	-
			See note 3.		
PAGE-LIM-LOW	PR	ISFATTR.SELECT.PLIM	-	-	-
			See note 3.		
PARAMETERS	CK	ISFATTR.CHECK.PARM	F	MVS.MODIFY.STC. *hcproc.hcstcid*	UPDATE
PARTNAME	JC	ISFATTR.JOBCL.PARTNAME	-	-	-
			*F	*jesx*.MODIFY.C	UPDATE
PARTNAME	JP	ISFATTR.SPOOL.SPARTN	-	-	-
			*F	*jesx*.MODIFY.G	UPDATE
PARTNAME	NO	ISFATTR.NODE.PARTNAM	-	-	-
			*F	*jesx*.MODIFY.NJE	UPDATE
PARTNAME	SP	ISFATTR.SPOOL.PARTNAME	-	-	-
			*F Q	*jesx*.MODIFY.Q	UPDATE
PASSWORD	LI	ISFATTR.LINE.PASSWORD	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-
PASSWORD	NS	ISFATTR.LOGON.PASSWORD	$T	*jesx*.MODIFY.LOGON	CONTROL
			-	-	-
PATH	NO	ISFATTR.NODE.PATH	-	-	-
			*F	*jesx*.MODIFY.NJE	UPDATE
PAU	PR PUN	ISFATTR.PROPTS.PAUSE	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
PDEFAULT	PR	ISFATTR.PROPTS.PDEFAULT	-	-	-
			*F	*jesx*.MODIFY.F	CONTROL

Table 101. Overtypeable Fields (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

When a set of related fields can be overtyped with the Overtype Extension pop-up, all of the fields in the set are protected by the same resource.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an overtypeable field does not apply in a particular JES environment, the command and OPERCMDS resource are shown as a hyphen (-).

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
PEN	NO	ISFATTR.NODE.PENCRYPT	$T	*jesx*.MODIFY.NODE	CONTROL
			-	-	-
PGN	DA	ISFATTR.JOB.PGN	RESET	MVS.RESET	UPDATE
PGN	JC	ISFATTR.JOBCL.PGN	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
PGNM	JC	ISFATTR.JOBCL.PGMRNAME	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
PI	RDR	ISFATTR.RDR.PRIOINC	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
PL	RDR	ISFATTR.RDR.PRIOLIM	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
PL	JC	ISFATTR.JOBCL.PROCLIB	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
PMG	NO	ISFATTR.NODE.PATHMGR	$T	*jesx*.MODIFY.NODE	CONTROL
			-	-	-
PORT	JDS	ISFATTR.OUTDESC.PORTNO	SSI		
			SSI		
PORT	NC	ISFATTR.NETOPTS.PORT	$T	*jesx*.MODIFY.SOCKET	CONTROL
			*F	*jesx*.MODIFY.SOCKET	UPDATE
PORT	NS	ISFATTR.NETOPTS.PORT	$T	*jesx*.MODIFY.SOCKET	CONTROL
			*F	*jesx*.MODIFY.NETSERV	UPDATE
PORTNO	OD	ISFATTR.OUTDESC.PORTNO	SSI		
			SSI		
PRMODE	JDS J0	ISFATTR.OUTPUT.PRMODE	-	-	-
			*F U	*jesx*.MODIFY.U	UPDATE
PRMODE	H O	ISFATTR.OUTPUT.PRMODE	$TO	*jesx*.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE
			-	-	-
PRMODE	JDS	ISFATTR.OUTPUT.PRMODE	-	-	-
			*F	*jesx*.MODIFY.U	UPDATE
PROT	SO	ISFATTR.OFFLOAD.PROTECT	$T	*jesx*.MODIFY.OFFLOAD	CONTROL
PRTDEF	NO	ISFATTR.NODE.PRTDEF	-	-	-
			*F	*jesx*.MODIFY.NJE	UPDATE
PRTDEST	I ST	ISFATTR.JOB.PRTDEST	$R	*jesx*.ROUTE.JOBOUT	UPDATE
PRTDEST	RDR	ISFATTR.RDR.PRTDEST	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-

Table 101. Overtypeable Fields (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

When a set of related fields can be overtyped with the Overtype Extension pop-up, all of the fields in the set are protected by the same resource.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an overtypeable field does not apply in a particular JES environment, the command and OPERCMDS resource are shown as a hyphen (-).

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
PRTOPTNS	OD	ISFATTR.OUTDESC.PRINTO	SSI		
			SSI		
PRTQUEUE	OD	ISFATTR.OUTDESC.PRINTQ	SSI		
			SSI		
PRTTSO	NO	ISFATTR.NODE.PRTTSO	-	-	-
			*F	*jesx*.MODIFY.NJE	UPDATE
PRTXWTR	NO	ISFATTR.NODE.PRTXWTR	-	-	-
			*F	*jesx*.MODIFY.NJE	UPDATE
PRTY	I ST	ISFATTR.JOB.PRTY	$T	*jesx*.MODIFY.BAT *jesx*.MODIFY.STC *jesx*.MODIFY.TSU	UPDATE
			*F J,P	*jesx*.MODIFY.JOBP	UPDATE
PRTY	H O	ISFATTR.OUTPUT.PRTY	$TO	*jesx*.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE
			-	-	-
PRV	NO	ISFATTR.NODE.PRIVATE	$T	*jesx*.MODIFY.NODE	CONTROL
PSEL	PR	ISFATTR.PROPTS.PRESELCT	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
PTYPE	NO	ISFATTR.NODE.PTYPE	-	-	-
			*F	*jesx*.MODIFY.NJE	UPDATE
PUNDEF	NO	ISFATTR.NODE.PUNDEF	-	-	-
			*F	*jesx*.MODIFY.NJE	UPDATE
PUNDEST	RDR	ISFATTR.RDR.PUNDEST	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
PWCNTL	NO	SFATTR.NODE.PWCNTL	-	-	-
			*F	*jesx*.MODIFY.NJE	UPDATE
QHLD	JC	ISFATTR.JOBCL.QHELD	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
QUIESCE	DA	ISFATTR.JOB.QUIESCE	RESET	MVS.RESET	UPDATE
RECV	NO	ISFATTR.NODE.RECEIVE	$T	*jesx*.MODIFY.NODE	CONTROL
			-	-	-
REGION	JC	ISFATTR.JOBCL.REGION	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
RES	SP	ISFATTR.SPOOL.RESERVED	$T	*jesx*.MODIFY.SPOOL	CONTROL
			-	-	-
REST	LI	ISFATTR.LINE.REST	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-

Table 101. Overtypeable Fields (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

When a set of related fields can be overtyped with the Overtype Extension pop-up, all of the fields in the set are protected by the same resource.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an overtypeable field does not apply in a particular JES environment, the command and OPERCMDS resource are shown as a hyphen (-).

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
REST	NC	ISFATTR.LINE.REST	$T	*jesx*.MODIFY.APPL *jesx*.MODIFY.SOCKET	CONTROL
			-	-	-
REST	NO	ISFATTR.NODE.REST	$T	*jesx*.MODIFY.NODE	CONTROL
			-	-	-
RESTART	LI	ISFATTR.PROPTS.RESTART	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-
RESTART	NS	ISFATTR.PROPTS.RESTART	$T	*jesx*.MODIFY.LOGON *jesx*.MODIFY.NETSRV	CONTROL
			-	-	-
REST-INT	LI	ISFATTR.PROPTS.RTIME	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-
REST-INT	NS	ISFATTR.PROPTS.RTIME	$T	*jesx*.MODIFY.LOGON *jesx*.MODIFY.NETSRV	CONTROL
			-	-	-
RETAINF	OD	ISFATTR.OUTDESC.RETAINF	SSI		
			SSI		
RETAINS	OD	ISFATTR.OUTDESC.RETAINS	SSI		
			SSI		
RETRYL	OD	ISFATTR.OUTDESC.RETRYL	SSI		
			SSI		
RETRYT	OD	ISFATTR.OUTDESC.RETRYT	SSI		
			SSI		
RST	JC	ISFATTR.JOBCL.RESTART	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
RTPD	SO	ISFATTR.OFFLOAD.RETENT	$T	*jesx*.MODIFY.OFFLOAD	CONTROL
ROOM	JDS OD	ISFATTR.OUTDESC.ROOM	SSI		
			SSI		
SAFF	I ST	ISFATTR.JOB.SYSAFF	$T	*jesx*.MODIFY.BAT *jesx*.MODIFY.STC *jesx*.MODIFY.TSU	UPDATE
SAFF	SP	ISFATTR.SPOOL.SYSAFF	$T	*jesx*.MODIFY.SPOOL	CONTROL
SAFF1	RDR	ISFATTR.RDR.SYSAFF	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
SBURST	PR	ISFATTR.SELECT.BURST	$T	*jesx*.MODIFY.DEV	UPDATE
			*S, *X	See note 3.	
SBURST	SO	ISFATTR.SELECT.BURST	$T	*jesx*.MODIFY.OFF	CONTROL

Table 101. Overtypeable Fields (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

When a set of related fields can be overtyped with the Overtype Extension pop-up, all of the fields in the set are protected by the same resource.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an overtypeable field does not apply in a particular JES environment, the command and OPERCMDS resource are shown as a hyphen (-).

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
SCHEDULING -ENV	JC	ISFATTR.JOBCL.SCHENV	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
SCHEDULING -ENV	I ST	ISFATTR.JOB.SCHENV	$T	*jesx*.MODIFY.BAT	UPDATE
SCLASS	PR PUN	ISFATTR.SELECT.CLASS	$T	*jesx*.MODIFY.DEV	UPDATE
			See note 3.		
SCLASS	SO	ISFATTR.SELECT.CLASS	$T	*jesx*.MODIFY.OFF	CONTROL
SCLASS1-8	SO	ISFATTR.SELECT.CLASS	$T	*jesx*.MODIFY.OFF	CONTROL
SCN	JC	ISFATTR.JOBCL.SCAN	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
SDEPTH	JC	ISFATTR.JOBCL.SDEPTH	-	-	-
			*F	*jesx*.MODIFY.C	UPDATE
SDEST1	PR	ISFATTR.SELECT.DEST	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
SDEST1	PUN	ISFATTR.SELECT.DEST	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
SDEST1	SO	ISFATTR.SELECT.DEST	$T	*jesx*.MODIFY.OFF	CONTROL
SDISP	SO	ISFATTR.SELECT.DISP	$T	*jesx*.MODIFY.OFF	CONTROL
SECURE	NO	ISFATTR.NETOPTS.SECURE	-	-	-
			*F	*jesx*.MODIFY.NJE	UPDATE
SECURE	NS	ISFATTR.NETOPTS.SECURE	$T	*jesx*.MODIFY.SOCKET	CONTROL
			-	-	-
SECURE	NC	ISFATTR.NETOPTS.SECURE	$T	*jesx*.MODIFY.SOCKET	CONTROL
			-	-	-
SELECT	PR PUN	ISFATTR.PROPTS.SELECT	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
SELECTMODE NAME	JP	ISFATTR.MEMBER.SELMNAME	-	-	-
			*F	*jesx*.MODIFY.G	UPDATE
SENDP	NO	ISFATTR.NODE.SENDP	$T	*jesx*.MODIFY.NODE	CONTROL
			-	-	-
SENTRS	NO	ISFATTR.NODE.SENTREST	$T	*jesx*.MODIFY.NODE	CONTROL
			-	-	-
SEP	PR	ISFATTR.PROPTS.SEP	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
SEP	PUN	ISFATTR.PROPTS.SEP	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-

Table 101. Overtypeable Fields (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

When a set of related fields can be overtyped with the Overtype Extension pop-up, all of the fields in the set are protected by the same resource.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an overtypeable field does not apply in a particular JES environment, the command and OPERCMDS resource are shown as a hyphen (-).

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
SEPCHAR	PR	ISFATTR.PROPTS.SEPCHARS	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
SEPDS	PR PUN RDR	ISFATTR.PROPTS.SEPDS	$T	*jesx*.MODIFY.DEV	UPDATE
			See note 3.		
SETUP	PR PUN	ISFATTR.PROPTS.SETUP	$T	*jesx*.MODIFY.DEV	UPDATE
			*F	*jesx*.MODIFY.W	
SETUP	PUN	ISFATTR.PROPTS.SETUP	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
SEVERITY	CK	ISFATTR.CHECK.SEVERITY	F	MVS.MODIFY.STC. *hcproc.hcstcid*	UPDATE
SFCB	PR	ISFATTR.SELECT.FCB	$T	*jesx*.MODIFY.DEV	UPDATE
			See note 3.		
SFCB	SO	ISFATTR.SELECT.FCB	$T	*jesx*.MODIFY.OFF	CONTROL
SFLH	SO	ISFATTR.SELECT.FLASH	$T	*jesx*.MODIFY.OFF	CONTROL
SFLH	PR	ISFATTR.SELECT.FLASH	$T	*jesx*.MODIFY.DEV	UPDATE
			*R, *S	See note 3.	
SFORMS	PR PUN	ISFATTR.SELECT.FORMS	$T	*jesx*.MODIFY.DEV	UPDATE
			See note 3.		
SFORMS	SO	ISFATTR.SELECT.FORMS	$T	*jesx*.MODIFY.OFF	CONTROL
SHOLD	SO	ISFATTR.SELECT.HOLD	$T	*jesx*.MODIFY.OFF	CONTROL
SJOBNAME	PR PUN	ISFATTR.SELECT.JOBNAME	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
SJOBNAME	SO	ISFATTR.SELECT.JOBNAME	$T	*jesx*.MODIFY.OFF	CONTROL
SOCKET	NS	ISFATTR.NETOPTS.SOCKET	$T	*jesx*.MODIFY.NETSRV	CONTROL
			*F	*jesx*.MODIFY.NETSERV	UPDATE
SODSP	LI	ISFATTR.SELECT.OUTDISP	$T	*jesx*.MODIFY.L	CONTROL
			-	-	-
SODSP	NC	ISFATTR.SELECT.ODISP	$T	*jesx*.MODIFY.L	CONTROL
			-	-	-
SODSP	SO	ISFATTR.SELECT.ODISP	$T	*jesx*.MODIFY.OFF	CONTROL
SOWNER	PR	ISFATTR.SELECT.OWNER	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
SOWNER	PUN	ISFATTR.SELECT.OWNER	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
SOWNER	SO	ISFATTR.SELECT.OWNER	$T	*jesx*.MODIFY.OFF	CONTROL

Table 101. Overtypeable Fields (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

When a set of related fields can be overtyped with the Overtype Extension pop-up, all of the fields in the set are protected by the same resource.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an overtypeable field does not apply in a particular JES environment, the command and OPERCMDS resource are shown as a hyphen (-).

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
SPEED	LI	ISFATTR.LINE.SPEED	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-
SPRMODE1	SO	ISFATTR.SELECT.PRMODE	$T	*jesx*.MODIFY.OFF	CONTROL
SPRMODE1	PR PUN RDR	ISFATTR.SELECT.PRMODE	$T	*jesx*.MODIFY.DEV	UPDATE
			See note 3.		
SRANGE	PR	ISFATTR.SELECT.RANGE	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
SRANGE	PUN	ISFATTR.SELECT.RANGE	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
SRANGE	SO	ISFATTR.SELECT.RANGE	$T	*jesx*.MODIFY.OFF	CONTROL
SRNUM	LI	ISFATTR.LINE.SRNUM	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-
SRNUM	NO	ISFATTR.NODE.SRNUM	-	-	-
			*F	*jesx*.MODIFY.NJE	UPDATE
SRVCLASS	DA	ISFATTR.JOB.SRVCLASS	RESET	MVS.RESET	UPDATE
SRVCLASS	I ST	ISFATTR.JOB.SRVCLS	$T	*jesx*.MODIFY.BAT *jesx*.MODIFY.STC *jesx*.MODIFY.TSU	CONTROL
			*F J	*jesx*.MODIFY.JOB	UPDATE
SRVCLASS	ENC	ISFATTR.ENCLAVE.SRVCLASS			
SRVNAME	NC	ISFATTR.NETOPTS.NETSRV	-	-	-
			*F	*jesx*.MODIFY.SOCKET	UPDATE
SSAFF	SO	ISFATTR.SELECT.SYSAFF	$T	*jesx*.MODIFY.OFF	CONTROL
SSCHEDULING-ENV	SO	ISFATTR.SELECT.SCHENV	$T	*jesx*.MODIFY.OFF	CONTROL
SSRVCLASS	SO	ISFATTR.SELECT.SRVCLS	$T	*jesx*.MODIFY.OFF	CONTROL
SSIGNON	NO	ISFATTR.NODE.SSIGNON	$T	*jesx*.MODIFY.NODE	CONTROL
			*F	*jesx*.MODIFY.NJE	UPDATE
STACK	NS	ISFATTR.NETOPTS.STACK	$T	*jesx*.MODIFY.NETSRV	CONTROL
			*F	*jesx*.MODIFY.NETSERV	
STNUM	LI	ISFATTR.LINE.STNUM	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-
SUBNET	NO	ISFATTR.NODE.SUBNET	$T	*jesx*.MODIFY.NODE	CONTROL
			-	-	-

Table 101. Overtypeable Fields (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

When a set of related fields can be overtyped with the Overtype Extension pop-up, all of the fields in the set are protected by the same resource.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an overtypeable field does not apply in a particular JES environment, the command and OPERCMDS resource are shown as a hyphen (-).

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
SUCS	PR	ISFATTR.SELECT.UCS	$T	*jesx*.MODIFY.DEV	UPDATE
			See note 3.		
SUCS	SO	ISFATTR.SELECT.UCS	$T	*jesx*.MODIFY.OFF	CONTROL
SUS	PUN	ISFATTR.PROPTS.SUSPEND	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
SVOL	SO	ISFATTR.SELECT.VOL	$T	*jesx*.MODIFY.OFF	CONTROL
SVOL	PR PUN	ISFATTR.SELECT.VOL	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
SWA	JC	ISFATTR.JOBCL.SWA	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
SWRITER	PR PUN	ISFATTR.SELECT.WRITER	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
SWRITER	SO	ISFATTR.SELECT.WRITER	$T	*jesx*.MODIFY.OFF	CONTROL
SYNCTOL	MAS	ISFATTR.MEMBER.SYNCTOL	$T	*jesx*.MODIFY.MASDEF	CONTROL
SYSSYM	JC	ISFATTR.JOBCL.SYSSYM	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
			-	-	-
TDEPTH	JC	ISFATTR.JOBCL.TDEPTH	-	-	-
			*F	*jesx*.MODIFY.C	UPDATE
TITLE	JDS OD	ISFATTR.OUTDESC.TITLE	SSI		
			SSI		
TP6	JC	ISFATTR.JOBCL.TYPE6	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
TP26	JC	ISFATTR.JOBCL.TYPE26	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
TR	LI NC	ISFATTR.PROPTS.TRACE	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-
TR	NO	ISFATTR.NODE.TRACE	$T	*jesx*.MODIFY.NODE	CONTROL
			-	-	-
TR	NS	ISFATTR.PROPTS.TRACE	$T	*jesx*.MODIFY.LOGON *jesx*.MODIFY.NETSRV	CONTROL
			-	-	-
TR	PR PUN	ISFATTR.PROPTS.TRACE	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
TR	RDR	ISFATTR.RDR.TRACE	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
TRANS	PR	ISFATTR.PROPTS.TRANS	$T	*jesx*.MODIFY.DEV	UPDATE
			*F	*jesx*.MODIFY.F	

Table 101. Overtypeable Fields (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

When a set of related fields can be overtyped with the Overtype Extension pop-up, all of the fields in the set are protected by the same resource.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an overtypeable field does not apply in a particular JES environment, the command and OPERCMDS resource are shown as a hyphen (-).

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
TRANS	NO	ISFATTR.NODE.TRANSMIT	$T	*jesx*.MODIFY.NODE	CONTROL
			-	-	-
TRANSP	LI	ISFATTR.LINE. TRANSPARENCY	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-
TRKCELL	PR	ISFATTR.PROPTS.TRKCELL	PR	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
UCS	H O	ISFATTR.OUTPUT.UCS	$TO	*jesx*.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE
			-	-	-
UCS	JDS J0	ISFATTR.OUTPUT.UCS	-	-	-
			*F	*jesx*.MODIFY.U	UDPATE
UCSV	PR	ISFATTR.PROPTS.UCSVERFY	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
UJP	JC	ISFATTR.JOBCL.IEFUJP	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
UNALLOC	INIT	ISFATTR.INIT.UNALLOC	-	-	-
			*F	*jesx*.MODIFY.G	UPDATE
UNIT	LI	ISFATTR.PROPTS.UNIT	$T	*jesx*.MODIFY.LINE	UPDATE
			-	-	-
UNIT	PR PUN	ISFATTR.PROPTS.UNIT	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
UNIT	SO	ISFATTR.PROPTS.UNIT	$T	*jesx*.MODIFY.OFFLOAD	CONTROL
UNIT	RDR	ISFATTR.RDR.UNIT	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-
USERDATA	OD	ISFATTR.OUTDESC.USERDATA	SSI		
			SSI		
USERDATA1	JDS	ISFATTR.OUTDESC.USERDATA	SSI		
			SSI		
USERDATE	CK	ISFATTR.CHECK.USERDATE	F	MVS.MODIFY.STC. *hcproc.hcstcid*	UPDATE
USERLIB	JDS	ISFATTR.OUTDESC.USERLIB	SSI		
			SSI		
USO	JC	ISFATTR.JOBCL.IEFUSO	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
VALIDATE	SO	ISFATTR.OFFLOAD.VALIDATE	$T	*jesx*.MODIFY.OFFLOAD	CONTROL
VERBOSE	CK	ISFATTR.CHECK.VERBOSE	F	MVS.MODIFY.STC. *hcproc.hcstcid*	UPDATE

Table 101. Overtypeable Fields (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

When a set of related fields can be overtyped with the Overtype Extension pop-up, all of the fields in the set are protected by the same resource.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an overtypeable field does not apply in a particular JES environment, the command and OPERCMDS resource are shown as a hyphen (-).

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
VERIFYP	NO	ISFATTR.NODE.VERIFYP	$T	*jesx*.MODIFY.NODE	CONTROL
			-	-	-
VOLS	SO	ISFATTR.OFFLOAD.VOLS	$T	*jesx*.MODIFY.OFFLOAD	CONTROL
VTR	LI	ISFATTR.PROPTS.VTRACE	$T	*jesx*.MODIFY.LINE	CONTROL
			-	-	-
VTR	NC	ISFATTR.PROPTS.VTRACE	$T	*jesx*.MODIFY.LINE	CONTROL
			*F	*jesx*.MODIFY.SOCKET	UPDATE
VTR	NS	ISFATTR.PROPTS.VTRACE	$T	*jesx*.MODIFY.NETSRV	CONTROL
			*F	*jesx*.MODIFY.NETSERV	UPDATE
WARN%	RM	ISFATTR.RESMON.WARNPCT	$T	*jesx*.MODIFY.*resource*[2]	CONTROL
WORK-SELECTION	LI NC	ISFATTR.PROPTS.WS	$T	*jesx*.MODIFY.L	CONTROL
			-	-	-
WORK-SELECTION	PR	ISFATTR.PROPTS.WS	$T	*jesx*.MODIFY.DEV	UPDATE
			*R	*jesx*.RESTART.DEV.*device*	
WORK-SELECTION	PUN	ISFATTR.PROPTS.WS	$T	*jesx*.MODIFY.DEV	UPDATE
			See note 3.		
WORK-SELECTION	SO	ISFATTR.PROPTS.WS	$T	*jesx*.MODIFY.OFF	CONTROL
WTOTYPE	CK	ISFATTR.CHECK.WTOTYPE	F	MVS.MODIFY.STC.*hcproc.hcstcid*	UPDATE
WTR	H O	ISFATTR.OUTPUT.WRITER	$TO	*jesx*.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE
			-	-	-
WTR	JDS J0	ISFATTR.OUTPUT.WRITER	SSI		
			SSI		
XBM	JC	ISFATTR.JOBCL.XBM	$T	*jesx*.MODIFY.JOBCLASS	CONTROL
XEQDEST	RDR	ISFATTR.RDR.XEQDEST	$T	*jesx*.MODIFY.DEV	UPDATE
			-	-	-

Table 101. Overtypeable Fields (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

When a set of related fields can be overtyped with the Overtype Extension pop-up, all of the fields in the set are protected by the same resource.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

When an overtypeable field does not apply in a particular JES environment, the command and OPERCMDS resource are shown as a hyphen (-).

Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)	Command, JES2 Command, JES3	OPERCMDS Resource, JES2 OPERCMDS Resource, JES3	Required Access
Notes: [1] SDSF uses the subsystem interface (SSI) when you overtype the C (JES output class) or DEST (JES print destination name) on the JDS panel. You can change the class or destination without releasing the output. In order to release output when the JESSPOOL class is enabled, the user must have ALTER authority to the JESSPOOL resource. This authority is implied for the JESSPOOL resources created by the user. [2] The SAF resource varies with the JES2 resource. Refer to "JES2 resources" on page 293. [3] In a JES3 environment, you must also type an action character when overtyping the field. The command issued and OPERCMDS resource depend on the action character that is used with the overtype. Refer to Table 102.					

Table 102. Actions with Overtypes on the PR and PUN Panels in a JES3 Environment

Action Character	Command	OPERCMDS Resource	Required Access
B, E, F	*RESTART	*jesx*.RESTART.DEV.*device*	UPDATE
S	*START	*jesx*.START.DEV.*device*	UPDATE
X	*CALL	*jesx*.CALL.*dspname*	UPDATE

Table 103. Overtypeable Fields Sorted by OPERCMDS Resource Name.

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the MVS/JES Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	MVS/JES Command	Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)
		SSI	ADDRESS	JDS OD	ISFATTR.OUTDESC.ADDRESS
		SSI	AFPPARMS	JDS OD	ISFATTR.OUTDESC.AFPPARMS
		SSI	BUILDING	JDS OD	ISFATTR.OUTDESC.BLDG
		SSI[1]	C	JDS J0	ISFATTR.OUTPUT.CLASS
		SSI	CC	JDS J0	ISFATTR.OUTPUT.COPYCNT
		SSI	COLORMAP	JDS OD	ISFATTR.OUTDESC. COLORMAP
		SSI	COMSETUP	JDS OD	ISFATTR.OUTDESC. COMSETUP
		SSI	DEPARTMENT	JDS OD	ISFATTR.OUTDESC.DEPT
		SSI[1]	DEST	JDS J0	ISFATTR.OUTPUT.DEST

Table 103. Overtypeable Fields Sorted by OPERCMDS Resource Name (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the MVS/JES Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	MVS/JES Command	Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)
		SSI	FORMDEF	JDS OD	ISFATTR.OUTDESC.FORMDEF
		SSI	FORMLEN	JDS OD	ISFATTR.OUTDESC.FORMLEN
		SSI	FORMS	JDS J0	ISFATTR.OUTPUT.FORMS
		SSI	INTRAY	JDS OD	ISFATTR.OUTDESC.INTRAY
		SSI	IP DESTINATION	OD	ISFATTR.OUTDESC.IPDEST
		SSI	NAME	JDS OD	ISFATTR.OUTDESC.NAME
		SSI	OCOPYCNT	JDS OD	ISFATTR.OUTDESC.OCOPYCNT
		SSI	OFFSETXB	JDS OD	ISFATTR.OUTDESC. OFFSETXB
		SSI	OFFSETXF	JDS OD	ISFATTR.OUTDESC. OFFSETXF
		SSI	OFFSETYB	JDS OD	ISFATTR.OUTDESC. OFFSETYB
		SSI	OFFSETYF	JDS OD	ISFATTR.OUTDESC. OFFSETYF
		SSI	NOTIFY	JDS OD	ISFATTR.OUTDESC.NOTIFY
		SSI	OUTBN	JDS	ISFATTR.OUTDESC.OUTBIN
		SSI	OUTBIN	OD	ISFATTR.OUTDESC.OUTBIN
		SSI	OVERLAYB	JDS OD	ISFATTR.OUTDESC. OVERLAYB
		SSI	OVERLAYF	JDS OD	ISFATTR.OUTDESC. OVERLAYF
		SSI	PAGEDEF	JDS OD	ISFATTR.OUTDESC.PAGEDEF
		SSI	PORT	JDS	ISFATTR.OUTDESC.PORTNO
		SSI	PORTNO	OD	ISFATTR.OUTDESC.PORTNO
		SSI	PRMODE	JDS J0	ISFATTR.OUTPUT.PRMODE
		SSI	PRTOPTNS	OD	ISFATTR.OUTDESC.PRINTO
		SSI	PRTQUEUE	OD	ISFATTR.OUTDESC.PRINTQ
		SSI	RETAINF	OD	ISFATTR.OUTDESC.RETAINF
		SSI	RETAINS	OD	ISFATTR.OUTDESC.RETAINS
		SSI	RETRYL	OD	ISFATTR.OUTDESC.RETRYL
		SSI	RETRYT	OD	ISFATTR.OUTDESC.RETRYT
		SSI	ROOM	JDS OD	ISFATTR.OUTDESC.ROOM
		SSI	TITLE	JDS OD	ISFATTR.OUTDESC.TITLE
		SSI	UCS	JDS J0	ISFATTR.OUTPUT.UCS
		SSI	USERDATA	OD	ISFATTR.OUTDESC.USERDATA
		SSI	USERDATA1	JDS	ISFATTR.OUTDESC.USERDATA
		SSI	USERLIB	JDS OD	ISFATTR.OUTDESC.USERLIB
			SRVCLASS	ENC	ISFATTR.ENCLAVE.SRVCLASS
		SSI	WTR	JDS J0	ISFATTR.OUTPUT.WRITER

Table 103. Overtypeable Fields Sorted by OPERCMDS Resource Name (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the MVS/JES Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	MVS/JES Command	Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)
jesx.CALL.*dspname*	UPDATE	*X. See note 3.	B	PUN	ISFATTR.PROPTS.BPAGE
jesx.CALL.*dspname*	UPDATE	*X. See note 3.	CB	PR	ISFATTR.PROPTS.CB
jesx.CALL.*dspname*	UPDATE	*X. See note 3.	CHAR1	PR	ISFATTR.PROPTS.CHAR
jesx.CALL.*dspname*	UPDATE	*X. See note 3.	CKPTPAGE	PR	ISFATTR.PROPTS.CKPTPAGE
jesx.CALL.*dspname*	UPDATE	*X. See note 3.	CKPTSEC	PR	ISFATTR.PROPTS.CKPTSEC
jesx.CALL.*dspname*	UPDATE	*X. See note 3.	COPIES	PR	ISFATTR.PROPTS.COPIES
jesx.CALL.*dspname*	UPDATE	*X. See note 3.	COPYMARK	PR	ISFATTR.PROPTS.COPYMARK
jesx.CALL.*dspname*	UPDATE	*X. See note 3.	LINE-LIM-HI	PR PUN	ISFATTR.SELECT.LIM
jesx.CALL.*dspname*	UPDATE	*X. See note 3.	LINE-LIM-LO	PR PUN	ISFATTR.SELECT.LIM
jesx.CALL.*dspname*	UPDATE	*X. See note 3.	NPRO	PR	ISFATTR.PROPTS.NPRO
jesx.CALL.*dspname*	UPDATE	*X. See note 3.	PAGE-LIM-HI	PR	ISFATTR.SELECT.PLIM
jesx.CALL.*dspname*	UPDATE	*X. See note 3.	PAGE-LIM-LO	PR	ISFATTR.SELECT.PLIM
jesx.CALL.*dspname*	UPDATE	*X. See note 3.	SBURST	PR	ISFATTR.SELECT.BURST
jesx.CALL.*dspname*	UPDATE	*X. See note 3.	SCLASS	PR PUN	ISFATTR.SELECT.CLASS
jesx.CALL.*dspname*	UPDATE	*X. See note 3.	SEPDS	PUN	ISFATTR.PROPTS.SEPDS
jesx.CALL.*dspname*	UPDATE	*X. See note 3.	SFCB	PR	ISFATTR.SELECT.FCB
jesx.CALL.*dspname*	UPDATE	*X. See note 3.	SFORMS	PR PUN	ISFATTR.SELECT.FORMS
jesx.CALL.*dspname*	UPDATE	*X. See note 3.	SPRMODE1	PR PUN	ISFATTR.SELECT.PRMODE
jesx.CALL.*dspname*	UPDATE	*X. See note 3.	SUCS	PR	ISFATTR.SELECT.UCS
jesx.CALL.*dspname*	UPDATE	*X. See note 3.	WORK-SELECTION	PUN	ISFATTR.PROPTS.WS
jesx.CALL.NJE	UPDATE	*X	NODE	LI NO	ISFATTR.LINE.NODE
jesx.MODIFY.*resource* [2]	CONTROL	$T	LIMIT	RM	ISFATTR.RESMON.LIMIT
jesx.MODIFY.*resource* [2]	CONTROL	$T	WARN%	RM	ISFATTR.RESMON.WARNPCT
jesx.MODIFY.APPL	CONTROL	$T	ANODE	NC	ISFATTR.NETOPTS.NODE
jesx.MODIFY.APPL	CONTROL	$T	COMPACT	NC	ISFATTR.NODE.COMPACT

Table 103. Overtypeable Fields Sorted by OPERCMDS Resource Name (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the MVS/JES Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	MVS/JES Command	Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)
jesx.MODIFY.APPL	CONTROL	$T	CONNECT	NC	ISFATTR.NETOPTS.CONNECT
jesx.MODIFY.APPL	CONTROL	$T	CONN-INT	NC	ISFATTR.NETOPTS.CTIME
jesx.MODIFY.APPL	CONTROL	$T	LINE	NC	ISFATTR.NODE.LINE
jesx.MODIFY.APPL	CONTROL	$T	LOGMODE	NC	ISFATTR.NODE.LOGMODE
jesx.MODIFY.APPL	CONTROL	$T	LOGON	NC	ISFATTR.NETOPTS.LOGON
jesx.MODIFY.APPL	CONTROL	$T	REST	NC	ISFATTR.LINE.REST
jesx.MODIFY.BAT	UPDATE	$T	SCHEDULING-ENV	I ST	ISFATTR.JOB.SCHENV
jesx.MODIFY.BAT *jesx*.MODIFY.STC *jesx*.MODIFY.TSU	UPDATE	$T	SAFF	I ST	ISFATTR.JOB.SYSAFF
jesx.MODIFY.BAT *jesx*.MODIFY.STC *jesx*.MODIFY.TSU	UPDATE	$T	C	I ST	ISFATTR.JOB.CLASS
jesx.MODIFY.BAT *jesx*.MODIFY.STC *jesx*.MODIFY.TSU	UPDATE	$T	PRTY	I ST	ISFATTR.JOB.PRTY
jesx.MODIFY.BAT *jesx*.MODIFY.STC *jesx*.MODIFY.TSU	CONTROL	$T	SRVCLASS	I ST	ISFATTR.JOB.SRVCLS
jesx.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE	$TO	BURST	H O	ISFATTR.OUTPUT.BURST
jesx.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE	$TO SSI[1]	C	H O	ISFATTR.OUTPUT.CLASS
jesx.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE	$TO SSI[1]	DEST	H O	ISFATTR.OUTPUT.DEST
jesx.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE	$TO	FCB	H O	ISFATTR.OUTPUT.FCB
jesx.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE	$TO	FLASH	H O	ISFATTR.OUTPUT.FLASH
jesx.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE	$TO	FORMS	H O	ISFATTR.OUTPUT.FORMS
jesx.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE	$TO	ODISP	H O	ISFATTR.OUTPUT.ODISP
jesx.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE	$TO	PRMODE	H O	ISFATTR.OUTPUT.PRMODE
jesx.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE	$TO	PRTY	H O	ISFATTR.OUTPUT.PRTY

Table 103. Overtypeable Fields Sorted by OPERCMDS Resource Name (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the MVS/JES Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	MVS/JES Command	Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)
jesx.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE	$TO	UCS	H O	ISFATTR.OUTPUT.UCS
jesx.MODIFY.BATOUT *jesx*.MODIFY.STCOUT *jesx*.MODIFY.TSUOUT	UPDATE	$TO	WTR	H O	ISFATTR.OUTPUT.WRITER
jesx.MODIFY.C	UPDATE	*F	JESLOG	JC	ISFATTR.JOBCL.JESLOG
jesx.MODIFY.C	UPDATE	*F	LOG	JC	ISFATTR.JOBCL.JLOG
jesx.MODIFY.C	UPDATE	*F	PARTNAME	JC	ISFATTR.JOBCL.PARTNAME
jesx.MODIFY.C	UPDATE	*F	SDEPTH	JC	ISFATTR.JOBCL.SDEPTH
jesx.MODIFY.C	UPDATE	*F	TDEPTH	JC	ISFATTR.JOBCL.TDEPTH
jesx.MODIFY.DEV	UPDATE	$T	ASIS	PR	ISFATTR.PROPTS.ASIS
jesx.MODIFY.DEV	UPDATE	$T	CCTL	PR PUN	ISFATTR.PROPTS.CCTL
jesx.MODIFY.DEV	UPDATE	$T	CHAR1–4	PR	ISFATTR.PROPTS.CHAR
jesx.MODIFY.DEV	UPDATE	$T	CMPCT	PR PUN	ISFATTR.PROPTS.CMPCT
jesx.MODIFY.DEV	UPDATE	$T	COMP	PR PUN	ISFATTR.PROPTS.COMPRESS
jesx.MODIFY.DEV	UPDATE	$T	COMPACT	PR PUN	ISFATTR.PROPTS.COMPACT
jesx.MODIFY.DEV	UPDATE	$T	CKPTLINE	PR PUN	ISFATTR.PROPTS.CKPTLINE
jesx.MODIFY.DEV	UPDATE	$T	CKPTMODE	PR	ISFATTR.PROPTS.CKPTMODE
jesx.MODIFY.DEV	UPDATE	$T	CKPTPAGE	PR PUN	ISFATTR.PROPTS.CKPTPAGE
jesx.MODIFY.DEV	UPDATE	$T	CKPTSEC	PR	ISFATTR.PROPTS.CKPTSEC
jesx.MODIFY.DEV	UPDATE	$T	COPYMARK	PR	ISFATTR.PROPTS.COPYMARK
jesx.MODIFY.DEV	UPDATE	$T	CPYMOD	PR	ISFATTR.PROPTS.COPYMOD
jesx.MODIFY.DEV	UPDATE	$T	DFCB	PR	ISFATTR.PROPTS.DEVFCB
jesx.MODIFY.DEV	UPDATE	$T	FCBL	PR	ISFATTR.PROPTS.FCBLOAD
jesx.MODIFY.DEV	UPDATE	$T	FSSNAME	PR	ISFATTR.PROPTS.FSSNAME
jesx.MODIFY.DEV	UPDATE	$T	HONORTRC	PR	ISFATTR.PROPTS.HONORTRC
jesx.MODIFY.DEV	UPDATE	$T	K	PR	ISFATTR.PROPTS.SPACE
jesx.MODIFY.DEV	UPDATE	$T	LINE-LIMIT	PR PUN	ISFATTR.SELECT.LIM
jesx.MODIFY.DEV	UPDATE	$T	LRECL	PR PUN	ISFATTR.PROPTS.LRECL
jesx.MODIFY.DEV	UPDATE	$T	M	PR	ISFATTR.PROPTS.MARK
jesx.MODIFY.DEV	UPDATE	$T	MODE	PR	ISFATTR.PROPTS.MODE
jesx.MODIFY.DEV	UPDATE	$T	NEWPAGE	PR	ISFATTR.PROPTS.NEWPAGE
jesx.MODIFY.DEV	UPDATE	$T	NPRO	PR	ISFATTR.PROPTS.NPRO
jesx.MODIFY.DEV	UPDATE	$T	PAGE-LIMIT	PR	ISFATTR.SELECT.PLIM
jesx.MODIFY.DEV	UPDATE	$T	PAU	PR PUN	ISFATTR.PROPTS.PAUSE
jesx.MODIFY.DEV	UPDATE	$T	PSEL	PR	ISFATTR.PROPTS.PRESELCT
jesx.MODIFY.DEV	UPDATE	$T	SBURST	PR SO	ISFATTR.SELECT.BURST
jesx.MODIFY.DEV	UPDATE	$T	SCLASS	PR PUN	ISFATTR.SELECT.CLASS
jesx.MODIFY.DEV	UPDATE	$T	SDEST1	PR PUN	ISFATTR.SELECT.DEST

Table 103. Overtypeable Fields Sorted by OPERCMDS Resource Name (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the MVS/JES Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	MVS/JES Command	Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)
jesx.MODIFY.DEV	UPDATE	$T	SELECT	PR PUN	ISFATTR.PROPTS.SELECT
jesx.MODIFY.DEV	UPDATE	$T	SEP	PR PUN	ISFATTR.PROPTS.SEP
jesx.MODIFY.DEV	UPDATE	$T	SEPCHAR	PR	ISFATTR.PROPTS.SEPCHARS
jesx.MODIFY.DEV	UPDATE	$T	SEPDS	PR PUN	ISFATTR.PROPTS.SEPDS
jesx.MODIFY.DEV	UPDATE	$T	SETUP	PR PUN	ISFATTR.PROPTS.SETUP
jesx.MODIFY.DEV	UPDATE	$T	SFCB	PR	ISFATTR.SELECT.FCB
jesx.MODIFY.DEV	UPDATE	$T	SFLH	PR	ISFATTR.SELECT.FLASH
jesx.MODIFY.DEV	UPDATE	$T	SFORMS	PR PUN	ISFATTR.SELECT.FORMS
jesx.MODIFY.DEV	UPDATE	$T	SJOBNAME	PR PUN	ISFATTR.SELECT.JOBNAME
jesx.MODIFY.DEV	UPDATE	$T	SOWNER	PR PUN	ISFATTR.SELECT.OWNER
jesx.MODIFY.DEV	UPDATE	$T	SPRMODE1	PR PUN	ISFATTR.SELECT.PRMODE
jesx.MODIFY.DEV	UPDATE	$T	SRANGE	PR PUN	ISFATTR.SELECT.RANGE
jesx.MODIFY.DEV	UPDATE	$T	SUCS	PR	ISFATTR.SELECT.UCS
jesx.MODIFY.DEV	UPDATE	$T	SUS	PR PUN	ISFATTR.SELECT.SUSPEND
jesx.MODIFY.DEV	UPDATE	$T	SVOL1	PR	ISFATTR.SELECT.VOL
jesx.MODIFY.DEV	UPDATE	$T	SWRITER	PR PUN	ISFATTR.SELECT.WRITER
jesx.MODIFY.DEV	UPDATE	$T	TR	PR PUN	ISFATTR.PROPTS.TRACE
jesx.MODIFY.DEV	UPDATE	$T	TRANS	PR	ISFATTR.PROPTS.TRANS
jesx.MODIFY.DEV	UPDATE	$T	TRKCELL	PR	ISFATTR.PROPTS.TRKCELL
jesx.MODIFY.DEV	UPDATE	$T	UCSV	PR	ISFATTR.PROPTS.UCSVERFY
jesx.MODIFY.DEV	UPDATE	$T	UNIT	PR PUN	ISFATTR.PROPTS.UNIT
jesx.MODIFY.DEV	UPDATE	$T	WORK-SELECTION	PR PUN	ISFATTR.PROPTS.WS
jesx.MODIFY.DEV	UPDATE	$T	FLS	PUN	ISFATTR.PROPTS.FLUSH
jesx.MODIFY.DEV	UPDATE	$T	LINE-LIMIT	PUN	ISFATTR.SELECT.LIM
jesx.MODIFY.DEV	UPDATE	$T	SVOL	PUN	ISFATTR.SELECT.VOL
jesx.MODIFY.DEV	UPDATE	$T	AUTHORITY	RDR	ISFATTR.RDR.AUTHORITY
jesx.MODIFY.DEV	UPDATE	$T	C	RDR	ISFATTR.RDR.CLASS
jesx.MODIFY.DEV	UPDATE	$T	HOLD	RDR	ISFATTR.RDR.HOLD
jesx.MODIFY.DEV	UPDATE	$T	MC	RDR	ISFATTR.RDR.RMCLASS
jesx.MODIFY.DEV	UPDATE	$T	PI	RDR	ISFATTR.RDR.PRIOINC
jesx.MODIFY.DEV	UPDATE	$T	PL	RDR	ISFATTR.RDR.PRIOLIM
jesx.MODIFY.DEV	UPDATE	$T	PRTDEST	RDR	ISFATTR.RDR.PRTDEST
jesx.MODIFY.DEV	UPDATE	$T	PUNDEST	RDR	ISFATTR.RDR.PUNDEST
jesx.MODIFY.DEV	UPDATE	$T	SAFF	RDR	ISFATTR.RDR.SYSAFF
jesx.MODIFY.DEV	UPDATE	$T	TR	RDR	ISFATTR.RDR.TRACE
jesx.MODIFY.DEV	UPDATE	$T	UNIT	RDR	ISFATTR.RDR.UNIT
jesx.MODIFY.DEV	UPDATE	$T	XEQDEST	RDR	ISFATTR.RDR.XEQDEST
jesx.MODIFY.F	UPDATE	*F	MODE	PR	ISFATTR.PROPTS.MODE

Table 103. Overtypeable Fields Sorted by OPERCMDS Resource Name (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the MVS/JES Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	MVS/JES Command	Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)
jesx.MODIFY.F	CONTROL	*F	PDEFAULT	PR	ISFATTR.PROPTS.PDEFAULT
jesx.MODIFY.F	UPDATE	*F	SETUP	PR	ISFATTR.PROPTS.SETUP
jesx.MODIFY.F	UPDATE	*F	TRANS	PR	ISFATTR.PROPTS.TRANS
jesx.MODIFY.G	UPDATE	*F	ALLOC	INIT	ISFATTR.INIT.ALLOC
jesx.MODIFY.G	UPDATE	*F	BARRIER	INIT	ISFATTR.INIT.BARRIER
jesx.MODIFY.G	UPDATE	*F	DEFCOUNT	INIT	ISFATTR.INIT.DEFCOUNT
jesx.MODIFY.C	UPDATE	*F	GROUP	INIT	ISFATTR.INIT.GROUP
jesx.MODIFY.G	UPDATE	*F	MODE	INIT	ISFATTR.INIT.MODE
jesx.MODIFY.G	UPDATE	*F	UNALLOC	INIT	ISFATTR.INIT.UNALLOC
jesx.MODIFY.G	UPDATE	*F	SELECTMODE NAME	JP	ISFATTR.MEMBER.SELMNAME
jesx.MODIFY.G	UPDATE	*F	PARTNAME	JP	ISFATTR.MEMBER.SPARTN
jesx.MODIFY.INITIATOR	CONTROL	$T	CLASSES	INIT	ISFATTR.SELECT.JOBCLASS
jesx.MODIFY.INITIATOR	CONTROL	$T	CLASS1-8	INIT	ISFATTR.SELECT.JOBCLASS
jesx.MODIFY.JOB	UPDATE	*F	C	I ST	ISFATTR.JOB.CLASS
jesx.MODIFY.JOB	UPDATE	*F	SRVLCASS	I ST	ISFATTR.JOB.SRVCLS
jesx.MODIFY.JOB	UPDATE	*F	C	I ST	ISFATTR.JOB.CLASS
jesx.MODIFY.JOBP	UPDATE	*F	PRTY	I ST	ISFATTR.JOB.PRTY
jesx.MODIFY.JOBCLASS	CONTROL	$T	ACCT	JC	ISFATTR.JOBCL.ACCT
jesx.MODIFY.JOBCLASS	CONTROL	$T	ACTIVE	JC	ISFATTR.JOBCL.ACTIVE
jesx.MODIFY.JOBCLASS	CONTROL	$T	AUTH	JC	ISFATTR.JOBCL.AUTH
jesx.MODIFY.JOBCLASS	CONTROL	$T	BLP	JC	ISFATTR.JOBCL.BLP
jesx.MODIFY.JOBCLASS	CONTROL	$T	COMMAND	JC	ISFATTR.JOBCL.COMMAND
jesx.MODIFY.JOBCLASS	CONTROL	$T	CPR	JC	ISFATTR.JOBCL.CONDPURG
jesx.MODIFY.JOBCLASS	CONTROL	$T	CPY	JC	ISFATTR.JOBCL.COPY
jesx.MODIFY.JOBCLASS	CONTROL	$T	DSENQSHR	JC	ISFATTR.JOBCL.DSENQSHR
jesx.MODIFY.JOBCLASS	CONTROL	$T	GROUP	JC	ISFATTR.JOBCL.GROUP
jesx.MODIFY.JOBCLASS	CONTROL	$T	HOLD	JC	ISFATTR.JOBCL.HOLD
jesx.MODIFY.JOBCLASS	CONTROL	$T	JCLIM	JC	ISFATTR.JOBCL.JCLIM
jesx.MODIFY.JOBCLASS	CONTROL	$T	JESLOG	JC	ISFATTR.JOBCL.JESLOG
jesx.MODIFY.JOBCLASS	CONTROL	$T	JOBRC	JC	ISFATTR.JOBCL.JOBRC
jesx.MODIFY.JOBCLASS	CONTROL	$T	JRNL	JC	ISFATTR.JOBCL.JOURNAL
jesx.MODIFY.JOBCLASS	CONTROL	$T	LOG	JC	ISFATTR.JOBCL.LOG
jesx.MODIFY.JOBCLASS	CONTROL	$T	MAX-TIME	JC	ISFATTR.JOBCL.TIME
jesx.MODIFY.JOBCLASS	CONTROL	$T	MC	JC	ISFATTR.JOBCL.MSGCLASS
jesx.MODIFY.JOBCLASS	CONTROL	$T	MODE	JC	ISFATTR.JOBCL.MODE
jesx.MODIFY.JOBCLASS	CONTROL	$T	MSGLV	JC	ISFATTR.JOBCL.MSGLEVEL
jesx.MODIFY.JOBCLASS	CONTROL	$T	ODISP	JC	ISFATTR.JOBCL.ODISP
jesx.MODIFY.JOBCLASS	CONTROL	$T	OUT	JC	ISFATTR.JOBCL.OUTPUT

Table 103. Overtypeable Fields Sorted by OPERCMDS Resource Name (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the MVS/JES Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	MVS/JES Command	Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)
jesx.MODIFY.JOBCLASS	CONTROL	$T	PGN	JC	ISFATTR.JOBCL.PGN
jesx.MODIFY.JOBCLASS	CONTROL	$T	PGNM	JC	ISFATTR.JOBCL.PGMRNAME
jesx.MODIFY.JOBCLASS	CONTROL	$T	QHLD	JC	ISFATTR.JOBCL.QHELD
jesx.MODIFY.JOBCLASS	CONTROL	$T	REGION	JC	ISFATTR.JOBCL.REGION
jesx.MODIFY.JOBCLASS	CONTROL	$T	RST	JC	ISFATTR.JOBCL.RESTART
jesx.MODIFY.JOBCLASS	CONTROL	$T	SCHEDULING-ENV	JC	ISFATTR.JOBCL.SCHENV
jesx.MODIFY.JOBCLASS	CONTROL	$T	SCN	JC	ISFATTR.JOBCL.SCAN
jesx.MODIFY.JOBCLASS	CONTROL	$T	SWA	JC	ISFATTR.JOBCL.SWA
jesx.MODIFY.JOBCLASS	CONTROL	$T	SYSSYM	JC	ISFATTR.JOBCL.SYSSYM
jesx.MODIFY.JOBCLASS	CONTROL	$T	TP6	JC	ISFATTR.JOBCL.TYPE6
jesx.MODIFY.JOBCLASS	CONTROL	$T	TP26	JC	ISFATTR.JOBCL.TYPE26
jesx.MODIFY.JOBCLASS	CONTROL	$T	UJP	JC	ISFATTR.JOBCL.IEFUJP
jesx.MODIFY.JOBCLASS	CONTROL	$T	USO	JC	ISFATTR.JOBCL.IEFUSO
jesx.MODIFY.JOBCLASS	CONTROL	$T	XBM	JC	ISFATTR.JOBCL.XBM
jesx.MODIFY.L	CONTROL	$T	LINE-LIMIT	LI NC	ISFATTR.SELECT.LIM
jesx.MODIFY.L	CONTROL	$T	PAGE-LIMIT	LI NC	ISFATTR.SELECT.PLIM
jesx.MODIFY.L	CONTROL	$T	SODSP	LI NC	ISFATTR.SELECT.OUTDISP
jesx.MODIFY.L	CONTROL	$T	WORK-SELECTION	LI NC	ISFATTR.PROPTS.WS
jesx.MODIFY.LINE	CONTROL	$T	ADISC	LI	ISFATTR.LINE.AUTODISC
jesx.MODIFY.LINE	CONTROL	$T	ANODE	NC	ISFATTR.NETOPTS.NODE
jesx.MODIFY.LINE	CONTROL	$T	CONNECT	NC	ISFATTR.NETOPTS.CONNECT
jesx.MODIFY.LINE	CONTROL	$T	CONN-INT	NC	ISFATTR.NETOPTS.CTIME
jesx.MODIFY.LINE	CONTROL	$T	CODE	LI	ISFATTR.LINE.CODE
jesx.MODIFY.LINE	CONTROL	$T	COMP	LI	ISFATTR.LINE.COMPRESS
jesx.MODIFY.LINE	CONTROL	$T	CONNECT	LI	ISFATTR.NETOPTS.CONNECT
jesx.MODIFY.LINE	CONTROL	$T	CONN-INT	LI	ISFATTR.NETOPTS.CTIME
jesx.MODIFY.LINE	CONTROL	$T	CTR	LI NC	ISFATTR.PROPTS.CTRACE
jesx.MODIFY.LINE	CONTROL	$T	DUPLEX	LI	ISFATTR.LINE.DUPLEX
jesx.MODIFY.LINE	CONTROL	$T	INTF	LI	ISFATTR.LINE.INTERFACE
jesx.MODIFY.LINE	CONTROL	$T	JRNUM	LI	ISFATTR.LINE.JRNUM
jesx.MODIFY.LINE	CONTROL	$T	JTNUM	LI	ISFATTR.LINE.JTNUM
jesx.MODIFY.LINE	CONTROL	$T	JTR	LI NC	ISFATTR.PROPTS.JTRACE
jesx.MODIFY.LINE	CONTROL	$T	LINECCHR	LI	ISFATTR.LINE.LINECCHR
jesx.MODIFY.LINE	CONTROL	$T	LOG	LI	ISFATTR.LINE.LOG
jesx.MODIFY.LINE	CONTROL	$T	REST	LI	ISFATTR.LINE.REST
jesx.MODIFY.LINE	CONTROL	$T	RESTART	LI	ISFATTR.PROPTS.RESTART
jesx.MODIFY.LINE	CONTROL	$T	REST-INT	LI	ISFATTR.PROPTS.RTIME

Table 103. Overtypeable Fields Sorted by OPERCMDS Resource Name (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the MVS/JES Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	MVS/JES Command	Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)
jesx.MODIFY.LINE	CONTROL	$T	SPEED	LI	ISFATTR.LINE.SPEED
jesx.MODIFY.LINE	CONTROL	$T	SRNUM	LI	ISFATTR.LINE.SRNUM
jesx.MODIFY.LINE	CONTROL	$T	STNUM	LI	ISFATTR.LINE.STNUM
jesx.MODIFY.LINE	CONTROL	$T	TR	LI NC	ISFATTR.PROPTS.TRACE
jesx.MODIFY.LINE	CONTROL	$T	TRANSP	LI	ISFATTR.LINE. TRANSPARENCY
jesx.MODIFY.LINE	CONTROL	$T	UNIT	LI	ISFATTR.PROPTS.UNIT
jesx.MODIFY.LINE	CONTROL	$T	VTR	LI NC	ISFATTR.PROPTS.VTRACE
jesx.MODIFY.LOGON	CONTROL	$T	APPL	NS	ISFATTR.NETOPTS.APPL
jesx.MODIFY.LOGON	CONTROL	$T	LOG	NS	ISFATTR.NETOPTS.LOG
jesx.MODIFY.LOGON	CONTROL	$T	PASSWORD	NS	ISFATTR.LOGON.PASSWORD
jesx.MODIFY.LOGON	CONTROL	$T	RESTART	NS	ISFATTR.PROPTS.RESTART
jesx.MODIFY.LOGON	CONTROL	$T	RESTART-INT	NS	ISFATTR.PROPTS.RTIME
jesx.MODIFY.LOGON	CONTROL	$T	TR	NS	ISFATTR.PROPTS.TRACE
jesx.MODIFY.MASDEF	CONTROL	$T	CKPTHOLD	MAS	ISFATTR.MEMBER.CKPTHOLD
jesx.MODIFY.MASDEF	CONTROL	$T	DORMANCY	MAS	ISFATTR.MEMBER.DORMANCY
jesx.MODIFY.MASDEF	CONTROL	$T	SYNCTOL	MAS	ISFATTR.MEMBER.SYNCTOL
jesx.MODIFY.NETSRV	CONTROL	$T	CTR	NS	ISFATTR.PROPTS.CTRACE
jesx.MODIFY.NETSRV	CONTROL	$T	JTR	NS	ISFATTR.PROPTS.JTRACE
jesx.MODIFY.NETSRV	CONTROL	$T	RESTART	NS	ISFATTR.PROPTS.RESTART
jesx.MODIFY.NETSRV	CONTROL	$T	RESTART-INT	NS	ISFATTR.PROPTS.RTIME
jesx.MODIFY.NETSRV	CONTROL	$T	SOCKET	NS	ISFATTR.NETOPTS.SOCKET
jesx.MODIFY.NETSRV	CONTROL	$T	STACK	NS	ISFATTR.NETOPTS.STACK
jesx.MODIFY.NETSRV	CONTROL	$T	TR	NS	ISFATTR.PROPTS.TRACE
jesx.MODIFY.NETSRV	CONTROL	$T	VTR	NS	ISFATTR.PROPTS.VTRACE
jesx.MODIFY.NETSERV	CONTROL	*F	CTR	NS	ISFATTR.PROPTS.CTRACE
jesx.MODIFY.NETSERV	UPDATE	*F	IPNAME	NS	ISFATTR.NETOPTS.HOSTNAME
jesx.MODIFY.NETSERV	UPDATE	*F	JTR	NS	ISFATTR.PROPTS.JTRACE
jesx.MODIFY.NETSERV	UPDATE	*F	PORT	NS	ISFATTR.NETOPTS.PORT
jesx.MODIFY.NETSERV	UPDATE	*F	SOCKET	NS	ISFATTR.NETOPTS.SOCKET
jesx.MODIFY.NETSERV	UPDATE	*F	STACK	NS	ISFATTR.NETOPTS.STACK
jesx.MODIFY.NETSERV	UPDATE	*F	TR	NS	ISFATTR.PROPTS.VTRACE
jesx.MODIFY.NJE	UPDATE	*F	HOLD	NO	ISFATTR.NODE.HOLD
jesx.MODIFY.NJE	UPDATE	*F	JRNUM	NO	ISFATTR.NODE.JRNUM
jesx.MODIFY.NJE	UPDATE	*F	JTNUM	NO	ISFATTR.NODE.JTNUM
jesx.MODIFY.NJE	UPDATE	*F	NHOLD	NO	ISFATTR.NODE.NETHOLD
jesx.MODIFY.NJE	UPDATE	*F	MAXRETRIES	NO	ISFATTR.NODE.MAXRETR
jesx.MODIFY.NJE	UPDATE	*F	PARTNAME	NO	ISFATTR.NODE.PARTNAM
jesx.MODIFY.NJE	UPDATE	*F	PATH	NO	ISFATTR.NODE.PATH

Table 103. Overtypeable Fields Sorted by OPERCMDS Resource Name (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the MVS/JES Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	MVS/JES Command	Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)
jesx.MODIFY.NJE	UPDATE	*F	PRTDEF	NO	ISFATTR.NODE.PRTDEF
jesx.MODIFY.NJE	UPDATE	*F	PRTTSO	NO	ISFATTR.NODE.PRTTSO
jesx.MODIFY.NJE	UPDATE	*F	PRTXWTR	NO	ISFATTR.NODE.PRTXWTR
jesx.MODIFY.NJE	UPDATE	*F	PTYPE	NO	ISFATTR.NODE.PTYPE
jesx.MODIFY.NJE	UPDATE	*F	PUNDEF	NO	ISFATTR.NODE.PUNDEF
jesx.MODIFY.NJE	UPDATE	*F	PWCNTL	NO	ISFATTR.NODE.PWCNTL
jesx.MODIFY.NJE	UPDATE	*F	SECURE	NO	ISFATTR.NODE.SECURE
jesx.MODIFY.NJE	UPDATE	*F	SRNUM	NO	ISFATTR.NODE.SRNUM
jesx.MODIFY.NJE	UPDATE	*F	SSIGNON	NO	ISFATTR.NODE.SSIGNON
jesx.MODIFY.NJE	UPDATE	*F	STNUM	NO	ISFATTR.NODE.STNUM
jesx.MODIFY.NODE	CONTROL	$T	AUTHORITY	NO	ISFATTR.NODE.AUTHORITY
jesx.MODIFY.NODE	CONTROL	$T	CONNECT	NO	ISFATTR.NETOPTS.CONNECT
jesx.MODIFY.NODE	CONTROL	$T	CONN-INT	NO	ISFATTR.NETOPTS.CTIME
jesx.MODIFY.NODE	CONTROL	$T	CP	NO	ISFATTR.NODE.COMPACT
jesx.MODIFY.NODE	CONTROL	$T	DIRECT	NO	SFATTR.NODE.DIRECT
jesx.MODIFY.NODE	CONTROL	$T	END	NO	ISFATTR.NODE.ENDNODE
jesx.MODIFY.NODE	CONTROL	$T	HOLD	NO	ISFATTR.NODE.HOLD
jesx.MODIFY.NODE	CONTROL	$T	LINE	NO	ISFATTR.NODE.LINE
jesx.MODIFY.NODE	CONTROL	$T	LOGMODE	NO	ISFATTR.NODE.LOGMODE
jesx.MODIFY.NODE	CONTROL	$T	NODENAME	NO	ISFATTR.NODE.LOGON
jesx.MODIFY.NODE	CONTROL	$T	NETSRV	NO	ISFATTR.NODE.NETSRV
jesx.MODIFY.NODE	CONTROL	$T	PEN	NO	ISFATTR.NODE.PENCRYPT
jesx.MODIFY.NODE	CONTROL	$T	PMG	NO	ISFATTR.NODE.PATHMGR
jesx.MODIFY.NODE	CONTROL	$T	PRV	NO	ISFATTR.NODE.PRIVATE
jesx.MODIFY.NODE	CONTROL	$T	RECV	NO	ISFATTR.NODE.RECEIVE
jesx.MODIFY.NODE	CONTROL	$T	REST	NO	ISFATTR.NODE.REST
jesx.MODIFY.NODE	CONTROL	$T	SENDP	NO	ISFATTR.NODE.SENDP
jesx.MODIFY.NODE	CONTROL	$T	SENTRS	NO	ISFATTR.NODE.SENTREST
jesx.MODIFY.NODE	CONTROL	$T	SSIGNON	NO	ISFATTR.NODE.SSIGNON
jesx.MODIFY.NODE	CONTROL	$T	SUBNET	NO	ISFATTR.NODE.SUBNET
jesx.MODIFY.NODE	CONTROL	$T	TR	NO	ISFATTR.NODE.TRACE
jesx.MODIFY.NODE	CONTROL	$T	TRANS	NO	ISFATTR.NODE.TRANSMIT
jesx.MODIFY.NODE	CONTROL	$T	VERIFYP	NO	ISFATTR.NODE.VERIFYP
jesx.MODIFY.OFF	CONTROL	$T	LINE-LIMIT	SO	ISFATTR.SELECT.LIM
jesx.MODIFY.OFF	CONTROL	$T	MBURST	SO	ISFATTR.MODIFY.BURST
jesx.MODIFY.OFF	CONTROL	$T	MCLASS	SO	ISFATTR.MODIFY.CLASS
jesx.MODIFY.OFF	CONTROL	$T	MDEST	SO	ISFATTR.MODIFY.DEST
jesx.MODIFY.OFF	CONTROL	$T	MFCB	SO	ISFATTR.MODIFY.FCB

Table 103. Overtypeable Fields Sorted by OPERCMDS Resource Name (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the MVS/JES Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	MVS/JES Command	Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)
jesx.MODIFY.OFF	CONTROL	$T	MFLH	SO	ISFATTR.MODIFY.FLASH
jesx.MODIFY.OFF	CONTROL	$T	MFORMS	SO	ISFATTR.MODIFY.FORMS
jesx.MODIFY.OFF	CONTROL	$T	MHOLD	SO	ISFATTR.MODIFY.HOLD
jesx.MODIFY.OFF	CONTROL	$T	MODSP	SO	ISFATTR.MODIFY.ODISP
jesx.MODIFY.OFF	CONTROL	$T	MPRMODE	SO	ISFATTR.MODIFY.PRMODE
jesx.MODIFY.OFF	CONTROL	$T	MSAFF	SO	ISFATTR.MODIFY.SYSAFF
jesx.MODIFY.OFF	CONTROL	$T	MUCS	SO	ISFATTR.MODIFY.UCS
jesx.MODIFY.OFF	CONTROL	$T	MWRITER	SO	ISFATTR.MODIFY.WRITER
jesx.MODIFY.OFF	CONTROL	$T	NOTIFY	SO	ISFATTR.OFFLOAD.NOTIFY
jesx.MODIFY.OFF	CONTROL	$T	PAGE-LIMIT	SO	ISFATTR.SELECT.PLIM
jesx.MODIFY.OFF	CONTROL	$T	SBURST	SO	ISFATTR.SELECT.BURST
jesx.MODIFY.OFF	CONTROL	$T	SCLASS	SO	ISFATTR.SELECT.CLASS
jesx.MODIFY.OFF	CONTROL	$T	SCLASS1-8	SO	ISFATTR.SELECT.CLASS
jesx.MODIFY.OFF	CONTROL	$T	SDEST1	SO	ISFATTR.SELECT.DEST
jesx.MODIFY.OFF	CONTROL	$T	SDISP	SO	ISFATTR.SELECT.DISP
jesx.MODIFY.OFF	CONTROL	$T	SRANGE	SO	ISFATTR.SELECT.RANGE
jesx.MODIFY.OFF	CONTROL	$T	SFCB	SO	ISFATTR.SELECT.FCB
jesx.MODIFY.OFF	CONTROL	$T	SFLH	SO	ISFATTR.SELECT.FLASH
jesx.MODIFY.OFF	CONTROL	$T	SFORMS	SO	ISFATTR.SELECT.FORMS
jesx.MODIFY.OFF	CONTROL	$T	SHOLD	SO	ISFATTR.SELECT.HOLD
jesx.MODIFY.OFF	CONTROL	$T	SJOBNAME	SO	ISFATTR.SELECT.JOBNAME
jesx.MODIFY.OFF	CONTROL	$T	SODSP	SO	ISFATTR.SELECT.ODISP
jesx.MODIFY.OFF	CONTROL	$T	SOWNER	SO	ISFATTR.SELECT.OWNER
jesx.MODIFY.OFF	CONTROL	$T	SPRMODE1	SO	ISFATTR.SELECT.PRMODE
jesx.MODIFY.OFF	CONTROL	$T	SSAFF	SO	ISFATTR.SELECT.SYSAFF
jesx.MODIFY.OFF	CONTROL	$T	SSCHEDULING-ENV	SO	ISFATTR.SELECT.SCHENV
jesx.MODIFY.OFF	CONTROL	$T	SSRVCLASS	SO	ISFATTR.SELECT.SRVCLS
jesx.MODIFY.OFF	CONTROL	$T	SUCS	SO	ISFATTR.SELECT.UCS
jesx.MODIFY.OFF	CONTROL	$T	SVOL	SO	ISFATTR.SELECT.VOL
jesx.MODIFY.OFF	CONTROL	$T	SWRITER	SO	ISFATTR.SELECT.WRITER
jesx.MODIFY.OFF	CONTROL	$T	WORK-SELECTION	SO	ISFATTR.PROPTS.WS
jesx.MODIFY.OFFLOAD	CONTROL	$T	ARCHIVE	SO	ISFATTR.OFFLOAD.ARCHIVE
jesx.MODIFY.OFFLOAD	CONTROL	$T	CRTIME	SO	ISFATTR.OFFLOAD.CRTIME
jesx.MODIFY.OFFLOAD	CONTROL	$T	DSNAME	SO	ISFATTR.OFFLOAD.DATASET
jesx.MODIFY.OFFLOAD	CONTROL	$T	LABEL	SO	ISFATTR.OFFLOAD.LABEL
jesx.MODIFY.OFFLOAD	CONTROL	$T	PROT	SO	ISFATTR.OFFLOAD.PROTECT
jesx.MODIFY.OFFLOAD	CONTROL	$T	RTPD	SO	ISFATTR.OFFLOAD.RETENT

Table 103. Overtypeable Fields Sorted by OPERCMDS Resource Name (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the MVS/JES Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	MVS/JES Command	Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)
jesx.MODIFY.OFFLOAD	CONTROL	$T	UNIT	SO	ISFATTR.PROPTS.UNIT
jesx.MODIFY.OFFLOAD	CONTROL	$T	VALIDATE	SO	ISFATTR.OFFLOAD.VALIDATE
jesx.MODIFY.Q	UPDATE	*F	MINPCT	SP	ISFATTR.SPOOL.MINPCT
jesx.MODIFY.Q	UPDATE	*F	OVERFNAM	SP	ISFATTR.SPOOL.OVFNAME
jesx.MODIFY.Q	UPDATE	*F	PARTNAME	SP	ISFATTR.SPOOL.PARTNAME
jesx.MODIFY.SOCKET	CONTROL	$T	ANODE	NC	ISFATTR.NETOPTS.NODE
jesx.MODIFY.SOCKET	CONTROL	$T	CONNECT	NC	ISFATTR.NETOPTS.CONNECT
jesx.MODIFY.SOCKET	CONTROL	$T	CONN-INT	NC	ISFATTR.NETOPTS.CTIME
jesx.MODIFY.SOCKET	UPDATE	*F	CTR	NC	ISFATTR.PROPTS.CTRACE
jesx.MODIFY.SOCKET	CONTROL	$T	IPNAME	NS	ISFATTR.NETOPTS.IPNAME
jesx.MODIFY.SOCKET	CONTROL	$T	IPNAME	NC	ISFATTR.NETOPTS.IPNAME
jesx.MODIFY.SOCKET	UPDATE	*F	IPNAME	NC	ISFATTR.NETOPTS.IPNAME
jesx.MODIFY.SOCKET	UPDATE	*F	JTR	NC	ISFATTR.PROPTS.JTRACE
jesx.MODIFY.SOCKET	CONTROL	$T	LINE	NC	ISFATTR.NODE.LINE
jesx.MODIFY.SOCKET	CONTROL	$T	NETSRV	NC	ISFATTR.NETOPTS.NETSRV
jesx.MODIFY.SOCKET	CONTROL	$T	PORT	NC NS	ISFATTR.NETOPTS.PORT
jesx.MODIFY.SOCKET	UPDATE	*F	PORT	NC	ISFATTR.NETOPTS.PORT
jesx.MODIFY.SOCKET	CONTROL	$T	REST	NC	ISFATTR.LINE.REST
jesx.MODIFY.SOCKET	CONTROL	*F	SRVNAME	NC	ISFATTR.NETOPTS.NETSRV
jesx.MODIFY.SOCKET	UPDATE	*F	VTR	NC	ISFATTR.PROPTS.VTRACE
jesx.MODIFY.SPOOL	CONTROL	$T	RES	SP	ISFATTR.SPOOL.SYSAFF
jesx.MODIFY.SPOOL	CONTROL	$T	SAFF	SP	ISFATTR.SPOOL.RESERVED
jesx.MODIFY.U	UPDATE	*F	BURST	JDS	ISFATTR.OUTPUT.BURST
jesx.MODIFY.U	UPDATE	*F	C	JDS	ISFATTR.OUTPUT.CLASS
jesx.MODIFY.U	UPDATE	*F	CC	JDS	ISFATTR.OUTPUT.COPYCNT
jesx.MODIFY.U	UPDATE	*F	CHARS	JDS	ISFATTR.OUTPUT.CHARS
jesx.MODIFY.U	UPDATE	*F	CPYMOD	JDS	ISFATTR.OUTPUT.COPYMOD
jesx.MODIFY.U	UPDATE	*F	CPYMOD	J0	ISFATTR.PRTOPTS.COPYMOD
jesx.MODIFY.U	UPDATE	*F	DEST	JDS	ISFATTR.OUTPUT.DEST
jesx.MODIFY.U	UPDATE	*F	FCB	JDS	ISFATTR.OUTPUT.FCB
jesx.MODIFY.U	UPDATE	*F	FLASH	JDS	ISFATTR.OUTPUT.FLASH
jesx.MODIFY.U	UPDATE	*F	FORMS	JDS	ISFATTR.OUTPUT.FORMS
jesx.MODIFY.U	UPDATE	*F	PRMODE	JDS	ISFATTR.OUTPUT.PRMODE
jesx.MODIFY.U	UPDATE	*F	UCS	JDS	ISFATTR.OUTPUT.UCS
jesx.MODIFY.W	UPDATE	*F	DGRPY	PR PUN	ISFATTR.PROPTS.DGRPY
jesx.MODIFY.W	UPDATE	*F	DYN	PR PUN	ISFATTR.PROPTS.DYN
jesx.MODIFY.W	UPDATE	*F	OPLOG	PR	ISFATTR.PROPTS.OPACTLOG
jesx.RESTART.DEV.*device*	UPDATE	*R. See note 3.	B	PUN	ISFATTR.PROPTS.BPAGE

Table 103. Overtypeable Fields Sorted by OPERCMDS Resource Name (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the MVS/JES Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	MVS/JES Command	Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)
jesx.RESTART.DEV.*device*	UPDATE	*R. See note 3.	CHAR1	PR	ISFATTR.PROPTS.CHAR
jesx.RESTART.DEV.*device*	UPDATE	*R. See note 3.	CKPTPAGE	PR	ISFATTR.PROPTS.CKPTPAGE
jesx.RESTART.DEV.*device*	UPDATE	*R. See note 3.	CKPTSEC	PR	ISFATTR.PROPTS.CKPTSEC
jesx.RESTART.DEV.*device*	UPDATE	*R. See note 3.	COPIES	PR PUN	ISFATTR.PROPTS.COPIES
jesx.RESTART.DEV.*device*	UPDATE	*R. See note 3.	COPYMARK	PR	ISFATTR.PROPTS.COPYMARK
jesx.RESTART.DEV.*device*	UPDATE	*R. See note 3.	LINE-LIM-HI	PR PUN	ISFATTR.SELECT.LIM
jesx.RESTART.DEV.*device*	UPDATE	*R. See note 3.	LINE-LIM-LO	PR PUN	ISFATTR.SELECT.LIM
jesx.RESTART.DEV.*device*	UPDATE	*R. See note 3.	NPRO	PR	ISFATTR.PROPTS.NPRO
jesx.RESTART.DEV.*device*	UPDATE	*R. See note 3.	PAGE-LIM-HI	PR	ISFATTR.SELECT.PLIM
jesx.RESTART.DEV.*device*	UPDATE	*R. See note 3.	PAGE-LIM-LO	PR	ISFATTR.SELECT.PLIM
jesx.RESTART.DEV.*device*	UPDATE	*R. See note 3.	SCLASS	PR PUN	ISFATTR.SELECT.CLASS
jesx.RESTART.DEV.*device*	UPDATE	*R. See note 3.	SEPDS	PR PUN	ISFATTR.PROPTS.SEPDS
jesx.RESTART.DEV.*device*	UPDATE	*R. See note 3.	SFCB	PR	ISFATTR.SELECT.FCB
jesx.RESTART.DEV.*device*	UPDATE	*R. See note 3.	SFLH	PR	ISFATTR.SELECT.FLASH
jesx.RESTART.DEV.*device*	UPDATE	*R. See note 3.	SFORMS	PR PUN	ISFATTR.SELECT.FORMS
jesx.RESTART.DEV.*device*	UPDATE	*R. See note 3.	SPRMODE1	PR	ISFATTR.SELECT.PRMODE
jesx.RESTART.DEV.*device*	UPDATE	*R. See note 3.	SUCS	PR	ISFATTR.SELECT.UCS
jesx.RESTART.DEV.*device*	UPDATE	*R	WORK-SELECTION	PR PUN	ISFATTR.PROPTS.WS
jesx.ROUTE.JOBOUT	UPDATE	$R	EXECNODE	I ST	ISFATTR.JOB.EXECNODE
jesx.ROUTE.JOBOUT	UPDATE	$R	PRTDEST	I ST	ISFATTR.JOB.PRTDEST
jesx.START.DEV.*device*	UPDATE	*S	B	PR PUN	ISFATTR.PROPTS.BPAGE
jesx.START.DEV.*device*	UPDATE	*S. See note 3.	CHAR1	PR	ISFATTR.PROPTS.CHAR
jesx.START.DEV.*device*	UPDATE	*S. See note 3.	CB	PR	ISFATTR.PROPTS.CB
jesx.START.DEV.*device*	UPDATE	*S. See note 3.	CKPTPAGE	PR	ISFATTR.PROPTS.CKPTPAGE

Table 103. Overtypeable Fields Sorted by OPERCMDS Resource Name (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the MVS/JES Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	MVS/JES Command	Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)
jesx.START.DEV.*device*	UPDATE	*S. See note 3.	CKPTSEC	PR	ISFATTR.PROPTS.CKPTSEC
jesx.START.DEV.*device*	UPDATE	*S. See note 3.	COPIES	PR PUN	ISFATTR.PROPTS.COPIES
jesx.START.DEV.*device*	UPDATE	*S. See note 3.	COPYMARK	PR	ISFATTR.PROPTS.COPYMARK
jesx.START.DEV.*device*	UPDATE	*S	CPYMOD	PR	ISFATTR.PROPTS.COPYMOD
jesx.START.DEV.*device*	UPDATE	*S. See note 3.	LINE-LIM-HI	PR PUN	ISFATTR.SELECT.LIM
jesx.START.DEV.*device*	UPDATE	*S. See note 3.	LINE-LIM-LO	PR PUN	ISFATTR.SELECT.LIM
jesx.START.DEV.*device*	UPDATE	*S. See note 3.	NPRO	PR	ISFATTR.PROPTS.NPRO
jesx.START.DEV.*device*	UPDATE	*S. See note 3.	PAGE-LIM-HI	PR PUN	ISFATTR.SELECT.PLIM
jesx.START.DEV.*device*	UPDATE	*S. See note 3.	PAGE-LIM-LO	PR PUN	ISFATTR.SELECT.PLIM
jesx.START.DEV.*device*	UPDATE	*S. See note 3.	SBURST	PR	ISFATTR.SELECT.BURST
jesx.START.DEV.*device*	UPDATE	*S. See note 3.	SCLASS	PR PUN	ISFATTR.SELECT.CLASS
jesx.START.DEV.*device*	UPDATE	*S. See note 3.	SEPDS	PUN	ISFATTR.PROPTS.SEPDS
jesx.START.DEV.*device*	UPDATE	*S. See note 3.	SFCB	PR	ISFATTR.SELECT.FCB
jesx.START.DEV.*device*	UPDATE	*S. See note 3.	SFLH	PR	ISFATTR.SELECT.FLASH
jesx.START.DEV.*device*	UPDATE	*S. See note 3.	SFORMS	PR PUN	ISFATTR.SELECT.FORMS
jesx.START.DEV.*device*	UPDATE	*S. See note 3.	SPRMODE1	PR PUN	ISFATTR.SELECT.PRMODE
jesx.START.DEV.*device*	UPDATE	*S. See note 3.	SUCS	PR	ISFATTR.SELECT.UCS
jesx.START.DEV.*device*	UPDATE	*S. See note 3.	WORK-SELECTION	PUN	ISFATTR.PROPTS.WS
jesx.START.NET	CONTROL	$S	APPLID	LI	ISFATTR.LINE.APPLID
MVS.MODIFY.STC.*hcproc.hcstcid*	UPDATE	MODIFY	CATEGORY	CK	ISFATTR.CHECK.CATEGORY
MVS.MODIFY.STC.*hcproc.hcstcid*	UPDATE	MODIFY	DEBUG	CK	ISFATTR.CHECK.DEBUG
MVS.MODIFY.STC.*hcproc.hcstcid*	UPDATE	MODIFY	EINTERVAL	CK	ISFATTR.CHECK.EINTERVAL
MVS.MODIFY.STC.*hcproc.hcstcid*	UPDATE	MODIFY	INTERVAL	CK	ISFATTR.CHECK.INTERVAL

Table 103. Overtypeable Fields Sorted by OPERCMDS Resource Name (continued).

The variable ***jesx*** should be replaced by the name of the targeted JES subsystem.

Replace ***hcproc*** and ***hcstcid*** with the IBM Health Checker for z/OS procedure name and started task ID.

Resources apply to the JES indicated by the command in the MVS/JES Command column: the $ command character indicates a JES2 command and the * command character indicates a JES3 command.

OPERCMDS Resource Name	Required Access	MVS/JES Command	Overtypeable Field	SDSF Panel	SDSF Resource Name (UPDATE Authority Required)
MVS.MODIFY.STC. *hcproc.hcstcid*	UPDATE	MODIFY	PARAMETERS	CK	ISFATTR.CHECK.PARM
MVS.MODIFY.STC. *hcproc.hcstcid*	UPDATE	MODIFY	SEVERITY	CK	ISFATTR.CHECK.SEVERITY
MVS.MODIFY.STC. *hcproc.hcstcid*	UPDATE	MODIFY	USERDATE	CK	ISFATTR.CHECK.USERDATE
MVS.MODIFY.STC. *hcproc.hcstcid*	UPDATE	MODIFY	VERBOSE	CK	ISFATTR.CHECK.VERBOSE
MVS.MODIFY.STC. *hcproc.hcstcid*	UPDATE	MODIFY	WTOTYPE	CK	ISFATTR.CHECK.WTOTYPE
MVS.MODIFY.WLM	UPDATE	MODIFY	System	RES	ISFATTR.RESOURCE.*system*
MVS.RESET	UPDATE	RESET	PGN	DA	ISFATTR.JOB.PGN
MVS.RESET	UPDATE	RESET	QUIESCE	DA	ISFATTR.JOB.QUIESCE
MVS.RESET	UPDATE	RESET	SRVCLASS	DA	ISFATTR.JOB.SRVCLASS
MVS.ROUTE	READ	RO	Any, when the system is other than the one the user is logged on to	DA INIT MAS PR	

Notes:

[1] SDSF uses the subsystem interface (SSI) when you overtype the C (JES output class) or DEST (JES print destination name) on the JDS panel. You can change the class or destination without releasing the output. In order to release output when the JESSPOOL class is enabled, the user must have ALTER authority to the JESSPOOL resource. This authority is implied for the JESSPOOL resources created by the user.

[2] The SAF resource varies with the JES2 resource. See "JES2 resources."

[3] In a JES3 environment, the command issued and OPERCMDS resource depend on the action character that is used with the overtype. See Table 102 on page 279.

Access authority

Multiple OPERCMDS class resources are often provided for the same overtypeable field, but they are for different panels. You choose the OPERCMDS resource that you need according to the panels you are protecting. In the table, *jesx* should be replaced by the name of the targeted JES subsystem.

To see how this information relates to the command levels for the action characters and resource names, see the CMDLEV parameter in "Group function parameters reference" on page 40. See also "Action characters and overtypeable fields for each command level" on page 73.

JES2 resources

The following table shows the SAF resources in the OPERCMDS class for the JES2 resources displayed on the RM panel.

Table 104. OPERCMDS Resources That Protect Overtyping JES2 Resources

JES2 Resource	OPERCMDS Resource	Required Access
BERT	*jesx*.MODIFY.CKPTSPACE	CONTROL
BSCB	*jesx*.MODIFY.TPDEF	CONTROL
BUFX	*jesx*.MODIFY.BUFDEF	CONTROL
CKVR	*jesx*.MODIFY.CKPTDEF	CONTROL
CMBS	*jesx*.MODIFY.CONDEF	CONTROL
CMDS	*jesx*.MODIFY.CONDEF	CONTROL
ICES	*jesx*.MODIFY.TPDEF	CONTROL
JNUM	*jesx*.MODIFY.JOBDEF	CONTROL
JOES	*jesx*.MODIFY.OUTDEF	CONTROL
JQES	*jesx*.MODIFY.JOBDEF	CONTROL
LBUF	*jesx*.MODIFY.BUFDEF	CONTROL
NHBS	*jesx*.MODIFY.NJEDEF	CONTROL
SMFB	*jesx*.MODIFY.SMFDEF	CONTROL
TBUF	Not applicable	
TGS	*jesx*.MODIFY.SPOOLDEF	CONTROL
TTAB	*jesx*.MODIFY.TRACEDEF	CONTROL
VTMB	*jesx*.MODIFY.TPDEF	CONTROL

Page data sets

Protecting page data sets

Protect page data sets by defining resource names in the SDSF class. The resources are shown in Table 105.

Table 105. SAF Resources for Page Data Sets

Action Characters and Overtypes	Resource Name	Class	Access Required
D	ISFPAG.*datasetname*	SDSF	READ
DC	ISFPAG.*datasetname*	SDSF	READ
DD	ISFPAG.*datasetname*	SDSF	READ
DL	ISFPAG.*datasetname*	SDSF	READ
DP	ISFPAG.*datasetname*	SDSF	READ
DS	ISFPAG.*datasetname*	SDSF	READ

To control access to the PAG panel, protect the PAG command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting page data sets

To protect all page data sets and permit a user to control them, define a generic profile as follows:

```
REDEFINE SDSF ISFPAG.** UACC(NONE)
PERMIT ISFPAG.** CLASS(SDSF) ID(userid) ACCESS(READ)
```

PARMLIB data sets

Protecting PARM data sets

Protect PARM data sets by defining resource names in the SDSF class. The resources are shown in Table 106.

Table 106. SAF Resources for PARM Data Sets

Action Characters and Overtypes	Resource Name	Class	Access Required
D	ISFPARM.*datasetname*	SDSF	READ
DE	ISFPARM.*datasetname*	SDSF	READ

To control access to the PARM panel, protect the PARM command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting PARM data sets

To protect all PARM data sets and permit a user to control them, define a generic profile as follows:

```
REDEFINE SDSF ISFPARM.** UACC(NONE)
PERMIT ISFPARM.** CLASS(SDSF) ID(userid) ACCESS(READ)
```

Printers

You can protect the printers displayed on the PR panel.

Authority to access the job on the printer is not checked.

Protecting printers

Protect printers by defining resource names in the WRITER class. The resources are shown in Table 107.

Table 107. Authority Required to Printer Resources for Actions and Overtypes

Action Character or Overtypeable Field	Resource Name	Class	Access
D action character	*jesx*.LOCAL.*device-name* for local printers *jesx*.RJE.*device-name* for remote printers	WRITER	READ
C action character	*jesx*.LOCAL.*device-name* for local printers *jesx*.RJE.*device-name* for remote printers	WRITER	ALTER
K action character, FSSName overtype	*jesx*.LOCAL.*device-name* *jesx*.RJE.*device-name*	WRITER	CONTROL
All others	*jesx*.LOCAL.*device-name* for local printers *jesx*.RJE.*device-name* for remote printers	WRITER	CONTROL

In the table,

jesx
: is the name of the JES subsystem the printer is on.

device-name
is the name of the printer.

To protect the MVS and JES commands generated by action characters or overtypes, see "Tables of action characters" on page 213 and "Tables of overtypeable fields" on page 258.

To control access to the PR panel, protect the PR command. This is described in "Authorized SDSF commands" on page 235.

Permitting access only while using SDSF

Users can be conditionally permitted to access the WRITER class resources so that they only can access printers through SDSF. See "Using conditional access" on page 209 for more information.

Examples of protecting printers

In the following examples, *jesx* is the name of the JES subsystem. For example, it might be *JES2*, *JESA*, or to protect all JES subsystems, *JES%*.

1. To protect all printers and punches, issue the following commands:
   ```
   RDEFINE WRITER jesx.** UACC(READ)
   PERMIT jesx.** CLASS(WRITER) ID(userid or groupid) ACCESS(ALTER)
   ```
2. To restrict printers to only be used through SDSF, issue the following command:
   ```
   PERMIT jesx.** CLASS(WRITER) ID(userid or groupid) ACCESS(ALTER)
   WHEN(CONSOLE(SDSF))
   ```
   You must have the CONSOLE class active, the SDSF console defined in the console class, and the user authorized to use the SDSF console through the CONSOLE class, as follows:
   ```
   SETROPTS CLASSACT(CONSOLE)
   RDEFINE CONSOLE SDSF UACC(NONE)
   PERMIT SDSF CLASS(CONSOLE) ID(userid or groupid) ACCESS(READ)
   ```

Processes (z/OS UNIX System Services)

You can protect the z/OS UNIX System Services (z/OS UNIX) processes displayed on the PS panel.

Protecting processes

Protect processes by defining resource names in the SDSF class. The resources are shown in Table 114 on page 301.

Table 108. Authority Required to z/OS UNIX Processes for Actions and Overtypes

Action Character or Overtypeable Field	Resource Name	Class	Access
D action character	ISFPROC.*owner.jobname*	SDSF	READ
All others	ISFPROC.*owner.jobname*	SDSF	ALTER

In the table,

owner
is the owner of the z/OS UNIX process.

jobname
is the jobname of the z/OS UNIX process.

To protect the MVS and JES commands generated, see "Tables of action characters" on page 213.

To control access to the PS panel, protect the PS command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting processes

To protect all processes issue the following commands:

```
RDEFINE SDSF ISFPROC.** UACC(NONE)
PERMIT ISFPROC.** CLASS(SDSF) ID(userid or groupid)
ACCESS(ALTER)
```

Punches

You can protect the punches displayed on the PUN panel.

Protecting punches

Protect punches by defining resource names in the WRITER class. The resources are shown in Table 109.

Table 109. Authority Required to Punch Resources for Actions and Overtypes

Action Character or Overtypeable Field	Resource Name	Class	Access
D action character	*jesx*.LOCAL.*device-name* for local punches	WRITER	READ
	jesx.RJE.*device-name* for remote punches		
C action character	*jesx*.LOCAL.*device-name* for local punches	WRITER	ALTER
	jesx.RJE.*device-name* for remote punches		
All others	*jesx*.LOCAL.*device-name* for local punches	WRITER	CONTROL
	jesx.RJE.*device-name* for remote punches		

In the table,

jesx
is the name of the JES subsystem.

device-name
is the name of the punch.

To protect the MVS and JES commands generated, see "Tables of action characters" on page 213 and "Tables of overtypeable fields" on page 258.

To control access to the PUN panel, protect the PUN command. This is described in "Authorized SDSF commands" on page 235.

Permitting access only while using SDSF

Users can be conditionally permitted to access the WRITER class resources so that they only can access punches through SDSF. With RACF, the user can be permitted

to access the WRITER profiles using the clause `WHEN(CONSOLE(SDSF))` with the PERMIT command. See "Using conditional access" on page 209 for more information.

Example of protecting punches

To protect all punches and printers issue the following commands:

```
RDEFINE WRITER jesx.** UACC(READ)
PERMIT jesx.** CLASS(WRITER) ID(userid or groupid) ACCESS(ALTER)
```

Readers

You can protect the readers displayed on the RDR panel.

Protecting readers

Protect readers by defining resource names in the SDSF class. The resources are shown in Table 110.

Table 110. Authority Required to Reader Resources for Actions and Overtypes

Action Character or Overtypeable Field	Resource Name	Class	Access
D action character	ISFRDR.*device-name.jesx*	SDSF	READ
C action character	ISFRDR.*device-name.jesx*	SDSF	ALTER
All others	ISFRDR.*device-name.jesx*	SDSF	CONTROL

In the table,

jesx
: is the name of the JES subsystem.

device-name
: is the name of the reader.

To protect the MVS and JES commands generated, see "Tables of action characters" on page 213 and "Tables of overtypeable fields" on page 258.

To control access to the RDR panel, protect the RDR command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting readers

To protect all readers issue the following commands:

```
RDEFINE SDSF ISFRDR.** UACC(NONE)
PERMIT ISFRDR.** CLASS(SDSF) ID(userid or groupid) ACCESS(CONTROL)
```

Resources defined to WLM

You can protect the WLM resources that are displayed on the RES panel.

Protecting WLM resources

Protect WLM resources by defining SAF resource names in the SDSF class. The SAF resources are shown in Table 111 on page 299.

Table 111. Authority Required to SAF Resources for WLM Resources

Action Character or Overtypeable Field	Resource Name	Class	Access
D action character	ISFRES.*resource.system*	SDSF	READ
Overtype system	ISFRES.*resource.system*	SDSF	ALTER

To protect the MVS commands generated by action characters or overtypeable fields, see "Tables of action characters" on page 213 and "Tables of overtypeable fields" on page 258.

To control access to the RES panel, protect the RES command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting resources

To protect all resources and permit a user to control them, define a generic profile as follows:

```
RDEFINE SDSF ISFRES.** UACC(NONE)
PERMIT ISFRES.** CLASS(SDSF) ID(userid or groupid) ACCESS(ALTER)
```

Scheduling environments

You can protect the WLM scheduling environments that are displayed on the SE panel.

Protecting scheduling environments

Protect scheduling environments by defining resource names in the SDSF class. The resources are shown in Table 112.

Table 112. Authority Required to Scheduling Environment Resource for Actions

Action Character or Overtypeable Field	Resource Name	Class	Access
D, R and ST action characters	ISFSE.*sched-env.system*	SDSF	READ

To protect the MVS command generated by the D action character, see "Tables of action characters" on page 213.

To protect the R and ST action characters, protect the RES and ST commands. To control access to the SE panel, protect the SE command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting scheduling environments

To protect all scheduling environments and permit a user to control them, define a generic profile as follows:

```
RDEFINE SDSF ISFSE.** UACC(NONE)
PERMIT ISFSE.** CLASS(SDSF) ID(userid or groupid) ACCESS(CONTROL)
```

SDSF server

The SDSF server is used to process ISFPARMS statements and to provide sysplex data on the sysplex-wide device panels (PR, INIT, PUN, RDR and so on). For more information, refer to Chapter 3, "Using the SDSF server," on page 107.

You can protect these aspects of the SDSF server:

- Use of the SERVER parameter on the SDSF command, which specifies a server name that overrides the default server name defined in ISFPARMS.
- Reverting from ISFPARMS in statement format to ISFPARMS in assembler macro format, when the server is not available or no ISFPARMS statements are defined.
- Use of the server operator commands.

If you are using the server with WebSphere MQ to provide sysplex data, you must also protect server access to WebSphere MQ queues. Refer to "WebSphere MQ" on page 306 for more information.

Protecting the SDSF server

The resources related to server processing of ISFPARMS are shown in Table 113.

Table 113. Authority Required to Server Functions

Function	Resource Name	Class	Access
Use of the SERVER parameter on the SDSF command	ISFCMD.OPT.SERVER	SDSF	READ
Reverting to ISFPARMS in assembler macro format	SERVER.NOPARM	SDSF	READ
MODIFY *server*,DISPLAY server command	*server-name*.MODIFY.DISPLAY	OPERCMDS	READ
All other server MODIFY commands	*server-name*.MODIFY.*modify-parm*	OPERCMDS	CONTROL

In the table,

server-name
: is the name of the SDSF server specified either by the ISFPMAC macro or SDSF command.

modify-parm
: is one of these parameters of the MODIFY command: DEBUG, DISPLAY, FOLDMSG, LOGCLASS, LOGTYPE, REFRESH, START, STOP, TRACE, TRCLASS. The MODIFY command is described in Chapter 3, "Using the SDSF server," on page 107.

The server START and STOP commands are protected by MVS. The resources are MVS.START.STC.*server-name* and MVS.STOP.STC.*server-name*, respectively. Both are in the OPERCMDS class and require UPDATE authority.

Examples of protecting the SDSF server

1. To allow SDSF to revert from the ISFPARMS defined with statements to the ISFPARMS defined with assembler macros, issue the following commands:

```
RDEFINE SDSF SERVER.NOPARM UACC(NONE)
PERMIT SERVER.NOPARM CLASS(SDSF) ID(userid or groupid) ACCESS(READ)
```

2. To protect use of all MODIFY command parameters for server SDSF, issue the following commands:

```
RDEFINE OPERCMDS SDSF.MODIFY.** UACC(NONE)
PERMIT SDSF.MODIFY.** CLASS(OPERCMDS) ID(userid) ACCESS(CONTROL)
```

Spool offloaders

You can protect the offloaders displayed on the SO panel (JES2 only).

Protecting spool offloaders

Protect spool offloaders by defining resource names in the SDSF class. The resources are shown in Table 114.

Table 114. Authority Required to Offloader Resources for Actions and Overtypes

Action Character or Overtypeable Field	Resource Name	Class	Access
D action character	ISFSO.*device-name.jesx*	SDSF	READ
C action character	ISFSO.*device-name.jesx*	SDSF	ALTER
All others	ISFSO.*device-name.jesx*	SDSF	CONTROL

In the table,

device-name
is the name of the offloader, transmitter, or receiver.

jesx
is the name of the JES2 subsystem.

To protect the MVS and JES2 commands generated, see "Tables of action characters" on page 213 and "Tables of overtypeable fields" on page 258.

To control access to the SO panel, protect the SO command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting spool offloaders

To protect all offloaders issue the following commands:

```
RDEFINE SDSF ISFSO.** UACC(NONE)
PERMIT ISFSO.** CLASS(SDSF) ID(userid or groupid) ACCESS(CONTROL)
```

Spool volumes

You can protect the spool volumes displayed on the SP panel.

Protecting spool volumes

Protect spool volumes by defining resource names in the SDSF class. The resources are shown in Table 114.

Table 115. Authority Required to Spool Volume Resources for Actions and Overtypes

Action Character or Overtypeable Field	Resource Name (JES2) Resource Name (JES3)	Class	Access
D, DL and J action character	ISFSP.*volser.jesx*	SDSF	READ
	ISFSP.*ddname.jesx* ISFSP.*partitionname.jesx*		
All others	ISFSP.*volser.jesx*	SDSF	CONTROL
	ISFSP.*ddname.jesx* ISFSP.*partitionname.jesx*		

In the table,

volser
is the volume serial of the spool volume.

ddname
is the ddname.

partitionname
is the name of the partition.

jesx
is the name of the JES subsystem.

To protect the MVS and JES commands generated, see "Tables of action characters" on page 213 and "Tables of overtypeable fields" on page 258.

To control access to the SP panel, protect the SP command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting spool volumes

To protect all spool volumes issue the following commands:

```
RDEFINE SDSF ISFSP.** UACC(NONE)
PERMIT ISFSP.** CLASS(SDSF) ID(userid or groupid) ACCESS(CONTROL)
```

SYSLOG

In a JES2 environment that is z/OS V1R11 or later, or a JES3 environment, you can control access to the SYSLOG that is displayed on the LOG panel by controlling:

- Access to the LOG command, which displays the LOG panel. This is explained in "Authorized SDSF commands" on page 235.
- Access to the JES logical log. JES, rather than SDSF, issues the SAF call to check user authorization. For more information, refer to "SYSLOG" on page 349.

Parameters of the LOG command allow users to choose the sysplex-wide OPERLOG rather than the single-system SYSLOG. For information on protecting the OPERLOG, see "OPERLOG" on page 255.

Protecting the logical log

Protect the logical log by defining a resource name in the JESSPOOL class. The resource is shown in Table 116 on page 303.

Table 116. Authority Required for Accessing the Logical Log

Function	Resource Name	Class	Access
Access to the JES logical log	*nodeid*.+MASTER+.SYSLOG.SYSTEM.*sysname*	JESSPOOL	READ

As an alternative to defining the JESSPOOL profiles, you can define the custom property Security.Syslog.UseSAFRecvr in ISFPARMS to force the SAF call to always succeed even when the profile is not defined. This may be useful as you migrate to using the new logical log. For more information, see "Customized properties (PROPLIST)" on page 91.

System Symbol information

Protecting system symbol information

Protect system symbol information by defining resource names in the SDSF class. The resources are shown in Table 117.

Table 117. SAF Resources for System Symbol Information

Action Characters and Overtypes	Resource Name	Class	Access Required
D	ISFSYM.*symbolname.sysname*	SDSF	READ

To control access to the SYM panel, protect the SYM command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting system symbol information

To protect all system symbol information and permit a user to control it, define a generic profile as follows:

```
REDEFINE SDSF ISFSYM.** UACC(NONE)
PERMIT ISFSYM.** CLASS(SDSF) ID(userid) ACCESS(READ)
```

System information

Protecting system information

Protect system information by defining resource names in the SDSF class. The resources are shown in Table 118.

Table 118. SAF Resources for System Information

Action Characters and Overtypes	Resource Name	Class	Access Required
D	ISFSYS.*sysplexname.systemname*	SDSF	READ
DAA	ISFSYS.*sysplexname.systemname*	SDSF	READ
DAL	ISFSYS.*sysplexname.systemname*	SDSF	READ
DALO	ISFSYS.*sysplexname.systemname*	SDSF	READ
DC	ISFSYS.*sysplexname.systemname*	SDSF	READ
DCEE	ISFSYS.*sysplexname.systemname*	SDSF	READ
DD	ISFSYS.*sysplexname.systemname*	SDSF	READ
DEM	ISFSYS.*sysplexname.systemname*	SDSF	READ

Table 118. SAF Resources for System Information (continued)

Action Characters and Overtypes	Resource Name	Class	Access Required
DG	ISFSYS.*sysplexname.systemname*	SDSF	READ
DI	ISFSYS.*sysplexname.systemname*	SDSF	READ
DIQP	ISFSYS.*sysplexname.systemname*	SDSF	READ
DLL	ISFSYS.*sysplexname.systemname*	SDSF	READ
DLO	ISFSYS.*sysplexname.systemname*	SDSF	READ
DLR	ISFSYS.*sysplexname.systemname*	SDSF	READ
DM	ISFSYS.*sysplexname.systemname*	SDSF	READ
DMP	ISFSYS.*sysplexname.systemname*	SDSF	READ
DO	ISFSYS.*sysplexname.systemname*	SDSF	READ
DP	ISFSYS.*sysplexname.systemname*	SDSF	READ
DPCD	ISFSYS.*sysplexname.systemname*	SDSF	READ
DPCI	ISFSYS.*sysplexname.systemname*	SDSF	READ
DSF	ISFSYS.*sysplexname.systemname*	SDSF	READ
DSL	ISFSYS.*sysplexname.systemname*	SDSF	READ
DSM	ISFSYS.*sysplexname.systemname*	SDSF	READ
DSY	ISFSYS.*sysplexname.systemname*	SDSF	READ
DT	ISFSYS.*sysplexname.systemname*	SDSF	READ
DTO	ISFSYS.*sysplexname.systemname*	SDSF	READ
DTR	ISFSYS.*sysplexname.systemname*	SDSF	READ
DTS	ISFSYS.*sysplexname.systemname*	SDSF	READ
DW	ISFSYS.*sysplexname.systemname*	SDSF	READ
DX	ISFSYS.*sysplexname.systemname*	SDSF	READ

To control access to the SYS panel, protect the SYS command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting system information

To protect system information and permit a user to control it, define a generic profile as follows:

```
REDEFINE SDSF ISFSYS.** UACC(NONE)
PERMIT ISFSYS.** CLASS(SDSF) ID(userid) ACCESS(READ)
```

System requests

You can protect the system requests displayed on the SR panel.

Protecting system requests

Protect system requests by defining resource names in the SDSF class. The resources are shown in Table 113 on page 300.

Table 119. Authority Required to System Request Resource for Action Characters

Action Character	Resource Name	Class	Access
D	ISFSR.*type.system.jobname*	SDSF	READ
C	ISFSR.ACTION.*system.jobname*	SDSF	READ
AI, R	ISFSR.REPLY.*system.jobname*	SDSF	READ

In the table,

type
is the message type, either ACTION or REPLY.

system
is the name of the originating system.

jobname
is the name of the issuing job.

To protect the MVS commands generated, see "Tables of action characters" on page 213.

To control access to the SR panel, protect the SR command. This is described in "Authorized SDSF commands" on page 235.

Example of protecting system requests

To protect all system requests issue the following commands:

```
RDEFINE SDSF ISFSR.** UACC(NONE)
PERMIT ISFSR.** CLASS(SDSF) ID(userid or groupid) ACCESS(READ)
```

User log (ULOG)

Users can browse the ULOG to see all system commands and responses issued during their user session, including commands generated by SDSF. If the installation activates message suppression attributes, all command responses may not be returned.

SDSF uses MVS console services to acquire an extended console for the user; all commands issued use that console identifier.

Protecting the ULOG

You protect the ULOG by:

- Controlling access to the ULOG command, which displays the ULOG panel. This is described in "Authorized SDSF commands" on page 235.
- Controlling access to the extended console that SDSF acquires. The extended console is protected by a resource in the OPERCMDS class, shown in Table 120.

Table 120. Resource that Protects the Extended Console

Function	Resource Name	Class
Extended console	MVS.MCSOPER.*console-name*	OPERCMDS

This resource is checked by SDSF. If no resource has been defined or if the OPERCMDS class is not active, SDSF allows activation of the extended console.

The console name used by SDSF defaults to the user ID. When SDSF needs to activate a console and the default console name is already in use, SDSF attempts to use a modified console name, which consists of the default name plus a single-character suffix. Users can change the console name with the SET CONSOLE command.

SDSF supplies an OPERPARM with master level authority when activating the console. Since SDSF supplies the OPERPARM, the user's OPERPARM segment (defined through RACF) is ignored.

When SDSF is using an extended console and commands are issued through the / (slash) command, some subsystems (such as NetView* and CICS*) require the console name to be defined to the subsystem.

For more information on the console used by SDSF, see "Issuing MVS and JES commands" on page 342. For more information on protecting the extended console, see *z/OS MVS Planning: Operations*.

Examples of protecting ULOG

1. To activate the OPERCMDS class and define a resource for the extended console, use the following RACF commands:

   ```
   RDEFINE OPERCMDS MVS.MCSOPER.console-name
   PERMIT MVS.MCSOPER.console-name ID(userid) ACCESS(READ)
   ```

2. To refresh the OPERCMDS class, issue the following:

   ```
   SETROPTS RACLIST(OPERCMDS) REFRESH
   ```

WebSphere MQ

You can protect the WebSphere MQ queues and commands used by SDSF to provide sysplex support for SDSF panels. WebSphere MQ is used only with z/OS V1R12 or lower systems, or if you have specifically requested it. For more information, refer to "Servers with server groups and WebSphere MQ" on page 111.

The following discussion assumes that you have already defined SAF security for WebSphere MQ, including **context** and **connection** security, as described in *WeSphere MQ for z/OS System Setup Guide*. If you have not defined WebSphere MQ security, do that before implementing the security described here. For an overview of SDSF's use of WebSphere MQ, see "Server communications with WebSphere MQ" on page 116.

Providing security for SDSF's use of WebSphere MQ consists of:

- Protecting the queues used by SDSF
- Allowing the SDSF server to define queues
- Defining connection and context security

Note: Examine the resources and sample commands in this section carefully before implementing them. Some of them affect the function of WebSphere MQ beyond SDSF's use of it.

Protecting the queues

Protect the queues used by SDSF by defining resources in the MQQUEUE class.

The client request queue, actually an alias for the server request queue, is defined by SDSF so that users' reading of the queue, although allowed by SAF, is prohibited by WebSphere MQ. The SAF profiles, then, should prevent access by the user to the server request queue, and allow access to the client request queue. The server must have access to both queues.

The resources are shown in Table 121.

Table 121. Authority Required for the WebSphere MQ Queues

			Required Access	
Queue	**Resource Name**	**Class**	**Server**	**Client**
Server request queue	*ssid*.*prefix*.SERVER.*server.system*.REQUESTQ	MQQUEUE	ALTER	None
Client request queue, used to send work to the server, and to send work from the server to remote servers	*ssid*.*prefix*.CLIENT.*server*.*system*.REQUESTQ	MQQUEUE	ALTER	UPDATE (Put only)
ReplyTo queue, used by the client to receive server responses	*ssid*.*prefix*.USER.*userid*.*	MQQUEUE	UPDATE	UPDATE
Model queue, used to create dynamic queues	*ssid*.*prefix*.MODEL.**	MQQUEUE	UPDATE	UPDATE

In the table,

ssid
: is the WebSphere MQ subsystem ID. This is the queue manager name specified on the COMM statement of ISFPARMS.

prefix
: is a string that identifies the queue name. It is defined by the QPREFIX parameter of the COMM statement in ISFPARMS.

The *ssid*.*prefix*.MODEL.** resource affects function outside of SDSF.

Allowing the server to communicate with remote servers

The transmission queue is used to send messages to remote SDSF servers. The SDSF server needs access to this queue.

This resource affects function outside of SDSF.

Table 122. Authority Required for Communication with Remote Servers

			Required Access	
Function	**Resource Name**	**Class**	**Server**	**Client**
Transmission queue, used to send messages to remote SDSF servers	*ssid*.XMIT.QUEUE	MQQUEUE	UPDATE	None

In the table, *ssid* is the WebSphere MQ subsystem ID. This is the queue manager name specified on the COMM statement of ISFPARMS.

Allowing the server to define queues

The resources that protect the ability of the SDSF server to define queues are shown in Table 123. In addition to the SDSF server, you may want the operator ID used by WebSphere MQ for commands entered from the console to have access to these resources. This ID is CSQOPR by default.

These resources affect function outside of SDSF.

Table 123. Authority Required for Server Definition of Queues

			Required Access	
Function	**Resource Name**	**Class**	**Server**	**Client**
Define queues	*ssid*.DEFINE.QMODEL	MQCMDS	ALTER	None
Define a queue alias	*ssid*.DEFINE.QALIAS	MQCMDS	ALTER	None
Define queues	*ssid*.QUEUE.*prefix*.MODEL.QUEUE	MQADMIN	ALTER	None
Model queue, used to create the temporary server RreplyTo queue	*ssid*.SYSTEM.COMMAND.REPLY.MODEL	MQQUEUE	ALTER	None
Command input queue, used to submit DEFINE commands	*ssid*.SYSTEM.COMMAND.INPUT	MQQUEUE	ALTER	None

In the table,

ssid
: is the WebSphere MQ subsystem ID. This is the queue manager name specified on the COMM statement of ISFPARMS.

prefix
: is a string that identifies the queue name. It is defined by the QPREFIX parameter of the COMM statement in ISFPARMS.

If you don't want the SDSF server to issue WebSphere MQ DEFINE commands to define queues, you can specify QDEFINE(NO) on the COMM statement in ISFPARMS. However, you will need to define some queues manually. See “COMM statement” on page 32 for more information.

Defining connection security for SDSF

Connection security may be used to control which users can connect to WebSphere MQ. When connection security is enabled, both the SDSF server and the SDSF client must have access to the local queue manager. These both use the batch/TSO adapter.

This resource affects function outside of SDSF.

Table 124. Authority Required for Connection Security

			Required Access	
Function	**Resource Name**	**Class**	**Server**	**Client**
Connection security	*ssid*.BATCH	MQCONN	READ	READ

In the table, *ssid* is the WebSphere MQ subsystem ID. This is the queue manager name specified on the COMM statement of ISFPARMS.

Defining context security for SDSF

Context security is used to protect setting of the identity context fields in the message header. The server needs this authority; clients should not have this authority. WebSphere MQ itself also needs to be given this authority.

This resource affects function outside of SDSF.

Table 125. Authority Required for Context Security

			Required Access	
Function	**Resource Name**	**Class**	**Server**	**Client**
Context security	*ssid*.CONTEXT	MQADMIN	UPDATE	None

In the table, *ssid* is the WebSphere MQ subsystem ID. This is the queue manager name specified on the COMM statement of ISFPARMS.

Assigning user IDs

All security is dependent on the user ID of the task attempting access to the profile, as follows:

- For the SDSF server, the user ID is assigned through the started task table or STARTED class
- For TSO users, the user ID is the TSO logon user ID
- For batch jobs, the user ID is the user ID the batch job is running under.

Example

The following example uses RACF commands to define security for SDSF's use of WebSphere MQ. In this example, the following values are used:

Item	Value
Server ID	SDSF
WebSphere MQ subsystem ID (queue manager name)	MQS1, MQS2
WebSphere MQ user ID	MQS
Queue prefix	ISF
Operator user ID (used by WebSphere MQ for commands entered from the console)	CSQOPR
Client user ID (RACF group)	ISFSPROG

1. Assign the user ID SDSF to the server:

```
SETR CLASSACT(STARTED)
RDEFINE STARTED SDSF*.* STDATA(USER(SDSF))
```

2. Allow the server access to its own queues, and allow WebSphere MQ access to those queues:

```
RDEFINE MQQUEUE MQS*.ISF.SERVER.** UACC(NONE)
PE MQS*.ISF.SERVER.** ID(SDSF) ACC(ALTER) CL(MQQUEUE)
PE MQS*.ISF.SERVER.** ID(MQS) ACC(ALTER) CL(MQQUEUE)
RDEFINE MQQUEUE MQS*.ISF.CLIENT.** UACC(NONE)
PE MQS*.ISF.CLIENT.** ID(SDSF) ACC(ALTER) CL(MQQUEUE)
```

3. Allow the user, ISFSPROG, access to the client request queue:

```
PE MQS*.ISF.CLIENT.** ID(ISFSPROG) ACC(UPDATE)  CL(MQQUEUE)
```

Note: You can eliminate this PERMIT command by specifying a UACC of UPDATE in the RDEFINE for the resource, that is,

```
RDEFINE MQQUEUE MQS*.ISF.CLIENT.** UACC(UPDATE)
```

4. Allow the server access to the user ReplyTo queue:

```
RDEFINE MQQUEUE MQS*.ISF.USER.** UACC(NONE)
PE MQS*.ISF.USER.** ID(SDSF) ACC(UPDATE) CL(MQQUEUE)
```

5. Allow the user access to the ReplyTo queue using the global access table:

```
RDEFINE GLOBAL MQQUEUE
RALTER GLOBAL MQQUEUE ADDMEM(MQS*.ISF.USER.&RACUID.**/UPDATE)
SETR GLOBAL(MQQUEUE)
```

6. Allow the server and user access to the model queue:

```
RDEFINE MQQUEUE MQS*.ISF.MODEL.** UACC(NONE)
PE MQS*.ISF.MODEL.** ID(SDSF) ACC(UPDATE) CL(MQQUEUE)
PE MQS*.ISF.MODEL.** ID(ISFSPROG) ACC(UPDATE) CL(MQQUEUE)
```

Note: You can eliminate the PERMIT commands by specifying a UACC of UPDATE in the RDEFINE for the resource, that is,

```
RDEFINE MQQUEUE MQS*.ISF.MODEL.** UACC(UPDATE)
```

7. Allow the server access to the transmission queue:

```
RDEF  MQQUEUE  MQS*.XMIT.QUEUE  UACC(NONE)
PE  MQS*.XMIT.QUEUE  CL(MQQUEUE)  ID(SDSF)  ACC(UPDATE)
```

8. Allow the server to define queues:

```
RDEFINE MQCMDS MQS*.DEFINE.QMODEL UACC(NONE)
PE MQS*.DEFINE.QMODEL ID(SDSF) ACC(ALTER) CL(MQCMDS)
PE MQS*.DEFINE.QMODEL ID(CSQOPR) ACC(ALTER) CL(MQCMDS)
RDEFINE MQCMDS MQS*.DEFINE.QALIAS UACC(NONE)
PE MQS*.DEFINE.QALIAS ID(SDSF) ACC(ALTER) CL(MQCMDS)
PE MQS*.DEFINE.QALIAS ID(CSQOPR) ACC(ALTER) CL(MQCMDS)
RDEFINE MQADMIN MQS*.QUEUE.ISF.** UACC(NONE)
PERMIT MQS*.QUEUE.ISF.** ID(SDSF) ACC(ALTER) CL(MQADMIN)
RDEFINE MQQUEUE MQS*.SYSTEM.** UACC(NONE)
PE MQQUEUE MQS*.SYSTEM.** ID(SDSF) ACC(ALTER) CL(MQQUEUE)
```

9. Define connection security for the server and user:

```
RDEFINE MQCONN MQS*.BATCH UACC(NONE)
PE MQS*.BATCH ID(SDSF) ACC(READ) CL(MQCONN)
PE MQS*.BATCH ID(ISFSPROG) ACC(READ) CL(MQCONN)
```

Note: You can eliminate the PERMIT commands by specifying a UACC of READ in the RDEFINE for the resource, that is,

```
RDEFINE MQCONN MQS*.BATCH UACC(READ)
```

10. Enable context security:

```
RDEFINE MQADMIN MQS*.CONTEXT UACC(NONE)
PE MQS*.CONTEXT ID(SDSF) ACC(UPDATE) CL(MQADMIN)
PE MQS*.CONTEXT ID(MQS) ACC(ALTER) CL(MQADMIN)
```

Chapter 8. Converting ISFPARMS to SAF security

This topic discusses converting from ISFPARMS to SAF for security. It includes discussions and RACF examples.

When setting up RACF security, you will be using the RACF resources described in this topic. You will also need some of ISFPARMS, as described in Chapter 2, "Using ISFPARMS for customization and security," on page 15. For tables showing SAF equivalents for the parameters of ISFGRP and GROUP, see Appendix B, "SAF equivalents for ISFPARMS," on page 561.

Getting started

A good first goal is a one-to-one conversion from ISFPARMS security to SAF security. This may cause you to create more profiles than are needed, so you will want to analyze the profiles and combine them where practical.

Your first task is to analyze your current security system to determine the kind of protection and authorization you need. In addition to making your SDSF security system easier to maintain, this analysis may result in improvements in the general security and auditability of your installation

The conversion of SDSF security to SAF may require the cooperation of different groups in your organization. Some security administrators work as system programmers and are knowledgeable of the ISFPARMS security implementation. Other security administrators are independent of the system programmers and have no knowledge of the SDSF product and its functions. So, depending on your organization, the system programmer and security administrator may need to work together in the conversion effort.

SDSF environment

The following questions may help you analyze your current SDSF security:

- Who is using SDSF, and how?

  You may find that SDSF is only used by system programmers, or that operators are using SDSF to facilitate their jobs, or that SDSF is used by everyone authorized to submit jobs on the system.
- What sort of authorizations are permitted through ISFPARMS?

  You may find that some users only have access to the LOG panel, or that there are groups of users able to manipulate each others jobs, or that users other than operators are authorized to issue operator commands through SDSF.
- Is there a conflict between security and productivity?
- Are there any security exposures, for example, operators logged on unattended terminals or in an unprotected environment?
- How will SDSF be used in the future?

Migration considerations

When migrating to SAF, you should be aware of the following:

© Copyright IBM Corp. 1997, 2016

OWNER Command. There is no protection for the OWNER command using ISFPARMS. This command can only be protected using SAF. If the command is not protected using SAF, then all users can use the OWNER command to further restrict the jobs that appear on their displays.

The OWNER keyword on the ISFGRP macro or GROUP statement can be used to limit the jobs that appear on the displays.

Destinations. When a user has no IDEST list in ISFPARMS, that user must have READ authority to the SDSF class resource ISFOPER.ANYDEST.*jesx*. Otherwise, no jobs will appear on the queues and the user's DEST value, when queried, will be displayed as either blanks or the character string ????????, depending on the JES release.

When an IDEST list is provided for a user, the user must have READ authorization to each SDSF class resource (ISFAUTH.DEST.*destname*) protecting the destination names in the IDEST list.

When DEST and IDEST lists are specified in ISFPARMS and SAF security is used for destination names, refer to "Destination names" on page 239 for information on how to protect destination names.

NOTIFY. There is no one-to-one RACF equivalent for setting CMDAUTH or DSPAUTH to NOTIFY in ISFPARMS. To obtain similar functions, a user must have access to the appropriate person's output by way of the JESSPOOL resource. For a RACF example of how to give this authority, see "Providing function comparable to NOTIFY authority" on page 248.

CMDLEV. Although you can migrate command protection from ISFPARMS CMDLEV protection to RACF OPERCMDS protection in a one-to-one fashion, it is not necessarily advisable to keep the hierarchy restriction of CMDLEV when using RACF. RACF provides a more flexible means of authorizing users to access various commands. Decide which commands your users need and then authorize the proper users or groups of users to access the appropriate OPERCMDS resources.

When using RACF security for command-level authorization, for every CMDLEV parameter you wish to authorize with RACF, you must permit the user to access all corresponding MVS and JES command resources in the OPERCMDS class at that command level and all command levels prior to it.

To review the various command levels for the action characters and resource names, see the CMDLEV parameter in "Group function parameters reference" on page 40 and "Action characters and overtypeable fields for each command level" on page 73.

Examples of RACF security for SDSF groups

This section explains SAF security for three SDSF groups that are common to most installations:

- Group 1— System programmers
- Group 2—Operators
- Group 3—End users

For each group, there are two sample GROUP statements shown, one for ISFPARMS security, and another for SAF security. These samples do not show the ISFPARMS macros not related to security, such as ISFPMAC and ISFTR.

The sample for each group has a table of SAF resources you can use as a guide to establish security for that group. The tables show profiles that provide security equivalent to that provided by the AUTH, CMDAUTH, CMDLEV, and DSPAUTH parameters of the GROUP statements shown. To provide authority comparable to the NOTIFY function, see "Providing function comparable to NOTIFY authority" on page 248.

For specific profile information, see Chapter 7, "Protecting SDSF functions," on page 211. Appendix C, "SDSF resource names for SAF security," on page 577 contains a list of all resource names.

Providing group authority

All users can access the JESSPOOL resources they own. Users do not need access authority to work with their own jobs and output.

You can provide authority to the SDSF resources by group by going from broad access (for example, RACF generic profiles) to limited access (RACF discrete profiles).

The profiles shown in the table for the system programmers group are very broad, generic profiles that will protect all resources. The system programmers group can be given unlimited authority to these profiles. The profiles shown in the operator table are restrictive and can limit the operator's authority. The profiles shown in the end user table are even more restrictive.

System programmers will need access to all profiles for each group in order to retain access to all resources. Likewise, the operators, in addition to having access to their own profiles, will also need access to all profiles defined for end users.

Group 1 — system programmers

Members of the ISFPARMS system programmers group have unlimited authority. They have access to all SDSF resources and can perform all SDSF tasks.

Shown below are two sample GROUP statements, one that can be used without SAF and another that can be used with SAF profiles to provide Group 1 authority:

Table 126. Sample GROUP statements, Group 1

Without SAF	With SAF
GROUP NAME(ISFSPROG), TSOAUTH(JCL,OPER,ACCT), AUTH(ALL), CMDAUTH(ALL), CMDLEV(7), DSPAUTH(ALL), DFIELD2(DAFLD2), GPLEN(2), ACTION(ALL), ACTIONBAR(YES), APPC(ON), OWNER(NONE), CONFIRM(ON), CURSOR(ON, DATE(MMDDYYYY, DATESEP(/), LOG(OPERACT), ISYS(NONE), DADFLT(IN,OUT,TRANS,STC,TSU,JOB), VALTAB(TRTAB), UPCTAB(TRTAB2), LANG(ENGLISH), DISPLAY(OFF)	GROUP NAME(ISFSPROG), DFIELD2(DAFLD2), ACTION(ALL), ACTIONBAR(YES), APPC(ON), CONFIRM(ON), CURSOR(ON), DATE(MMDDYYYY), DATESEP(/), LOG(OPERACT), DADFLT(IN,OUT,TRANS,STC,TSU,JOB), VALTAB(TRTAB), UPCTAB(TRTAB2), LANG(ENGLISH), DISPLAY(OFF)

To control membership in the group, which is done with TSOAUTH in the "Without SAF" case, use the profile shown in Table 127.

Table 127. Profile for Membership in Group 1 — System Programmers

Function	Class	Resource Profile	Access
Membership	SDSF	GROUP.ISFSPROG.*server-name*	READ

For guidance on providing security equivalent to that provided by the AUTH, CMDAUTH, CMDLEV, and DSPAUTH parameters, see the generic profiles shown in Table 128 on page 315.

You can use one generic profile to protect all resources in a particular class. The ISF*.** profile also provides destination operator authority to the JESSPOOL resources and protects the OWNER command, which cannot be protected in an ISFGRP macro. The OPERCMDS profiles shown protect all JES and MVS commands, even those that are not issued from within an SDSF session.

Users in ISFPARMS Group 1 must have access to those SAF resources defined for ISFPARMS Group 1, Group 2, and Group 3.

Table 128. Profiles for Function of Group 1 — System Programmers

Function	Class	Resource Profile	Access
SDSF commands Command line commands (/) Operator authority to JESSPOOL Overtypeable fields Job classes WLM resources Scheduling environments Initiators Lines Network connections Network servers Nodes Offloaders (JES2 only) MAS or Jesplex members Readers System requests on SR Spool volumes z/OS UNIX processes Enclaves JES2 resources (JES2 only) Destination names APF panel DYNX panel ENQ panel LNK panel LPA panel PARM panel PAG panel SYM panel SYS panel	SDSF	ISF*.**	ALTER
All action characters, extended console, and server MODIFY command	OPERCMDS	*jes* .** MVS.** *server*.MODIFY.**	CONTROL
Printers and punches (local and remote)	WRITER	*jes* .**	ALTER
IBM Health Checker for z/OS checks	XFACILIT	HZS.**	CONTROL
Log stream used to record check history	LOGSTRM	*log-stream-name*	READ
MVS system logger	LOGSTRM	SYSPLEX.OPERLOG	READ

Group 2 — operators

Members of the ISFPARMS operators group have the same authority as Group 1, except for some restrictions.

Group 2 members cannot issue the TRACE, INPUT, and ABEND commands or look at everyone's output

Shown below are two sample GROUP statements, one that can be used without SAF to provide Group 2 authority, and another that can be used with SAF profiles to provide Group 2 authority:

Table 129. Sample GROUP statements, Group 2

Without SAF	With SAF
GROUP NAME(ISFOPER), TSOAUTH(JCL,OPER), AUTH(ALLOPER), CMDAUTH(ALL), CMDLEV(7), DSPAUTH(USERID,NOTIFY,AMSG), GPLEN(2), ACTION(ALL), ACTIONBAR(YES), APPC(ON), OWNER(NONE), CONFIRM(ON), CURSOR(ON), DATE(MMDDYYYY), DATESEP(/), LOG(OPERACT), ISYS(NONE), DADFLT(IN,OUT,TRANS,STC,TSU,JOB), VALTAB(TRTAB), UPCTAB(TRTAB2), LANG(ENGLISH), DISPLAY(OFF)	GROUP NAME(ISFOPER), ACTION(ALL), ACTIONBAR(YES), APPC(ON), CONFIRM(ON), CURSOR(ON), DATE(MMDDYYYY), DATESEP(/), LOG(OPERACT), DADFLT(IN,OUT,TRANS,STC,TSU,JOB), VALTAB(TRTAB), UPCTAB(TRTAB2), LANG(ENGLISH), DISPLAY(OFF)

To control membership in the group, which is done with TSOAUTH in the "Without SAF" case, use the profile shown in Table 130.

Table 130. Profiles for Membership in Group 2 — Operators

Function	Class	Resource Profile	Access
Membership	SDSF	GROUP.ISFOPER.*server-name*	READ

For guidance on providing security equivalent to that provided by the AUTH, CMDAUTH, CMDLEV, and DSPAUTH parameters shown in the first sample, see the generic profiles shown in Table 131.

The SAF profile ISFCMD.FILTER.OWNER protects the OWNER command, which cannot be protected with ISFPARMS. Refer also to the notes below the table.

Users in Group 2 must have access to those SAF resources defined for ISFPARMS Group 2 and Group 3.

Table 131. Profiles for Function of Group 2 — Operators

Function	Class	Resource Profile	Access
SDSF commands	SDSF	ISFCMD.DSP.** ISFCMD.ODSP.** ISFCMD.FILTER.ACTION ISFCMD.FILTER.DEST ISFCMD.FILTER.FINDLIM ISFCMD.FILTER.PREFIX ISFCMD.FILTER.SYSID ISFCMD.FILTER.SYSNAME ISFCMD.FILTER.OWNER ISFCMD.FILTER.RSYS (See note 1.)	READ
Command line commands (/)	SDSF	ISFOPER.SYSTEM	READ

Table 131. Profiles for Function of Group 2 — Operators (continued)

Function	Class	Resource Profile	Access
All MVS and JES commands generated by action characters and overtypeable fields	OPERCMDS	*jesx*.** MVS.** (See note 2.)	CONTROL
Destinations	SDSF	ISFOPER.ANYDEST.*jesx*	READ
Overtypeable fields	SDSF	ISFATTR.**	UPDATE
Job classes	SDSF	ISFJOBCL.**	CONTROL
Initiators	SDSF	ISFINIT.**	CONTROL
Lines	SDSF	ISFLINE.**	CONTROL
Network connections	SDSF	ISFAPPL.** ISFLINE.** ISFSOCK.**	CONTROL
Network servers	SDSF	ISFNS.**	CONTROL
Nodes	SDSF	ISFNODE.**	CONTROL
Offloaders (JES2 only)	SDSF	ISFSO.**	CONTROL
MAS or Jesplex members	SDSF	ISFMEMB.**	ALTER
Readers	SDSF	ISFRDR.**	CONTROL
Printers and punches (local and remote)	WRITER	*jesx*.**	ALTER
WLM resources	SDSF	ISFRES.**	CONTROL
Scheduling environments	SDSF	ISFSE.**	READ
System requests	SDSF	ISFSR.**	READ
z/OS UNIX processes	SDSF	ISFPROC.**	ALTER
Enclaves	SDSF	ISFENC.**	ALTER
APF panel	SDSF	ISFAPF.**	READ
DYNX panel	SDSF	ISFDYNX.**	READ
LNK panel	SDSF	ISFLNK.**	READ
LPA panel	SDSF	ISFLPA.**	READ
PAG panel	SDSF	ISFPAG.**	READ
PARM panel	SDSF	ISFPARM.**	READ
SYS panel	SDSF	ISFSYS.**	READ
ENQ panel	SDSF	ISFENQ.**	READ
SYM panel	SDSF	ISFSYM.**	READ
JES2 resources (JES2 only)	SDSF	ISFRM.**	CONTROL
DSPAUTH=AMSG	SDSF	ISFAUTH.DEST.**.DATASET.*dsname* (See note 3.)	READ
Checks	XFACILIT	HZS.**	CONTROL
Log stream used to record check history	LOGSTRM	*log-stream-name*	READ
Action characters and overtypeable fields for jobs and output groups	SDSF	ISFAUTH.DEST.* ISFAUTH.DEST.*.*	ALTER

Table 131. Profiles for Function of Group 2 — Operators (continued)

Function	Class	Resource Profile	Access
MVS system logger	LOGSTRM	SYSPLEX.OPERLOG	READ

Note:

1. As an alternative to the profiles listed, you can define these profiles
   ISFCMD.**
   ISFCMD.MAINT.**
   ISFCMD.FILTER.INPUT

   but not give Group 2 users access to ISFCMD.MAINT.** and ISFCMD.FILTER.INPUT so that they will not be authorized to use the TRACE, INPUT, and ABEND commands that these profiles protect. This is the approach taken in "Summary of profiles for group 1, group 2, and group 3" on page 321
2. These profiles protect all JES and MVS commands, even those that are not issued from within SDSF; some are outside the scope of SDSF.
3. You must define profiles for each system message data set. See the appropriate JES initialization and tuning guide for a list of message data set names. You can use the destination operator interface for global access to JESSPOOL resources. See "Destination operator authority" on page 240 for more information.

Group 3 — end users

Members of the ISFPARMS end users group can display some SDSF panels, issue a subset of action characters, and overtype some fields. They are also, by default, authorized to work with their own jobs and output.

Group 3 is more restrictive than Group 2. Group 3 members cannot:

- Display the device or system resource panels
- Issue the ACTION, DEST, FINDLIM, PREFIX, SYSID or RSYS commands
- Issue action characters or overtype fields that affect devices or system resources
- Display other people's jobs (that is, jobs with names that are different from their user ID)

Shown below are two sample GROUP statements, one that can be used without SAF to provide Group 3 authority, and another that can be used with SAF profiles to provide Group 3 authority:

Table 132. Sample GROUP statements, Group 3

Without SAF	With SAF
GROUP NAME(ISFUSER), TSOAUTH(JCL), AUTH(ALLUSER), CMDAUTH(USERID,NOTIFY), CMDLEV(2), AUPDT(10), DSPAUTH(USERID,NOTIFY), PREFIX(USERID), ACTION(11,12,USER), ACTIONBAR(YES), APPC(ON), CONFIRM(ON), CURSOR(ON), DATE(MMDDYYYY), DATESEP(/), DADFLT(IN,OUT,TRANS,STC,TSU,JOB), VALTAB(TRTAB), UPCTAB(TRTAB2), LANG(ENGLISH), DISPLAY(OFF)	GROUP NAME(ISFUSER), AUPDT(10), PREFIX(USERID), ACTION(11,12,USER), ACTIONBAR(YES), APPC(ON), CONFIRM(ON), CURSOR(ON), DATE(MMDDYYYY), DATESEP(/), DADFLT(IN,OUT,TRANS,STC,TSU,JOB), VALTAB(TRTAB), UPCTAB(TRTAB2), LANG(ENGLISH), DISPLAY(OFF)

To control membership in the group, which is done with TSOAUTH in the "Without SAF" case, use the profile shown in Table 133.

Table 133. Profile for Membership in Group 3 — End Users

Function	Class	Resource Profile	Access
Membership	SDSF	GROUP.ISFUSER.*server-name*	READ

Use the generic profiles shown in Table 134 as a guide to providing security equivalent to the AUTH, CMDAUTH, CMDLEV, and DSPAUTH parameters shown in the first sample.

Users are authorized to access their own jobs even if they are not permitted to access any JESSPOOL resources.

Users in Group 3 must have access to only those SAF resources defined for ISFPARMS Group 3.

Table 134. Profiles for Function of Group 3 — End Users

Function	JES	Class	Resource Profile	Access
SDSF commands	Both	SDSF	ISFCMD.DSP.**	READ

Table 134. Profiles for Function of Group 3 — End Users (continued)

Function	JES	Class	Resource Profile	Access
Action characters for CMDLEV=2	JES2	OPERCMDS	*jesx*.DISPLAY.BAT	READ
			jesx.DISPLAY.TSU	READ
			jesx.DISPLAY.STC	READ
			jesx.DISPLAY.INITIATOR	READ
			jesx.DISPLAY.DEV	READ
			jesx.MSEND.CMD	READ
			jesx.DISPLAY.BATOUT	READ
			jesx.DISPLAY.TSUOUT	READ
			jesx.DISPLAY.STCOUT	READ
			MVS.DISPLAY.WLM	READ
			jesx.MODIFY.BATOUT	UPDATE
			jesx.MODIFY.TSUOUT	UPDATE
			jesx.MODIFY.STCOUT	UPDATE
			MVS.CANCEL.ATX.*	UPDATE
			MVS.CANCEL.TSU.*	UPDATE
			jesx.CANCEL.BAT	UPDATE
			jesx.CANCEL.TSU	UPDATE
			jesx.CANCEL.STC	UPDATE
			jesx.CANCEL.BATOUT	UPDATE
			jesx.CANCEL.TSUOUT	UPDATE
			jesx.CANCEL.STCOUT	UPDATE
			jesx.CANCEL.DEV	UPDATE
			jesx.RELEASE.BATOUT	UPDATE
			jesx.RELEASE.STCOUT	UPDATE
			jesx.RELEASE.TSUOUT	UPDATE
			jesx.RESTART.DEV	UPDATE
			jesx.RESTART.BAT	UPDATE
			jesx.MODIFYHOLD.BAT	UPDATE
			jesx.MODIFYHOLD.STC	UPDATE
			jesx.MODIFYHOLD.TSU	UPDATE
			jesx.ROUTE.JOBOUT	UPDATE
	JES3 (see note)	OPERCMDS	*jesx*.CALL.DISPLAY	UPDATE
			jesx.DISPLAY.A	READ
			jesx.DISPLAY.S	READ
			jesx.DISPLAY.JOB	READ
			jesx.DISPLAY.JOBE	READ
			jesx.MODIFY.JOB	UPDATE
			jesx.MODIFY.U	UPDATE
			jesx.RESTART.DEV.*main*	CONTROL
			MVS.CANCEL.ATX.*	UPDATE
			MVS.CANCEL.TSU.*	UPDATE

Table 134. Profiles for Function of Group 3 — End Users (continued)

Function	JES	Class	Resource Profile	Access
Overtypeable fields for CMDLEV=2	JES2	SDSF	ISFATTR.JOB.PRTDEST	UPDATE
			ISFATTR.OUTDESC.*	UPDATE
			ISFATTR.OUTPUT.BURST	UPDATE
			ISFATTR.OUTPUT.CLASS	UPDATE
			ISFATTR.OUTPUT.DEST	UPDATE
			ISFATTR.OUTPUT.FCB	UPDATE
			ISFATTR.OUTPUT.FLASH	UPDATE
			ISFATTR.OUTPUT.FORMS	UPDATE
			ISFATTR.OUTPUT.PRMODE	UPDATE
			ISFATTR.OUTPUT.UCS	UPDATE
			ISFATTR.OUTPUT.WRITER	UPDATE
	JES3	SDSF	ISFATTR.JOB.PRTDEST	UPDATE
			ISFATTR.OUTDESC.*	UPDATE
			ISFATTR.OUTPUT.BURST	UPDATE
			ISFATTR.OUTPUT.CHARS	UPDATE
			ISFATTR.OUTPUT.CLASS	UIPDATE
			ISFATTR.OUTPUT.COPYCNT	UPDATE
			ISFATTR.OUTPUT.COPYMOD	UPDATE
			ISFATTR.OUTPUT.DEST	UPDATE
			ISFATTR.OUTPUT.FCB	UPDATE
			ISFATTR.OUTPUT.FLASH	UPDATE
			ISFATTR.OUTPUT.FORMS	UPDATE
			ISFATTR.OUTPUT.PRMODE	UPDATE
			ISFATTR.OUTPUT.UCS	UPDATE
			ISFATTR.OUTPUT.WRITER	UPDATE
Destinations	Both	SDSF	ISFOPER.ANYDEST.*jesx*	READ

Note: Because the JES3 environment requires that you use SAF for security, SDSF does not document command levels for those action characters. The OPERCMDS resources shown here for JES3 protect action characters that provide function similar to the action characters that are available with CMDLEV=2 in a JES2 environment.

Summary of profiles for group 1, group 2, and group 3

Table 135 shows the resources for the previous three groups: system programmers, operators, and end users.

Table 135. Profiles for Groups 1, 2, and 3

Resource Name	Class	Access Level	Group	JES
**	JESSPOOL			Both
ISF*.**	SDSF	ALTER	1	Both
ISFCMD.FILTER.INPUT	SDSF	READ	1	Both
ISFCMD.MAINT.**	SDSF	READ	1	Both
jesx.**	OPERCMDS	CONTROL	1,2	Both
MVS.**	OPERCMDS	CONTROL	1,2	Both
jesx.**	WRITER	ALTER	1,2	Both
ISFAUTH.DEST.*	SDSF	ALTER	1,2	Both
ISFAUTH.DEST.*.*	SDSF	ALTER	1,2	Both
HZS.**	XFACILIT	CONTROL	1,2	Both
ISFAPPL.**	SDSF	CONTROL	1,2	Both
ISFAPF.**	SDSF	READ	1,2	Both

Table 135. Profiles for Groups 1, 2, and 3 (continued)

Resource Name	Class	Access Level	Group	JES
ISFDYNX.**	SDSF	READ	1,2	Both
ISFENC.**	SDSF	ALTER	1,2	Both
ISFENQ.**	SDSF	READ	1,2	Both
ISFINIT.**	SDSF	CONTROL	1,2	Both
ISFJOBCL.**	SDSF	CONTROL	1,2	JES2
ISFLINE.**	SDSF	CONTROL	1,2	Both
ISFNS.**	SDSF	CONTROL	1,2	Both
ISFNODE.**	SDSF	CONTROL	1,2	Both
ISFSO.**	SDSF	CONTROL	1,2	JES2
ISFLNK.**	SDSF	READ	1,2	Both
ISFLPA.**	SDSF	READ	1,2	Both
ISFMEMB.**	SDSF	ALTER	1,2	Both
ISFPAG.**	SDSF	READ	1,2	Both
ISFPARM.**	SDSF	READ	1,2	Both
ISFPROC.**	SDSF	ALTER	1,2	Both
ISFRES.**	SDSF	CONTROL	1,2	Both
ISFRM.**	SDSF	CONTROL	1,2	JES2
ISFSE.**	SDSF	READ	1,2,3	Both
ISFSOCK.**	SDSF	CONTROL	1,2	Both
ISFSP.**	SDSF	CONTROL	1,2	Both
ISFSR.**	SDSF	READ	1,2	Both
ISFSYM.**	SDSF	READ	1,2	Both
ISFSYS.**	SDSF	READ	1,2	Both
ISFAUTH.DEST.**.DATASET.*dsname*	SDSF	READ	1,2	Both
ISFCMD.**	SDSF	READ	1,2	Both
ISFOPER.DEST.*jesx*	SDSF	READ	1,2	Both
ISFOPER.SYSTEM	SDSF	READ	1,2	Both
ISFATTR.**	SDSF	UPDATE	1,2	Both
log-stream-name	LOGSTRM	READ	1,2	Both
SYSPLEX.OPERLOG	LOGSTRM	READ	1,2	Both
jesx.DISPLAY.BAT	OPERCMDS	READ	1,2,3	JES2
jesx.DISPLAY.TSU	OPERCMDS	READ	1,2,3	JES2
jesx.DISPLAY.STC	OPERCMDS	READ	1,2,3	JES2
jesx.MSEND.CMD	OPERCMDS	READ	1,2,3	Both
jesx.DISPLAY.BATOUT	OPERCMDS	READ	1,2,3	JES2
jesx.DISPLAY.TSUOUT	OPERCMDS	READ	1,2,3	JES2
jesx.DISPLAY.STCOUT	OPERCMDS	READ	1,2,3	JES2
jesx.MODIFY.BATOUT	OPERCMDS	UPDATE	1,2,3	JES2
jesx.MODIFY.TSUOUT	OPERCMDS	UPDATE	1,2,3	JES2
jesx.MODIFY.STCOUT	OPERCMDS	UPDATE	1,2,3	JES2

Table 135. Profiles for Groups 1, 2, and 3 (continued)

Resource Name	Class	Access Level	Group	JES
MVS.CANCEL.ATX.*	OPERCMDS	UPDATE	1,2,3	Both
MVS.CANCEL.TSU.*	OPERCMDS	UPDATE	1,2,3	Both
MVS.MODIFY.WLM	OPERCMDS	UPDATE	1,2	Both
jesx.CANCEL.BAT	OPERCMDS	UPDATE	1,2,3	JES2
jesx.CANCEL.TSU	OPERCMDS	UPDATE	1,2,3	JES2
jesx.CANCEL.STC	OPERCMDS	UPDATE	1,2,3	JES2
jesx.RELEASE.BATOUT	OPERCMDS	UPDATE	1,2,3	JES2
jesx.RELEASE.STCOUT	OPERCMDS	UPDATE	1,2,3	JES2
jesx.RELEASE.TSUOUT	OPERCMDS	UPDATE	1,2,3	JES2
jesx.RESTART.DEV	OPERCMDS	UPDATE	1,2,3	JES2
jesx.RESTART.BAT	OPERCMDS	CONTROL	1,2,3	JES2
jesx.MODIFYHOLD.BAT	OPERCMDS	UPDATE	1,2,3	JES2
jesx.MODIFYHOLD.STC	OPERCMDS	UPDATE	1,2,3	JES2
jesx.MODIFYHOLD.TSU	OPERCMDS	UPDATE	1,2,3	JES2
jesx.ROUTE.JOBOUT	OPERCMDS	UPDATE	1,2,3	JES2
ISFCMD.DSP.**	SDSF	READ	1,2,3	Both
ISFOPER.ANYDEST.*jesx*	SDSF	READ	1,2,3	Both
ISFATTR.JOB.PRTDEST	SDSF	UPDATE	1,2,3	JES2
ISFATTR.OUTDESC.*	SDSF	UPDATE	1,2,3	Both
ISFATTR.OUTPUT.BURST	SDSF	UPDATE	1,2,3	Both
ISFATTR.OUTPUT.CHARS	SDSF	UPDATE	1,2,3	JES3
ISFATTR.OUTPUT.CLASS	SDSF	UPDATE	1,2,3	Both
ISFATTR.OUTPUT.COPYCNT	SDSF	UPDATE	1,2,3	Both
ISFATTR.OUTPUT.COPYMOD	SDSF	UPDATE	1,2,3	JES3
ISFATTR.OUTPUT.DEST	SDSF	UPDATE	1,2,3	Both
ISFATTR.OUTPUT.FCB	SDSF	UPDATE	1,2,3	Both
ISFATTR.OUTPUT.FLASH	SDSF	UPDATE	1,2,3	Both
ISFATTR.OUTPUT.FORMS	SDSF	UPDATE	1,2,3	Both
ISFATTR.OUTPUT.ODISP	SDSF	UPDATE	1,2,3	JES2
ISFATTR.OUTPUT.PRMODE	SDSF	UPDATE	1,2,3	Both
ISFATTR.OUTPUT.UCS	SDSF	UPDATE	1,2,3	Both
ISFATTR.OUTPUT.WRITER	SDSF	UPDATE	1,2,3	Both

Using the Security Conversion Assist

To help you convert from ISFPARMS to RACF for SDSF security, SDSF provides the Security Conversion Assist. This generates sample RACF commands for most of the ISFGRP/GROUP parameters of your ISFPARMS. For a complete list of supported ISFGRP/GROUP parameters, see "Supported ISFGRP/GROUP parameters" on page 327.

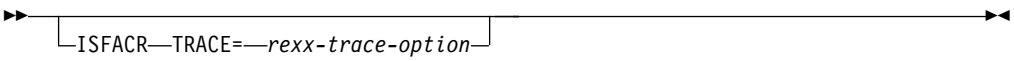
The Security Conversion Assist is intended to give you a headstart on the conversion to RACF security. It does not provide a complete, automated conversion. Use the generated RACF commands as a sample to help you understand RACF security, or review, modify and issue them to provide RACF security.

The Security Conversion Assist requires RACF on the system on which you run it. You must have TSO authority of JCL, ACCT and OPER.

Steps in using the Security Conversion Assist

About this task

Invoke the Security Conversion Assist by issuing the ISFACR command from any ISPF command line. The syntax of the command is as follows:



The ISFACR command displays a menu of steps that you select in sequence. The steps are:

1. **Define a profile**. This step lets you specify such things as the ISFPARMS and RACF commands data sets, the CLIST library, and RACF group names.
2. **Convert ISFPARMS to profile descriptions.** This step analyzes the ISFPARMS source file and:
   a. If ISFPARMS is in statement format, creates a copy of it that is in assembler macro format in data set *userid*.PARMI.SDSF.
   b. Produces an intermediate output file for profile descriptions. The file is named in your profile. The profile descriptions contain, in plain text, the RACF profiles that are produced by Security Conversion Assist. To be sure the required profile descriptions are present, check the file that is created against the tables in Appendix C, "SDSF resource names for SAF security," on page 577. Profile descriptions are explained in detail in "Profile descriptions" on page 325.

   This step also checks the RACF database for the presence of the user IDs that are found in name tables (ISFNTBL/NTBL) in the ISFPARMS.

   A pop-up lets you run this step in the foreground or in batch.
3. **Review profile descriptions.** This step allows you to study and modify the profile descriptions, to make sure that the proper RACF profiles are created in a subsequent step. Some profile descriptions may be marked with the word `CHANGE`; you may correct these in this step or wait and correct the generated RACF commands. Refer to"Profile descriptions" on page 325 for more information.
4. **Convert profile descriptions to RACF commands.** This step translates the profile descriptions into RACF commands and writes them to the data set specified in your profile. For each command that is in the process of being created, the procedure checks if the profile is already in the RACF database. If so, no command is generated.

   The Security Conversion Assist allows you to select a specific class for which RACF commands are to be generated. Specifying ALL causes commands for all classes to be generated.

   A pop-up lets you run this conversion in the foreground or in batch.

5. **Review RACF commands.** Use this step to review the generated commands. See "RACF commands" on page 326 for a discussion of what to look for.

   You may want to simply use the generated commands as a sample to help you understand RACF security. Or, after carefully reviewing and modifying the commands, you may want to issue them to provide RACF security.

If you run the Security Conversion Assist multiple times with the same prefix for generated group names, it creates commands to delete groups defined with the previous run. You specify the prefix in the profile, which is option 1 of the menu.

Required data sets

To use the Security Conversion Assist, you will need the following data sets:

Table 136. Data Sets Required by the Security Conversion Assist

Data Set	Characteristics
For an assembler macro version of the source ISFPARMS, if they are in statement format; *userid*.PARMI.SDSF	Sequential, record length at least 80
For profile descriptions, by default *userid*.IN.SDSF	Sequential, record length at least 80
For generated RACF commands, by default *userid*.SDSFRACF.CLIST	Sequential file, record length at least 133

Profile descriptions

The generated profile descriptions are in order by class, as follows: JESSPOOL, OPERCMDS, SDSF, WRITER, GSDSF, GROUP, GLOBAL, and RACFVARS. To help you identify what the profile description corresponds to, each profile description includes descriptive text on the third line. For example, a profile description for an SDSF command would show `SDSF_COMMANDS` on the third line.

The parts of a profile description are shown in Table 137.

Table 137. Profile Description Example

Statement	Description
`Class= SDSF`	Class name
`ISFCMD.ODSP.SYSLOG.JES2`	Profile name
`SDSF_COMMANDS`	Descriptive text
`ISF`	Owner
`NONE`	UACC
`NOWARNING`	WARNING|NOWARNING
`ALL`	AUDIT
`MEMBERS`	Heading for members
`---`	Entry for group class or for general resource grouping class
`ACCESS LIST`	Heading for access list
`* READ`	Entry in access list with access level
`CONDITIONAL ACCESS LIST`	Heading for conditional access list

Note that the headings for members, access list, and conditional access list are shown even when there are no entries.

The access list information for some profile descriptions will have the word `CHANGE` instead of a user ID or group ID. You can update this access list information with the correct user ID or group ID here, or wait and make the changes in the RACF

commands that are generated from the profile descriptions. Figure 12 shows an example of a profile description that needs to be changed. You would replace the word CHANGE on the next-to-last line with a user or group ID.

```
Class= JESSPOOL
*.+MASTER+.*.*.*.SYSLOG
SYSLOG
MAQ
READ
NOWARNING
ALL
MEMBERS
ACCESS LIST
CHANGE ALTER
CONDITIONAL ACCESS LIST
```

Figure 12. Example of a Profile Description to Change

The profile descriptions that may require a change to the access list information are shown in Table 138.

Table 138. Profile Descriptions That May Require a Change to the Access List

Class	Profile
JESSPOOL	*.+MASTER+.*.*.*.SYSLOG *.+MASTER+.SYSLOG.* *.*.*.*.JESTRACE *.*.JESNEWS.*.D*.JESNEWS *.*.SYSLOG.*
OPERCMDS	*.UPDATE.JESNEWS JES2.* JES3.* MVS.* MVS.START.STC.*
GLOBAL	Class= GLOBAL JESSPOOL
RACFVARS	Class= RACFVARS &RACLNDE

RACF commands

When reviewing the generated RACF commands, you should look for:

- Any access list entries that are marked with the word `CHANGE`. These must be updated. For example, the `CHANGE` in the following PERMIT command needs to be changed to a user ID or group name.

  ```
  PERMIT *.+MASTER+.*.*.*.SYSLOG CLASS(JESSPOOL) ID(CHANGE) ACCESS(ALTER)
  ```

- Commands that are commented out. All destructive commands, such as RDELETE, REMOVE and CLASSACT, are commented out. To issue them, you need to remove the comment.
- Any commands that have an inappropriate scope. For example, you may want to change an * to a node ID to reduce the scope.
- Impact of use of the Generic Owner facility. It is assumed that you will use the Generic Owner facility. This facility has a great impact on the ownership of RACF profiles, in particular, the JESSPOOL profiles. If you use this facility it will be impossible to create a more specific profile with a different owner. Remove the associated RACF command when you do not plan to use Generic Owner.

- Make sure that the RACF classes are defined GENERIC; if not, none of the profiles will work. When one of the classes is not defined GENERIC, first make sure that there is no profile left in that class, then define the class to be generic. Subsequently run appropriate commands to get back the profiles that were previously defined in that RACF class.
- The &RACUID entry in the GLOBAL profile. You must change the ampersand (&) to a double ampersand (i.e. `&&RACUID`). &RACUID will be treated as a variable. Therefore, before processing the commands, you must change the & to an &&.

Supported ISFGRP/GROUP parameters

The following GROUP statement or ISFGRP macro parameters are supported by the Security Conversion Assist:

Table 139. GROUP/ISFGRP Parameters Supported

Parameter	Notes
TSOAUTH	The authorities must be defined in RACF TSOAUTH class profiles instead of in SYS1.UADS. This parameter is only partially converted to RACF. It will be used for the ISFPARMS file that stays in your system after the conversion. A RACF group entry will be generated that uses the name specified for the group in ISFPARMS (either with the value for the GROUP NAME keyword or the label on the ISFGRP macro); if no name was specified, the conversion assist generates a name using the prefix defined in the conversion assist profile followed by a T and a sequence number. When a TSOAUTH profile is defined with UACC(READ), the results of the conversion assist are ambiguous. The profile description will contain an * as an ID entry in a group definition. When the profile description is converted to RACF commands, the * is converted into RACF commands that change the UACC for the profile involved.
AUTH	
CMDAUTH	Only values of ALL and GROUP are supported. For ALL, profiles in the SDSF class are generated. For GROUP, profiles in the JESSPOOL class are generated. When ALL is used and XCMD is also defined for this group, a direct translation to RACF profiles is not possible because of the way SDSF checks profiles. Once the authority to give all commands is recognized, no further checks are done by SDSF.
CMDLEV	The levels are used to generate corresponding RACF profiles in the GSDSF class.
DSPAUTH	Only values of ALL and GROUP are supported. For ALL, profiles in the SDSF class are generated. For GROUP, profiles in the JESSPOOL class are generated. When ALL is used and XDSP is also defined for this group, a direct translation to RACF profiles is not possible because of the way SDSF checks on profiles. Once the authority to display all output is recognized, no further checks are done by SDSF.
GPREF	The Security Conversion Assist looks for this parameter when GROUP is specified in the CMDAUTH and DSPAUTH parameter. The appropriate RACF profiles in the JESSPOOL class are generated.
ICMD	
IDSP	
IUID	The Security Conversion Assist searches the ISFPARMS source file to find all the group names defined in the ISFNTBL macros. Because it is likely that these group names already exist in your RACF database as RACF GROUP names, group names are defined that consist of the prefix specified in your profile followed by a sequence number. The IUID value is copied into the corresponding RACF profile in the Descriptive text field. This is an important field when it comes to do cross checking later.

Table 139. GROUP/ISFGRP Parameters Supported (continued)

Parameter	Notes
XCMD	When authorization is given using the ISFOPER... and ISFAUTH... profiles in the SDSF class, you may find that XCMD is not converted correctly.
XDSP	When authorization is given using the ISFOPER... and ISFAUTH... profiles in the SDSF class, you may find that XDSP is not converted correctly.
XDSPD	

Usage notes:

- The RACF profile for the OWNER command, ISFCMD.FILTER.OWNER, is defined as UACC(READ), because authority to the OWNER command is not controlled by ISFPARMS.
- The RACF profiles produced do not conform to the Enhanced Generics Standard.
- On a qualifier with 'nodeid', the Security Conversion Assist uses an '*' instead.
- The Security Conversion Assist does not convert destination security (e.g. IDEST lists) and does not handle output groups.

Diagnosing security

You can use the security trace function to understand and diagnose SDSF security provided by either ISFPARMS or SAF. The security trace function provides simple messages that are included in the ULOG or issued as write-to-programmer messages. For more information, refer to “Diagnosing security” on page 18.

Chapter 9. Using installation exit routines

Programming Interface Information

This topic describes how to use an installation exit routine to customize your security authorization strategy.

Note: SDSF's support for installation exits can change. With each new release of SDSF, you should review your exit routines to ensure that they still function correctly, and make changes as necessary. For the most common uses, SDSF's installation exits have been superseded by custom properties in ISFPARMS, which are significantly easier to define and maintain. For more information, see "Customized properties (PROPLIST)" on page 91.

Installation exit routines

You can write installation exit routines for the set of installation exit points provided by SDSF. These routines can supplement the authorization you established with ISFPARMS and the SAF security interface. Your installation exit routines supply customized authorization processing for your installation and return to SDSF their authorization decisions.

The PROPLIST and PROPERTY statements in ISFPARMS provide an alternative to some of the customization available through the exit routines. For more information, see "Customized properties (PROPLIST)" on page 91.

Using the ISFUSER module

You add your installation exit routines to the ISFUSER module supplied by SDSF in member ISFUSER of the ISF.SISFSRC data set. As supplied, module ISFUSER performs no authorization functions and is always present, whether you add installation exit routines or not.

Instructions for the use of module ISFUSER are contained in the module, which indicates where you should add the code to be used for each exit point. The module also has information about the function codes and registers used in the exit point interface. Note that the pre-SAF exit will be the first exit point.

ISFUSER is called and must return in 31-bit mode. To install the ISFUSER module after adding installation exit routines, perform SMP RECEIVE and APPLY.

ISFUPRM macro

The installation exit routine can use parameters supplied in the ISFUPRM macro, which maps the user parameter area. A pointer to the user parameter area is passed to ISFUSER upon entry. The user parameter area contains such information as:

- User ID, logon procedure name, and terminal name
- User authority based on ISFGRP macro or GROUP specifications
- Prefix and group prefix information defined in ISFGRP macros or GROUP statements
- Pointers to include and exclude lists defined in ISFGRP macros or GROUP statements

© Copyright IBM Corp. 1997, 2016

- Pointers to the primary and alternate field lists defined in ISFFLD macros or FLD statements
- Pointers to destination name tables and user selected node/remote names defined in ISFNTBL macros or NTBL statements
- Trace table information
- Job information

Installation exit points

The installation exit points within SDSF link to the ISFUSER module at entry point ISFUSER. SDSF provides the following exit points for installation routines to customize authorization:

Exit Point	Use to Control
Initialization	Who can use SDSF
Command Authority	Which commands users can issue
SYSOUT Display Authority	For which jobs users can display output
SDSF termination	Termination processing
Pre-SAF	How the SAF authorization decision is to be made
Post-SAF	Accept or ignore result of SAF authorization
SAF indeterminate	Action for SAF indeterminate responses
Table build	What is displayed on tabular panels

These exit points are described in detail in the remainder of this topic. The descriptions include input, output (if any), and return codes.

SAF considerations for exit points

For information about the SAF resources used for SDSF security, see Chapter 7, "Protecting SDSF functions," on page 211.

The Command Authority and SYSOUT Display Authority exits are not taken when SAF makes an authorization decision. Because the JES3 environment requires SAF security, these exits do not apply in the JES3 environment. Instead, use the pre-SAF exit.

The SYSOUT Display Authority exit routine is not invoked for a user who has authority to access the SDSF class resource ISFOPER.DEST.*jesx* (users who have destination operator authority).

If the installation wants to maintain the functions of these installation exit routines while using SAF for security decisions, then the exit routine code should be moved to one of the other exit points available with SDSF.

Use the SDSF exits for SAF calls made by SDSF. SAF calls may be made by other components; for example, JES2 makes a SAF call for a resource in the JESSPOOL class when you browse a data set. You cannot affect SAF calls made by other components with the SDSF exits.

Initialization exit point

This exit is taken during SDSF initialization after all of the authorization parameters from ISFPARMS and the ISPF profile have been moved into the user parameter area. The initialization exit routine can control authorization to use SDSF.

The initialization exit routine also controls use of the table build exits and the source of information for the Display Active Users panel.

The initialization exit point may not be the first exit called by SDSF. In particular, security related exits such as pre-SAF and post-SAF are called prior to the initialization exit point.

If you want table build exits taken, your initialization exit routine must set exit flags for each tabular panel. When you set an exit flag to B'1', the table build exit is taken. See "Table build exit point" on page 337 for more information.

In addition, your initialization exit can set the following to B'1' to perform other functions:

Field	Description
UPRSFLAG.UPRNORMF	Derive information for the DA panel directly from MVS control blocks rather than from RMF
UPRSFLAG.UPRNORMS	Disable use of sysplex DA
UPRSFLAG.UPRSNOCS	Allows an EMCS console to be shared if it is already active. See UPRSFLG5.UPRS5CSX for controlling sharing of the EMCS console across address spaces.
UPRCKLIM	Sets the default maximum number of instances for each health check that will be read from the logstream for the CKH panel. Users can override this with the SET CKLIM command.
UPROFLG1.UPRO1DYZ	Specifies that the columns related to a zAAP are shown on the DA panel only if a zAAP is defined in the set of systems being shown, and the columns related to a zIIP are shown on the DA panel only if a zIIP is defined in the set of systems being shown.
UPROFLG1.UPRO1GHO	Append a generic pattern-matching character to the job specified with the H command, unless the prefix already ends with a generic character or is already the maximum length (8 characters). For example, if the user enters H GREER, this setting would result in a prefix of H GREER*.
UPROFLG1.UPRO1GPF	Append a generic pattern-matching character to the prefix specified with the PREFIX command, unless the prefix already ends with a generic character or is already the maximum length (8 characters). For example, if the user enters PREFIX JONES, this would result in a prefix of JONES*.
UPROFLG1.UPRO1GST	Append a generic pattern-matching character to the job specified with the ST command, unless the prefix already ends with a generic character or is already the maximum length (8 characters). For example, if the user enters ST GREER, this setting would result in a prefix of ST GREER*.

Field	Description
UPROFLG1.UPRO1LNF	Specifies the SAF logging option to use when a job's data sets are browsed from an SDSF panel, with the exceptions of the JDS and OD panels. If the value is TRUE, the SAF logging setting is LOG=NOFAIL (rather than the default, LOG=ASIS).
UPROFLG1.UPRO1SFW	Controls issuing a warning message when a SAF no-decision is converted to a failure
UPROFLG2.UPRO2DNL	Affects normalization of the CPU% column on the DA panel. If the value is TRUE, the CPU% column is normalized using the LPAR value for CPU busy for the system. If the value is FALSE, the CPU% column is normalized with the MVS value for CPU busy for the system. The LPAR value takes into account several states related to PR/SM. The LPAR value requires RMF. If the LPAR value is not available, SDSF uses the MVS value to normalize the CPU% colunm. FALSE is the default.
UPROFLG2.UPRO2DU8	Controls how device names are formatted on the PUN panel. If the value is TRUE, the device names are shown in a shortened format. Otherwise, the name is shown with dots between subtypes.
UPROFLG2.UPRO2DR8	Controls how device names are formatted on the RDR panel. If the value is TRUE, the device names are shown in a shortened format. Otherwise, the name is shown with dots between subtypes.
UPROFLG2.UPRO2NMD	Disables modification of the console name when console activation fails due to the console being in use. A value of TRUE disables the function and a value of FALSE enables it. FALSE is the default.
UPROFLG2.UPRO2NPS	Disables point-and-shoot fields such as column titles.
UPROFLG3.UPRO3JPC	Controls the scope of the CK panel.
UPROFLG3.UPRO3JPD	Controls the scope of the DA panel.
UPROFLG3.UPRO3JPE	Controls the scope of the ENC panel.
UPROFLG3.UPRO3JPP	Controls the scope of the PS panel.
UPROFLG3.UPRO3NOD	Controls whether duplicate SYSOUT data sets are included when you browse or print a job.
UPROFLG4.UPRO4JSM	Controls scope of the SYM panel.
UPRSFLG3.UPRS3MEM	Restrict user access to jobs that have run or will run on another member in a MAS configuration
UPRSFLG3.UPRS3NOF	Bypass all filtering for DA, H, I, O and ST, including include and exclude lists set in ISFPARMS
UPRSFLG3.UPRS3SWP	When browsing job data sets, do not gather data from in-core buffers if the job is swapped out. This is ignored for systems other than the one you are logged onto.
UPRSFLG4.UPRS4NCM	Disable use of communications between servers in a server group
UPROFLG5.UPRO5JEN	Controls scope of the ENQ panel.

Field	Description
UPRSFLG5.UPRS5CSX	Allow sharing of an EMCS console if it is in use but was activated in a different address space than the user. Console sharing means that commands will be issued using that console, and any responses will be directed to the ULOG for the task that has activated the console. The option to allowing sharing is effective only when console sharing is permitted. See UPRSFLAG.UPRSNOCS.
UPRSFLG5.UPRS5DSI	Specifies that the system SIO rate is included on the title line of the DA panel, but the system zAAP use is not.
UPRSFLG6.UPRS6JS3	ON if SDSF is running under JES3. [1]
UPRS6FSY	Controls the use of system symbols with filtering.
UPRSSNME	Contains the JES subsystem name for the JES that SDSF is running under. [1].
UPXCONSF	Names the list of suffixes to use when modifying the console name when the console activation fails due to the console being in use. The default is $#@12345.

Note:

1. SDSF invokes other exit points prior to the initialization exit point (such as the pre-SAF and post-SAF calls). Fields listed for the initialization exit point are not available for exit points that are invoked earlier.

Input

- Function code (X'00') in register 0
- Address of user parameters (ISFUPRM) in register 1

Return codes

00 Allows the user to use SDSF.

Nonzero

The user is not authorized to use SDSF. Message ISF024I is issued.

Command authority exit point

This exit is taken only when the SAF indeterminate exit routine or the pre-SAF exit routine sets the return code to X'04'. It is not taken when SAF makes an authorization decision. Because the JES3 environment requires SAF security, this exit does not apply in the JES3 environment. Instead, use the pre-SAF exit.

This exit is taken prior to SDSF issuing a command on behalf of the user. The command to be issued could be in response to a field being overtyped on a tabular panel, an action character being entered, or a command entered with the / command.

Input

- Function code (X'04') in register 0
- Address of user parameters (ISFUPRM) in register 1
- Authorization attribute in field UPRARETC, as follows:
  - X'10'— User is not authorized to the job.
  - X'14'— User is not authorized to the command.
  - X'24'— User is not authorized to the printer.
  - X'28'— User is not authorized to the initiator.
  - X'2C'— User is not authorized to the system.
  - X'30'— User is not authorized to the type.
  - X'34'— User is not authorized to the device.

X'38'— User is not authorized to the node.
X'3C'— User is not authorized to the scheduling environment.
X'40'— User is not authorized to the WLM resource.
X'44'— User is not authorized to the job class.
X'48'— User is not authorized to the spool volume.
X'4C'— User is not authorized to the enclave.
X'50'— User is not authorized to the process.
X'54'— User is not authorized to the JES2 resource.
X'58'— User is not authorized to the IBM Health Checker for z/OS check.
X'60'— User is not authorized to LPA resource.
X'64'— User not authorized to LNK resource .
X'68'— User is not authorized to APF resource.
X'6c'— User is not authorized to parmlib resource.
X'70'— User is not authorized to page resource.
X'74'— User is not authorized to system resource.
X'78'— User is not authorized to SRCH resource.
X'7c'— User is not authorized to SYM resource.
X'80'— User is not authorized to ENQ resource.

Output

Authorization attribute in field UPRARETC, as described above. If this field is nonzero on entry to the exit routine, the exit routine must zero this field and set the return code to X'00' to allow the command to be issued.

Return codes

00 Allows the command to be issued.

Nonzero

The user is not authorized to issue the command.

SYSOUT display authority exit point

This exit is taken only when the SAF indeterminate exit routine or the pre-SAF exit routine sets the return code to X'04'. It is not taken when SAF makes an authorization decision. Because the JES3 environment requires SAF security, this exit does not apply in the JES3 environment. Instead, use the pre-SAF exit.

This exit is taken prior to SDSF displaying a SYSOUT data set. It is taken as a result of an S, V, or X action character being used.

If the display of SYSOUT data sets with DSIDs less than 101 is restricted by AMDEST, AMSG, or GRPMSG values in the DSPAUTH parameter of an ISFGRP macro or GROUP statement, a return code of zero in field UPRARETC overrides this restriction but does not authorize the user to display SYSOUT with DSIDs of 101 or greater.

Input

- Function code (X'08') in register 0
- Address of user parameters (ISFUPRM) in register 1
- Authorization attribute in field UPRARETC, as follows:
  X'00'— User is authorized to display the data set.
  X'10'— User is not authorized to the job.

Output

If this field is nonzero on entry to the exit routine, the exit routine must zero this field and set the return code to X'00' to allow the user to display the data.

Return codes

00 Allows the data to be displayed.

Nonzero
The user is not authorized to see the data.

SDSF termination exit point

This exit is taken during SDSF termination prior to any data sets being closed or storage being freed.

Input

- Function code (X'0C') in register 0
- Address of user parameters (ISFUPRM) in register 1

Return codes

No return codes are expected from this exit.

Pre-SAF exit point

This exit is taken prior to the call to SAF and prior to the initialization exit. It allows the installation to control how the authorization decision is to be made. It is taken only for SAF calls done by SDSF. In addition to the SAF calls done by SDSF, SAF calls may be made by other components.

Input

- Function code (X'10') in register 0
- Address of user parameters (ISFUPRM) in register 1
- SAF class name being checked is in field UPRCLASS
- Resource name area is pointed to by UPRRSCN. The first halfword is the length of the resource name which is followed by the resource name.
- Authorization required for the resource is in field UPRATTR. The values are:
  X'02'— READ
  X'04'— UPDATE
  X'08'— CONTROL
  X'80'— ALTER

Return codes

00 Use SAF to make the authorization decision. This is the default return code.

04 Bypass the SAF call and fall back to ISFPARMS for the authorization decision. The command authority and SYSOUT display authority exits will be invoked.

08 Bypass the SAF call and fail the request. The user is not authorized to the resource.

Other Same as return code 08, but IBM recommends that the return code be explicitly set to 08 to indicate that the request is to be failed.

Post-SAF exit point

The post-SAF exit point is taken after SDSF has invoked SAF to allow the installation to control how the authorization decision is to be made. The installation can use the exit to perform additional authorization checking or auditing, accept or ignore the SAF decision, or indicate that ISFPARMS is to be used for authorization.

If the exit indicates that the decision is to be ignored or failed, SAF logging of the request will already have been done.

The exit is taken only for SAF calls done by SDSF. In addition to the SAF calls done by SDSF, SAF calls may be made by other components.

Input

- Function code (X'1C') in register 0.

Address of user parameters (ISFUPRM) in general register 1.

- UPRSAFRC contains the original SAF return code.
- UPRSAFPL contains the address of the RACROUTE parameter list used for the SAF call.
- UPRCLASS contains the SAF class name that was checked.
- UPRRSCN contains the address of the resource name area. The resource name area consists of a halfword length containing the length of the resource name, followed by the resource name.
- UPRATTR contains the authorization attribute required for the resource, as follows:
  X'02'Read
  X'04'Update
  X'08'Control
  X'80'Alter

Return codes

00 Accept the SAF decision and process according to the original SAF return code. If the original SAF return code was 04, the SDSF SAF indeterminate exit will be called.

04 Ignore the SAF decision and process as if SAF set a return code of 04. The SDSF SAF indeterminate exit will be called.

08 Ignore the SAF decision. The user is not authorized to the resource.

12 Ignore the SAF decision. The user is authorized to the resource.

Other Same as return code 08, but IBM recommends that 08 be explicitly set to indicate that the SAF return code is to be ignored.

SAF indeterminate exit point

This exit is taken when SAF cannot make an authorization decision and returns an indeterminate response. This can occur when the requested class is not active or when no profile is defined.

SDSF does not take this exit when security checking is being performed for operator destination access to a resource. This is because further SAF checking will be performed against the JESSPOOL resource itself, even if SAF could not determine if the user had operator destination access to the resource. This exit is not suppressed when the SAF call is made to the actual JESSPOOL resource.

Input

- Function code (X'14') in register 0.
- Address of user parameters (ISFUPRM) in register 1.
- Class name is in field UPRCLASS.

- Resource name area is pointed to by UPRRSCN. The first halfword is the length of the resource name which is followed by the resource name.
- Authorization required for the resource is in field UPRATTR. The values are:
  X'02'— READ
  X'04'— UPDATE
  X'08'— CONTROL
  X'80'— ALTER

Return codes

00 User is allowed to access the resource.

04 Use ISFPARMS to determine authorization and allow access when ISFPARMS has no corresponding protection scheme. The SAF call is bypassed. This is the default return code if the user does not change the installation exit routine.

Note: This does not apply to the SERVER option on the SDSF command, which is protected by the ISFCMD.OPT.SERVER resource, as described in "SDSF server" on page 300.

08 User is not allowed to access the resource.

Table build exit point

The table build exit point is used during the table display build to determine if a row should be displayed. It is taken for every row to be included in a tabular panel. It allows the installation to suppress a row from being displayed on a tabular panel. The DEST, OWNER and PREFIX filters are applied before the user exit, but filters set by the FILTER command are applied after the exit.

The table build exit is taken only if enabled through an initialization exit routine. For each tabular display, a flag, defined in bytes UPRUXFLG, UPRXFLG2, UPRXFLG3, UPRXFLG4, UPRXFLG5, and UPRXFLG6 specifies whether the table build exit is to be taken. The initialization exit routine must turn on the corresponding flag for each table row exit that is to be used.

When the table row exit is taken, a display-dependent work area, which describes the row to be displayed, is passed to the table build exit routine.

Input

- Function code (X'18') in register 0.
- Address of user parameters (ISFUPRM) in register 1.
- Address of a display-dependent work area that describes the row to be displayed in field UPRUXWA@. The work area is mapped by the UXTEBPRM DSECT (which is expanded as part of the ISFUPRM macro).

Return codes

00 Row is displayed. This is the default return code.

Nonzero
Row is not displayed.

End of Programming Interface Information

Chapter 10. Installation and configuration considerations

This topic discusses special considerations for JES and WebSphere MQ.

JES2 considerations

DESTDEF considerations

The JES2 DESTDEF initialization statement controls how destination names are displayed and controlled. The values of DESTDEF control how SDSF processes destinations.

If DESTDEF SHOWUSER=WITHLOCAL is coded, then destinations of the form *local-node.userid*, which are otherwise displayed as *userid*, are displayed as LOCAL.*userid*.

If you changed the field list definitions for the PR display and you coded a default width for the destination column in the ISFFLD macro or FLD statement (that is a width of 'D'), then the length of the column will be 18 rather than 8 to accommodate the longer destination name that will be displayed.

SDSF server group with multiple levels of JES2

The servers in a server group may process JES2 subsystems that are at different maintenance levels. For this to function properly, you should:

1. For each JES2, assemble SDSF against the correct JES2 level, that is, the level of the JES2 it is running on. Ensure that the reassembled libraries can be accessed by the server proc.
2. Add a steplib to the server JCL referencing the SISFLOAD, SISFMOD1 and SISFLPA that are appropriate for that level of JES2. The SISFLOAD, SISFMOD1 and SISFLPA can be created with ISFASMP as described in “Sample SMP/E UCLIN jobs for automatic reassembly” on page 351. Be sure that any data sets in the server steplib are authorized.

SDSF with a secondary JES2 subsystem

SDSF can process data from a secondary JES2 subsystem. This allows you to use SDSF for JES subsystems that you may be testing.

All SDSF functions are available when processing a secondary JES, with the following restrictions:

- The LOG command displays all SYSLOG data sets on spool. Since MVS allocates the SYSLOG data sets using the primary JES, there may be no SYSLOG data sets on the secondary spool. This may lead to no data being shown when the LOG display is accessed. However, if OPERLOG is active, the LOG command will display the log data from the OPERLOG regardless of the JES being processed.
- The C, O, and P action characters, and the C and DEST overtypes will not be available on the Job Data Set (JDS) display.

© Copyright IBM Corp. 1997, 2016

Installation considerations

When SDSF was installed, it was assembled using macros for your JES. If your secondary JES level is different than the level used during the install, you will need to reinstall SDSF using the macros that correspond to the secondary JES. To do this:

1. Create a separate SMP/E target zone for each secondary JES and its associated SDSF. You should also create different SDSF data set names to distinguish the SDSF levels on your system.
2. Run an SMP/E APPLY to create the SDSF load libraries that correspond to the secondary JES.
3. Add a steplib to your TSO logon procedure that references the new SISFLOAD, SISFMOD1 and SISFLPA that you created in the step above.

Once the reinstall is complete, or if your JES levels are the same:

- When using ISFPARMS in statement format:
  1. Create a new ISFPRM*xx* member and change the JESNAME parameter of the OPTIONS statement to reference the secondary JES
  2. Start a second server using a different server name using the new member as input. For example, if you created member ISFPRMJA for the secondary JES, you could start a server using the `S SDSF.SDSFA,M=JA` command.
  3. Access the second server using the SERVER option of the SDSF command. For example, `SDSF SERVER(SDSFA)` would cause SDSF to connect to the SDSFA server.
- When using ISFPARMS in assembler macro format:
  1. Create a new ISFPARMS and change the JESNAME parameter of ISFPMAC to reference your secondary JES
  2. Assemble and link-edit the ISFPARMS into a new load library
  3. Add the library as a steplib to your TSO logon procedure. Note that this library must be APF authorized.

JES3 considerations

SDSF may be invoked on either a local or global processor.

SDSF retrieves information about the JES being processed, including the JES3 global system name, during initialization. As a result, if a JES3 DSI is done to move the global system, SDSF users must re-access SDSF so that initialization can take place.

ISFPARMS must be in the statement format (parmlib member ISFPRM*xx*) rather than the assembler macro format. ISFPRM*xx* is processed by the SDSF server, which must be started. If the SDSF server is not started, SDSF uses the default assembler macro ISFPARMS shipped with SDSF, regardless of any other modified ISFPARMS you might have.

SDSF security must be provided by SAF rather than ISFPARMS.

For more information, refer to the red paper, *Using SDSF in a JES3 Environment*, REDP-4531-00.

For new SDSF function to be available, both the processor from which SDSF is invoked and the JES3 global processor must have SDSF at the level that provides the new function.

Getting started running SDSF in the JES3 environment

The following tasks are associated with running SDSF in a JES3 environment.

Task	Reference Information
Prepare ISFPRM*xx*. If you are beginning with an ISFPARMS in assembler macro format, convert it to statement format.	"ISFPARMS in the JES3 environment" "Converting ISFPARMS assembler macros to statements" on page 17
Start the SDSF server.	Chapter 3, "Using the SDSF server," on page 107
Implement SAF security.	"SAF in the JES3 environment"
If you have SDSF user exits, review and revise them as necessary for the JES3 environment. If you have implemented the command authority or SYSOUT display authority user exits, reimplement them in the pre-SAF user exit.	"Installation exit points" on page 330

ISFPARMS in the JES3 environment

The statements in parmlib member ISFPRM*xx* are largely the same for JES2 and JES3 environments. If you have a mixed JES2 and JES3 environment, you can use a single ISFPRM*xx* parmlib member. When processing ISFPRM*xx*, SDSF ignores statements and keywords that do not apply to the current JES type, such as:

- Any keywords that define security. Security must be provided only through SAF.
- The SERVERGROUP, SERVER and COMM statements, which are used for sysplex support of the device panels and browsing SYSLOG. The related TIMEOUT parameter of the OPTIONS statement is also ignored.
- Statements and keywords that define field lists for panels that are not supported in the JES3 environment.

A JES3NAME parameter of the OPTIONS statement allows you to specify the JES3 that is to be processed. The syntax is as follows:

```
JES3NAME (*) | (JES-name)
```

JES-name, the name of the JES3 subsystem, can be 1 to 8 characters. The default is *, which requests the JES system the user is currently running under.

The details of the differences for the JES3 environment are included in the descriptions of the ISFPARMS statements in Chapter 2, "Using ISFPARMS for customization and security," on page 15.

To assist you in defining SAF security, code a new custom property in ISFPARMS. Setting the property Security.SAFNoDec.WarnMsg to TRUE causes an SDSF message to be issued whenever a SAF no-decision result (return code 04) is converted to a failure. The message includes the class name, resource name and access level being checked. Once you have defined the SAF security, set the value to FALSE. See "Customized properties (PROPLIST)" on page 91 for more information.

SAF in the JES3 environment

In a JES3 environment, SAF is required for SDSF security. When a request is made to access a resource, and the profile that protects the resource is not defined, or the associated class is not active, SDSF fails the request. So, you must define all of the profiles, and activate all of the classes, that are used for SDSF security. The

ISFPARMS custom property, Security.SAFNoDec.WarnMsg, can be helpful in identifying missing classes and profiles. See "Customized properties (PROPLIST)" on page 91 for more information.

If you already have SDSF security defined using SAF in a JES2 environment, evaluate and update the SAF security for the JES3 environment as follows:

- Add profiles for the new resources in the OPERCMDS class that protect action characters and overtypable columns in the JES3 environment. For more information, see "New resources in the JES3 environment."
- Ensure that you are using SAF for all aspects of security, rather than a mix of SAF and ISFPARMS. For example:
  - Every user must belong to a group in ISFPARMS, and in the JES3 environment, group membership must be controlled through SAF, rather than with GROUP statements in ISFPARMS. Use the GROUP.*group-name.server-name* profile in the SDSF class. See "Using SAF to control group membership" on page 36 for more information.
  - The SDSF class must be activated. In addition to group membership, resources in the SDSF class control access to other things that you might previously have controlled with ISFPARMS, such as access to SDSF panels and overtypeable fields.

If you have not previously implemented SAF security, but have implemented security using ISFPARMS in a JES2 environment, you could begin by converting ISFPARMS to SAF, then making updates for the JES3 environment. See Chapter 8, "Converting ISFPARMS to SAF security," on page 311 for more information.

If you have not previously implemented SDSF security, you might begin with the SDSF class. Resources in the SDSF class control membership in groups in ISFPARMS, access to SDSF panels, objects that are displayed on those panels, and overtypeable fields. See Chapter 5, "Using SAF for security," on page 191 for an introduction to SAF security for SDSF.

New resources in the JES3 environment: The SAF resources that protect the action characters and overtypeable columns in the JES3 environment vary slightly from the resources used in the JES2 environment. In addition, there are new action characters and overtypeable columns in the JES3 environment. The SAF resources for the JES3 environment are described along with the SAF resources for the JES2 environment, in Chapter 7, "Protecting SDSF functions," on page 211. In addition, the information that describes enhancements in each release includes a discussion of the SAF resources for any new JES3 support for that release. See Chapter 1, "Exploiting new functions," on page 1.

SDSF considerations

SDSF does not support more than a single instance of SDSF executing under the same task control block (TCB).

Issuing MVS and JES commands

SDSF uses a console when issuing MVS or JES commands that were entered with a / command. The console used varies.

- If the user session log (for display on the ULOG panel) is active, SDSF uses an extended console. See "Extended console name" on page 343 for more information.

- If the user log is not active, SDSF uses a console ID of 0.

System programmers can control the console used by SDSF with parameters in ISFPARMS. EMCSREQ specifies whether an EMCS console must be used. EMCSAUTH specifies whether SDSF activates the EMCS console with MASTER or SYS,IO,CONS authority.

Users can request that SDSF use a console ID of 0 with the i parameter on the / command (i/*command*). For this to be accepted, a console ID of 0 must be allowed by the setting for EMCSREQ in ISFPARMS.

Installations should control use of the / command as they would a console with master authority. The user session log (ULOG) can be protected with the AUTH parameter in ISFPARMS or with SAF. For more information, see "Group function parameters reference" on page 40 or "User log (ULOG)" on page 305. For information on protecting consoles, see *z/OS MVS Planning: Operations*.

Extended console name

The name of the extended console used by SDSF defaults to the user ID. Users can change it with the SET CONSOLE command.

When SDSF needs to activate an extended console and the default console name is in use (for example, when you invoke SDSF from a REXX exec while also using SDSF interactively) SDSF attempts to activate a new console with a different name, which is derived by modifying the default console name. To modify the name, SDSF appends a single-character suffix. SDSF can try up to 32 different characters until a unique console name is obtained. The original console name must be fewer than 8 characters long.

You can control console name modification with:

- The SET CONMOD (ON|OFF) command, which turns console name modification on and off.
- In ISFPARMS, the custom property Console.EMCS.ConModChars, which specifies the characters to be used as the suffix. By default, the characters are $#@12345.
- In ISFPARMS, the custom property Console.EMCS.NoConMod, which turns console name modification off.
- In a REXX exec, with the ISFCONMOD special variable.
- In a Java program, with ISFRequestSettings.

If console name modification is off and the default extended console name is in use, SDSF attempts to share the console. For example, if you use ISPF split screen and access SDSF in multiple logical screens, SDSF shares the console activated in the first logical screen with subsequent logical screens. As a result, ULOG in the first logical screen contains system messages for all of the logical screens. SDSF shares the console only when the console is activated in the same address space. If the console cannot be shared, activation of the console fails.

RMF considerations

The following require that RMF Monitor I be started:

- The DA panel, in a JES3 environment
- A sysplex-wide DA panel, in a JES2 environment

By default, Monitor I is started when you start RMF.

In addition, modules in the SISFLOAD data set must be made accessible to the RMF started task on each system in the sysplex.

If ISF.SISFLOAD is installed in the link list or link pack area, no action is necessary. RMF will be able to load the SDSF modules it needs from the LNKLST or LPA.

If you are running SDSF in a TSO STEPLIB, you will need to add a steplib to the RMF started task procedure. Add the following statement to your RMF procedure JCL for each system in the sysplex:

```
//STEPLIB  DD  DSN=ISF.SISFLOAD,DISP=SHR
```

RMF Monitor I is also needed to obtain the LPAR and zAAP views of CPU utilization displayed on the title line of the DA panel, and the values for the SzAAP% and SzIIP% columns on the DA panel.

RMF protects the services that SDSF uses to gather data for the DA panel with the SAF profile ERBSDS.MON2DATA or a generic–style profile, ERBSDS.*, both in the FACILITY class. If either profile exists, SDSF users must have READ access to it to access the RMF data on the DA panel. However, the profiles are optional. If neither profile exists, then SDSF uses can access the data. For details, see *z/OS RMF User's Guide*.

WebSphere MQ considerations

With a sysplex that includes one or more systems at the z/OS V1R12 level or lower, you can use WebSphere MQ along with the SDSF server to provide sysplex-wide data on SDSF panels.

WebSphere MQ must be up and operational on all systems that are to participate, with communication between queue managers configured using channels, as described in *WebSphere MQ Intercommunication*.

For more information, refer to "Using the server for sysplex data" on page 110.

z/OS libraries

SDSF's sysplex support requires the following:

- The WebSphere MQ load libraries, including SCSQLOAD, must be accessible to both the SDSF server and the SDSF client. This can be done by adding the libraries to the lnklst, or with a steplib.
- If you use a steplib, all libraries in the steplib concatentation must be APF-authorized, including SCSQLOAD.

WebSphere MQ configuration

A minimal amount of WebSphere MQ configuration is required for SDSF.

- Review the values in the WebSphere MQ system parameter module, CSQZPARM. In particular, you may need to increase the number of background

and foreground connections, which are defined with the IDBACK and IDFORE parameters. The SDSF server establishes several connections with WebSphere MQ: at least 11 and at most approximately 31. Your IDBACK value should reflect this usage. Similarly, the SDSF client establishes a connection with WebSphere MQ for each SDSF logical session. Multiple SDSF sessions can be started using ISPF's split screen. Your IDFORE value may need to be adjusted to accommodate this. See *WebSphere MQ for z/OS System Setup Guide* for more information.

Table 140. Summary of Possible WebSphere MQ System Parameters Changes

Parameter	WebSphere MQ Default	Possible Changes for SDSF
IDBACK	20	Change to reflect SDSF server connections with WebSphere MQ, up to 31
IDFORE	100	Change to reflect the maximum number of SDSF client logical sessions that are connected to WebSphere MQ

- To separate the SDSF message usage from your existing applications, you may want to define a separate queue manager to be used by SDSF. You control which queue manager SDSF uses by coding its name on the COMM statement associated with the server definition in ISFPARMS. See "SERVER statement" on page 31.
- Review your WebSphere MQ page sets. Storage estimates are described in "Storage estimates."
- If your installation has a large number of devices, you may need MQSeries® APAR PQ33000 installed on your system. This APAR provides support for very large MQSeries messages.

Storage estimates

The following are estimates of the number of SDSF messages and their sizes, for use in defining the page sets for your WebSphere MQ queue manager.

The number of messages is proportional to the number of users, the number of requests for data (caused, for example, by a user pressing Enter), and the number of servers in the server group.

The size of the messages varies with the data being requested. In general, a request is approximately 300 bytes. A response consists of a 300-byte header followed by the response data. The response varies with the panel and the number of rows returned. SDSF compresses the response data, so the actual data sent through WebSphere MQ may be less than the maximum. The effectiveness of compression relates directly to the contents of the data being returned.

Table 141. Message Size Estimates

Message Type	Size
Request	300 bytes

Table 141. Message Size Estimates (continued)

Message Type	Size
	Check: 1600 bytes for each IBM Health Checker for z/OS check returned; 100 bytes for each line of check output
	Enclave: 400 bytes for each enclave returned, and 800 bytes for each enclave detail returned (I action character)
	Initiator: 300 bytes for each initiator returned
	JES2 resource: 230 bytes for each resource returned
	Line: 500 bytes for each line returned
	Node: 350 bytes for each node returned
	Printer: 1K bytes for each printer returned
	Process: 500 bytes for each process returned
	Punch: 800 bytes for each punch returned
	Reader: 500 bytes for each reader returned
	Spool offload: 1K bytes for each offloader returned

Queues

You do not define any queues for SDSF's use of WebSphere MQ; they are defined for you. The queues SDSF uses are:

- A model queue, which is used in creating other queues. The SDSF server defines this if it is not already defined.
- Temporary, dynamic queues used to communicate between the SDSF server and the user. WebSphere MQ creates these request queues with the use of the model queue.

For more information, see "Queues used by SDSF" on page 116. The DEFINE commands used to define the queues are logged in the server log. See "Logging" on page 109 for details.

Protecting the queues

You use SAF to protect access to the WebSphere MQ queues used by SDSF. For details, see "WebSphere MQ" on page 306.

Communication between queue managers in a non-clustered environment

This section applies only if you are **not** using WebSphere MQ clustering. If you are using clustering, you should not use the queue manager alias technique described in this section.

SDSF recommends the use of clustering with the WebSphere MQ queue managers. Clustering is a configuration technique that provides these benefits in SDSF:

- Significant reduction in the WebSphere MQ definitions required to link queue managers together
- Improved awareness of the status of SDSF servers in the server group.

If you do not use WebSphere MQ clustering, the addition of queues used by SDSF may require you to perform some WebSphere MQ configuration so that the queue managers for those queues can communicate. Communication between queue

managers is described in *WebSphere MQ Intercommunication*, SC33–1872. Refer to that document when configuring your WebSphere MQ environment. This section provides a brief introduction.

When a queue manager needs to put messages on a queue managed by a different queue manager, it locates the target queue by the queue name and the queue manager name. For example, SDSF's server request queue is accessed by all SDSF servers in the server group. To locate that queue, an WebSphere MQ queue manager would need the:

- Queue name, *queue-prefix*.CLIENT.*server.system*.REQUESTQ
- Queue manager name, which is specified on the COMM statement in ISFPARMS.

In a non-clustered environment, there are several ways to define the remote queues and queue managers to the local queue managers. SDSF is designed to simplify this task by allowing you to use the queue manager alias technique. A queue manager alias relates a queue manager name to a transmission queue. (The transmission queue is a special kind of queue on which messages are stored until they can be transmitted to the remote queue manager. WebSphere MQ uses a channel and a transmission queue on the remote system to ensure that message gets routed properly.) The queue manager alias is convenient because only a single definition is needed to route all requests to all queues managed by a remote queue manager. If you don't use a queue manager alias, you need a remote queue definition for each remote *queue*. This results in many more definitions.

Example:
This example shows the form of the queue manager alias that is useful with SDSF.

You have two queue managers, MQS1 and MQS2. You have the following queue manager alias definitions:

- For MQS1:

  ```
  DEFINE QREMOTE ('MQS2') RNAME('') RQMNAME('MQS2') XMITQ('MQS1.TO.MQS2.XMITQ')
  ```

  This defintion causes all messages targeted for queue manager MQS2 to be put on the transmission queue MQS1.TO.MQS2.XMITQ.
- For MQS2:

  ```
  DEFINE QREMOTE ('MQS1') RNAME('') RQMNAME('MQS1') XMITQ('MQS2.TO.MQS1.XMITQ')
  ```

  This definition causes all messages targeted for queue manager MQS1 to be put on the transmission queue MQS2.TO.MQS1.XMITQ.

See *WebSphere MQ Intercommunication* for more information.

ISPF considerations

z/OS provides sample ISPF primary option menus with SDSF and other elements and features already added under option 13.14, as described in the program directory. If you want to add SDSF to your own customized ISPF menu, you should add text to the body for the SDSF menu option, for example:

```
S  SDSF         System Display and Search Facility
```

and update the ZSEL statement in the PROC section to invoke SDSF with the ISFISP entry point, as shown in the following except. The lines added for SDSF are shown in **bold**.

```
.
.
.
 7,'PGM(ISPYXDR) PARM(&ZTAPPLID) SCRNAME(DTEST) NOCHECK'
 8,'PANEL(ISRLPRIM) SCRNAME(LMF)'
 9,'PANEL(ISRDIIS) ADDPOP'
10,'PGM(ISRSCLM) SCRNAME(SCLM) NOCHECK'
11,'PGM(ISRUDA) PARM(ISRWORK) SCRNAME(WORK)'
 S,'PANEL(ISFSDOP2) NEWAPPL(ISF) SCRNAME(SDSF)'
 X,EXIT
SP,'PGM(ISPSAM) PARM(PNS)'
' ',' '
   *,'?' )
IF (&ZCMD = 'S')
   &ZSEL = 'PGM(ISFISP) NOCHECK NEWAPPL(ISF) SCRNAME(SDSF)'
IF (&ZCMD = 'S.')
   &ZSEL = 'PGM(ISFISP) NOCHECK NEWAPPL(ISF) SCRNAME(SDSF)'
```

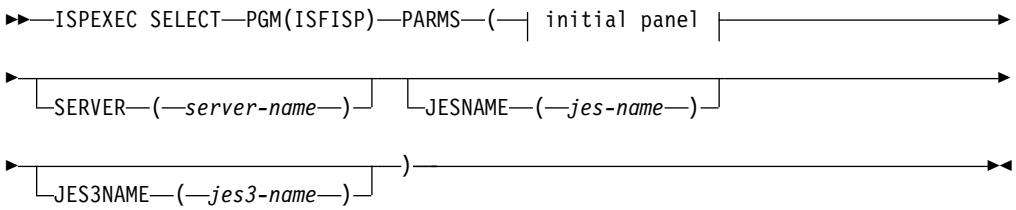
Note: The IF statements are required. Failure to include this logic may result in an incorrect number of rows being displayed on split screens, a failure to process additional options specified on the S command, or message ISF922E. The IF statements must be added after the ZSEL statement.

If you want to be able to invoke SDSF as a command from within ISPF, you can add SDSF to the ISPF command table. For example, you could add this entry:

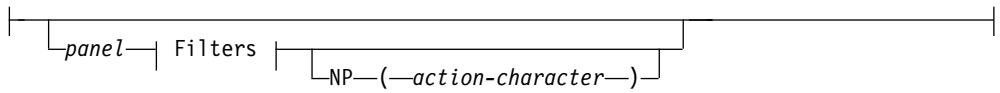
```
Verb    T  Action
SDSF    0  SELECT PGM(ISFISP) NEWAPPL(ISF) SCRNAME(SDSF)
```

ISFISP entry point

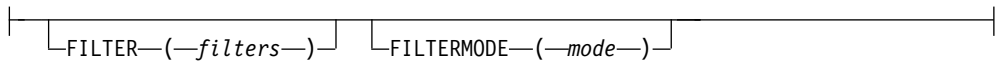
When you invoke SDSF as an ISPF dialog using the ISFISP entry point, you can specify parameters to specify an initial panel and other values. The syntax of the ISPEXEC service is as follows:



Initial panel:



Filters:



panel
Is the command to access a panel, for example, DA or ST.

server-name
Is the name of the local SDSF server.

jes-name
Is the name of the JES2 subsystem to process.

jes3-name
Is the name of the JES3 subsystem to process.

filters
Is the set of filters for the panel, up to 25. This is valid only when ISPF is invoked from a web client.

A filter consists of a column title, operand and value. The operand can be EQ (equal), NE (not equal), LT (less than), LE (less than or equal to), GT (greater than) or GE (greater than or equal to). To specify multiple filters for a single column, use the same column title with the second and subsequent filters.

Filter criteria remain in effect until you add new filters or turn filtering off. Filter criteria are saved in the ISPF profile when SDSF ends.

mode
Is the relationship between filters:

AND
The row must match all filters.

OR The row must match any filter.

This is valid only when ISPF is invoked from a web client.

action-character
Is an action character to be applied to the tabular panel. If building the panel or applying the filters results in more than one row, or if the panel is not a tabular panel, the action character is ignored. This is valid only when ISPF is invoked from a web client.

Specifying that SDSF should process JES2

By default, SDSF determines whether to process JES2 or JES3. You can specify that SDSF should not do that determination and process JES2 by invoking it with an alternate command: use ISFISP2 rather than ISFISP in the PROC section of an ISPF panel, and SDSF2 rather than SDSF in an ISPF command table.

SYSLOG

Beginning with z/OS V1R11, SDSF uses a JES logical log to provide the SYSLOG panel for JES2 and JES3 environments. SDSF does not use the HASPINDX data set.

For lower level systems, SDSF uses the HASPINDX data set for the SYSLOG panel. Even with z/OS V1R11 and higher systems, it is possible to force the use of HASPINDX, with the PROPERTY statement in ISFPARMS. However, you should be aware that it is IBM's intention to eventually remove the HASPINDX-based SYSLOG when all systems support the logical log.

You can specify the system being processed with the SYSID command.

Logical Log

The logical log removes constraints on the number of jobs and data sets that can be indexed, and should result in better performance than use of the HASPINDX data set. If you have only z/OS V1R11 or higher systems in the MAS, you can delete any HASPINDX data sets.

A SAF resource controls access to the JES logical log. The resource is *nodeid*.+MASTER+.SYSLOG.SYSTEM.*sysname* in the JESSPOOL class. READ access is required. This resource is in addition to the ISFCMD.ODSP.SYSLOG resource in the SDSF class that was already used.

HASPINDX

The HASPINDX data set contains information related to all SYSLOG jobs and data sets on the spool. The first time a user accesses the SYSLOG panel, SDSF determines which HASPINDX data set to use and allocates it if necessary.

Configuration options for the HASPINDX data set

If the same data set name is specified in ISFPARMS for multiple systems, those systems will share the HASPINDX data set. Sharing the HASPINDX data set across systems reduces the number of data sets that are needed, but may result in contention and reduced performance, depending on how much activity there is on the SDSF SYSLOG. For that reason, IBM does not recommend sharing the HASPINDX data set. Note that HASPINDX data sets are typically small and that sharing of the HASPINDX data set results in no functional, usability or other external difference to end users.

If you have high SYSLOG activity and many users of the SDSF SYSLOG panel, you should consider allocating multiple HASPINDX data sets. There is no limit to the number of HASPINDX data sets you can create. A reasonable approach might be to create one for each system. You might also choose to create one for each SDSF user.

The configuration options for the HASPINDX data set are summarized below.

Table 142. Configuration Options for the HASPINDX Data Set

Configuration	Method	Pros and Cons
One for each system	Specify a unique name in ISFPARMS on each system	Performance may be better than with a shared HASPINDX data set, but several data sets are needed. Note that the use of system symbols may simplify this approach. See “Conditional processing” on page 20 for more information.
Shared across systems	Specify the same name in ISFPARMS on several systems	Fewer data sets are needed, but performance may be degraded.
One for each user or batch job	Preallocate a data set to ddname HASPINDX	May result in the best performance, but many data sets are needed.

When the logical SYSLOG is used rather than HASPINDX, there is no limit to the number of SYSLOG jobs or data sets that can be processed. There is also no effect based on the number of concurrent users accessing SYSLOG.

Specifying the HASPINDX data set

The HASPINDX data set can be specified in two ways:

- In ISFPARMS, with the INDEX and INDXVOL keywords of the ISFPMAC macro or OPTIONS statement.
- By preallocating the ddname HASPINDX prior to the SYSLOG panel being accessed. SDSF then uses the data set referenced by that ddname.

Sharing HASPINDX in a mixed environment

During SYSLOG processing, SDSF checks the level of HASPINDX, which is maintained in a special record within the data set. If the level of HASPINDX is not

compatible with the level expected by SDSF, SDSF formats the data set. Formatting involves initializing the data set and preprocessing the SYSLOG data sets on spool. The time required varies based on the size of the HASPINDX data set and the number of SYSLOG jobs and data sets on spool.

Allocating the HASPINDX data set

Sample job ISFISALC contains an allocation for the ISF.HASPINDX data set.

The space required for the HASPINDX data set varies based on the number of SYSLOG data sets on the spool and the number of records within each data set. The HASPINDX block size determines how many data sets can be processed; the larger the block size, the more data sets it can contain. The HASPINDX space determines how many syslog records can be processed; the more space, the more records that can be recorded.

For most installations, the SDSF default block size of 4096 and a space of 5 cylinders will be adequate. The space must be contiguous. The values can be changed at any time after SDSF installation if the HASPINDX data set becomes full.

Reassembling SDSF for JES2 maintenance

Some portions of SDSF are dependent on JES2 macros and JES2 maintenance may change the structure of their associated control blocks. You should reassemble SDSF when JES2 maintenance changes the macros. This can be done automatically by SMP/E when maintenance is shipped, or you can do it manually using the ISFASMP sample job. No reassemblies are needed in the JES3 environment.

SDSF provides code that, at initialization, checks the assembled level of all SDSF modules against the level of the JES2 system. If a mismatch occurs, SDSF abends the initialization.

Reassembling SDSF with ISFASMP

You can reassemble the entire SDSF product with the sample SMP/E usermod shipped with the SDSF/JES2 feature dependent FMID. The ISFASMP job contains a sample SMP/E user modification. When you apply it, it causes all assembler parts of SDSF to be reassembled. It uses the same SYSLIBs during the initial installation of SDSF and JES2. Thus, you retain any maintenance you applied through SMP/E.

Sample SMP/E UCLIN jobs for automatic reassembly

SDSF provides a sample job that causes SMP/E to reassemble SDSF whenever maintenance affects JES2 macros. The job is member ISFISUCL of the sample JCL library data set, ISF.SISFJCL1, shipped with the SDSF/JES2 feature dependent FMID. It takes UCLIN from another member of the data set as input. By default, the job specifies member ISFZZUCL, which is an alias for the member that contains UCLIN for the current level of z/OS JES2. Other members in the data set contain UCLIN for the other supported levels of z/OS JES2.

Instructions in the prolog of ISFISUCL describe the changes you need to make to the job. After making the required changes, run the job against the zone containing SDSF and JES2. Maintenance applied to any JES2 macro that has been included in the UCLIN job will cause the affected SDSF parts to be reassembled.

The UCLIN jobs create a GENASM subentry in the MAC entry for each JES2 macro that is referenced by SDSF. The GENASM subentries created specify the

SDSF modules that reference the JES2 macro. The UCLIN ADD function is used instead of REP so that any existing GENASM subentries are not replaced by these jobs.

This facility remains in effect as long as the CSI entry describing the JES2 macro is not replaced. If the entry is replaced, the UCLIN job must be rerun to restore the required GENASM entries.

z/OSMF considerations

IBM® z/OS Management Facility (z/OSMF) provides a framework for managing various aspects of a z/OS system through a web browser interface. By streamlining some traditional tasks and automating others, z/OSMF can help to simplify some areas of z/OS system management.

The SDSF task of z/OSMF lets you see key summary information about your sysplex in graphical form, work with jobs and checks for IBM z/OS Health Checker, and issue system commands. It includes function that is analogous to these functions of z/OS SDSF:

- DA, H, I, O, ST, Job Data Set and Output Data Set (browse) panels, for jobs and job output
- CK and Health Check History panels, for health checks
- APF, LNK, LPA, PAG, PARM, and SYS panels
- ULOG panel, for command and message responses issued during the current session
- Editing JCL
- Action characters for controlling jobs and checks
- Overtypeable fields, for modifying the attributes of jobs and checks
- Slash (/) command, for issuing system commands
- PREFIX, DEST, OWNER, SYSNAME, FILTER and SORT commands, for filtering and sorting tabular data
- ARRANGE command, for customizing the order and widths of columns

To select the SDSF task, expand the Jobs and Resources category in the navigation area and select **SDSF**.

Requirements

The SDSF task requires:

- z/OSMF APAR PM98630 and SDSF APAR PM86303. The SDSF APAR is functional only when the z/OSMF APAR is also installed.
- A TSO logon proc and related settings, which you specify with the **SDSF Settings** task in the z/OSMF Settings category. The TSO logon proc is used to launch a TSO address space that is created on behalf of the user. For details on the settings, refer to "Defining required settings for the SDSF task" on page 353.
- For remote systems to be displayed on the overview page of the SDSF task, SDSF communications must be active (through either WebSphere MQ or XCF) with a minimum level of z/OS SDSF V2R1. For more information, refer to "Using the server for sysplex data" on page 110.

Adding the SDSF task to z/OSMF

To add the SDSF task to z/OSMF, you import a properties file through the Import Manager task of z/OSMF, which is in the z/OSMF Administration category. This process is described in the z/OSMF online help.

The properties file for SDSF is /usr/lpp/sdsf/zosmf/sdsf.properties. Specify this file name in the Import dialog.

The import is generally a one-time process. The SDSF plugin only needs to be imported the first time you are installing SDSF or after you have deleted the plugin and want to restore it.

Defining required settings for the SDSF task

Before selecting the SDSF task the first time, you must define some settings. To do this, expand the z/OSMF Settings category, then select **SDSF Settings**. Supply a value for each required field. The settings are saved across sessions.

For TSO logon proc, you can specify any logon proc for which the user is authorized that is suitable for SDSF, as long as it contains either a //SYSEXEC or //SYSPROC that references data set ISF.SISFEXEC.

You do not need to create a new logon procedure for exclusive use of the z/OSMF SDSF task.

Example: The following is a sample logon proc that can be used by the SDSF task, with the minimum allocations. You may need to adjust the data set name for your installation.

```
//SDSF EXEC PGM=IKJEFT01,DYNAMNBR=500
//SYSEXEC DD DISP=SHR,DSN=ISF.SISFEXEC
```

The SDSF task does not use ISPF, so ISPF allocations are not required. If you use a logon proc that includes ISPF allocations, ensure that the allocations can be shared between the launched TSO address space and a standard TSO login. In particular, ensure that any ISPF profile allocation will allow both the launched TSO address space and the standard login to proceed.

If your logon proc invokes an initial command (using the PARM= keyword on the EXEC statement), the command must return to the TSO READY prompt. When the logon completes, SDSF waits fro the TSO READY prompt before proceeding.

If the logon fails when launching the SDSF task, click the TSO Messages link to view TSO messages that were issued during logon. Common errors preventing a successful launch include a JCL error in the logon proc, an invalid account number, and a missing ISF.SISFEXEC data set in the //SYSEXEC concatentation.

Reviewing z/OS Unix System Services settings

SDSF uses the z/OS Unix System Services interprocess communications (IPC) message queue for communications between SDSF and z/OSMF. The maximum message size is controlled by the size of the queue defined by the IPCMSGQBYTES option of PARMLIB member BPXPRM*xx*.

Message sizes used by SDSF vary based on the request type and amount of data returned in the response. You should review the setting of IPCMSGQBYTES on your system (using the D OMVS,O operator command) to ensure it is large enough for the messages sent by SDSF.

For details, refer to the topic about BPXPRM*xx* in *z/OS MVS Initialization and Tuning Reference*.

Protecting SDSF function in z/OSMF

The function provided by the SDSF task in z/OSMF is protected just as z/OS SDSF is protected, with the same SAF resources and ISFPARMS parameters. To control which group in ISFPARMS a user is assigned to, you can use either:

- SAF.

  To determine group membership, SDSF checks the SAF resource GROUP.*group-name*.*server-name* in the SDSF class. This is explained in detail in "Using SAF to control group membership" on page 36.
- The TSOAUTH parameter of the ISFGRP statement in ISFPARMS.

  For more information, refer to "Group membership" on page 36.

Using SAF is the recommended approach, as it is more dynamic and allows you to assign users to the same group regardless of the environment from which they invoke SDSF.

To hide SDSF functions that you are not authorized to, you can use Authorization preferences in the SDSF Settings task.

Using the SDSF task in z/OSMF has these new requirements:

- You must have access to resources in the ZMFAPLA class.

Table 143. Resources in the ZMFAPLA Class

z/OSMF Task	Resource	Access Required	Class
SDSF	*saf-prefix*.ZOSMF.IBMSDSF.JOBS	READ	ZMFAPLA
SDSF Settings	*saf-prefix*.ZOSMF.IBMSDSF.SETTINGS	READ	ZMFAPLA

  You configure *saf-prefix* in z/OSMF. The default is IZUDFLT.
- Your user ID must be connected to the IZUUSER group.

Access to views in z/OSMF is protected in the same way as the corresponding panel command in z/OS SDSF. For a description of protecting SDSF commands with SAF, refer to "Authorized SDSF commands" on page 235. For a description of protecting SDSF commands with the AUTH parameter of ISFPARMS, refer to "Group function parameters reference" on page 40.

Table 144. z/OSMF Views and Corresponding z/OS SDSF Commands

View	SDSF Command
Active Jobs	DA
All Jobs	ST
Input Queue	I
Output Queue	O
Held Output Queue	H
Job Data Sets	None (? action character)

Table 144. z/OSMF Views and Corresponding z/OS SDSF Commands (continued)

View	SDSF Command
Health Checks	CK
Health Check History	None (L action character)
APF Data Sets	APF
LNK Data Sets	LNK
LPA Data Sets	LPA
PAG Data Sets	PG
PARM Data Sets	PARM
System Information	SYS

The Overview tab requires access to these SDSF functions:

- The DA and SP panels for the System Activity graph
- The CK panel for the Health Check graph
- The slash (/) command, for the System command line.

Modifying values in tables or on property sheets in z/OSMF is protected in the same way as overtyping fields in z/OS SDSF. The columns have the same titles in z/OSMF as in z/OS SDSF. For more information about protecting overtyping fields with SAF, refer to "Overtypeable fields" on page 255. With ISFPARMS, overtyping fields is protected with the CMDLEV parameter. Refer to "Action characters and overtypeable fields for each command level" on page 73 for more information.

Actions on tables in z/OSMF are protected in the same way as the corresponding action characters in z/OS SDSF. The tables that follow show the actions for each of the tables in the SDSF task, and the corresponding action character for z/OS SDSF. For a complete discussion of protecting action characters with SAF, refer to "Action characters" on page 211. With ISFPARMS, action characters are protected with the CMDLEV parameter. Refer to "Action characters and overtypeable fields for each command level" on page 73 for more information.

Table 145. Actions on the Active Jobs Table

Action on the Active Jobs Table	Action Character on the DA Panel
Browse All Job Data Sets	S
Cancel Job	C
Cancel Job and Print Available Data Sets (JES3 only)	CP
Cancel Job and Produce a Dump	CD
Cancel Job and Purge Output	P
Cancel Job and Restart Automatically	CA
Cancel Job, Restart, and Hold Prior to Execution (JES2 only)	EC
Cancel Job, Restart Automatically, and Produce a Dump	CDA
Cancel Job, Take a Dump and Keep Job Output	CDP
Cancel Protected Job and Purge Output (JES2 only)	PP
Display DDNAMES of All Spool Data Sets that Contain Data (JES3 only)	DSD
Display DDNAMES of Data Sets in Spool Hold Status that Contain Data (JES3 only)	DSH
Display Extended Information (JES3 only)	DX

Table 145. Actions on the Active Jobs Table (continued)

Action on the Active Jobs Table	Action Character on the DA Panel
Display Information Short Form	D
Display Information Long Form	DL
Display Job Data Sets	?
Display Line, Page and Record Counts (JES3 only)	DE
Display List of Mains Authorized for Job (JES3 only)	DM
Display Output Long Form (JES2 only)	LL
Display Output Short Form (JES2 only)	L
Display Output Status (JES3 only)	L
Display Output Status Include Output On the Held Queue (JES3 only)	LH
Display Output Status Include SNA/NJE Output (JES3 only)	LB
Display Output Status Include TCP/IP Job Output (JES3 only)	LT
Display Spool Partition Name (JES3 only)	DSP
Edit JCL	SJ
Hold Job	H
Main Device Scheduling Allocate Queue (JES3 only)	DMA
Main Device Scheduling Error Queue (JES3 only)	DME
Main Device Scheduling Restart Queue (JES3 only)	DMR
Main Device Scheduling System Select Queue (JES3 only)	DMSS
Main Device Scheduling System Verify Queue (JES3 only)	DMSV
Main Device Scheduling Unavailable Volumes (JES3 only)	DMU
Release Job	A
Restart Job	E
Restart Job After Current Step Completes	ES
Restart and Hold Job After Current Step Completes (JES2 only)	ESH
Run Job	J
Spin Job	W

Table 146. Actions on the All Jobs Table

Action on the All Jobs Table	Action Character on the ST Panel
Browse All Job Data Sets	S
Cancel Job	C
Cancel Job and Print Available Data Sets (JES3 only)	CP
Cancel Job and Produce a Dump	CD
Cancel Job and Purge Output	P
Cancel Job and Restart Automatically	CA
Cancel Job, Restart, and Hold Prior to Execution (JES2 only)	EC
Cancel Job, Restart Automatically, and Produce a Dump	CDA
Cancel Job, Take a Dump and Keep Job Output (JES3 only)	CDP
Cancel Protected Job and Purge Output (JES2 only)	PP

Table 146. Actions on the All Jobs Table (continued)

Action on the All Jobs Table	Action Character on the ST Panel
Display DDNAMES of All Spool Data Sets that Contain Data (JES3 only)	DSD
Display DDNAMES of Data Sets in Spool Hold Status that Contain Data (JES3 only)	DSH
Display Information Short Form	D
Display Extended Information (JES3 only)	DX
Display Information Long Form (JES2 only)	DL
Display Job Data Sets	?
Display Line, Page and Record Counts (JES3 only)	DE
Display List of Mains Authorized for Job (JES3 only)	DM
Display Output Short Form (JES2 only)	L
Display Output Long Form (JES2 only)	LL
Display Output Status (JES3 only)	L
Display Output Status Include Output On the Held Queue (JES3 only)	LH
Display Output Status Include SNA/NJE Output (JES3 only)	LB
Display Output Status Include TCP/IP Job Output (JES3 only)	LT
Display Spool Partition Name (JES3 only)	DSP
Edit JCL	SJ
Hold Job	H
Main Device Scheduling Allocate Queue (JES3 only)	DMA
Main Device Scheduling Error Queue (JES3 only)	DME
Main Device Scheduling Restart Queue (JES3 only)	DMR
Main Device Schedling System Select Queue (JES3 only)	DMSS
Main Device Scheduling System Verify Queue (JES3 only)	DMSV
Main Device Scheduling Unavailable Volumes (JES3 only)	DMU
Release Job	A
Restart Job	E
Restart Job After Current Step Completes (JES2 only)	ES
Restart and Hold Job After Current Step Completes (JES2 only)	ESH
Run Job	J
Spin Job	W

Table 147. Actions on the Input Queue Table

Action on the Input Queue Table	Action Character on the I Panel
Browse All Job Data Sets	S
Cancel Job	C
Cancel Job and Restart Automatically	CA
Cancel Job and Print Available Data Sets (JES3 only)	CP
Cancel Job and Produce a Dump	CD
Cancel Job and Purge Output	P
Cancel Job, Restart, and Hold Prior to Execution (JES2 only)	EC

Table 147. Actions on the Input Queue Table (continued)

Action on the Input Queue Table	Action Character on the I Panel
Cancel Job, Restart Automatically, and Produce a Dump	CDA
Cancel Job, Take a Dump and Keep Job Output (JES3 only)	CDP
Cancel Protected Job and Purge Output	PP
Display DDNAMES of All Spool Data Sets that Contain Data (JES3 only)	DSD
Display DDNAMES of Data Sets in Spool Hold Status that Contain Data (JES3 only)	DSH
Display Extended Information (JES3 only)	DX
Display Information Short Form	D
Display Information Long Form	DL
Display Job Data Sets	?
Display Line, Page and Record Counts (JES3 only)	DE
Display List of Mains Authorized for Job (JES3 only)	DM
Display Output Short Form	L
Display Output Long Form	LL
Display Output Status Include Output On the Held Queue (JES3 only)	LH
Display Output Status Include SNA/NJE Output (JES3 only)	LB
Display Output Status Include TCP/IP Job Output (JES3 only)	LT
Display Spool Partition Name (JES3 only)	DSP
Edit JCL	SJ
Hold Job	H
Main Device Scheduling Allocate Queue (JES3 only)	DMA
Main Device Scheduling Error Queue (JES3 only)	DME
Main Device Scheduling Restart Queue (JES3 only)	DMR
Main Device Schedling System Select Queue (JES3 only)	DMSS
Main Device Scheduling System Verify Queue (JES3 only)	DMSV
Main Device Scheduling Unavailable Volumes (JES3 only)	DMU
Release Job	A
Restart Job	E
Restart Job After Current Step Completes (JES2 only)	ES
Restart and Hold Job After Current Step Completes (JES2 only)	ESH
Run Job	J
Spin Job	W

Table 148. Actions on the Output Queue Table

Action on the Output Queue Table	Action Character on the O Panel
Browse All Job Data Sets	S
Display Job Data Sets	?
Display Output Status Short Form (JES2 only)	L
Display Output Status Long Form (JES2 only)	LL
Edit JCL	SJ

Table 148. Actions on the Output Queue Table (continued)

Action on the Output Queue Table	Action Character on the O Panel
Hold Output (JES2 only)	H
Purge Output (JES2 only)	C
Release Output (JES2 only)	A

Table 149. Actions on the Held Output Queue Table

Action on the Held Output Queue Table	Action Character on the H Panel
Browse All Job Data Sets	S
Display Job Data Sets	?
Display Output Status Short Form (JES2 only)	L
Display Output Status Long Form (JES2 only)	LL
Edit JCL	SJ
Hold Output (JES2 only)	H
Purge Output (JES2 only)	PO
Release Output (JES2 only)	A
Release Output to Print and Purge (JES2 only)	O
Release Output to Print and Keep (JES2 only)	OK

Table 150. Actions on the Job Data Set Table

Action on the Job Data Set Table	Action Character on the JDS Panel
Browse All Data Sets	S
Display Information (JES3 only)	D
Edit JCL	SJ
Hold Output	H
Purge Output	C
Release Output	O

Table 151. Actions on the Health Check Table

Action on the Health Check Table	Action Character on the CK Panel
Activate	A
Browse	S
Deactivate	H
Delete	P
Delete Force	PF
Display Diagnostics	DD
Display History	L
Display Information Short Form	D
Display Information Long Form	DL

Table 151. Actions on the Health Check Table (continued)

Action on the Health Check Table	Action Character on the CK Panel
Display Outdated Policies	DPO
Display Policies	DP
Display Status	DS
Refresh	E
Remove Categories	U
Run	R

Table 152. Actions on the Health Check History Table

Action on the Health Check History Table	Action Character on the Health Check History Panel
Browse	S

Table 153. Actions on the System Table

Action on the System Table	Action Character on the SYS Panel
Display IPL information	D
Display all address spaces	DAA
Display address space list	DAL
Display allocated list	DALO
Display consoles	DC
Display language environment options	DCEE
Display dump information	DD
Display EMCS	DEM
Display GRS information	DG
Display IOS information	DI
Display IQP information	DIQP
Display LLA information	DLL
Display system logger information	DLO
Display LOGREC information	DLR
Display configuration information	DM
Display MPF information	DMP
Display OMVS options	DO
Display product registration	DP
Display PCIE dev info	DPCD
Display PCIE product info	DPCI
Display SMF status	DSF
Display SLIP information	DSL
Display SMS information	DSM
Display system symbols	DSY

Table 153. Actions on the System Table (continued)

Action on the System Table	Action Character on the SYS Panel
Display time	DT
Display TSO options	DTO
Display trace	DTR
Display TSO address spaces	DTS
Display WLM information	DW
Display XCF sysplex information	DX

Table 154. Actions on the APF Table

Action on the APF Table	Action Character on the APF Panel
Display	D
Display all	DA

Table 155. Actions on the PAG Table

Action on the PAG Table	Action Character on the PAG Panel
Display page data set	D
Display common page data	DC
Display page delete information	DD
Display local page data sets	DL
Display PLPA data sets	DP
Display storage class memory	DS

Table 156. Actions on the LNK Table

Action on the LNK Table	Action Character on the LNK Panel
Display link data set	D
Display link names	DN

Table 157. Actions on the PARM Table

Action on the PARM Table	Action Character on the PARM Panel
Display parm data set	D
Display errors in parm data	DE

Diagnosing problems in z/OSMF

TSO messages: In addition to z/OSMF messages that are displayed in a message window, TSO messages may be issued in response to starting a session or other interactions. To display these messages, click **TSO Messages.**

Log files in z/OSMF: The directory of the z/OSMF log file is configurable, as described in *IBM z/OS Management Facility Configuration Guide*.

Determining the level of the SDSF plugin: From the z/OSMF Welcome panel, click the About link. Find the IBM SDSF plugin in the list. The associated information contains the SDSF FMID and the service level of the plugin.

Removing the SDSF task from z/OSMF

To remove the SDSF and SDSF Settings tasks from z/OSMF, use the Import Manager task to import properties file /usr/lpp/sdsf/zosmf/sdsfDelete.properties.

Using the SDSF classic interface

SDSF function is available through the z/OSMF ISPF task. To use the ISPF task, select ISPF in the z/OS Classic Interfaces category.

You can link to SDSF function that is available through the z/OSMF ISPF task from other function in z/OSMF. To do that, register the SDSF function as an event handler for z/OSMF events. For more information, refer to *IBM z/OS Management Facility Configuration Guide*.

Chapter 11. Introduction to SDSF application services

Table 158 summarizes the interfaces that allow you to access SDSF function.

Table 158. Summary of SDSF Interfaces

Interface	Description	Refer to
Batch	Program name SDSF provides basic support for SDSF commands and action characters. Program name ISFAFD adds support for overtyping fields.	Chapter 12, "Using SDSF in batch," on page 369
REXX	Allows you to exploit the power and simplicity of REXX. Full support for SDSF function.	Chapter 13, "Using SDSF with the REXX programming language," on page 377
Java	Allows you to create Java applications that exploit SDSF function. Full support for SDSF function.	Chapter 14, "Using SDSF with the Java programming language," on page 451

This topic provides a quick introduction to SDSF function and terminology for people who are not already experienced users of SDSF but want to exploit SDSF's batch, REXX or Java interfaces. It does not provide complete information for using SDSF function. For that, you must refer to the SDSF online help, which you can access with these commands:

Table 159. Commands for Getting Help on Using SDSF

Command	Description
HELP	Context-sensitive help. Includes menus, a table of contents and an index.
TUTOR	Interactive tutorial on basic SDSF function.
REXXHELP	Help that is specific to using SDSF with REXX.
COLSHELP	List of the columns on SDSF panels. You can filter and sort the list.
SEARCH	Search SDSF's help and tutorial.

SDSF panels

When you use SDSF interactively, SDSF displays data on panels. There are panels for active jobs, output groups, printers, initiators and so on. Most SDSF panels are tabular, that is, they display data in rows and columns.

Figure 13 on page 364 uses a sample tabular panel to show the layout of an SDSF panel.

```
Display  Filter  View  Print  Options  Search  Help 1
------------------------------------------------------------------------------
 SDSF SAMPLE  SYS1 2                                    3  LINE 1-22 (31)
 COMMAND INPUT ===> _   4                                 SCROLL ===> PAGE
 PREFIX=*  DEST=(ALL)  OWNER=SHERRYF  FILTERS=2 5
 NP    JOBNAME   6      ProcStep JobID    Owner    C Pos DP PGN Real Paging
 7     CATALOG  CATALOG  IEFPROC                    NS  FF     3228   0.00
       TAPEPOL  PROC01   M02XF83L TSU19596 TAPEPOL   OT  FF  69    0   0.00
       TANDA    E52TOOL1 M02SA06L TSU18751 TANDA     OT  FF  63 1488   0.00
       KSHEL    PROC01   M02PV317 TSU07739 KSHEL     OT  FF  60    0   0.00
```

© Copyright IBM Corp. 1997, 2016

Figure 13. A Sample SDSF Tabular Panel

See	Name	Description
1	**Action bar**	The action bar permits you to select a pull-down menu to accomplish various SDSF tasks.
2	**Title line**	The title line shows the panel name as well as other information.
3	**Message area**	Short error and confirmation messages appear here.
4	**Command line**	The command line lets you enter SDSF, MVS, or JES commands.
5	**Message and information lines**	Longer messages appear below the command line. The information lines display responses when you issue some SDSF commands. The example shows the response to SET DISPLAY, which displays settings for filters.
6	**Data area**	The data area contains the system data. On tabular panels, the data is in columns and rows. Each row represents a single job, TSO user, data set, device or system resource, depending on the panel. The column titles may be customized by the system programmer. For that reason, when using the programming interfaces, you refer to columns by their internal *names* rather than by their titles. The names cannot be modified. When customizing the columns, system programmers can define a primary list of columns, which is shown when the panel is first displayed, and an alternate list, which you display with the ? command. Typically, the alternate list contains all of the columns in the primary list plus some additional columns. The additional columns may require additional work by SDSF to retrieve the data. These columns are referred to as *delayed* or *delayed-access*. The first column is the *fixed field*; when you scroll right or left, it remains in the same position. In the sample panel, the `JOBNAME` field is fixed. The REXX and Java interfaces allow you to control which columns are included when you access a panel. Typically, you want to include only those columns that are required.
6	**NP column**	Input field for brief commands, known as action characters.

When you use SDSF interactively, the panels you are authorized to are listed on the SDSF Primary Option Menu. (A few panels that you access from other panels are not shown.)

SDSF commands

You use SDSF commands to access SDSF panels, filter and sort the data on SDSF panels, issue system commands and set various options. For a list of SDSF commands, see "SDSF commands reference" on page 414.

The details of command syntax are in the online help. For quick access to command syntax, you can use this SEARCH command from the SDSF command line:

```
SEARCH 'FORMAT: command-name'
```

where *command-name* is the command name, for example, DA or PREFIX.

- REXX: For panel commands, see "Issuing commands with ISFEXEC" on page 380. For other commands, use special variables. See "Using special variables to invoke SDSF function" on page 410.

- Java: For panel commands, create an instance of a runner for the desired panel. For filter commands, use request settings. See “Using runners and request settings” on page 456.

Taking action against an object

You take action against or display more information about an object, such as a job or a device, with action characters. Action characters are short commands, usually one or two characters. When using SDSF interactively, you type action characters in the NP column.

To display valid action characters with a description, use the `SET ACTION` command. To display just a list of action characters, use `SET ACTION SHORT`. This example shows the results of SET ACTION SHORT:

```
  SDSF INPUT QUEUE DISPLAY ALL CLASSES                    LINES 1-5 (5)
  COMMAND INPUT ===>                                    SCROLL ===> HALF
ACTION=//,=,+,?,A,C,CA,CD,CDA,D,E,H,L,P,PP,Q,S,SB,SE,SJ,X,XC,
ACTION=XD,XDC,XF,XFC,XS,XSC
  NP   JOBNAME  JOBID    OWNER    PRTY C  POS  PRTDEST              RMT  NODE
       ISF2CMDS JOB08765 DLR        7 H   16  LOCAL                        1
       ISF2ALL  JOB08871 DLR        7 H    3  LOCAL                        1
       ISF2FILT JOB08883 DLR        7 H   14  LOCAL                        1
```

A few action characters access a secondary panel. For example, you use the ? action character on a job-related panel to display the Job Data Set panel, which lets you work with individual data sets.

- REXX: Use ISFACT. See “Issuing action characters and modifying columns with ISFACT” on page 388.
- Java: Use methods. See “Actions and overtypes” on page 455.

Changing values in columns

You can change the values in some columns by typing over them. SDSF refers to this as *overtyping*. Some overtypeable columns are part of a set of values. For example, there are eight SFORMS values for printers. When using SDSF interactively, you work with a set of related values by typing a + alone in the column. So, to overtype multiple SFORMS, you type a + in the SFORMS column to display a pop-up on which you can modify all eight values.

- REXX: Use ISFACT. See “Issuing action characters and modifying columns with ISFACT” on page 388.
- Java: Use the requestPropertyChange method. See “Actions and overtypes” on page 455.

Filtering and sorting panel information

You can limit the data on your SDSF panels by using SDSF commands.

Table 160 on page 366 provides a high-level introduction. For important details, including syntax, refer to the online help. For quick access to information about a command, use this SEARCH command from the SDSF command line:

```
SEARCH 'FORMAT: command-name'
```

Table 160. Summary of Commands for Filtering

Command	Use	Panels
DEST	Filter data by destination. You set a single value that filters all of the affected panels.	H, I, J0, O, PR, PUN, ST
FILTER	Filter data on any column or combination of columns. You can set a unique filter for each panel. For more information, refer to "Complex filters."	Tabular, OPERLOG
OWNER	Filter data by owning user ID (primarily). You set a single value that filters all of the affected panels.	DA, H, I, J0, O, PS, ST
PREFIX	Filter data by job name (primarily). You set a single value that filters all of the affected panels.	DA, H, I, O, PS, ST
SELECT	Fast path to display a specific object without changing other saved filters	Tabular panels
SYSNAME	Limit rows to include selected systems in a sysplex. You set a single value that filters all of the affected panels.	CK, DA, ENC, INIT, LI, NO, PR, PS, PUN, RDR, RM and SO

Filtering the data can reduce storage and improve performance. For best results, use the PREFIX, OWNER, DEST or SYSNAME commands, or parameters on the panel commands. Use the FILTER command, which SDSF processes after the data is gathered, if you cannot accomplish the desired filtering using the other commands.

You can sort panels on up to two columns, in ascending or descending order, with the SORT command.

Complex filters

You can use the Filter function to define up to 25 filters with boolean operators. The filter criteria are column, operator and value, and can include pattern matching. For example, to see jobs with a job name that begins with JERRY, you could use `FILTER JOBNAME EQ JERRY*`.

When entering multiple filters, you can specify AND or OR to define the relationship between filters.

When using SDSF interactively under ISPF, type `FILTER ?` to display the Filter pop-up, then type values on the pop-up or select from lists of valid values.

To turn off filtering use `FILTER OFF`.

To display the number of filters in effect, use `SET DISPLAY`.

- REXX: Use special variables. See "Using special variables to invoke SDSF function" on page 410.
- Java: Use request settings. See "Using runners and request settings" on page 456.

Issuing MVS or JES commands

You can issue any MVS and JES command from the SDSF command line. Type a slash (/) followed by the command, for example, `/d r,l`. When using SDSF interactively, you can specify a longer command by typing / by itself to display the pop-up, and then typing the command on the pop-up.

The messages issued in response are displayed on the information lines of the panel. The complete set of responses is in the user session log (ULOG). You can set a delay interval, which is the maximum amount of time SDSF will wait for messages, with this command: `SET DELAY seconds`. The default is 1 second. A delay of 0 specifies that messages issued in response to / commands should not be displayed on the message lines.

- REXX: Use ISFSLASH. See "Issuing system commands with ISFSLASH" on page 408.
- Java: Use the runner ISFRunner. See "Using runners and request settings" on page 456.

Browsing job output and checks

You browse job output, and checks for IBM Health Checker for z/OS, by using the S action character. The resulting panel includes the JES job log, JCL for the job, and job-related messages. To browse just the JCL for the job, use the `SJ` action character.

- REXX: Use a combination of action characters, with ISFACT, and special REXX variables. See "Browsing output" on page 394.
- Java: Allocate the spool data sets for a job with the browseAllocate method, then read them using standard utilities.

Printing from SDSF Panels

You print the output of jobs, and checks for IBM Health Checker for z/OS, with the X action character.

You can print to SYSOUT, a data set, or a print file (specified with a *ddname*), with different forms of the X action character. For example, to print to a data set, you would use `XD`.

- REXX: Use a combination of the X action character, with ISFACT, and special REXX variables. See "Printing output" on page 399.
- Java: Use a combination of print methods and associated print settings.

Chapter 12. Using SDSF in batch

Using batch processing, you can issue often-repeated SDSF commands by creating a list of the commands as control statements. In the list, you specify the SDSF panel you wish to use and the operation you wish to perform on it.

Note that you can also invoke SDSF function using the REXX programming language, which provides more power and flexibility. See Chapter 13, "Using SDSF with the REXX programming language," on page 377 for more information.

Invoking SDSF in batch

Invoke SDSF on an EXEC statement with one of two program names:

- SDSF, which supports commands and action characters.
- ISFAFD, which supports commands, action characters, and overtyping of fields on tabular and other panels, such as the print panels.

Follow the EXEC statement with an ISFIN DD for batch input, and an ISFOUT DD for the batch output.

For example, a batch job to invoke program name ISFAFD might use these statements:

```
//       EXEC PGM=ISFAFD
//ISFOUT DD SYSOUT=*
//ISFIN  DD *
```

The DCB attributes for ISFIN are RECFM=FB, LRECL=80, and the BLKSIZE is any multiple of 80. The DCB attribute for ISFOUT is RECFM=FBA. The LRECL is the screen width + 1 , and the BLKSIZE is any multiple of the LRECL.

To change screen width and depth of the batch output, use `PARM='++xxxx,yyyy'`, following the program name, where *xxxx* is the depth of the screen (number of lines) and *yyyy* is the width (number of characters). For example, to set the depth to 32 and the width to 1000, use:

```
//       EXEC PGM=SDSF,PARM='++32,1000'
//ISFOUT DD SYSOUT=*
//ISFIN  DD *
```

If you do not use the `PARM` statement, the width defaults to 132 and the depth to 60. The maximum for width and depth is 9999.

You can change the name of the SDSF server when invoking SDSF in batch. In the following example, the server name is SDSFT.

```
// EXEC PGM=SDSF,PARM='SERVER(SDSFT)'
```

If you add the server name when invoking SDSF in batch, you cannot combine it with changes to the dimensions of the screen.

A return code of 0016 when SDSF is invoked in batch indicates that the user could not be placed in any of the groups defined with ISFPARMS. See for a description of ISFPARMS.

© Copyright IBM Corp. 1997, 2016

Specifying that SDSF should process JES2

When you invoke SDSF with either program name SDSF or ISFAFD, SDSF determines whether to process JES2 or JES3. You can request that SDSF not do that determination and process JES2. For this purpose, use the alternate program name SDSF2 or ISFAFD2.

Using program name SDSF

SDSF panels and commands

To access a panel and display its contents, use the panel command and ++ALL. For example, to select the H panel and display its contents, use:

```
H
++ALL
```

When ++ALL is specified, anything else on the card is ignored.

To move around on the panel, you can use scroll commands (RIGHT, LEFT, UP, DOWN, TOP, BOTTOM).

Use any SDSF command as you would enter it on the command line, following the syntax described in the online help. The maximum length of a command is 42 characters: only the first 42 characters of each record in ISFIN will be processed. Note that you cannot use commands that require ISPF, such as commands that display pop-ups.

Action characters

To use an action character, code ++*action-character* in your batch job.

To prevent a confirmation pop-up from being displayed for destructive action characters, use the SET CONFIRM OFF command.

You must do a successful FIND prior to issuing an action character. This protects you from issuing an action character against the wrong row.

To allow for an unsuccessful FIND, you should follow each action character with a RESET command, which clears pending action characters. For example, to find job jobxyz on the O panel, browse it with the S action character and issue a RESET in case the job is not found, you would use:

```
O
FIND 'jobxyz'
++S
RESET
```

Using program name ISFAFD

When you invoke SDSF with program name ISFAFD, it works the same as when you invoke it with program name SDSF, with these differences:

- Action characters do not require a successful FIND
- Overtypes and PF keys are supported
- The contents of a panel are not updated until you explicitly refresh the panel. You do this with the AFD REFRESH command.

- Attribute bytes (used to define characteristics of fields such as color and conditioning for input) are present on the SDSF panels. These attribute bytes are translated out when you invoke SDSF with program name SDSF.

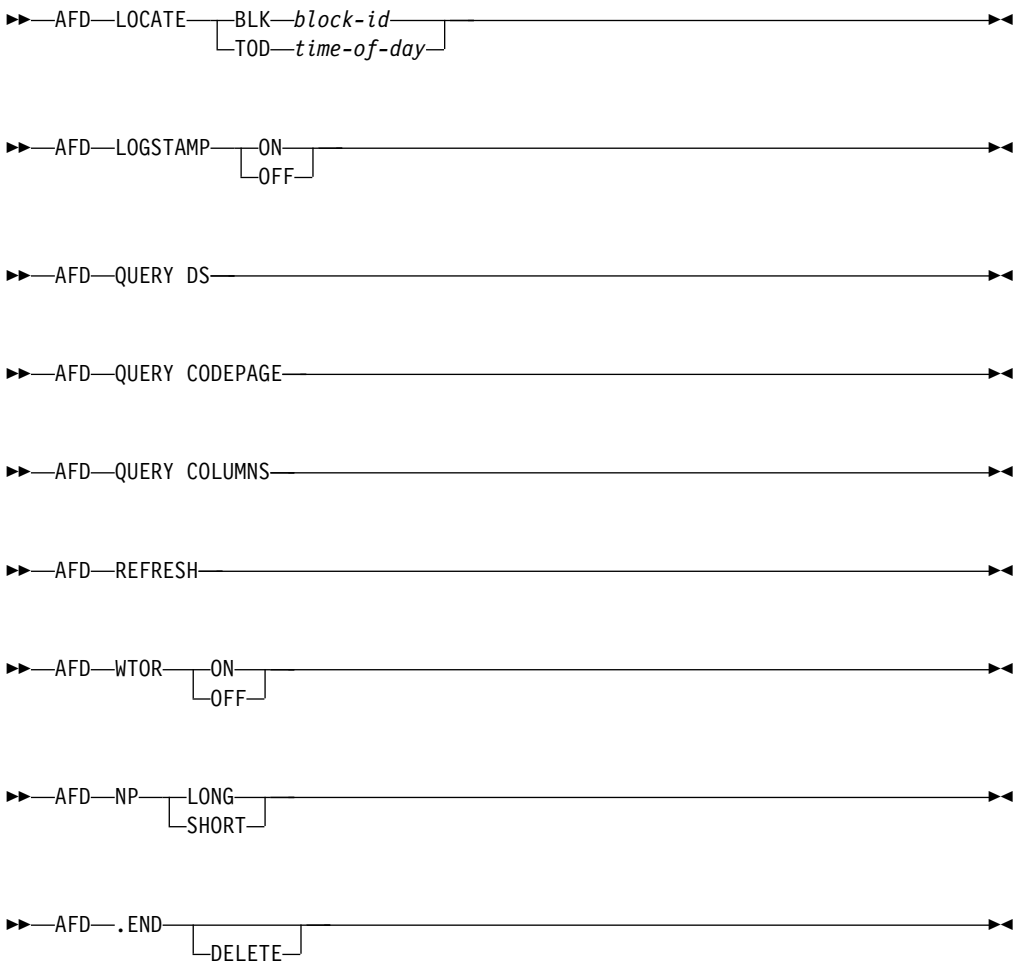
Commands

With program name ISFAFD, you can use the SDSF commands as you would with program name SDSF. You can also use the AFD command, which is described on page "AFD command."

AFD command

Use the AFD command when running SDSF in batch mode with program name ISFAFD.

The syntax of the command is shown below.



LOGSTAMP
controls the addition of a log stamp prefix for each record in the OPERLOG or SYSLOG when printing the log with SDSF's PRINT function. The logstamp is added only when printing to a ddname (for example, PRINT FILE). LOGSTAMP ON causes the log stamp prefix to be added; LOGSTAMP OFF causes the log stamp prefix to not be added. The log stamp of the OPERLOG is a 32-byte prefix. The log stamp varies with the type of log being processed,

that is, OPERLOG, the logical form of SYSLOG (the default for z/OS V1R11 and higher), or the HASPINDX-based form of SYSLOG, for lower levels of JES2.

The log stamp is described in Table 161.

Table 161. Contents of the Log Stamp

Word	SYSLOG (logical)	SYSLOG (HASPINDX-based)	OPERLOG
1-2	STCKE for record	First time in block	Local TOD value returned by IXGBRWSE
3-4	Job key and data set key	Job key and data set key	Block ID returned by IXGBRWSE
5	Relative record number within data set	Relative record number within data set	Relative record number within block
6	1. Byte 1: level 2. Bytes 2–4: reserved	1. Byte 1: level 2. Byte 2: reserved 3. Bytes 3,4: reserved	1. Byte 1: level 2. Bytes 2–4: reserved
7	Reserved	Reserved	1. Byte 1: Control 2. Byte 2: Color 3. Byte 3: Highlight 4. Byte 4: Intensity
8	Reserved	Reserved	Reserved

LOCATE BLK *block-id*
scrolls the OPERLOG to the first record in the log block identified by *block-id*. *block-id* is 16 hexadecimal digits.

LOCATE TOD *time-of-day*
scrolls the OPERLOG to the first record for the time of day identified by *time-of-day*. *time-of-day* is 16 hexadecimal digits.

QUERY DS
displays information about the current data set or log on the message line. The information includes record count, record length, and carriage control. For SYSLOG and OPERLOG, the information also includes the length of the logstamp. (The record count is not displayed for the SYSLOG or OPERLOG panel. In cases where the record length is not available to SDSF, SDSF uses the maximum record length for the job plus 1, or if that is unknown, the screen width plus 1.) This command is valid only on browse panels.

QUERY CODEPAGE
displays the code page that is in use on the message line. If the installation has defined its own code page in ISFPARMS, rather than naming one in the ISFTR macro or TRTAB statement, the code page value is displayed as `N/A`.

QUERY COLUMNS
displays information about the columns on the current tabular panel, using the message lines. The format is as follows:

- Overtypeable columns: '`title`'=(O,`length`)
- Overtypeable columns with related columns: '`title`'=(O,`length`,`number-of-values`)
- Non-overtypeable columns: '`title`'=(N)

REFRESH
requests that SDSF refresh the current display.

WTOR
controls the display of WTORs at the bottom of the Log panel. WTOR ON turns on the display of WTORs on the Log panel. SDSF shows those WTORs defined for the user by the ACTION command or the ACTION parameter of ISFPARMS. WTOR OFF turns off the display of WTORs on the Log panel.

NP controls the width of the NP column.

NP LONG sets the NP column on all tabular panels to the extended width, which is 10 characters on the PR display and the PUN display, and 5 characters on all other displays.

NP SHORT sets the NP column to the standard width.

.END
assigns a label, .END, to the current top line of the SYSLOG or OPERLOG. `.END` overrides the ending line value when printing the SYSLOG or OPERLOG with the PRINT command.

Use the DELETE keyword to delete a previously assigned label.

Note: You can also temporarily extend the NP column on a single tabular panel by typing a + in the NP column. Then, to reset the NP column, use the RESET command.

Examples:

- `AFD WTOR OFF`

  This command turns off the display of WTORs at the bottom of the Log panel.
- `AFD QUERY DS`

  Entered when the current panel is the SYSLOG, this command displays information about the SYSLOG on the message line, for example:

  ```
  AFD QUERY DS LRECL=130,LSLEN=32,CCTL=NONE
  ```
- `AFD LOCATE BLK 1A45B3218C32D862`

  This command scrolls the OPERLOG panel to the first record for the log block with an ID of X'1A45B3218C32D862'.
- `AFD NP LONG`

  This command sets the width of the NP column on all SDSF tabular displays to the extended width.
- `AFD QUERY CODEPAGE`

  This command displays the code page in use on the message line, for example:

  ```
  AFD QUERY CODEPAGE=CP00037
  ```
- `AFD .END`

  This command assigns the label .END to the current top line of the SYSLOG or OPERLOG. To use this label with PRINT, you could then:

  1. Scroll the log so that the current top line is the line with which you want to begin printing.
  2. Issue `PRINT * 99999999`

  SDSF would then print from the current top line to the line that was previously marked with .END.

PF keys

With program name ISFAFD, you can use selected PF keys by coding `++AFD PF`*xx*, where *xx* is the 2-digit PF key number. For example, to perform a repeat-find, you would code:

```
++AFD PF05
```

The PF keys you can use are:

PF03 End the current panel

PF05 Repeat the previous FIND

Action characters

The syntax for action characters is the same as for program name SDSF: see "Action characters" on page 370. However, because a successful FIND is not required, the action character will always be issued against the top row on the panel. To avoid issuing action characters against the wrong row, you might want to first set filters to be sure that only the appropriate row or rows is displayed.

The block action character (//) is not valid with program name ISFAFD.

Overtypeable fields

You can overtype columns on tabular panels and on other SDSF panels, such as panels for printing.

Overtyping columns on tabular panels

You can overtype columns on any tabular panel except OD. The syntax for overtyping columns on tabular panels is the column title followed by = and the new value, all within <>. Enclose the column title and value in single quotation marks.

For example, on the O display, to change the forms for job JFROSTA to STD, change the destination to KGNVMC.JFROST, and refresh the screen, you would use:

```
O
FIND 'JFROSTA'
++<'FORM'='STD'><'DEST'='KGNVMC.JFROST'>
AFD REFRESH
```

You can abbreviate column titles to the shortest title that is unique for the display. If you want the overtypes to be continued on the next card, use a trailing comma.

Where it is valid when using SDSF interactively, you can combine an action character and overtypes; the action character must precede the overtypes. For example, on the H display, to release job SMOSES with the O action character, change the class to A, and refresh the screen, you would use:

```
H
FIND 'SMOSES'
++O<'C'='A'>AFD REFRESH
```

Although you cannot overtype output descriptors on the OD panel, you can overtype most of them on the JDS panel. The JDS panel supports only the first value for output descriptors with multiple values (such as ADDRESS and NOTIFY). To modify the other values for these fields, overtype the first value with a +, then specify the values on the Overtype Extension pop-up. To erase an output descriptor on the JDS panel, type a comma (,) in the field.

Overtyping fields on other panels

You can overtype fields on any other panels that do not require ISPF, such as the print panels, the system command extension pop-up, and the Overtype Extension pop-up.

The syntax for providing values on other types of SDSF panels is similar to the syntax for overtyping fields on tabular panels, except that no column name is used, only =*value*, within <>. The values are positional; in other words, the first value supplied goes into the first field on the panel, the second value supplied goes into the second field on the panel, and so on. On panels with a command line (for example, the print panels), the command line is not counted as an input field.

Use `++AFD END` or `++AFD PF03` to end processing of the panel.

For example, on the Open Print panel, to specify H as the class and 3 as the number of copies (the first and second fields) you would use:

```
PRINT S
++<='H'><='3'>
++AFD PF03
```

To skip a field on the panel, specify < > with no enclosed text. For example, on the Open Print panel, to specify H as the class and STD as the forms (the first and third fields), you would use:

```
PRINT S
++<='H'>< ><='STD'>
++AFD PF03
```

To blank a field, specify <=' '> (a blank enclosed in single quotation marks).

When entering a data set name on the Open Print Data Set panel, enclose it in three sets of single quotes to indicate that it is a fully qualified name. Enclose the data set name in one set of single quotes if you want the TSO prefix to be added.

Notes on using program name ISFAFD

- You can use a trailing comma as a continuation character, so that you can continue overtypes across several cards. The continuation character is required when overtypes that must be processed together (for example, values on a print panel) are specified on multiple cards. To enter a data set name, member name, and disposition on the Open Print Data Set panel, you could use:

  ```
  PRINT D
  ++<='droyek.sdsfdata.december'>,
  <='report'>,
  <='old'>
  ++AFD PF03
  ```

- You can include blank lines, or comments, enclosed in `/* */` on separate lines; they will be ignored when the input is processed.
- To avoid an error message (`AFD CURSOR`*row,column*) set SET CURSOR to OFF, so that the cursor always returns to the command line.

Security and SDSF in batch

To protect use of SDSF in batch you control which group of users a user is assigned to. You do this either through SAF or ISFPARMS. SAF is recommended because it is dynamic and because it allows you to assign users to the same group regardless of the environment from which they invoke SDSF (interactive, batch, REXX or Java).

Using SAF

To use SAF for determining group membership, you assign a name to the group. SDSF then checks the SAF resource GROUP.*group-name.server-name*. This is explained in detail in "Using SAF to control group membership" on page 36.

Using ISFPARMS

You can use parameters in the ISFGRP macro or GROUP statement to determine group membership. These allow you to control membership based on user ID, logon procedure, terminal name, or TSO authority.

When an SDSF batch session is started, it establishes the following values for these criteria:

User ID
: Set to the user ID from the ACEE (accessor environment element), provided it contains a valid user ID **OR** Set to the job name minus the last character.

Logon proc name
: Set to `BATCH` for program name SDSF, and `AFD` for program name ISFAFD.

Terminal name
: Set to `BATCH` for program name SDSF, and the LU name for program name ISFAFD.

TSOAUTH for ISFGRP
: Set to JCL authority.

So, for example, to restrict a group from running SDSF in batch, you could code an XLPROC keyword on ISFGRP to exclude the logon procedure name `BATCH`. Similarly, you could code an ILPROC keyword to assign batch jobs to a specific ISFGRP.

Figure 14 contains sample ISFPARMS statements to assign SDSF batch jobs to the group ISFBATCH.

```
          ISFPMAC
ISFSPROG  ISFGRP TSOAUTH=(JCL,OPER,ACCT),...
ISFOPER   ISFGRP TSOAUTH=(JCL,OPER),...
ISFUSER   ISFGRP TSOAUTH=(JCL),...,XLPROC=BATCH
ISFBATCH  ISFGRP TSOAUTH=(JCL),...,ILPROC=BATCH
BATCH     ISFNTBL BATCH,1
```

Figure 14. Sample ISFPARMS to Restrict Batch

Chapter 13. Using SDSF with the REXX programming language

Programming Interface Information

This topic describes how to access SDSF data and function with the REXX programming language, and how to protect the use of SDSF through REXX.

Using SDSF with REXX provides a simpler and more powerful alternative to using SDSF in batch, which is described in Chapter 12, "Using SDSF in batch," on page 369.

Table 162 outlines how to access SDSF function with REXX.

Table 162. Using SDSF with REXX

To:	Use:	For more information:
Add and delete the SDSF host command environment	ISFCALLS()	"Adding the SDSF host command environment with ISFCALLS" on page 379
Issue SDSF commands to access tabular panels and other information	ISFEXEC	"Issuing commands with ISFEXEC" on page 380
Issue action characters and overtype columns	ISFACT	"Issuing action characters and modifying columns with ISFACT" on page 388
Browse output	ISFBROWSE or ISFACT and special variables	"Browsing output" on page 394
Print output	ISFACT and special variables	"Printing output" on page 399
Browse the SYSLOG and OPERLOG	ISFLOG	"Browsing the system log with ISFLOG" on page 403
Issue system commands	ISFSLASH	"Issuing system commands with ISFSLASH" on page 408
Issue SDSF commands for filtering and options, and check messages	Special REXX variables	"Using special variables to invoke SDSF function" on page 410
Drop specified special variables	ISFRESET()	"Dropping special variables with ISFRESET" on page 414

For examples of REXX execs, refer to "Examples of REXX execs" on page 429.

You must be authorized to use SDSF with REXX and you must be authorized to the SDSF functions that you invoke from REXX. In some cases, invoking an SDSF function from REXX when you are not authorized to the function will cause the exec to fail and the invocation of SDSF to end.

System programmers should be sure to define ISFPARMS group membership so that SDSF users have the proper authorization when invoking SDSF with REXX. For more information, see "Security and REXX" on page 448

Other sources of information

In addition to this information, you may want to refer to these other sources for information about using REXX with SDSF:

© Copyright IBM Corp. 1997, 2016

- REXXHELP. Type this command (or REXXH for short) on any command line when using SDSF under ISPF. In addition to examples and usage information, the online help for REXX also includes links to descriptions of commands, action characters and overtypable columns and column values, which is not included in this information.

  To search SDSF's help, including the help for REXX, use the SEARCH command. You can type SEARCH followed by up to four words on the SDSF command line when using SDSF under ISPF.

  If you are not already familiar with SDSF, you should begin with the SDSF help. To display a brief, interactive tutorial, use the TUTOR command.
- ISPF models that you can download from the Internet. In addition to the same examples as are included in this information, the models help with the syntax of REXX commands such as ISFEXEC and ISFACT. See the SDSF page at http://www.ibm.com/systems/z/os/zos/features/sdsf/.
- Implementing REXX Support in SDSF, SG24-7419-00. This Redbook includes more complete and sophisticated examples than those in this information. The following is a brief table of contents:
  - Chapter 1. Issuing a system command
  - Chapter 2. Copying SYSOUT to a PDS
  - Chapter 3. Bulk job update processor
  - Chapter 4. SDSF support for the COBOL language
  - Chapter 5. Searching for a message in SYSLOG
  - Chapter 6. Viewing SYSLOG
  - Chapter 7. Reviewing execution of a job
  - Chapter 8. Remote control from other systems
  - Chapter 9. JOB schedule and control
  - Chapter 10. SDSF data in graphics
  - Chapter 11. Extended uses
  - Appendix A. REXX variables for SDSF host commands
  - Appendix B. Additional material

Programming practices

Be aware that many of the things you work with in a REXX exec, such as the list of columns on an SDSF panel, the contents of the title line of a panel, and the contents of responses to SDSF commands such as WHO, may change over time. You should design your REXX execs to minimize the impact of those changes. For example, rather than making assumptions about the contents of a panel, you can query special REXX variables that SDSF provides.

Following these guidelines for variable names will reduce the potential for conflicts between REXX variables you create and special and column variables used by SDSF:

- Do not use variable names that begin with ISF. SDSF reserves that prefix for the names of special REXX variables.
- Use the PREFIX option of the ISFEXEC and ISFACT commands to force unique variable names. See the description of options in “Issuing panel commands with ISFEXEC” on page 382 for more information.
- Isolate SDSF environment calls to a REXX procedure to limit the scope of the variable names.

- When referencing a panel command that contains embedded blanks or special characters (such as on ISFEXEC and ISFACT), enclose the command in single quotes. When referencing the PARM panel on ISFACT, enclose the panel name in single quotes so that it is not interpreted as the PARM keyword of ISFACT.

Remember that SDSF may add special variables and columns with a new release or service, so that even if you do not currently have a conflict with variable names, one could occur in the future. To reduce your risk, always specify the columns to be returned using the ISFCOLS special variable.

Quick start

Procedure

In a very simple REXX exec, you might do the following:

1. Add the SDSF host command environment.
   ```
   rc=isfcalls('ON')
   ```
2. Access a panel with "ISFEXEC *panel-command*". This creates stem variables for each row and column on that panel. To access the Status panel, you could use:
   ```
   Address SDSF "ISFEXEC ST"
   ```
3. Find the job you want to work with by examining the JNAME stem variables created for the JOBNAME column. (You refer to columns not by their titles, but by the same names that you would use in defining a field list in ISFPARMS. See Chapter 4, “Columns on the SDSF panels,” on page 133.)
   ```
   do ix=1 to JNAME.0    /* Loop for all rows returned */
     if pos("RJONES",JNAME.ix) = 1 then
   ```
4. Take an action or modify a value for the job with "ISFACT *operands*". *operands* is made up of:
   - The panel command that you used previously with ISFEXEC
   - A TOKEN.*number* variable that was created by the ISFEXEC command and identifies the row that represents the job
   - Parameters that define the action or modification. In this example, you supply the P action character in the NP column to cancel the job.
     ```
     Address SDSF "ISFACT ST TOKEN('"TOKEN.ix"') PARM(NP P)"
     ```
5. Delete the host command environment (after closing the do loop).
   ```
   end
   rc=isfcalls('OFF')
   ```

What to do next

Of course, in an actual exec, you would have more complex logic and error checking. This would require the use of special REXX variables to do things like examine messages issued, filter rows on the panel, and define the columns to include. For more examples, see “Examples of REXX execs” on page 429.

Adding the SDSF host command environment with ISFCALLS

Using SDSF with REXX requires that you add a host command environment prior to any other SDSF host environment commands. The host command environment is what allows you to use `Address SDSF` on the ISFEXEC and ISFACT commands. You add the host command environment with the ISFCALLS() function.

You should delete the host command environment, again using ISFCALLS, prior to the termination of the exec.

The syntax of the ISFCALLS() function is:

ON adds the SDSF host command environment

OFF
deletes the SDSF host command environment

SSTYPE=JES2
requests that SDSF process JES2 rather than determining whether to process JES2 or JES3.

Result codes

The ISFCALLS() function sets the following result codes:

00 Function completed successfully

01 Host command environment query failed, environment not added

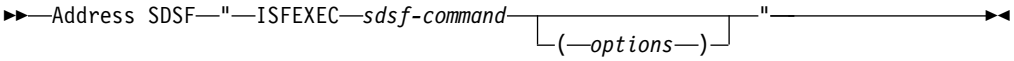
02 Host command environment add failed

03 Host command environment delete failed

04 Options syntax error, or options not defined

Issuing commands with ISFEXEC

You issue commands with the ISFEXEC host command as follows:



sdsf-command
is a supported SDSF command, including any parameters. If the command contains special characters or blanks, enclose it in single quotation marks. The supported commands are:

- The commands that access SDSF tabular panels (for example, DA and ST). For more information, see "Issuing panel commands with ISFEXEC" on page 382.
- The WHO and QUERY commands. For more information, see "Issuing WHO and QUERY commands with ISFEXEC" on page 387.
- The slash (/) command, which allows you to enter system commands. Although this is supported, the recommended method for issuing system commands is with ISFSLASH. For more information, see "Issuing system commands with ISFSLASH" on page 408 or "Issuing system commands with ISFEXEC" on page 387.

Commands entered with the ISFEXEC command generally have a maximum length, including any parameters, of 42 characters (the same as the command input area when using SDSF interactively). Slash (/) commands entered with the ISFEXEC command can have operands up to 126 characters long.

Note that for function associated with other SDSF commands, such as filtering and setting options, you use special variables rather than ISFEXEC. See "Using special variables to invoke SDSF function" on page 410.

For a complete list of the SDSF commands, see “SDSF commands reference” on page 414. For the syntax of the commands, see the online help.

options
is an optional list of options for the command. The closing parenthesis is optional. The options that you use depend on the type of the command you issue, and are explained in the topics that follow. The following option is of general use as you develop a REXX exec:

VERBOSE
adds diagnostic messages to the ISFMSG2 stem variable. The messages describe each row variable created by SDSF.

Examples of using ISFEXEC

The following examples illustrate how to issue a command with ISFEXEC. For more complete examples, see “Examples of REXX execs” on page 429.

1. Issue the DA command and create variables for the DA panel.
   ```
   Address SDSF "ISFEXEC DA"
   ```
2. Issue the CK command with the ALL parameter and create variables for the CK panel.
   ```
   Address SDSF "ISFEXEC CK ALL"
   ```
3. Issue the ST command and create variables for the alternate field list.
   ```
   Address SDSF "ISFEXEC ST (ALTERNATE)"
   ```
   Note: Delayed-access columns are not included. These require the DELAYED option.
4. Issue the ST command and create variables for the alternate field list, including delayed-access columns.
   ```
   Address SDSF "ISFEXEC ST (ALTERNATE DELAYED)"
   ```
5. Issue the O command, with filters for class A and forms 1234.
   ```
   Address SDSF "ISFEXEC OA 1234"
   ```
6. Issue the WHO command.
   ```
   Address SDSF "ISFEXEC WHO"
   ```

Return codes for ISFEXEC

After the ISFEXEC host environment command completes, a return code is set in the REXX variable RC. The values are:

00 The request completed successfully.

04 The request completed successfully but not all functions were performed.

08 An incorrect or invalid parameter was specified for an option or command.

12 A syntax error occurred in parsing a host environment command.

16 The user is not authorized to invoke SDSF.

20 A request failed due to an environmental error.

24 Insufficient storage was available to complete a request.

Note that a return code of 0 indicates that SDSF successfully processed the ISFEXEC command. It does not indicate that specific functions were authorized or that commands were executed. Check the ISFMSG and ISFMSG2 variables to determine if a request completed. See “Messages” on page 382 for more information.

Messages

Messages issued in response to a command or special variable are available in these special variables:

ISFMSG
contains the SDSF short message

ISFMSG2
is a stem variable that contains SDSF numbered messages. ISFMSG2.0 contains the number of stem variables that follow.

Issuing panel commands with ISFEXEC

You can issue the commands that access SDSF tabular panels with ISFEXEC. Tabular panels display data in rows and columns.

For information on non-tabular panels, see:

- "Browsing the system log with ISFLOG" on page 403
- The discussion of the ISFULOG special variable in "Issuing system commands with ISFSLASH" on page 408.

ALTERNATE
requests the alternate field list. For a discussion of primary and alternate field lists, see "Variable field lists (FLD or ISFFLD)" on page 85.

DELAYED
specifies that delayed-access columns be included. Delayed-access columns require I/O to retrieve the data. If you do not include this option, delayed-access columns are omitted. Omitting delayed-access columns may improve performance. For information on which columns are delayed-access, see

- Chapter 4, "Columns on the SDSF panels," on page 133
- The COLSHELP command in SDSF

NOMODIFY
specifies that row tokens for use in modifying rows should not be returned. Use this to improve performance if you will not be modifying any values.

PRIMARY
requests the primary field list.

If you specify both PRIMARY and ALTERNATE, or neither PRIMARY nor ALTERNATE, the primary and alternate field lists are merged. For a discussion of primary and alternate field lists, see "Variable field lists (FLD or ISFFLD)" on page 85.

PREFIX *value*
specifies a prefix for column name and TOKEN variables that are created; use this to ensure that variable names do not conflict. The prefix can be up to 24 characters long, and should not begin with ISF. It does not affect the names of special variables.

VERBOSE
adds diagnostic messages to the ISFMSG2 stem variable. The messages describe each row variable created by SDSF.

Controlling the columns included on panels

By default, tabular panels accessed with REXX include the columns in both the primary and alternate field lists defined in ISFPARMS, except any "delayed-access" columns. You can control the columns that are included on SDSF panels as

described in Table 163. Limiting the columns that are included limits the columns for which SDSF creates REXX variables. Limiting the columns to just those that are required can make the exec process more quickly.

Table 163. Controlling the Columns on SDSF Panels

To Specify:	Use:	Default:	For More Information:
Primary, alternate or merged field list	Options on ISFEXEC	Merged	"Options for panel commands"
Delayed-access columns	Option on ISFEXEC	Not included	"Options for panel commands"
List of columns by column name	ISFCOLS variable		"Special variables for panels and panel commands" on page 385

Options for panel commands

You can use the following options with panel commands on ISFEXEC. Combine the options if necessary. For example, you could specify both ALTERNATE and DELAYED to include delayed-access columns that are in the alternate field list. Note that by default, the primary and alternate field lists are both included. That is, if you specify both PRIMARY and ALTERNATE, or neither PRIMARY nor ALTERNATE, the primary and alternate field lists are merged.

Panel data returned

The ISFEXEC command returns data as follows:

- Column data. The column data is returned in stem variables in this format: *column-name.row-number*, where:

  column-name
  : is the name of the column. The first column returned is always the fixed field. The column name is different than the column title that is displayed when using SDSF interactively. It is the same name that is used in the FLD statements in ISFPARMS. For more information:

    – Refer to Chapter 4, "Columns on the SDSF panels," on page 133 for a list of column names and titles

    – When running SDSF under ISPF, issue the COLSHELP command. COLSHELP provides column names, titles, descriptions and information about values.

    – SDSF online help, for column titles, plus information about values for overtypeable and other columns.

    If you specify a prefix with the PREFIX option, the column-name variable begins with the prefix. For an example, see "List job data sets" on page 432.

  row-number
  : is the row number.

    The value for stem variable number 0 is a count of the number of variables returned. This count is the same for all columns. It is also in special variable ISFROWS.

    For overtypeable columns with related values, a sub-stem is added to the row number to indicate the number of the related value, as follows:

    column-name.row-number.value-number

    So, for example, the SFORMS column in the PR panel has values SFORMS.1.0 (which contains a count of the values) and SFORMS.1.1 through SFORMS.1.8. The value in SFORMS.1.2 is displayed in column SFORM2.

SDSF panel data is the same in the REXX environment as in the interactive environment, with a few exceptions. For details, see "Data formats - differences between REXX and interactive SDSF."

The related ISFCOLS and ISFTITLES special variables return column names and column titles. See "Special variables for panels and panel commands" on page 385 for more information.

- Title line. The contents of the title line are returned in the ISFTLINE special variable. The title line includes the name of the panel and, in some cases, additional information. For a description of the contents of the title line for an SDSF panel, see the help for fields for the panel.
- Tokens to identify each row. These are returned in the TOKEN stem variable. For example, variable TOKEN.2 contains a string that identifies row two on the panel being processed.

  If you specify a prefix with the PREFIX option, the name of the stem variable containing tokens begins with the prefix. For example, if the prefix is JDS_, the name of the stem variable is JDS_TOKEN.

  Use the token as input to the ISFACT command when taking an action or modifying a value for that row. See "Issuing action characters and modifying columns with ISFACT" on page 388 for more information.

Example:
The following example shows data returned in the stem variables.

```
JNAME.0=45
JOBID.0=45
OWNERID.0=45
     .
     .
remaining 0 variables
     .
     .
JNAME.1=BURDINE3
JOBID.1=JOB04922
OWNERID.1=BURDINE
     .
     .
remaining variables
     .
     .
```

This example shows data for a column with related values, the SFORMS column on the Printer panel.

```
SFORMS.1=STD
SFORMS.1.1=STD (This the same value as is in SFORMS.1)
SFORMS.1.2=NAR
SFORMS.1.3=REC
.
.
```

Data formats - differences between REXX and interactive SDSF: SDSF panel data is the same in the REXX environment as in the interactive environment, with a few exceptions.

- Numbers:
  - Do not include commas.
  - Are never scaled, as they are not restricted by column widths. They never include scaling characters such as T or M. However, some values are formatted with units. For example, values in the MemLimit column on the DA panel are formatted with MB, PB and so on.

  - Are formatted as three asterisks in cases of invalid or overflow data that would be displayed as all asterisks when SDSF is used interactively.
- Dates and times:
  - If formatted by SDSF, are in *yyyy.ddd* format (dates) and either *hh*:*mm*:*ss* or *hh*:*mm*:*ss.th* format (times). To convert them to a different format, you can use the REXX date() function.
  - Are formatted as N/A in cases of invalid dates that would be displayed as N/A embedded in asterisks when SDSF is used interactively.

Special variables for panels and panel commands

There are a number of special variables that are useful when working with panels and panel commands. Where the variable corresponds to an SDSF command that you would use when using SDSF interactively, the parameters for the variable are the same as for the command, with the exception that the ? parameter is not supported in REXX. Substitute the variable for the command, for example:

```
Command: PREFIX NEIL*
Variable: isfprefix="neil*"
```

For more information on special REXX variables, see "Using special variables to invoke SDSF function" on page 410 and "Special variables reference" on page 420. For the syntax of SDSF commands, see the online help.

For secondary panels like JDS that you access with an action character from another panel, you use different special variables than the ones described in this topic. See "Using secondary panels" on page 392.

Use these special variables when working with panels and panel commands:

ISFACTIONS
: specifies whether the action characters for the current panel should be returned in the ISFRESP stem variable. The values in the ISFRESP variable are in this format: ACTION=*action*, where *action* is the action character or the action character and a description, depending on the option specified on ISFACTIONS. See the SET ACTION command in the online help for the valid options. See "List action characters" on page 444 for an example.

ISFAPPC
: specifies whether transaction data should be included on the panel. See the APPC command in the online help. (JES2 only)

ISFCOLS
: **Input**: Specifies the set of columns for which SDSF should create variables, in this format:

  '*column-name column-name*...'

  The column names are different than the column titles that are displayed when using SDSF interactively. They are the names used in the FLD statements in ISFPARMS. For a list of column names, see Chapter 4, "Columns on the SDSF panels," on page 133, or, when running SDSF under ISPF, issue the COLSHELP command.

  Each column name you specify must exist in the current field list. Any name specified in the ISFCOLS variable that is not in the current field list will be ignored. The order of the columns is not significant. See "Controlling the columns included on panels" on page 382 for more information.

The fixed field (the first column on each SDSF panel when using SDSF interactively) is optional, since it will always be included regardless of the setting of ISFCOLS.

If the ISFCOLS variable is not defined, SDSF creates variables for each column in the field list that is not delayed-access, including the fixed field.

Output: Lists the columns that were processed, in this format:

column-name column-name...

The names are separated by a blank. The fixed field is always listed first.

When working with a secondary panel (a panel accessed with an action character) use the ISFCOLS2 variable. See "Using secondary panels" on page 392 for more information.

ISFCOLUMNGROUPS
contains a list of column grouping information for the columns listed in the ISFCOLS variable. The group values are a way of categorizing SDSF columns. The values are: NONE, ACCT (accounting), ACTIVITY, ADVANCED, GENERAL, INPUT, JES2, JES3, OUTPUT (printer), OUTPUN (punch), PERF (performance), PRINTING, RUNTIME, SECURITY, SCHED (scheduling), SELECT, STATUS and STATWLM (workload management status).

ISFDCOLS
contains a list of the delayed-access columns that were returned and for which SDSF should create variables, in this format:

column-name column-name...

When working with a secondary panel (a panel accessed with an action character) use the ISFDCOLS2 variable. See "Using secondary panels" on page 392 for more information.

Unlike ISFCOLS, ISFDCOLS is an output-only variable.

ISFDISPLAY
contains the filtering and sorting criteria, for example,

```
PREFIX=* DEST=(ALL)  OWNER=*  SYSNAME=SYS1
```

See the SET DISPLAY command in the online help.

ISFDISPLAYMODE
sets the format of the ISFDISPLAY special variable. See the SET DISPLAY command in the online help. The OFF parameter is not valid in REXX.

ISFRCOLS
contains the list of columns that have related values. For information on modifying related values, see "Modifying related fields" on page 389.

ISFROWS
contains the number of rows created for a tabular panel. (This is also found in the zero stem of the column variables, for example, JNAME.0.)

ISFSORT
specifies the sort criteria (up to 10 columns, with ascending or descending order). Use column names rather than column titles. Assigning the value to null (`isfsort=""`) sorts the panel using the fixed field (the first column). See the SORT command in the online help for the syntax.

ISFTIMEOUT
specifies the response timeout value for sysplex requests. See the SET TIMEOUT command in the online help.

ISFTITLES
contains the column titles for the columns on the panel. The titles are listed in the same order as the column names in the ISFCOLS variable. The titles are enclosed in single quotation marks and separated by blanks.

When working with a secondary panel, accessed with an action character, use the ISFTITLES2 variable. See "Using secondary panels" on page 392 for more information.

ISFTLINE
contains the title line from the tabular panel being processed.

ISFUCOLS
contains the list of modifiable columns for the panel. All modifiable columns are included, regardless of whether the user is authorized to modify them.

When working with a secondary panel, accessed with an action character, use the ISFUCOLS2 variable. See "Using secondary panels" on page 392 for more information.

ROWACTIVE
is a stem variable that indicates whether the object (for example, the job or the printer) is active. The value is either Y (active) or N (inactive). ROWACTIVE.0 contains a count of the number of stem variables that follow.

Issuing WHO and QUERY commands with ISFEXEC

You can issue the WHO and QUERY commands with ISFEXEC:

- WHO provides information about the user and the environment
- QUERY lists SDSF data such as the commands for which you are authorized.

Responses are returned in the ISFRESP stem variable. For the WHO command, the responses are in *keyword=value* format, for example, USERID=RJONES. For more information on using special REXX variables, see "Using special variables to invoke SDSF function" on page 410.

For a description of the WHO and QUERY commands, see the online help.

For an example, see "Issue the WHO command" on page 447.

Issuing system commands with ISFEXEC

Although using ISFSLASH is the recommended method, you can use ISFEXEC to issue slash (/) commands.

To preserve lowercase and special characters in the command text, enclose it in single quotation marks, being sure that the quotation marks are passed to SDSF and not removed by REXX, for example:

```
Address SDSF "ISFEXEC '/f test,''abc'''"
```

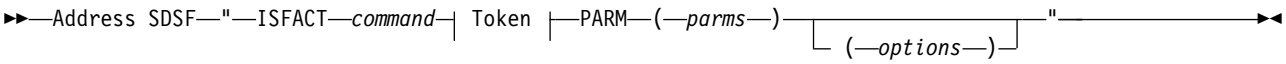
The W and I prefix parameters of the slash (/) command are not supported. Use the WAIT and INTERNAL options instead. See "Options for slash (/) commands" on page 409 for more information.

For a description of special variables to use with slash (/) commands, see "Special variables for slash (/) commands" on page 409.

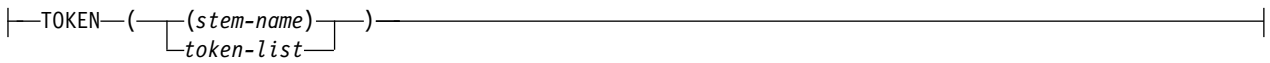
For information on using ISFSLASH, see "Issuing system commands with ISFSLASH" on page 408.

Issuing action characters and modifying columns with ISFACT

You invoke SDSF action characters and modify column values using the ISFACT host environment command, as follows:



Token:



command
is the command for the panel. It must be the same SDSF command, including any parameters, that was previously entered with the ISFEXEC command. When referencing the PARM panel, enclose PARM in single quotes to avoid ambiguity with the PARM option.

stem-name
is the name of a stem variable that identifies the rows to be acted upon. The stem variable contains one or more row tokens previously set by ISFEXEC or ISFACT in the returned TOKEN. stem variable and must correspond to the panel accessed with *command*. The tokens must not be folded to upper case or enclosed in single quotation marks. For more information on tokens, see "Using tokens" on page 390. The variable *stem-name* should:

- End with a period, to allow the commands to be put into compound variables
- Not begin with the characters ISF
- Be no longer than 128 characters

The 0 variable in the stem must contain a count of the number of variables in the stem.

token-list
is one or more tokens that identifies the row to be acted upon, in the format *'token1', 'token2', ...,'tokenN'*. Each token was previously set by ISFEXEC or ISFACT in the returned TOKEN. stem variable and must correspond to the panel accessed with *command*. Enclose the token in single quotation marks that are not removed by REXX.

For more information, see "Using tokens" on page 390.

parms
is the list of parameters that specifies the action characters and modifications, in the form:

column1 value1 column2 value2 ... columnN valueN

where

column1, column2, columnN
are either:

- NP, when issuing an action character
- column names, when modifying values. The column names are different than the titles that are displayed when using SDSF interactively. They

are the same names that you use on FLD statements in ISFPARMS. For a list of column names, see Chapter 4, "Columns on the SDSF panels," on page 133, or, when running SDSF under ISPF, issue the COLSHELP command.

The column must be in the current field list for the panel; use column-related options on the ISFACT command, such as ALTERNATE, if necessary. For more information, see "Controlling the columns included on panels" on page 382.

If you name a column multiple times, SDSF processes only the last one.

value1, value2, valueN
are either:

- an action character, when the column is NP. The SDSF action characters are described in the online help. Most of the action characters are supported with REXX. Table 168 on page 418 shows the exceptions. The action characters for browsing and printing output have special restrictions and requirements. See "Browsing output" on page 394 and "Printing output" on page 399.
- a value, when modifying a value in a column other than NP. If the value contains special characters, you must enclose it in quotation marks. Lowercase characters are folded to upper case, even if they are enclosed in quotation marks.

  The fields that can be modified, or overtyped, are described in the help for each panel.

  For information on modifying sets of related fields, see "Modifying related fields."

The resulting command cannot exceed the maximum allowed by z/OS.

options
is an optional list of options. See "Options for action characters and overtypeable fields" on page 391 for more information.

Modifying related fields

When working with sets of related fields, such as the four selection destinations on the Printer panel, add a plus (+) before the column name to indicate that the value is in addition to any other values for the same column. Use this syntax for each value. When using SDSF interactively, you work with related fields through the overtype extension pop-up, which you access by typing the + character in the overtypeable column.

For example, `PARM(SDESTN1 D1 +SDESTN1 D2 +SDESTN1 D3)` indicates that the SDESTN1 column is to be modified with the values D1,D2,D3.

SDSF accepts a + sign for the first column in the set of columns, for example, `PARM(+SDESTN1 D1 +SDESTN1 D2)`. This is equivalent to `PARM(SDESTN1 D1 +SDESTN1 D2)`. However, subsequently specifying the first column in the set without a + sign resets the values. For example, `PARM(SDESTN1 D1 +SDESTN1 D2 SDESTN1 D11)` would result in the column being modified with the single value D11. This is because SDSF processes the last occurrence of the column name. Since the last occurrence does not have the + sign, it is interpreted as a complete replacement.

If the same column is specified more than once, the last occurrence is used for the action except when the + sign is used with the column name.

Special variables ISFRCOLS and ISFRCOL2 contain lists of columns with related fields for the current panel and a secondary panel, respectively.

Using tokens

A token consists of a variable-length string that may contain special characters. You must not modify it.

A token cannot be shared by different users. The user who references a token with a host command must be the same user who created it.

When a token references a secondary panel (such as JDS), all subsequent tokens must also refer to the secondary panel using the same row from the primary panel.

Tokens represent jobs at the time that they are generated and are intended to be used soon after they are generated, rather than saved for later use. If the row to be acted upon no longer exists when the host command is issued, SDSF considers the row token invalid. You should not change the associated panel, for example, by changing filtering.

The format of tokens may change incompatibly with service or new releases of SDSF.

Examples of using ISFACT

The following examples illustrate how to issue an action character and modify a column, after having first issued the appropriate panel command with ISFEXEC. For more complete examples, see "Examples of REXX execs" on page 429.

1. Issue the P action character for row 4 on the H panel.

   ```
   Address SDSF "ISFACT H TOKEN('"TOKEN.4"') PARM(NP P)"
   ```

2. Issue the P action character for rows 1 and 2 on the H panel.

   ```
   Address SDSF "ISFACT ST TOKEN('"TOKEN.1"','"TOKEN.2"') PARM(NP P)"
   ```

3. Issue the P action character for the row the number of which is represented by variable *ix* on the H panel.

   ```
   Address SDSF "ISFACT H TOKEN('"TOKEN.ix"') PARM(NP P)"
   ```

4. Modify the priority of multiple jobs.

   ```
   Address SDSF "ISFACT ST TOKEN((TOKEN.)) PARM(JPRIO 10)"
   ```

   For this type of usage, you would use command parameters or special variables to limit the panel to just those jobs you want to modify. For a complete example, see "Modify a value for a set of jobs" on page 434.

5. Issue the P action character for rows that are identified by tokens in the stem variable JSTEM.

   ```
   Address SDSF "ISFACT ST TOKEN((JSTEM.)) PARM(NP P)"
   ```

   For this type of usage, you would use logic to set the values in the stem variable JSTEM. to the tokens, in stem variable TOKEN., for those jobs you want to modify. For a complete example, see "Modify a value for a set of jobs" on page 434.

6. For row 2 on the O panel, modify the class to A and the forms to 1234.

   ```
   Address SDSF "ISFACT O TOKEN('"TOKEN.2"') PARM(OCLASS A FORMS 1234)"
   ```

7. Allocate all data sets in the job represented by row 5 on the ST panel.

   ```
   Address SDSF "ISFACT ST TOKEN('"TOKEN.5"') PARM(NP SA)"
   ```

Return codes for ISFACT

After the ISFACT host environment command completes, a return code is set in the REXX variable RC. The values are:

00 The request completed successfully.

04 The request completed successfully but not all functions were performed.

08 An incorrect or invalid parameter was specified for an option or command.

12 A syntax error occurred in parsing a host environment command.

16 The user is not authorized to invoke SDSF.

20 A request failed due to an environmental error.

24 Insufficient storage was available to complete a request.

Note that a return code of 0 indicates that SDSF successfully processed the ISFACT command. It does not indicate that specific functions were authorized or that commands were executed. Check the ISFMSG and ISFMSG2 variables to determine if a request completed. See "Messages" on page 382 for more information.

Options for action characters and overtypeable fields

You can use the following options with ISFACT:

ALTERNATE
requests the alternate field list for the panel

ALTERNATE2
requests the alternate field list for the secondary panel

DELAYED
specifies that delayed-access columns be included on the panel

DELAYED2
specifies that delayed-access columns be included on the secondary panel

NOMODIFY2
specifies that row tokens for use in modifying rows should not be returned on the secondary panel. Use this to improve performance if you will not be modifying any values.

PRIMARY
requests the primary field list.

If you specify both PRIMARY and ALTERNATE, or neither PRIMARY nor ALTERNATE, the primary and alternate field lists are merged, and all the column variables for the panel are available.

PRIMARY2
requests the primary field list for a secondary panel.

If you specify both PRIMARY2 and ALTERNATE2, or neither PRIMARY2 nor ALTERNATE2, the primary and alternate field lists are merged, and all the column variables for the panel are available.

PREFIX *value*
specifies a prefix for column name and TOKEN variables that are created; use this to ensure that variable names do not conflict. The prefix can be up to 24 characters long, and should not begin with ISF. It does not affect the names of special variables.

VERBOSE
adds diagnostic messages to the ISFMSG2 stem variable. The messages describe each row variable created by SDSF.

WAIT
specifies that SDSF should wait the full delay interval before retrieving responses to a comand. This option is strongly recommended to ensure the responses are accessible in the ISFULOG special variable. The delay interval is specified with the ISFDELAY variable.

Using secondary panels

Secondary panels are accessed with an action character from another panel. For example, when you use the ? action character from the Status panel to access the Job Data Set (JDS) panel, JDS is a secondary panel. For secondary panels, ISFACT returns column and row data in the same way that ISFEXEC does. See "Panel data returned" on page 383 for more information.

Special variables for secondary panels

There are a number of special variables that you can use when working with secondary panels.

ISFACTIONS
specifies whether the action characters for the current panel should be returned in the ISFRESP stem variable. The values in the ISFRESP variable are in this format: ACTION=*action*, where *action* is the action character or the action character and a description, depending on the option specified on ISFACTIONS. See the SET ACTION command for the valid options. See "List action characters" on page 444 for an example.

ISFAPPC
specifies whether transaction data should be included on the panel. See the APPC command.

ISFCOLS2
Input: Specifies the set of columns on the secondary panel for which SDSF should create variables, in this format:

'*column-name column-name*...'

The column names are different than the column titles that are displayed when using SDSF interactively. They are the names used in the FLD statements in ISFPARMS. For a list of column names, see Chapter 4, "Columns on the SDSF panels," on page 133, or, when running SDSF under ISPF, issue the COLSHELP command.

Each column name you specify must exist in the current field list. Any name specified in the ISFCOLS2 variable that is not in the current field list will be ignored.

The fixed field (the first column on each SDSF panel when using SDSF interactively) is optional, since it will always be included regardless of the setting of ISFCOLS2.

If the ISFCOLS2 variable is not defined, SDSF creates variables for each column on the secondary panel that is in the field list and is not delayed-access, including the fixed field.

Output: Lists the columns on the secondary panel that were processed, in this format:

column-name column-name...

The names are separated by a blank. The fixed field is always listed first.

Note: the column names do not include the prefix.

ISFDCOLS2
contains the list of delayed-access columns for the secondary panel, in this format:

column-name column-name...

ISFDISPLAY
contains the filtering and sorting criteria, for example,

```
PREFIX=* DEST=(ALL)  OWNER=*  SYSNAME=SYS1
```

See the SET DISPLAY command.

ISFDISPLAYMODE
sets the format of the ISFDISPLAY special variable. See the SET DISPLAY command in the online help. The OFF parameter is not valid in REXX.

ISFFILTER2
specifies filter criteria to be applied to the returned variables. Use column names rather than column titles. See the FILTER command in the online help.

ISFRCOLS2
contains the list of related fields (such as Address-Line1 through 4) for the secondary panel, in this format:

column-name column-name...

ISFROWS2
contains the number of rows created for the secondary panel. (This is also found in the column variables, for example, DDNAME.0.)

ISFSORT2
specifies the sort criteria (up to 10 columns, with ascending or descending order). Use column names rather than column titles. Assigning the value to null sorts the panel using the fixed field (the first column). See the SORT command for other syntax.

ISFTIMEOUT
specifies the response timeout value for sysplex requests. See the SET TIMEOUT command. (JES2 only)

ISFTITLES2
contains the column titles for the secondary panel. The titles are listed in the same order as the column names in the ISFCOLS2 variable. Each title is enclosed in single quotation marks and separated by a blank.

ISFTLINE
contains the title line from the tabular panel being processed

ISFUCOLS2
contains the list of modifiable columns for the secondary panel. All modifiable columns are included, regardless of whether the user is authorized to modify them.

ISFULOG
is a stem variable that contains the command echo and responses for system commands generated by action characters, including SAF authorization messages (if supported by the external security manager). Use the WAIT option on the ISFACT command to ensure that the command responses are available in the ISFULOG stem variable.

For more information on special REXX variables, see “Using special variables to invoke SDSF function” on page 410 and “Special variables reference” on page 420.

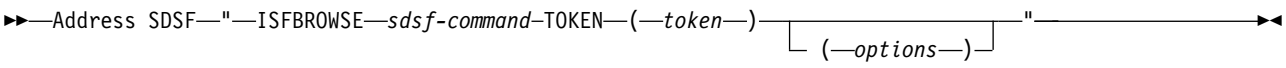
Browsing output

To browse the output of jobs and checks, you use a combination of host commands, action characters and special REXX variables. For details, refer to the appropriate topic:

- “Browsing output with ISFBROWSE.” You can use this approach to browse the output of jobs and checks. The output data is returned in the ISFLINE stem variable.
- “Browsing jobs with an external utility” on page 397. You can use this approach to browse job output. You allocate the output data sets with special REXX-only action characters, then browse the data sets using EXECIO or a similar utility.
- “Browsing checks with the S action character” on page 398. You can use this approach to browse the output of checks. The output data is returned in the ISFLINE stem variable.

Browsing output with ISFBROWSE

You can browse the output of jobs and checks using the ISFBROWSE host command, as follows:



sdsf-command
is the command for the panel. It must be the same SDSF command, including any parameters, that was previously entered with the ISFEXEC command.

token
is a token that identifies the row to be acted upon. The token was previously set by ISFEXEC or ISFACT and must correspond to the panel accessed with *sdsf-command*. Enclose the token in single quotation marks that are not removed by REXX.

For more information, see “Using tokens” on page 390.

options
is an optional list of options. The closing parenthesis is optional.

JCL
Browse just the JCL (jobs only)

NOCLOSE
Leave the data set open for subsequent requests, to avoid the overhead of closing, unallocating, re-allocating, and re-opening the data set. To undo the allocations, use ISFBROWSE without NOCLOSE and set special variable ISFSTARTLINETOKEN.

VERBOSE
Add diagnostic messages to stem variable isfmsg2. The messages describe each variable created by SDSF. This can be useful for troubleshooting as you develop REXX execs.

Examples of using ISFBROWSE

The following examples show ISFBROWSE commands you would use after having first issued the appropriate panel command with ISFEXEC. For more complete examples, see "Examples of REXX execs" on page 429.

1. Browse the output for a check on the CK panel. The number of the row is represented by ix.

   ```
   Address SDSF "ISFBROWSE CK TOKEN('"TOKEN.ix"')"
   ```

2. Browse just the JCL for a job on the ST panel. The number of the row is represented by x.

   ```
   Address SDSF "ISFBROWSE ST TOKEN('"TOKEN.x"') (JCL)"
   ```

3. Browse the output for a job on the DA panel. Leave the data sets open for subsequent browse requests. The number of the row is represented by ix.

   ```
   Address SDSF "ISFBROWSE DA TOKEN('"TOKEN.ix"') (NOCLOSE)"
   ```

Special variables for use with the ISFBROWSE command

There are a number of special variables that you can use with the ISFBROWSE command. For information on special REXX variables, see "Using special variables to invoke SDSF function" on page 410 and "Special variables reference" on page 420.

Several of the special variables provide function that corresponds to scrolling through the data, including repositioning to the next or previous data set. For example, you might specify a number of lines that you want to retrieve with each browse request, using ISFLINELIM, then use logic and other special variables to advance through the data, as shown below:

```
isflinelim = 500
   do until isfnextlinetoken=''
      Address SDSF "ISFBROWSE ST "TOKEN('"token.x"')"
      /****************************/
      /* Loop through the lines   */
      /****************************/
      do ix=1 to isfline.0
        say isfline.ix
      end
   isfstartlinetoken = isfnextlinetoken
end
```

Use these special variables with the ISFBROWSE command:

ISFDUPDS
controls whether duplicate SYSOUT data sets are included. Values are ON and OFF.

ISFFIRSTLINEDSID
is the data set identifier of the data set associated with the first line that was returned.

ISFFIRSTLINERECNO
is the record number within the data set of the first line that was returned.

ISFFIRSTLINETOKEN
is a token corresponding to the first line of the data that was returned.

ISFINPUT
controls whether SYSIN data sets are included. Values are ON and OFF.

ISFLASTLINEDSID
is the data set identifier of the data set associated with the last line that was returned.

ISFLASTLINERECNO
is the record number within the data set of the last line that was returned.

ISFNEXTLINETOKEN
is a token corresponding to the next unread line of the data that was returned. It is null when an end-of-file condition is encountered.

ISFLINE
contains the data that is returned. It is a stem variable. ISFLINE.0 contains the number of variables.

ISFLINELIM
limits the number of ISFLINE stem variables that may be created. The valid values are 0-99999999. A value of zero indicates no limit.

ISFSTARTLINETOKEN
specifies the starting line for the data to be returned. Assign a value by setting the variable to either the ISFFIRSTLINETOKEN or ISFNEXTLINETOKEN special variable.

Use these special variables with the ISFBROWSE command for find and scroll functions:

ISFFIND
contains a string to be found, up to 255 characters. The find operation is not sensitive to case. Use this with a value of FINDNEXT or FINDPREV in the ISFSCROLLTYPE special variable.

ISFFINDENDCOL
specifies the column by which the string specified with the ISFFIND special variable must end. It must be less than ISFFINDSTARTCOL.

ISFFINDLIM
specifies the maximum number of lines to search for the string specified with the ISFFIND special variable. Valid values are 1000 through 9999999.

ISFFINDSTARTCOL
specifies the column in which the string specified with the ISFFIND special variable must start.

ISFSCROLL
is used to reposition the first line of data that is returned.

- For repositioning a number of lines, specify an integer to be used as an offset from the value in the ISFSTARTLINETOKEN special variable. Then, specify a value of UP or DOWN for the ISFSCROLLTYPE special variable. If ISFSTARTLINETOKEN is not specified, the offset is applied to the top of the data set.
- For repositioning to another data set, specify a number to be used as the number of data sets and specify a value of NEXT or PREV for the ISFSCROLLTYPE special variable. The data returned begins with the first line of the data set. ISFSCROLL defaults to 1 and can be omitted when you specify ISFSCROLLTYPE with NEXT or PREV.

ISFSCROLLTYPE
is used to reposition the first line of data that is returned. Specify one of these values:

UP or DOWN
is used with the value in the ISFSCROLL special variable to reposition a number of lines. DOWN is the default.

NEXT or PREV
is used with the value in the ISFSCROLL special variable to reposition a number of data sets.

TOP specifies that the first record returned is the first record of the data. This is the default.

BOT requests the bottom, or most recent, data. The last line returned is the last line of data. The first line returned is a function of the value of the ISFLINELIM special variable. For example, if you use BOT with a value of 100 for ISFLINELIM, the last 100 lines of data are returned.

FINDNEXT
is used with the value in the ISFFIND special variable to reposition to the next line that contains that string. If the ISFSTARTLINETOKEN special variable is not specified, the search begins with the top line.

FINDPREV
is used with the value in the ISFFIND special variable to reposition to the previous line that contains that string. If the ISFSTARTLINETOKEN special variable is not specified, the search begins with the top line, wraps to the bottom and then searches from there.

Return codes for ISFBROWSE

After the ISFBROWSE host environment command completes, a return code is set in the REXX variable RC. The values are:

00 The request completed successfully.

04 The request completed successfully but not all functions were performed.

08 An incorrect or invalid parameter was specified for an option or command.

12 A syntax error occurred in parsing a host environment command.

16 The user is not authorized to invoke SDSF.

20 A request failed due to an environmental error.

24 Insufficient storage was available to complete a request.

Note that a return code of 0 indicates that SDSF successfully processed the ISFBROWSE command. It does not indicate that specific functions were authorized or that commands were executed. Check the ISFMSG and ISFMSG2 variables to determine if a request completed. See “Messages” on page 382 for more information.

Browsing jobs with an external utility

To browse job output from the DA, H, I, JDS, O and ST panels using EXECIO or similar utility, you first allocate the output data sets with special REXX-only action characters. The action characters are:

SA Allocate all data sets associated with the item. On the DA, I or ST panels, this will be all data sets in the job. On the O and H panels, it will be all data sets in the output group. On the JDS panel, it will be a single data set.

SJA
Allocate the JCL data set

The following special variables describe the results of the allocation that you use with EXECIO or a similar utility:

ISFDDNAME
is a stem variable that contains the system-generated DDNAME returned by allocation that is referenced on EXECIO or other utility. It is not the application specified DDNAME that is contained in the DDNAME.x stem variable returned by ISFACT. ISFDDNAME.0 contains a count of the number of variables that follow.

ISFDSNAME
is a stem variable that contains the application-specified data set name that has been allocated by SDSF. The variables have a one-to-one correspondence with the variables in ISFDDNAME. Thus, the REXX caller can associate the data set being processed with the system generated DDNAME that has been allocated. ISFDSNAME.0 contains a count of the number of variables that follow.

ISFLRECL
is a stem variable that contains the logical record length for the allocated data set and corresponds to the DDNAME listed in ISFDDNAME. ISFLRECL.0 contains a count of the number of variables that follow.

ISFRECFM
is a stem variable that contains the record format for the allocated data set and corresponds to the DDNAME listed in ISFDDNAME. ISFRECFM.0 contains a count of the number of variables that follow.

You can also use these special variables:

ISFDUPDS
controls whether duplicate SYSOUT data sets are included.

ISFINPUT
controls whether SYSIN data sets are included.

Usage notes

- SDSF allocates SYSOUT data sets using the FREE=CLOSE attribute. This causes the system to free the allocation when the data set is closed by the application. If an application causes a data set to be allocated but does not open it, it should free the allocation explicitly. Failure to free the data sets may result in the allocation limit being reached and further allocations being rejected.
- The REXX caller should also ensure that the DYNAMNBR JCL keyword is set to a high enough limit to accommodate all of the expected allocations done by the exec.
- You can use the FINIS option of EXECIO to close the data set when EXECIO completes.

Browsing checks with the S action character

To browse check output from the CK or CKH panel, you can use the S action character on the ISFACT command, along with the following special variable:

ISFLINE
is a stem variable that contains lines of data in response to a browse request. ISFLINE.0 contains the number of stem variables that follow.

Examples of browsing output

See “Browse job output with EXECIO” on page 435 and “Browse check output from the CK panel” on page 441.

Printing output

To print the output of jobs and checks through REXX, you use a combination of the X action character, with ISFACT, and special REXX variables. The PRINT command is not supported in the REXX environment.

The forms of the X action character are:

X and XC
Print all data sets using default settings; XC closes the print file after printing.

XS and XSC
Print all data sets to SYSOUT using attributes specified in special variables; XSC closes the print file after printing.

The special variables define the attributes of the SYSOUT print file. They correspond to the fields on the Open Print pop-up. The special variables are:

Table 164. Special REXX Variables for Printing to SYSOUT

Variable	Purpose
ISFPRTCCASA	How SDSF handles carriage control for printing: **ON** Always insert ASA carriage control characters **OFF** Handle carriage control based on the record format of the data set bring printed: • If the record format includes A, then the print function uses ASA (ANSI) carriage control. • If the record format includes M, then the print function uses machine carriage control. • Otherwise, SDSF removes carriage control characters if they are present in the source.
ISFPRTCLASS	SYSOUT class
ISFPRTCOPIES	Copies class
ISFPRTDEST	Destination
ISFPRTFCB	FCB
ISFPRTFORMDEF	FORMDEF
ISFPRTFORMS	Forms
ISFPRTLRECL	Logical record length
ISFPRTOUTDESNAME	Output descriptor name to be used when creating the file
ISFPRTPAGEDEF	PAGEDEF
ISFPRTPRTMODE	Process mode
ISFPRTRECFM	Record format
ISFPRTSOURCEATTS	Whether to use attributes of the source for printing
ISFPRTUCS	UCS
ISFPRTWRITER	Writer name

XD and XDC
Print all data sets to a data set using attributes specified in special variables; XDC closes the print file after printing.

The special variables name attributes of the data set. They correspond to the fields on the Open Print Data Set pop-up.

Table 165. Special REXX Variables for Printing to a Data Set

Variable	Purpose	Default
ISFPRTCCASA	How SDSF handles carriage control for printing. For details, refer to the description of ISFPRTCCASA in Table 164 on page 399.	
ISFPRTBLKSIZE	Block size for new data sets	003120
ISFPRTDATACLAS	Data class for new data sets	
ISFPRTDIRBLKS	Number of directory blocks for new data sets	
ISFPRTDISP	Allocation disposition for data sets	
ISFPRTDSNAME	Data set name. If the name is not enclosed in quotation mark, the name begins with the current user ID.	
ISFPRTDSNTYPE	Data set name type: **LIBRARY or LIB** allocates a partitioned data set extended (PDSE) **PDS** allocates a partitioned data set **LARGE** allocates a large format data set **EXTREQ** indicates that an extended sequential data set is required **EXTPREF** indicates that an extended sequential data set is preferred **EXTREQ** indicates an extended sequential data set is required **BASIC** indicates that neither an extended nor a large format data set is to be allocated **BASIC** indicates that neither an extended nor a large format data set is to be allocated.	A partitioned or sequential data set is allocated based on the data set characteristics that are provided.
ISFPRTEXTATTR	Extended attributes option: **NO** The data set cannot have extended attributes and reside in EAS **OPT** The data set can have extended attributes and reside in EAS.	Based on the data type
ISFPRTLRECL	LRECL for new data sets	0000240
ISFPRTMEMBER	Member name	
ISFPRTMGMTCLAS	Management class for new data sets	
ISFPRTPRIMARY	Primary space allocation for new data sets	00000500
ISFPRTRECFM	Record format	VBA
ISFPRTSECONDARY	Secondary space allocation for new data sets	00000500

Table 165. Special REXX Variables for Printing to a Data Set (continued)

Variable	Purpose	Default
ISFPRTSPACETYPE	Space units for allocating for new data sets	BLKS
ISFPRTSTORCLAS	Storage class for new data sets	
ISFPRTUNIT	Unit for new data sets	
ISFPRTVOLSER	Volume serial for new data sets	

XF and XFC
Print all data sets to a file (DDNAME) using attributes specified in special variables; XFC closes the print file after printing. The special variables name attributes of the file.

Table 166. Special Variables for Printing to a File

Variable	Purpose
ISFPRTDDNAME	DDNAME

In the event of an error, such as the data being invalid or missing, SDSF issues a message that is available in the ISFMSG2 stem variable. In addition, the ISFMSG variable may contain a short error message.

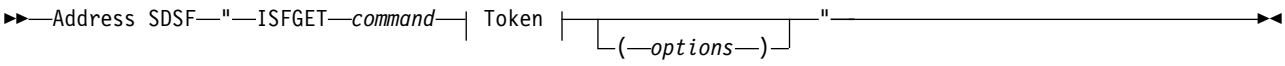
Note that the print data set is always closed after the request regardless of whether the X action character includes the C option. This is because all SDSF requests are independent; the print data set is closed when SDSF terminates.

Examples of printing

See "Print to SYSOUT" on page 444.

Getting all of the values for a single row

You can request all of the column values for a specific row using the ISFGET host environment command, as follows:



Token:

command
is the command for the panel. It must be the same SDSF command, including any parameters, that was previously entered with the ISFEXEC command.

token
identifies the row to be acted upon. The token was previously set by ISFEXEC or ISFACT for the panel accessed with *command*. Enclose the token in single quotation marks. For more information on tokens, see "Using tokens" on page 390.

Return codes for ISFGET

After the ISFGET host environment command completes, a return code is set in the REXX variable RC. The values are:

00 The request completed successfully.

04 The request completed successfully but not all functions were performed.

08 An incorrect or invalid parameter was specified for an option or command.

12 A syntax error occurred parsing a host environment command.

16 The user is not authorized to invoke SDSF.

20 A request failed due to an environmental error.

24 Insufficient storage was available to complete a request.

Note that a return code of 0 indicates that SDSF successfully processed the ISFGET command. It does not indicate that specific functions were authorized or that commands were executed. Check the ISFMSG and ISFMSG2 variables to determine if a request completed. See "Messages" on page 382 for more information.

Data returned for ISFGET

When you use an action character to access a secondary panel, such as JDS, ISFACT returns column and row data in the same way that ISFEXEC does. See "Panel data returned" on page 383 for more information.

Options for getting all of the values for a row

You can use the following options with ISFGET:

ALTERNATE
requests the alternate field list for the panel

ALTERNATE2
requests the alternate field list for the secondary panel

DELAYED
specifies that delayed-access columns be included on the panel

DELAYED2
specifies that delayed-access columns be included on the secondary panel

NOMODIFY2
specifies that row tokens for use in modifying rows should not be returned on the secondary panel. Use this to improve performance if you will not be modifying any values.

PRIMARY
requests the primary field list.

If you specify both PRIMARY and ALTERNATE, or neither PRIMARY nor ALTERNATE, the primary and alternate field lists are merged, and all the column variables for the panel are available.

PRIMARY2
requests the primary field list for a secondary panel.

If you specify both PRIMARY2 and ALTERNATE2, or neither PRIMARY2 nor ALTERNATE2, the primary and alternate field lists are merged, and all the column variables for the panel are available.

PREFIX *value*
specifies a prefix for column name and TOKEN variables that are created; use this to ensure that variable names do not conflict. The prefix can be up to 24 characters long, and should not begin with ISF.

VERBOSE
adds diagnostic messages to the ISFMSG2 stem variable. The messages describe each row variable created by SDSF.

Special variables with ISFGET

For information on special REXX variables, see "Using special variables to invoke SDSF function" on page 410 and "Special variables reference" on page 420.

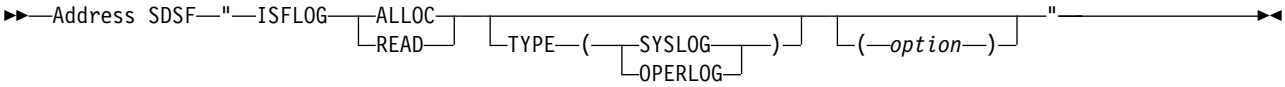
Browsing the system log with ISFLOG

You browse both the single-system SYSLOG and the sysplex-wide OPERLOG using the ISFLOG host environment command.

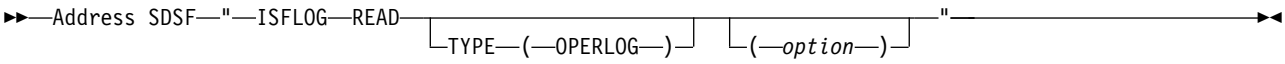
When used with the SYSLOG, the ISFLOG command processes the JES logical log. For more information, refer to "SYSLOG" on page 349.

The syntax of the ISFLOG command is as follows:

SYSLOG



OPERLOG



ALLOC
indicates that the logical SYSLOG is to be allocated for use with a utility such as EXECIO. The allocation is done with the FREE=CLOSE option so that the file is automatically de-allocated when closed.

Use ALLOC with these special stem variables:

- ISFDDNAME contains the ddname that is returned
- ISFDSNAME contains the data set name that is returned

READ
indicates that the system log is to be read. The records are returned in the ISFLINE stem variable. ISFLINE.0 contains the number of variables.

By default, SDSF retrieves the records for the current day. You can customize the results with these special variables:

- ISFLINELIM sets a limit on the number of variables created.
- ISFLOGSTARTTIME, ISFLOGSTARTDATE, ISFLOGSTOPTIME and ISFLOGSTOPDATE define the date and time range for the records. Use them to ensure that your date and time range is reasonable, so that an excessive number of variables is not created.

When these special variables are used, SDSF positions the SYSLOG as near as possible to the requested record. However, due to the precision used for time stamps and the time the record is actually written to SYSLOG, it is possible that this may be several lines away from the desired record.

- Variables that allow you to simulate scrolling through the data. These include ISFSCROLL, ISFSCROLLYPE, ISFNEXTLINETOKEN and ISFSTARTLINETOKEN.

For details on the special variables, refer to "Special variables for use with the ISFLOG command."

TYPE(SYSLOG | OPERLOG)
is optional and names the type of system log to be used:

SYSLOG
specifies the single-system SYSLOG. Use the special variable ISFSYSID to indicate the member to be processed.

OPERLOG
specifies the sysplex-wide OPERLOG.

option
is optional. See "Options for the ISFLOG command."

Use the special variable ISFSYSID to indicate the member to be processed.

Examples of using ISFLOG

The following examples illustrate how to use the ISFLOG command.

1. Allocate the logical SYSLOG for use with EXECIO.
   ```
   Address SDSF "ISFLOG ALLOC TYPE(SYSLOG)"
   ```
2. Read the logical SYSLOG into the ISFLINE special variable.
   ```
   Address SDSF "ISFLOG READ TYPE(SYSLOG)"
   ```
3. Read the OPERLOG into the ISFLINE special variable.
   ```
   Address SDSF "ISFLOG READ TYPE(OPERLOG)"
   ```
4. Read the logical SYSLOG into the ISFLINE special variable and the WTORS into the ISFWTOR special variable.
   ```
   Address SDSF "ISFLOG READ TYPE(SYSLOG) (WTOR)"
   ```

See also "Work with the last 24 hours of SYSLOG" on page 445 and "Work with the current day of the system log" on page 446.

Options for the ISFLOG command

VERBOSE
adds diagnostic messages to the ISFMSG2 stem variable. The messages describe each row variable created by SDSF.

WTOR
causes any WTORs to be returned in the ISFWTOR. stem variable.

Special variables for use with the ISFLOG command

There are a number of special variables that you can use with the ISFLOG command. For information on special REXX variables, see "Using special variables to invoke SDSF function" on page 410 and "Special variables reference" on page 420.

Several of the special variables provide function that corresponds to scrolling through the data. For example, you might specify a number of lines that you want to retrieve with each browse request, using ISFLINELIM, then use logic and other special variables to advance through the data, as shown below:

```
isflinelim = 500
   do until isfnextlinetoken=''
      Address SDSF "ISFLOG READ TYPE(SYSLOG)"
      /***************************/
      /* Loop through the lines    /
      /***************************/
      do ix=1 to isfline.0
        say isfline.ix
      end
   isfstartlinetoken = isfnextlinetoken
end
```

Use these special variables with the ISFLOG command:

ISFCOLOR
is a stem variable containing a single-character abbreviation for the color for each line. The possible values come from first letter of these colors: Red, Green, Blue, White, Yellow, Turquoise, Pink. OPERLOG only.

ISFDATE
specifies the date format, including the separator character, for special variables that take a date as input. It accepts any format valid with the SET DATE command. See the SET DATE command in the online help for the valid formats.

ISFDESCODE
is a stem variable containing the descriptor codes for each line. When there are multiple descriptor codes, they are turned in a list, separated by blanks. OPERLOG only.

ISFFIRSTLINEDATE
is the date associated with the first line that was returned.

ISFFIRSTLINEDSID
is the data set identifier of the data set associated with the first line that was returned. SYSLOG only.

ISFFIRSTLINEJOBID
is the job ID associated with the first line that was returned. SYSLOG only.

ISFFIRSTLINERECNO
is the record number within the data set of the first line that was returned. SYSLOG only.

ISFFIRSTLINETIME
is the time associated with the first line that was returned.

ISFFIRSTLINETOKEN
is a token corresponding to the first line of the data that was returned.

ISFHIGHLIGHT
is a stem variable containing a single-character abbreviation for the highlighting for each line. The possible values come from the first letter of these highlight values: Blink, Reverse, Underline and None. OPERLOG only.

ISFINTENSITY
is a stem variable containing a single-character abbreviation for the intensity for each line. The possible values come from the first letter of these intensities: High and Low. OPERLOG only.

ISFLASTLINEDATE
is the date associated with the last line that was returned.

ISFLASTLINEDSID
is the data set identifier of the data set associated with the last line that was returned. SYSLOG only.

ISFLASTLINEJOBID
is the job ID associated with the last line that was returned. SYSLOG only.

ISFLASTLINERECNO
is the record number within the data set of the last line that was returned. SYSLOG only.

ISFLASTLINETIME
is the time associated with the last line that was returned.

ISFLINE
contains the data that is returned. It is a stem variable. ISFLINE.0 contains the number of variables.

ISFLINELIM
limits the number of ISFLINE stem variables that may be created. The valid values are 0-99999999. A value of zero indicates no limit.

ISFLOGSTARTDATE
specifies the starting date for records returned by the ISFLOG command, in the current date format (see the ISFDATE special variable) or *yyyy.ddd*. Leading zeros are not required. It must be less than the ending date. The default is the current day.

ISFLOGSTARTTIME
specifies the starting time for records returned by the ISFLOG command, in *hh*:*mm*:*ss*.*th* format. Only *hh*:*mm* is required. Leading zeros are not required. This is the local time corresponding to the first record to be returned. It must be less than the ending time. The default is 00:00:00.00.

ISFLOGSTOPDATE
specifies the ending date for records returned by the ISFLOG command, in the current date format (see the ISFDATE special variable) or *yyyy.ddd*. Leading zeros are not required. The default is the current day.

ISFLOGSTOPTIME
specifies the ending time for records returned by the ISFLOG command, in *hh*:*mm*:*ss*.*th* format. Only *hh*:*mm* is required. Leading zeros are not required. This is the local time corresponding to the last record to be returned. The default is 23:59:59.99.

ISFNEXTLINETOKEN
is a token corresponding to the next unread line of the data that was returned. It is null when an end-of-file condition is encountered.

ISFSTARTLINETOKEN
specifies the starting line for the data to be returned. Assign a value by setting the variable to either the ISFFIRSTLINETOKEN or ISFNEXTLINETOKEN special variable.

ISFSYSID
with the SYSLOG, names the member to be processed by the ISFLOG command. See the SYSID command in the online help.

ISFWTOR
is a stem variable that contains the WTORs, if requested with the WTOR option. ISFWTOR.0 contains the number of variables.

Use these special variables with the ISFLOG command for find and scroll functions:

ISFFIND
contains a string to be found, up to 255 characters. The find operation is not sensitive to case. Use this with a value of FINDNEXT or FINDPREV in the ISFSCROLLTYPE special variable.

ISFFINDENDCOL
specifies the column by which the string specified with the ISFFIND special variable must end. It must be less than ISFFINDSTARTCOL.

ISFFINDLIM
specifies the maximum number of lines to search for the string specified with the ISFFIND special variable. Valid values are 1000 through 9999999.

ISFFINDSTARTCOL
specifies the column in which the string specified with the ISFFIND special variable must start.

ISFSCROLL
is used to reposition the first line of data that is returned. Specify an integer to be used as an offset from the value in the ISFSTARTLINETOKEN special variable. Then, specify a value of UP or DOWN for the ISFSCROLLTYPE special variable. If ISFSTARTLINETOKEN is not specified, the offset is applied to the top of the data set.

ISFSCROLLTYPE
is used to reposition the first line of data that is returned. Specify one of these values:

UP or DOWN
is used with the value in the ISFSCROLL special variable. DOWN is the default.

TOP specifies that the first record returned is the first record of the data. This is the default.

BOT requests the bottom, or most recent, data. The last line returned is the last line of data. The first line returned is a function of the value of the ISFLINELIM special variable. For example, if you use BOT with a value of 100 for ISFLINELIM, the last 100 lines of data are returned.

FINDNEXT
is used with the value in the ISFFIND special variable to reposition to the next line that contains that string. If the ISFSTARTLINETOKEN special variable is not specified, the search begins with the top line.

FINDPREV
is used with the value in the ISFFIND special variable to reposition to the previous line that contains that string. If the ISFSTARTLINETOKEN special variable is not specified, the search begins with the top line, wraps to the bottom and then searches from there.

Return codes for ISFLOG

After the ISFLOG host environment command completes, a return code is set in the REXX variable RC. The values are:

00 The request completed successfully.

04 The request completed successfully but not all functions were performed.

08 An incorrect or invalid parameter was specified for an option or command.

12 A syntax error occurred parsing a host environment command.

16 The user is not authorized to invoke SDSF.

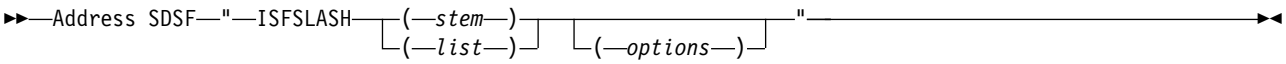
20 A request failed due to an environmental error.

24 Insufficient storage was available to complete a request.

Note that a return code of 0 indicates that SDSF successfully processed the ISFLOG command. It does not indicate that specific functions were authorized or that commands were executed. Check the ISFMSG and ISFMSG2 variables to determine if a request completed. See "Messages" on page 382 for more information.

Issuing system commands with ISFSLASH

You issue system commands using the ISFSLASH host environment command as follows:



stem
is the name of a stem variable containing the list of system commands to be issued. The 0 variable of the stem must contain a count of the variables in the stem. The variable *stem* should:

- End in a period, to allow the commands to be put into compound variables
- Be enclosed in parentheses, to indicate that it is a stem variable
- Be 1 to 128 characters long
- Not start with the characters ISF

list
is a list of one or more system commands to be issued, separated by a blank or a comma.

Enclose a command in single quotation marks, whether you are issuing it directly through ISFSLASH or using a stem variable, if the command:

- Contains special characters or embedded blanks
- Requires mixed case. Although SDSF preserves the case of the command text, Consoles folds the text to uppercase in issuing the command, unless it is enclosed in single quotation marks.

The W and I prefix parameters of the slash (/) command are not supported. Use the WAIT and INTERNAL options instead. See "Options for slash (/) commands" on page 409 for more information.

The system commands can be up to 126 characters in length (the maximum length allowed by Consoles).

Examples of using ISFSLASH

The following examples illustrate how to issue a command with ISFSLASH.

1. Issue a single command. Wait the full delay interval (specified with variable ISFDELAY) for responses, rather than returning when the first response is received.

   ```
   Address SDSF ISFSLASH "$da (WAIT)"
   ```

2. Issue a single command using a stem variable.

   ```
   cmd.0=1
   cmd.1='d r,l'
   Address SDSF ISFSLASH "(cmd.)"
   ```

3. Issue multiple commands. Because the commands contain blanks, enclose them in single quotation marks.

   ```
   Address SDSF ISFSLASH "$da , 'd a,l' 'd t'"
   ```

4. Issue multiple commands using a stem variable. SDSF will wait the full delay interval for the response.

   ```
   mycmd.0=2
   mycmd.1='$DA'
   mycmd.2='d t'
   Address SDSF ISFSLASH "("mycmd.") (WAIT)"
   ```

See also “Issue system commands using ISFSLASH” on page 445.

Options for slash (/) commands

INTERNAL
specifies that console ID 0 (INTERNAL) should be used to issue the command

WAIT
specifies that SDSF should wait the full delay interval before retrieving responses. This option is strongly recommended to ensure the responses are accessible in the ISFULOG special variable. The delay interval is specified with the ISFDELAY variable.

Special variables for slash (/) commands

Use special variables to set options such as the delay limit and the console name. Where the variable is associated with an SDSF command, the parameters for the variable are the same as for the command, with the exception that the ? parameter is not supported in REXX. Substitute the variable for the command, for example:

Command: SET DELAY 5
Variable: isfdelay="5"

For the syntax of commands, see the online help. For information on special REXX variables, see “Using special variables to invoke SDSF function” on page 410 and “Special variables reference” on page 420.

ISFCMDLIM
limits the number of commands that may be issued through ISFSLASH. The limit is a value from 0-99999999 where 0 means no limit. The default is 0. If the number of stem variables exceeds the limit, all commands up to and including the limit are issued.

ISFCONMOD
controls console name modification. By default it is on, which means that, when SDSF needs to activate an extended console and the default console name is already in use, SDSF attempts to activate a new console with a modified name. For more information, refer to the SET CONMOD command in the online help and “Extended console name” on page 343.

If you run a REXX exec while using SDSF interactively, you should not disable console modification, to avoid an activation failure caused by the required console already being in use.

ISFCONS
specifies a name for the extended console for the user session log (ISFULOG stem variable). Refer to the SET CONSOLE command in the online help for more information.

If you run a REXX exec while using SDSF interactively and you have disabled console modification, you should specify a unique console name with ISFCONS, to avoid an activation failure caused by the required console already being in use.

ISFDELAY
specifies the response delay limit for system commands. Refer to the SET DELAY command in the online help for more information.

ISFULOG
is a stem variable that contains the MVS system command echo and any responses generated during the session, including SAF authorization messages (if supported by the external security manager). ISFULOG.0 contains a count of the number of stem variables that follow.

For more information, see "Issuing commands with ISFEXEC" on page 380.

Return codes for ISFSLASH

After the ISFSLASH host environment command completes, a return code is set in the REXX variable RC. The values are:

00 The request completed successfully.

04 The request completed successfully but not all functions were performed.

08 An incorrect or invalid parameter was specified for an option or command.

12 A syntax error occurred in parsing a host environment command.

16 The user is not authorized to invoke SDSF.

20 A request failed due to an environmental error.

24 Insufficient storage was available to complete a request.

Note that a return code of 0 indicates that SDSF successfully processed the ISFEXEC command. It does not indicate that specific functions were authorized or that commands were executed. Check the ISFMSG and ISFMSG2 variables to determine if a request completed. See "Messages" on page 382 for more information.

Using special variables to invoke SDSF function

Much of the function that SDSF commands provide when you use SDSF interactively is supported in the REXX environment by special REXX variables.

The special variables use the following format:

The parameters for the variable are the same as for the associated command, with the exception that the ? parameter is not supported in REXX. The values of special

variables are not saved across sessions (or invocations) in the REXX environment. The special variable names are not affected by the PREFIX option of ISFEXEC or ISFACT.

Special variable names are not case-sensitive.

Values specified with special variables do not have the 42–character (or, in the case of slash commands, 126–character) limit that commands entered with ISFEXEC have.

Where the variable is associated with an SDSF command, the parameters for the variable are the same as for the command, with the exception that the ? parameter is not supported in REXX. Substitute the variable for the command, for example:

Command: PREFIX RJONES*
Variable: isfprefix="RJONES*"

For the syntax of commands, see the online help. For a complete list of special REXX variables, see "Special variables reference" on page 420.

To drop SDSF special variables (that is, unassign the variables and restore them to their original undefined state) use the ISFRESET() function. The option to use with ISFRESET corresponds to the variable type (Input, InOut or Output), indicated in the table. The variables for printing are all type Input. For more information, see "Dropping special variables with ISFRESET" on page 414.

The variables are grouped here by command type:

- "SDSF command"
- "Filter commands"
- "Options commands" on page 412
- "Trace commands" on page 413

SDSF command

Use the following special variables for function that is equivalent to the parameters on the SDSF command.

ISFSERVER
names the SDSF server. See the SERVER parameter in "Global initialization parameters (OPTIONS or ISFPMAC)" on page 23.

ISFJESNAME
names the JES2 subsystem to process. See the JESNAME parameter in "Global initialization parameters (OPTIONS or ISFPMAC)" on page 23.

ISFJES3NAME
names the JES3 subsystem to process. See the JES3NAME parameter in "Global initialization parameters (OPTIONS or ISFPMAC)" on page 23.

Filter commands

Use the following special variables for function that is equivalent to the filter commands.

ISFDEST
specifies up to four destinations to be used for filtering. Each destination can be up to the maximum acceptable length for a destination. See the DEST command in the online help for syntax, but note these differences:

- The length of the value specified with ISFDEST can exceed the 42-character limit of the DEST command
- When specifying multiple destinations with ISFDEST, separate the destinations with a blank. Do not use the + operand used with the command.

ISFFILTER
specifies filter criteria to be applied to the returned variables. Use the column names rather than the column titles. See the FILTER command in the online help. Use ISFFILTERMODE to specify the AND or OR relationship between filters.

ISFFILTERMODE
specifies a relationship between filters, both within a column and between columns. The relationship can be either AND or OR.

ISFINPUT
controls whether SYSIN data sets are returned. See the INPUT command in the online help.

ISFOWNER
specifies the owner to be used to limit the returned variables. See the OWNER command in the online help.

ISFPREFIX
specifies the job name to be used to limit the returned variables. See the PREFIX command in the online help.

ISFSYSNAME
specifies the system to be used to limit sysplex requests. See the SYSNAME command in the online help.

Options commands

Use the following special variables for function that is equivalent to the options commands, such as the SET commands.

ISFACTIONS
specifies whether the action characters for the current panel should be returned in the ISFRESP stem variable. The values in the ISFRESP variable are in this format: ACTION=*action*, where *action* is the action character or the action character and a description, depending on the option specified on ISFACTIONS. See the SET ACTION command in the online help for the valid options. See "List action characters" on page 444 for an example.

ISFCKLIM
specifies the limit for the number of instances of a check to be shown on the CKH panel.

ISFCONMOD
controls console name modification. By default it is on, which means that, when SDSF needs to activate an extended console and the default console name is already in use, SDSF attempts to activate a new console with a modified name. For more information, refer to the SET CONMOD command in the online help and "Extended console name" on page 343.

If you run a REXX exec while using SDSF interactively, you should not disable console modification, to avoid an activation failure caused by the required console already being in use.

ISFCONS
specifies a name for the extended console for the user session log (ISFULOG stem variable). Refer to the SET CONSOLE command in the online help for more information.

If you run a REXX exec while using SDSF interactively and you have disabled console modification, you should specify a unique console name with ISFCONS, to avoid an activation failure caused by the required console already being in use.

ISFDATE
specifies the date format, including the separator character, for special variables used with the ISFLOG command that take a date as input. See the SET DATE command in the online help for the valid formats.

ISFDELAY
specifies the timeout for command responses. See the SET DELAY command in the online help.

ISFDISPLAY
contains the filtering and sorting criteria, for example,

```
PREFIX=* DEST=(ALL)  OWNER=*  SYSNAME=
```

See the SET DISPLAY command in the online help.

ISFDISPLAYMODE
sets the format of the ISFDISPLAY special variable. See the SET DISPLAY command in the online help. The OFF parameter is not valid in REXX.

ISFDUPDS
controls whether duplicate SYSOUT data sets are included.

ISFINPUT
controls whether SYSIN data sets are returned. See the INPUT command in the online help.

ISFSCHARS
specifies generic and placeholder characters used for pattern matching. See the SET SCHARS command in the online help.

ISFTIMEOUT
specifies the timeout interval for sysplex data. See the SET TIMEOUT command in the online help.

Trace commands

Use the following special variables for function that is equivalent to the SET SECTRACE command.

ISFSECTRACE
specifies an option to be used when enabling SDSF security trace

ISFMSG2
contains security trace messages, if you specified ISFSECTRACE ON

ISFULOG
contains security trace messages, if you specified ISFSECTRACE WTP

For more information, refer to “Diagnosing security” on page 18.

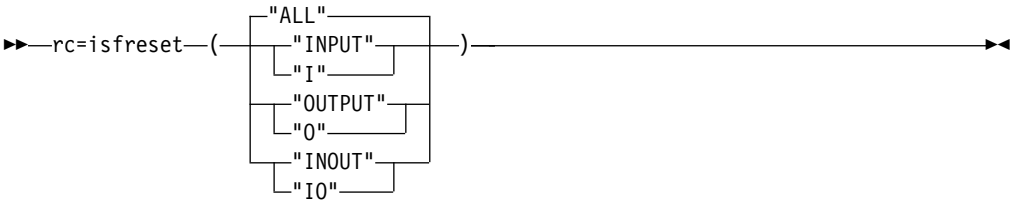
Use the following special variables for function that is equivalent to the TRACE command.

ISFTRACE
specifies a trace option to be used when enabling SDSF trace

ISFTRMASK
specifies a trace mask to be used when enabling SDSF trace

Dropping special variables with ISFRESET

You drop special variables using the ISFRESET() function. This unassigns the variables and restores them to their original undefined state. The syntax of ISFRESET is as follows:



ALL
all special variables. ALL is the default.

INPUT or I
all input special variables.

OUTPUT or O
all output special variables.

INOUT or IO
all input/output special variables.

ISFRESET does not require access to SDSF and so no authorization is required to use it. ISFRESET is not dependent on ISFCALLS and can be issued at any point in the exec. However, it is most useful when issued prior to an Address SDSF command.

For a complete list of special variables, refer to "Special variables reference" on page 420.

Result codes for ISFRESET

After the ISFRESET completes, a result code is set in the REXX variable RC. The values are:

0 The request completed successfully.

1 Environment error (for example, REXX is not running).

2 Syntax error occurred, for example, invalid parameter.

SDSF with REXX reference

This topic describes the REXX support for SDSF function.

SDSF commands reference

The SDSF commands and their use in REXX are described in Table 167 on page 415. For the syntax of the commands, see the online help. For quick access to command syntax, use this SEARCH command from the SDSF command line:

```
SEARCH 'FORMAT: command-name'
```

where *command-name* is the command name, for example, DA or PREFIX.

Table 167. SDSF Commands and REXX

Command	Purpose	Use on ISFEXEC	Use on ISFACT	REXX Variable	Notes
/	Issue MVS command	Yes	No		The preferred method is to use ISFSLASH.
?	Switch between primary and alternate field lists	No	No		Not supported in REXX. See the PRIMARY, ALTERNATE and DELAYED options of the ISFEXEC command and the PRIMARY2, ALTERNATE2 and DELAYED2 options of the ISFACT command.
?	Display output data set information from browse	No	No		Not supported in REXX
ABEND	Force SDSF abend	No	No		Not supported in REXX
ACTION	Control WTORs displayed on the SYSLOG	No	No		
AFD	Invoke SDSF with program ISFAFD	No	No		Not supported in REXX
APF	Display the APF panel	Yes	Yes		
APPC	Control the display of transaction data	No	No	ISFAPPC	
ARRANGE	Control the order of panel columns	No	No		Not supported in REXX
AS	Display the AS panel	Yes	Yes		
BOOK	Invoke BookManager®	No	No		Not supported in REXX
BOTTOM	Scroll to the bottom	No	No	ISFSCROLL, ISFSCROLLTYPE	Supported for browse only
CK	Display the CK panel	Yes	Yes		
COLS	Display the scale line	No	No		Not supported in REXX
DA	Display the DA panel	Yes	Yes		
DEST	Specify destinations for filtering	No	No	ISFDEST	The length of the value can exceed the 42-character limit of the DEST command. When specifying multiple destinations (up to 4), separate them with a blank. Do not use the + operand.
DOWN	Scroll down	No	No	ISFSCROLL, ISFSCROLLTYPE	Supported only for browsing with ISFBROWSE and ISFLOG.
DYNX	Display the DYNX panel	Yes	Yes		
ENC	Display the ENC panel	Yes	Yes		
ENQ	Display the ENQ panel	Yes	Yes		
END	Return to the previous panel	No	No		Not supported in REXX

Table 167. SDSF Commands and REXX (continued)

Command	Purpose	Use on ISFEXEC	Use on ISFACT	REXX Variable	Notes
FILTER	Filter data	No	No	ISFFILTER, ISFFILTER2, ISFFILTERMODE	There is no limit to the number of filters you can set with ISFFILTER or ISFFILTER2. Supported for tabular panels.
FIND	Find a string	No	No	ISFFIND	Supported only for browsing with ISFBROWSE and ISFLOG
FINDLIM	Set the number of lines to search	No	No	ISFFINDLIM	Supported only for browsing with ISFBROWSE and ISFLOG
H	Display the H panel	Yes	Yes		
I	Display the I panel	Yes	Yes		
INIT	Display the INIT panel	Yes	Yes		
INPUT	Control inclusion of input data sets in browse	No	No	ISFINPUT	
JC	Display the JC panel	Yes	Yes		
JP	Display the JP panel	Yes	Yes		
J0	Display the J0 panel	Yes	Yes		
LEFT	Scroll left	No	No		Not supported in REXX
LI	Display the LINES panel	Yes	Yes		
LNK	Display the LNK panel	Yes	Yes		
LPA	Display the LPA panel	Yes	Yes		
LOCATE	Locate a line or column	No	No		Not supported in REXX
LOG	Display the SYSLOG and Operlog	No	No		Use the ISFLOG command
LOGLIM	Limit the Operlog	No	No		
MAS	Display the MAS panel	Yes	Yes		
NC	Display the NC panel	Yes	Yes		
NEXT	Skip to the next data set	No	No	ISFSCROLL, ISFSCROLLTYPE	Use with ISFBROWSE
NO	Display the NODES panel	Yes	Yes		
NS	Display the NS panel	Yes	Yes		
O	Display the O panel	Yes	Yes		
OWNER	Limit the jobs by owner	No	No	ISFOWNER	
PAG	Display the PAG panel	Yes	Yes		
PARM	Display the PARM panel. Enclose PARM in single quotes when using ISFACT.	Yes	Yes		
PANELID	Display panel ID	No	No		Not supported in REXX
PR	Display the PR panel	Yes	Yes		
PREFIX	Filter jobs by name	No	No	ISFPREFIX	

Table 167. SDSF Commands and REXX (continued)

Command	Purpose	Use on ISFEXEC	Use on ISFACT	REXX Variable	Notes
PREV	Skip to the previous data set	No	No	ISFSCROLL, ISFSCROLLTYPE	Use with ISFBROWSE
PRINT	Print data or the screen	No	No		Not supported in REXX
PS	Display the PS panel	Yes	Yes		
PUN	Display the PUN panel	Yes	Yes		
QUERY	List SDSF data	Yes	No		Responses returned in ISFRESP stem
RDR	Display the RDR panel	Yes	Yes		
RES	Display the RES panel	Yes	Yes		
RESET	Clear pending actions	No	No		Not supported in REXX
RIGHT	Scroll right	No	No		Not supported in REXX
RM	Display the RM panel	Yes	Yes		
RSYS	Limit WTORs on SYSLOG by system	No	No		
SE	Display the SE panel	Yes	Yes		
SELECT	Display selected rows	No	No		Not supported in REXX
SET ACTION	Display action characters	No	No	ISFACTIONS	
SET BROWSE	Set default browse action character	No	No		Not supported in REXX
SET CKLIM	Set limit for instances on the CKH panel	No	No	ISFCKLIM	
SET CMODE	Set mode for sysplex communications	No	No	ISFCMODE	
SET CONFIRM	Set confirmation of destructive actions	No	No		Not supported in REXX
SET CONMOD	Set the modification of the extended console name	No	No	ISFCONMOD	
SET CONSOLE	Specify extended console	No	No	ISFCONS	
SET CSORT	Control cursor-sensitive sort	No	No		Not supported in REXX
SET CURSOR	Set cursor placement	No	No		Not supported in REXX
SET DATE	Set date format	No	No	ISFDATE	
SET DELAY	Set timeout value	No	No	ISFDELAY	
SET DISPLAY	Set display of values	No	No	ISFDISPLAY	
SET DUPDS	Set display of duplicate SYSOUT data sets when browsing or printing job data sets	No	No	ISFDUPDS	Duplicate SYSOUT data sets are displayed by default
SET LANGUAGE	Set language for help	No	No		Not supported in REXX
SET LOG	Set default Log panel	No	No		Not supported in REXX

Table 167. SDSF Commands and REXX (continued)

Command	Purpose	Use on ISFEXEC	Use on ISFACT	REXX Variable	Notes
SET PRTCCASA	Set how SDSF handles carriage control for printing	No	No	ISFPRTCCASA	
SET SCHARS	Set wildcard characters	No	No	ISFSCHARS	
SET SCREEN	Set colors	No	No		Not supported in REXX
SET SHELF	Set default bookshelf	No	No		Not supported in REXX
SET TIMEOUT	Set timeout for SYSPLEX function	No	No	ISFTIMEOUT	
SO	Display the SO panel	Yes	Yes		
SORT	Sort a tabular panel	No	No	ISFSORT, ISFSORT2	
SP	Display the SP panel	Yes	Yes		
SR	Display the SR panel	Yes	Yes		
ST	Display the ST panel	Yes	Yes		
SYM	Display the SYM panel	Yes	Yes		
SYS	Display the SYS panel	Yes	Yes		
SYSID	Assign a SYSID for SYSLOG	No	No	ISFSYSID	
SYSNAME	Limit data by system	No	No	ISFSYSNAME	
TOP	Scroll to the top	No	No	ISFSCROLL, ISFSCROLLTYPE	Supported for browse only
TRACE	Enable SDSF tracing	No	No	ISFTRACE ISFTRMASK	
TUTOR	Invoke the SDSF tutorial	No	No		Not supported in REXX
ULOG	Display the ULOG panel	No	No	ISFULOG stem variable	Use the WAIT option on the ISFACT command to ensure that the command responses are available in the ISFULOG stem variable.
UP	Scroll up	No	No	ISFSCROLL, ISFSCROLLTYPE	Supported only for browsing with ISFBROWSE and ISFLOG
WHO	List environmental data	Yes	No		Responses returned in ISFRESP stem variables

Action character reference

The action characters that are available when you use SDSF interactively are available when you use SDSF with REXX. The exceptions are described in Table 168. For information about the available action characters, see the online help.

Table 168. Action Characters Not Supported with REXX

Panel	Not supported	Comments
APF	//, =, +	

Table 168. Action Characters Not Supported with REXX (continued)

Panel	Not supported	Comments
AS	//, =, +	
CK (checks for IBM Health Checker for z/OS)	//, =, +, SB, SBI, SBO, SE, SEI, SEI	Results for S (browse) are returned in the ISFLINE stem variable. For more information, see "Browsing checks with the S action character" on page 398.
CKH (history of a check)	//, =, +	Results for S (browse) are returned in the ISFLINE stem variable. For more information, see "Browsing checks with the S action character" on page 398.
DA (active jobs)	//, =, +, N, Q, S, SB, SE, SJ	For browse, use SA (browse allocate) and SJA (browse allocate JCL) or the ISFBROWSE command. For more information, see "Browsing output" on page 394.
DYNX	//, =, +	
ENC (WLM enclaves)	//, =, +, I	
ENQ	//, =, +	
H (held output queue)	//, =, +, Q, S, SB, SE, SJ	For browse, use SA (browse allocate) and SJA (browse allocate JCL) or the ISFBROWSE command. For more information, see "Browsing output" on page 394.
I (input queue)	//, =, +, I, Q, S, SB, SE, SJ	For browse, use SA (browse allocate) and SJA (browse allocate JCL) or the ISFBROWSE command. For more information, see "Browsing output" on page 394.
INIT (initiators)	//, =, +	
JC (job classes)	//, =, +, ST	
JDS (job data sets)	//, =, +, Q, S, SB, SE, SJ	For browse, use SA (browse allocate) and SJA (browse allocate JCL) or the ISFBROWSE command. For more information, see "Browsing output" on page 394.
JP (members in the JESPLEX)	//, =, +	
J0 (JES3 job 0)	//, =, +, S, SB, SE	Use the ISFBROWSE command.
LI (lines)	//, =, +	
LNK	//, =, +	
LPA	//, =, +	
MAS (members in the MAS)	//, =, +	
NC (network connections)	//, =, +	
NO (nodes)	//, =, +	
NS (network servers)	//, =, +	

Table 168. Action Characters Not Supported with REXX (continued)

Panel	Not supported	Comments
O (output Queue)	//, =, +, Q, S, SB, SE, SJ	For browse, use SA (browse allocate) and SJA (browse allocate JCL) or the ISFBROWSE command. For more information, see "Browsing output" on page 394.
PAG	//, =, +	
PARM	//, =, +	
PR (printers)	//, =, +	
PS (z/OS Unix processes)	//, =, +	
PUN (punches)	//, =, +	
RDR (readers)	//, =, +	
RES (WLM Resources)	//, =, +	
RM (JES2 resources)	//, =, +	
SE (WLM scheduling environments)	//, =, +, R, ST	
SO (spool offloaders)	//, =, +	
SP (spool volumes)	//, =, +	
SR (system requests)	//, =, +, R with no command	
ST (status of all jobs)	//, =, +, Q, I, S, SB, SE, SJ	For browse, use SA (browse allocate) and SJA (browse allocate JCL) or the ISFBROWSE command. For more information, see "Browsing output" on page 394.
SYM	//, =, +	
SYS	//, =, +	

Special variables reference

Table 169 shows the special REXX variables, with the exception of the variables for printing, which are shown in "Printing output" on page 399.

Table 169. Special REXX Variables

Variable	Type	Associated Command	Description	Comments
ISFACTIONS	Input	SET ACTION	Controls the display of action characters for current panel	Action characters and optional descriptions are returned in the ISFRESP stem variables.
ISFAPPC	Input	APPC	Controls the display of APPC transactions	
ISFCMDLIM	Input	Slash (/)	Limits the number of commands that may be issued through ISFSLASH	
ISFCKLIM	Input	SET CKLIM	Sets the maximum number of instances of a check to display on the CKH panel	
ISFCMODE	Input	SET CMODE	Sets the mode for sysplex communication	

Table 169. Special REXX Variables (continued)

Variable	Type	Associated Command	Description	Comments
ISFCOLOR	Output		Stem variable containing the color of each line. The possible values are the first letters of the colors Red, Green, Blue, White, Yellow, Turquoise, Pink.	OPERLOG only
ISFCOLS	InOut		Input: sets the list of columns to be returned Output: contains list of columns that are returned	Limits the columns (and so the variables) that are created
ISFCOLS2	InOut		Input: sets the list of columns to be returned for a secondary panel Output: contains the list of columns that are returned for a secondary panel	Limits the columns (and so the variables) that are created
ISFCOLUMNGROUPS	Output		Lists column grouping information for the columns listed in the ISFCOLS variable. The group values are a way of categorizing SDSF columns. The values are: NONE, ACCT (accounting), ACTIVITY, ADVANCED, GENERAL, INPUT, JES2, JES3, OUTPUT (printer), OUTPUN (punch), PERF (performance), PRINTING, RUNTIME, SECURITY, SCHED (scheduling), SELECT, STATUS and STATWLM (workload management status).	
ISFCONMOD	Input	SET CONMOD	Controls the automatic modification of the extended console name when SDSF needs to activate a console (for issuing system commands and for the ULOG) and the default console name is in use	
ISFCONS	Input	SET CONSOLE	Sets the console name	If you have disabled console modification, you should change the console name when running a REXX exec while running SDSF interactively, to avoid an activation failure because the console is already in use.
ISFDATE	Input	SET DATE	Sets the date format for input on special variables	Does not affect the date format for returned stem variables

Table 169. Special REXX Variables (continued)

Variable	Type	Associated Command	Description	Comments
ISFDCOLS	Output		Contains the list of delayed access columns for the panel	
ISFDCOLS2	Output		Contains the list of delayed access columns for the secondary panel	
ISFDDNAME	Output, Stem		Stem variable that contains the system-generated DDNAME of an allocated SYSOUT data set. ISFDDNAME.0 contains a count of the number of variables that follow.	Set in response to a browse allocation action character, such as SA and SJA
ISFDELAY	Input	SET DELAY	Sets the response delay limit for system commands	
ISFDESCODE			Stem variable containing the descriptor codes for each line. When there are multiple descriptor codes, they are returned in a list, separated by blanks.	OPERLOG only
ISFDEST	Input	DEST	Sets the destinations to be used for filtering	Allows up to four destinations, with each being up to the maximum acceptable length for a destination
ISFDIAG	Output		Intended for use by IBM service personnel	See "Diagnosing errors in a REXX exec" on page 449.
ISFDISPLAY	Output		Contains the SET DISPLAY response for tabular panels	
ISFDISPLAYMODE	Input	SET DISPLAY	Sets the format of the ISFDISPLAY special variable	The value OFF is not valid with REXX.
ISFDSNAME	Output, Stem		Stem variable that contains the application-specified data set name (that is, the data set name as shown on the Job Data Set panel). Corresponds to the DDNAME listed in ISFDDNAME. The variables have a one-to-one correspondence with the ISFDDNAME stem variables. ISFDSNAME.0 contains a count of the number of variables that follow.	Set in response to a browse allocation action character, such as SA and SJA
ISFDUPDS	Input	SET DUPDS	Controls whether duplicate SYSOUT data sets are included when browsing or printing	
ISFFILTER	Input	FILTER	Sets filter criteria	Use column names rather than column titles. Supported with tabular panels.

Table 169. Special REXX Variables (continued)

Variable	Type	Associated Command	Description	Comments
ISFFILTER2	Input	FILTER	Sets filter criteria for a secondary panel	Use column names rather than column titles.
ISFFILTERMODE	Input	FILTER	Sets the relationship between filters	
ISFFILTERMODE2	Input	FILTER	Sets the relationship between filters for a secondary panel	
ISFFIND	Input	FIND	String to be found (up to 255 characters).	Use when browsing with ISFBROWSE or ISFLOG.
ISFFINDENDCOL	Input	FIND	Column in which the string specified with ISFFIND must end.	Use when browsing with ISFBROWSE or ISFLOG.
ISFFINDLIM	Input	FINDLIM	Maximum number of lines to search for the string specified with ISFFIND. 1000 to 9999999.	Use when browsing with ISFBROWSE or ISFLOG.
ISFFINDSTARTCOL	Input	FIND	Column in which the string specified with ISFFIND must start.	Use when browsing with ISFBROWSE or ISFLOG.
ISFFIRSTLINEDATE	Output		Date associated with the first line that was returned.	Use when browsing the log.
ISFFIRSTLINEDSID	Output		Data set identifier of the data set associated with the first line that was returned.	Use when browsing. Not valid with OPERLOG.
ISFFIRSTLINEJOBID	Output		Job ID associated with the first line that was returned.	Use when browsing the SYSLOG.
ISFFIRSTLINERECNO	Output		Record number within the data set of the first line that was returned.	Use when browsing. Not valid with OPERLOG.
ISFFIRSTLINETIME	Output		Time associated with the first line that was returned.	Use when browsing the log.
ISFFIRSTLINETOKEN	Output		Token corresponding to the first line of the data that was returned.	Use when browsing with ISFBROWSE or ISFLOG.
ISFHIGHLIGHT	Output		Stem variable containing the highlighting of each line. The possible values are the first letters of Blink, Reverse, Underline and None.	OPERLOG only
ISFINPUT	Input	INPUT	Controls which data sets will be returned	
ISFINTENSITY	Output		Stem variable containing the intensity of each line. The possible values are the first letters of High and Low.	OPERLOG only
ISFJESNAME	Input		Sets the JES subsystem to be processed	Equivalent to the value specified on the JESNAME option of the SDSF command (JES2 only).

Table 169. Special REXX Variables (continued)

Variable	Type	Associated Command	Description	Comments
ISFJES3NAME	Input		Sets the JES subsystem to be processed	Equivalent to the value specified on the JES3NAME option of the SDSF command (JES3 only).
ISFLASTLINEDATE	Output		Date associated with the last line that was returned.	Use when browsing the log.
ISFLASTLINEDSID	Output		Data set identifier of the data set associated with the last line that was returned.	Use when browsing. Not valid with OPERLOG.
ISFLASTLINEJOBID	Output		Job ID associated with the last line that was returned.	Use when browsing the SYSLOG.
ISFLASTLINERECNO	Output		Record number within the data set of the last line that was returned.	Use when browsing. Not valid with OPERLOG.
ISFLASTLINETIME	Output		Time associated with the last line that was returned.	Use when browsing the log.
ISFLINE	Output, Stem		Stem variable that contains the result of a browse request. ISFLINE.0 contains a count of the number of variables that follow.	Use when browsing the log or a check.
ISFLINELIM	Input		Limits the number of ISFLINE stem variables that may be created. The valid range is 0-99999999. A value of zero indicates no limit.	If the variable is not defined or null, there is no limit.
ISFLOGSTARTTIME	Input		Specifies the starting time for records returned by the ISFLOG command, in *hh*:*mm*:*ss.th* format. Only *hh*:*mm* is required. This is the local time corresponding to the first record to be returned.	If the variable is not defined or the value is null, the starting time is 00:00:00.00.
ISFLOGSTARTDATE	Input		Specifies the starting date for records returned by the ISFLOG command, in the current date format or either of these formats: *yyyy.ddd* or *yy.ddd*.	The default is the current day.
ISFLOGSTOPTIME	Input		Specifies the ending time for records returned by the ISFLOG command, in *hh*:*mm*:*ss.th* format. Only *hh*:*mm* is required. This is the local time corresponding to the last record to be returned.	If the variable is not defined or the value is null, the ending time is 23:59:59.99.
ISFLOGSTOPDATE	Input		Specifies the ending date for records returned by the ISFLOG command, in the current date format or either of these formats: *yyyy.ddd* or *yy.ddd*.	The default is the current day.

Table 169. Special REXX Variables (continued)

Variable	Type	Associated Command	Description	Comments
ISFLRECL	Output, Stem		Stem variable that contains the logical record length for the allocated data set and corresponds to the DDNAME listed in ISFDNAME. ISFLRECL.0 contains a count of the number of variables that follow.	
ISFMSG	Output		Contains the SDSF short message, if any, set on the completion of each request	Check at the completion of each request.
ISFMSG2	Output, Stem		Stem variable that is set to any numbered messages that may have been issued in response to the request. ISFMSG2.0 contains the count of message variables that follow. The message variables contain the oldest message first.	Check at the completion of each request.
ISFNEXTLINETOKEN	Output		Token corresponding to the next unread line of the data. It is null when an end-of-file condition is encountered.	Use when browsing with ISFBROWSE or ISFLOG.
ISFOWNER	Input	OWNER	Sets the owner to be used for filtering	Use the default SDSF generic characters unless you change them with the ISFSCHARS variable.
ISFPREFIX	Input	PREFIX	Sets the job name prefix to be used for filtering	Uses the default SDSF generic characters unless you change them with the ISFSCHARS variable.
ISFPRTBLKSIZE	Input		Block size for new data sets	Use with XD and XDC action characters.
ISFPRTCCASA	Input	SET PRTCCASA	Sets how SDSF handles carriage control for printing	Use with ISFPRTRECFM.
ISFPRTCLASS	Input		SYSOUT class	Use with X, XC, XS and XSC action characters.
ISFPRTCOPIES	Input		Copies class	Use with X, XC, XS and XSC action characters.
ISFPRTDATACLAS	Input		Data class for new data sets	Use with XD and XDC action characters.
ISFPRTDDNAME	Input		DDNAME	Use with XF and XFC action characters.
ISFPRTDEST	Input		Destination	Use with X, XC, XS and XSC action characters.
ISFPRTDIRBLKS	Input		Number of directory blocks for new data sets	Use with XD and XDC action characters.

Table 169. Special REXX Variables (continued)

Variable	Type	Associated Command	Description	Comments
ISFPRTDISP	Input		Allocation disposition for data sets	Use with XD and XDC action characters.
ISFPRTDSNAME	Input		Data set name. If the name is not enclosed in quotation mark, the name begins with the current user ID.	Use with XD and XDC action characters.
ISFPRTFCB	Input		FCB	Use with X, XC, XS and XSC action characters.
ISFPRTFORMDEF	Input		FORMDEF	Use with X, XC, XS and XSC action characters.
ISFPRTFORMS	Input		Forms	Use with X, XC, XS and XSC action characters.
ISFPRTLRECL	Input		Logical record length	Use with XD, XDC, XS and XSC action characters.
ISFPRTMEMBER	Input		Member name	Use with XD and XDC action characters.
ISFPRTMGMTCLAS	Input		Management class for new data sets	Use with XD and XDC action characters.
ISFPRTOUTDESNAME	Input		Output descriptor name to be used when creating the file	Use with X, XC, XS and XSC action characters.
ISFPRTPAGEDEF	Input		PAGEDEF	Use with X, XC, XS and XSC action characters.
ISFPRTPRIMARY	Input		Primary space allocation for new data sets	Use with XD and XDC action characters.
ISFPRTPRTMODE	Input		Process mode	Use with X, XC, XS and XSC action characters.
ISFPRTRECFM	Input		Record format	Use with XD, XDC, XS and XSC action characters.
ISFPRTSECONDARY	Input		Secondary space allocation for new data sets	Use with XD and XDC action characters.
ISFPRTSOURCEATTS	Input		Whether to use attributes of the source for printing	Use with the XS and XSC action characters.
ISFPRTSPACETYPE	Input		Space units for allocating for new data sets	Use with XD and XDC action characters.
ISFPRTSTORCLAS	Input		Storage class for new data sets	Use with XD and XDC action characters.
ISFPRTUCS	Input		UCS	Use with X, XC, XS and XSC action characters.
ISFPRTUNIT	Input		Unit for new data sets	Use with XD and XDC action characters.
ISFPRTVOLSER	Input		Volume serial for new data sets	Use with XD and XDC action characters.
ISFPRTWRITER	Input		Writer name	Use with the XS and XSC action characters.

Table 169. Special REXX Variables (continued)

Variable	Type	Associated Command	Description	Comments
ISFRCOLS	Output		Contains a list of columns with related fields	Related fields are sets of related columns, such as SFORMS and SFORM2-8 on the Printer panel.
ISFRCOLS2	Output		Contains a list of columns with related fields for a secondary panel	
ISFRECFM	Output, Stem		Stem variable that contains the record format for the allocated data set and corresponds to the DDNAME listed in ISFDDNAME. ISFRECFM.0 contains a count of the number of variables that follow.	
ISFRESP	Output, Stem		Stem variable that contains responses from commands. ISFRESP.0 contains the count of the response variables that follow.	Commands such as WHO use the ISFRESP stem variables to provide the command response.
ISFROWS	Output		Contains the number of rows created by a request for a tabular panel	Equivalent to the zero stem for each of the column variables
ISFROWS2	Output		Contains the number of rows created by a request for a secondary panel	Equivalent to the zero stem for each of the column variables
ISFSCHARS	Input	SET SCHARS	Sets the generic and placeholder characters to be used in pattern matching	
ISFSCROLL	Input	Scrolling commands	Repositions the first line of data that is returned	Use when browsing with ISFBROWSE or ISFLOG.
ISFSCROLLTYPE	Input	Scrolling commands	Repositions the first line of data that is returned	Use with ISFSCROLL.
ISFSECTRACE	Input	SET SECTRACE	Controls tracing of SDSF security	Use with ISFMSG2 or ISFULOG.
ISFSERVER	Input		Sets the SDSF server to be used when initializing SDSF	Corresponds to the SERVER option on the SDSF command
ISFSORT	Input	SORT	Sets the sort criteria	Use the column names instead of the column titles. To sort using the fixed field, assign the value to null.
ISFSORT2	Input	SORT	Sets the sort criteria for a secondary panel	Use the column names instead of the column titles. To sort using the fixed field, assign the value to null.
ISFSTARTLINETOKEN	Input		Starting line for the data to be returned.	Specify this value by setting the variable to either ISFFIRSTLINETOKEN or ISFNEXTLINETOKEN.

Table 169. Special REXX Variables (continued)

Variable	Type	Associated Command	Description	Comments
ISFSYSID	Input	SYSID	Specifies the member to be processed by the ISFLOG command	
ISFSYSNAME	Input	SYSNAME	Sets the system name to be used for filtering sysplex requests	Use the default SDSF generic characters unless you have changed them with the ISFSCHARS variable.
ISFTIMEOUT	Input	SET TIMEOUT	Sets the response timeout value for sysplex requests	JES2 only
ISFTITLES	Output		Contains the column titles associated with the variables that are returned	The titles are listed in the same order as the column names in the ISFCOLS variable. Titles are enclosed by single quotation marks and separated by blanks.
ISFTITLES2	Output		Contains the column titles associated with the variables that are returned for the secondary panel	The titles are listed in the same order as the column names in the ISFCOLS2 variable. Titles are enclosed by single quotation marks and separated by blanks.
ISFTLINE	Output		Contains the title line from the tabular panel	The title line frequently contains dynamic data related to the panel being accessed. The format of the data may vary and is subject to change at any time.
ISFTRACE	Input	TRACE	Sets the trace option to be used when enabling SDSF trace	This variable is intended to be used for the trace option since two trace commands are necessary to enable tracing. However, any operand acceptable to the trace command will be accepted for this variable.
ISFTRMASK	Input	TRACE	Sets the trace mask to be used when enabling SDSF trace	This variable is intended to be used for a trace mask since two trace commands are necessary to enable tracing: one to enable trace and the other for the mask. However, any non-blank operand acceptable to the trace command will be accepted for this variable. This variable is ignored if the value is null.
ISFUCOLS	Output		Contains the list of modifiable columns for the panel	Contains the columns defined as modifiable, but you may not necessarily be authorized to modify them. Authorization is not determined until you attempt to modify a column.

Table 169. Special REXX Variables (continued)

Variable	Type	Associated Command	Description	Comments
ISFUCOLS2	Output		Contains the list of modifiable columns for the secondary panel	Contains the columns defined as modifiable, but you may not necessarily be authorized to modify them. Authorization is not determined until you attempt to modify a column.
ISFULOG	Output, Stem		Stem variable that contains the MVS system command echo and any responses generated during the session, including SAF authorization messages. The ISFULOG.0 stem variable contains a count of the variables that follow.	The ISFULOG stem variables are formatted in the same manner as the ULOG panel. Use the WAIT option on the ISFACT command to ensure that the command responses are available in the ISFULOG stem variable.
ROWACTIVE	Output, Stem		Stem variable that indicates whether the object (for example, the job or the printer) is active. The value is either Y (active) or N (inactive). ROWACTIVE.0 contains a count of the number of stem variables that follow.	

Examples of REXX execs

The examples in this topic contain just the SDSF-specific portions of the execs.

For information about other examples, see "Other sources of information" on page 377.

Access an SDSF panel

1. Access the ST panel, then list the column variables.

```
/* REXX */
rc=isfcalls('ON')
     /* Access the ST panel */
Address SDSF "ISFEXEC ST"
if rc<>0 then
  Exit rc
     /* Get fixed field name from first word */
     /* of isfcols special variable          */
fixedField = word(isfcols,1)
Say "Number of rows returned:" isfrows
       /* Process all rows */
do ix=1 to isfrows
  Say "Now processing job:" value(fixedField"."ix)
         /* List all columns for row */
  do jx=1 to words(isfcols)
    col = word(isfcols,jx)
    Say "  Column" col"."ix "has the value:" value(col"."ix)
  end
end
rc=isfcalls('OFF')
```

2. Use the ISFCOLS special variable to limit the columns to Job Name and Owner, then access the ST panel. Add the following statement to the exec in example 1, prior to the ISFEXEC command.

```
ISFCOLS = 'JNAME OWNERID'
```

Cancel a job

Cancel all jobs with a certain job name using the P action character. First, access the ST panel to create the row variables for each job and the associated tokens. Loop through the rows, checking the job name for each in the JNAME variables. When the desired job name is found, use the ISFACT command to issue the P action character.

```
/* REXX */
 rc=isfcalls('ON')
       /* Set the jobname prefix and owner */
 isfprefix="**"
 isfowner="*"
     /* Access the ST panel.  A TOKEN variable is  */
     /* created for each row which is subsequently */
     /* needed to perform actions                  */
 Address SDSF "ISFEXEC ST"
 lrc=rc
 call msgrtn  /* List any error messages */
 if lrc<>0 then
   exit 20
     /* Find all jobs starting with RJONES and cancel them */
 numrows=isfrows
 do ix=1 to numrows    /* Loop for all rows returned */
  if pos("RJONES",JNAME.ix) = 1 then  /* If this is desired row */
   do
     /* Issue the P action character for the job    */
     /* identified by the token variable.  Note     */
     /* the token must be enclosed in single quotes */
     Address SDSF "ISFACT ST TOKEN('"TOKEN.ix"') PARM(NP P)"
     lrc=rc
     call msgrtn
     if lrc<>0 then
       exit 20
   end
end
rc=isfcalls('OFF')
Exit
     /* Subroutine to list error messages */
 msgrtn: procedure expose isfmsg isfmsg2.
      /* The isfmsg variable contains a short message */
   if isfmsg<>"" then
   Say "isfmsg is:" isfmsg
      /* The isfmsg2 stem contains additional descriptive */
      /* error messages                                   */
   do ix=1 to isfmsg2.0
   Say "isfmsg2."ix "is:" isfmsg2.ix
 end
 return
```

Cancel a set of jobs

After setting the special variables isfprefix and isfowner to limit the jobs returned, use ISFEXEC to access the ST panel. Then use ISFACT to issue the P action character for all of the jobs returned.

```
/* REXX */
rc=isfcalls('ON')
     /* Set the jobname prefix and owner */
isfprefix="ctest*"
isfowner="weber"
    /* Access the ST panel.  A TOKEN variable is  */
    /* created for each row which is subsequently */
    /* needed to perform actions                  */
Address SDSF "ISFEXEC ST"
lrc=rc
call msgrtn  /* List any error messages */
if lrc<>0 then
  exit 20
/* The tokens have already been assigned to the TOKEN stem  */
/* by ISFEXEC.  TOKEN.0 has the count of tokens.  All rows  */
/* returned by ISFEXEC will be canceled with the single     */
/* invocation of ISFACT.                                    */
Address SDSF "ISFACT ST TOKEN((TOKEN.)) PARM(NP P)"
lrc=rc
call msgrtn
if lrc<>0 then
  exit 20
rc=isfcalls('OFF')
Exit
       /* Subroutine to list error messages */
msgrtn: procedure expose isfmsg isfmsg2.
       /* The isfmsg variable contains a short message */
   if isfmsg<>"" then
      Say "isfmsg is:" isfmsg
       /* The isfmsg2 stem contains additional descriptive */
       /* error messages                                   */
  do ix=1 to isfmsg2.0
    Say "isfmsg2."ix "is:" isfmsg2.ix
  end
  return
```

List job data sets

Access the O panel to create the row variables and the associated tokens. Loop through the rows, checking the job name (JNAME) variables. When the desired job name is found, use the ISFACT command to issue the ? action character. Then, loop through the rows to list the data sets.

```
/* REXX */
rc=isfcalls('ON')
     /* Access the ST panel.  A TOKEN variable is  */
     /* created for each row which is subsequently */
     /* needed to perform actions                  */
 Address SDSF "ISFEXEC ST"
 lrc=rc
 call msgrtn  /* List any error messages */
 if lrc<>0 then
   exit 20
     /* Find a job starting with RJONES and list data sets */
 numrows=isfrows
 do ix=1 to numrows    /* Loop for all rows returned */
  if pos("RJONES",JNAME.ix) = 1 then  /* If this is desired row */
    do
      /* Issue the ? action character for the job    */
      /* identified by the token variable.  Note     */
      /* the token must be enclosed in single quotes */
      /* Use the prefix option to ensure unique      */
      /* variables are created, beginning with JDS_  */
      Address  SDSF "ISFACT ST TOKEN('"TOKEN.ix"') PARM(NP ?)",
"("prefix JDS_
      lrc=rc
      call msgrtn
      if lrc<>0 then
        exit 20
      do jx=1 to JDS_DDNAME.0   /* loop for all rows returned */
        say "DDNAME is " JDS_DDNAME.jx
      end
      lrc=rc
      call msgrtn
       if lrc<>0 then
         exit 20
    end
 end
  rc=isfcalls('OFF')
  Exit
      /* Subroutine to list error messages */
msgrtn: procedure expose isfmsg isfmsg2.
      /* The isfmsg variable contains a short message */
if isfmsg<>"" then
  Say "isfmsg is:" isfmsg
     /* The isfmsg2 stem contains additional descriptive */
     /* error messages                                   */
do ix=1 to isfmsg2.0
  Say "isfmsg2."ix "is:" isfmsg2.ix
end
return
```

Modify values in columns

Modify a value

Using ISFEXEC, access the O panel. Then, for jobs with a particular owner (RJONES), use ISFACT to change the class to A and forms to 1234.

```
/* REXX */
rc=isfcalls('ON')
     /* Access the O display */
Address SDSF "ISFEXEC O"
lrc=rc
call msgrtn
if lrc<>0 then
  exit 20
     /* Find all jobs owned by RJONES */
 do ix=1 to OWNERID.0
  if OWNERID.ix = "RJONES" then    /* If this is desired row */
    do
      /* Issue the action against the row identified by */
      /* the token.  The PARM contains the column name  */
      /* to be modified and the data to use.            */
      Address SDSF "ISFACT O TOKEN('"TOKEN.ix"')",
         "PARM(OCLASS A FORMS 1234)"
       lrc=rc
       call msgrtn
       if lrc<>0 then
        exit 20
    end
end
rc=isfcalls('OFF')
exit
     /* Subroutine to list error messages */
msgrtn: procedure expose isfmsg isfmsg2.
     /* The isfmsg variable contains a short message */
     /*********************************************/
if isfmsg<>"" then
  Say "isfmsg is:" isfmsg
     /* The isfmsg2 stem contains additional descriptive */
     /* error messages                                   */
do ix=1 to isfmsg2.0
  Say "isfmsg2."ix "is:" isfmsg2.ix
end
return
```

Modify a set of values

When a column has a set of related values, you use a +column syntax on the ISFACT statement to show that you are supplying multiple values. This example shows the ISFACT statement to supply multiple values for SDESTN1 on the PR column. You could use it with an exec like the one in the first example. Note that if you queried the contents of the columns, SDESTN1 would contain only the first value. The second value would be in SDESTN2.

```
  Address "SDSF ISFACT PR TOKEN('"TOKEN.ix"')",
    "PARM(SDESTN1 D1 +SDESTN1 D2)"
```

Modify a value for a set of jobs

After setting the special variables isfprefix and isfowner to limit the jobs returned, use ISFEXEC to access the ST panel. Then use ISFACT to change the priority of those jobs to 10.

```
/* REXX */
rc=isfcalls("on")
isfprefix="**"
isfowner="ken"
Address SDSF "ISFEXEC ST"
if rc=0 then
  do
    /* The tokens have already been assigned to the TOKEN stem  */
    /* by ISFEXEC.  TOKEN.0 has the count of tokens.  All rows  */
    /* returned by ISFEXEC will be changed with the single      */
    /* invocation of ISFACT.                                    */
    Address SDSF "ISFACT ST TOKEN((token.)) PARM(JPRIO 10)"
    /* List messages returned by ISFACT */
    do ix=1 to isfmsg2.0
      Say isfmsg2.ix
    end
    /* List returned command responses */
    do ix=1 to isfulog.0
      Say isfulog.ix
    end
  end
 rc=isfcalls("off")
```

Browse job output with EXECIO

Using ISFEXEC, access the ST panel to create the row variables for jobs. Then, for each job with a name that matches a desired string (RJONES1), use ISFACT to issue the SA action character. SA allocates the job data sets and sets the ISFDDNAME special variable to the DDNAME for each data set that has been allocated. Use the ISFDDNAME variable as input on the EXECIO command and list the contents of the data sets.

```
/* REXX */
rc=isfcalls('ON')
     /* Access the ST display */
Address SDSF "ISFEXEC ST"
lrc=rc
call msgrtn
if lrc<>0 then
  exit 20
     /* Loop for all RJONES jobs */
do ix=1 to JNAME.0
  if JNAME.ix = "RJONES" then
    do
      /* Issue the SA action against the row to */
      /* allocate all data sets in the job.     */
      Address SDSF "ISFACT ST TOKEN('"TOKEN.ix"') PARM(NP SA)"
      lrc=rc
      call msgrtn
      if lrc<>0 then
        exit 20
      /* The data set name for each allocated data    */
      /* set is contained in the isfdsname stem.  The */
      /* ddname returned by allocation is contained   */
      /* in the isfddname stem.                       */
      Say "Number of data sets allocated:" value(isfdsname".0")
      /* Read the records from each data set and list them */
      do jx=1 to isfddname.0
        Say "Now reading" isfdsname.jx
        "EXECIO * DISKR" isfddname.jx "(STEM line. FINIS"
        Say "  Lines read:" line.0
        do kx = 1 to line.0
          Say "  line."kx "is:" line.kx
        end
      end
    end
 end
 rc=isfcalls('OFF')
 exit
     /* Subroutine to list error messages */
msgrtn: procedure expose isfmsg isfmsg2.
     /* The isfmsg variable contains a short message */
if isfmsg<>"" then
  Say "isfmsg is:" isfmsg
     /* The isfmsg2 stem contains additional descriptive */
     /* error messages                                   */
do ix=1 to isfmsg2.0
  Say "isfmsg2."ix "is:" isfmsg2.ix
end
return
```

Browse job output with ISFBROWSE (basic)

Using ISFEXEC, access the ST panel to create the row variables for jobs. Then, for each job with a name that matches a desired string (RJONES), use the ISFBROWSE command to display the output for that job.

```
/* REXX */

rc=isfcalls("on")
```

```
     /************************/
     /* Access the ST display */
     /************************/
Address SDSF "ISFEXEC ST"
lrc=rc
call msgrtn
if lrc<>0 then
  exit 20
     /*****************************/
     /* Loop for all RJONES jobs */
     /*****************************/
do ix=1 to JNAME.0
  if JNAME.ix = "RJONES" then
    do
        Address SDSF "ISFBROWSE ST TOKEN('"token.ix"')"
        call msgrtn
        if rc>4 then
            exit 20
            /*****************************/
            /* Loop through the lines   */
            /*****************************/
        do jx=1 to isfline.0
         say isfline.jx
        end
    end
end

rc=isfcalls("off")

exit

     /************************************/
     /* Subroutine to list error messages */
     /************************************/
msgrtn: procedure expose isfmsg isfmsg2.
     /************************************************/
     /* The isfmsg variable contains a short message */
     /************************************************/
if isfmsg<>"" then
  Say "isfmsg is:" isfmsg
     /*****************************************************/
     /* The isfmsg2 stem contains additional descriptive */
     /* error messages                                    */
     /*****************************************************/
do ix=1 to isfmsg2.0
  Say "isfmsg2."ix "is:" isfmsg2.ix
end

return
```

Browse job output with ISFBROWSE

From the ST panel, for each job with the name RJONES, use the ISFBROWSE command to display the output. Use the isflinelim variable to limit the number of REXX variables returned by SDSF. Set the isfstartlinetoken variable to the returned value isfnextlinetoken, to allow the browse to continue with the next line in the display.

```
/* REXX */
rc=isfcalls("on")
     /************************/
     /* Access the ST display */
     /************************/
Address SDSF "ISFEXEC ST"
lrc=rc
call msgrtn
```

```
if lrc<>0 then
  exit 20
     /*****************************/
     /* Loop for all RJONES jobs */
     /*****************************/
do ix=1 to JNAME.0
  if JNAME.ix = "RJONES" then
    do
      isflinelim = 500
      do until isfnextlinetoken=''
        Address SDSF "ISFBROWSE ST TOKEN('"token.ix"')"
        if rc>4 then
          do
            call msgrtn
            exit 20
          end
            /*****************************/
            /* Loop through the lines   */
            /*****************************/
        do jx=1 to isfline.0
         say isfline.jx
        end
            /*****************************/
            /* Set start for next browse */
            /*****************************/
        isfstartlinetoken = isfnextlinetoken
      end
    end
end
rc=isfcalls("off")
exit
     /************************************/
     /* Subroutine to list error messages */
     /************************************/
msgrtn: procedure expose isfmsg isfmsg2.
     /************************************************/
     /* The isfmsg variable contains a short message */
     /************************************************/
if isfmsg<>"" then
  Say "isfmsg is:" isfmsg
     /*************************************************/
     /* The isfmsg2 stem contains additional messages    */
     /*************************************************/
do ix=1 to isfmsg2.0
  Say "isfmsg2."ix "is:" isfmsg2.ix
end
return
```

Browse a single data set with EXECIO

Using ISFEXEC, access the ST panel to create the row variables for jobs. Then, find an active job named RJONES. Use ISFACT to issue the ? action character and list the job's data sets, adding the prefix option to ensure that you create unique variables. Find the message log data set, allocate it, and read it using EXECIO.

```
/* REXX */
rc=isfcalls('ON')
     /* Access the ST display */
Address SDSF "ISFEXEC ST"
lrc=rc
call msgrtn
if lrc<>0 then
  exit 20
     /* Loop for all running RJONES jobs */
do ix=1 to JNAME.0
  if JNAME.ix = "RJONES" & ,
     QUEUE.ix = "EXECUTION" & ,
     ACTSYS.ix <> "" then
    do
      /* Issue the ? (JDS) action against the  */
      /* row to list the data sets in the job. */
      Address SDSF "ISFACT ST TOKEN('"TOKEN.ix"') PARM(NP ?)" ,
         "( prefix jds_"
      lrc=rc
      call msgrtn
      if lrc<>0 then
        exit 20
      /* Find the JESMSGLG data set and allocate it */
      /* using the SA action character              */
      do jx=1 to jds_DDNAME.0
        if jds_DDNAME.jx = "JESMSGLG" then
          do
            Address SDSF "ISFACT ST TOKEN('"jds_TOKEN.jx"')" ,
              "PARM(NP SA)"
            lrc=rc
            call msgrtn
            if lrc<>0 then
              exit 20
            /* Read the records from the data set and list them. */
            /* The ddname for each allocated data set will be in */
            /* the isfddname stem.  Since the SA action was done */
            /* from JDS, only one data set will be allocated.    */
            do kx=1 to isfddname.0
              Say "Now reading" isfdsname.kx
              "EXECIO * DISKR" isfddname.kx "(STEM line. FINIS"
              Say "  Lines read:" line.0
              do lx = 1 to line.0
                Say "  line."lx "is:" line.lx
              end
            end
          end
      end
 end
 rc=isfcalls('OFF')
 exit
     /* Subroutine to list error messages */
msgrtn: procedure expose isfmsg isfmsg2.
     /* The isfmsg variable contains a short message */
if isfmsg<>"" then
  Say "isfmsg is:" isfmsg
     /* The isfmsg2 stem contains additional descriptive */
     /* error messages                                   */
 do ix=1 to isfmsg2.0
  Say "isfmsg2."ix "is:" isfmsg2.ix
end
return
```

Browse a single data set with ISFBROWSE

Using ISFEXEC, access the ST panel to create the row variables for jobs. Then, find an active job named RJONES. Use ISFACT to issue the ? action character and list

the job's data sets, adding the prefix option to ensure that you create unique variables. Find the message log data set, and read it using ISFBROWSE.

```
/* REXX */

rc=isfcalls('ON')

     /*************************/
     /* Access the ST display */
     /*************************/
Address SDSF "ISFEXEC ST"
lrc=rc
call msgrtn
if lrc<>0 then
  exit 20

    /***********************************/
    /* Loop for all running RJONES jobs */
    /***********************************/
do ix=1 to JNAME.0

  if JNAME.ix = "RJONES" & ,
     QUEUE.ix = "EXECUTION" & ,
     ACTSYS.ix <> "" then
    do
      /*****************************************/
      /* Issue the ? (JDS) action against the  */
      /* row to list the data sets in the job. */
      /*****************************************/
      Address SDSF "ISFACT ST TOKEN('"TOKEN.ix"') PARM(NP ?)" ,
         "( prefix jds_"
      lrc=rc
      call msgrtn
      if lrc<>0 then
        exit 20

      /*********************************************/
      /* Find the JESMSGLG data set and read it     */
      /* using ISFBROWSE.  Use isflinelim to limit  */
      /* the number of REXX variables returned.     */
      /*********************************************/
      isflinelim=500
      do jx=1 to jds_DDNAME.0

        if jds_DDNAME.jx = "JESMSGLG" then
          do
           /*****************************************************/
           /* Read the records from the data set.               */
           /*****************************************************/
             total_lines = 0
             do until isfnextlinetoken=''

               Address SDSF "ISFBROWSE ST TOKEN('"jds_TOKEN.jx"')"

               do kx=1 to isfline.0
                  Say "Line" total_lines+kx "is:" isfline.kx
               end

               total_lines = total_lines + isfline.0
                  /*****************************/
                  /* Set start for next browse */
                  /*****************************/
               isfstartlinetoken = isfnextlinetoken

             end

             Say "  Lines read:" total_lines
```

```
          end
       end
     end
end

rc=isfcalls('OFF')

exit

     /************************************/
     /* Subroutine to list error messages */
     /************************************/
msgrtn: procedure expose isfmsg isfmsg2.

     /*********************************************/
     /* The isfmsg variable contains a short message */
     /*********************************************/
if isfmsg<>"" then
  Say "isfmsg is:" isfmsg

     /*****************************************************/
     /* The isfmsg2 stem contains additional descriptive */
     /* error messages                                    */
     /*****************************************************/
do ix=1 to isfmsg2.0
  Say "isfmsg2."ix "is:" isfmsg2.ix
end

return
```

Browse check output from the CK panel

Using ISFEXEC, access the CK panel with the E parameter, which requests only exception checks. For the RACF_GRS_RNL check on SY1, which found an exception, use ISFACT to issue the S action to browse the check. Browsing a check causes the ISFLINE special variable stem variables to be created. List the contents of ISFLINE.

```
/* REXX */
rc=isfcalls('ON')
     /* Access the CK panel and filter by exceptions */
Address SDSF "ISFEXEC CK E"
lrc=rc
call msgrtn
if lrc<>0 then
  exit 20
found=0
     /* Find the RACF_GRS_RNL check that is running on SY1 */
do ix=1 to NAME.0 while found=0
  if NAME.ix = "RACF_GRS_RNL" & SYSNAME.ix = "SY1" then
    do
      found=1
      /* Issue the S action against the check.  This will */
      /* return the check output in the isfline stem.     */
      Address SDSF "ISFACT CK TOKEN('"TOKEN.ix"') PARM(NP S)"
      lrc=rc
      call msgrtn
      if lrc<>0 then
        exit 20
      /* List each line of check output */
      do jx=1 to isfline.0
        Say "Check line" jx":" isfline.jx
      end
    end
end
 if found=0 then
  say "Check not found"
 rc=isfcalls('OFF')
exit
     /* Subroutine to list error messages */
msgrtn: procedure expose isfmsg isfmsg2.
     /* The isfmsg variable contains a short message */
if isfmsg<>"" then
  Say "isfmsg is:" isfmsg
     /* The isfmsg2 stem contains additional descriptive */
     /* error messages                                   */
do ix=1 to isfmsg2.0
  Say "isfmsg2."ix "is:" isfmsg2.ix
end
return
```

Browse check output from the CK panel using ISFBROWSE

Using ISFEXEC, access the CK panel with E parameter, which requests only exception checks. For the RACF_GRS_RNL check on SY1, use ISFBROWSE to browse the check. Browsing a check causes the ISFLINE special variable stem variables to be created. List the contents of ISFLINE.

```
/* REXX */
rc=isfcalls('ON')
     /*********************************************/
     /* Access the CK panel and filter by exceptions */
     /*********************************************/
Address SDSF "ISFEXEC CK E"
lrc=rc
call msgrtn
if lrc<>0 then
```

```
  exit 20
found=0
     /*****************************************************/
     /* Find the RACF_GRS_RNL check that is running on SY1 */
     /*****************************************************/
do ix=1 to NAME.0 while found=0
  if NAME.ix = "RACF_GRS_RNL" & SYSNAME.ix = "SY1" then
    do
      found=1
      /*************************************************/
      /* Issue ISFBROWSE against the check.  This will    */
      /* return the check output in the isfline stem.     */
      /*************************************************/
      Address SDSF "ISFBROWSE CK TOKEN('"TOKEN.ix"')"
      lrc=rc
      call msgrtn
      if lrc<>0 then
        exit 20
      /*********************************/
      /* List each line of check output */
      /*********************************/
      do jx=1 to isfline.0
        Say "Check line" jx":" isfline.jx
      end
    end
 end
if found=0 then
  say "Check not found"
rc=isfcalls('OFF')
exit
     /************************************/
     /* Subroutine to list error messages */
     /************************************/
msgrtn: procedure expose isfmsg isfmsg2.
     /***********************************************/
     /* The isfmsg variable contains a short message */
     /***********************************************/
if isfmsg<>"" then
  Say "isfmsg is:" isfmsg
     /*****************************************************/
     /* The isfmsg2 stem contains additional descriptive */
     /* error messages                                    */
     /*****************************************************/
do ix=1 to isfmsg2.0
  Say "isfmsg2."ix "is:" isfmsg2.ix
end
return
```

Browse check output from the CKH panel

Use ISFEXEC to access the CK panel, then, for a check with owner IBMSDSF, use ISFACT to display the history. From the history, for any instance with a non-zero result (an exception), use ISFACT to browse the check output.

```
/* REXX */
isfcklim = 999   /* set the limit of checks returned to 999 */
rc=isfcalls("on")
Address SDSF "ISFEXEC CK"
 do ix=1 to name.0   /* Loop for all checks */
  if pos("IBMSDSF",owner.ix) > 0 then  /* If desired  check */
    do
      Address SDSF "ISFACT CK PARM(NP L) TOKEN('"token.ix"') (PREFIX",
       " CK_)"
      do jx=1 to ck_name.0
        if ck_result.jx <> 0 then
          do
            Address SDSF "ISFACT CK PARM(NP S) TOKEN('"ck_token.jx"')",
            "(PREFIX CKH_)"
            say "Now processing check" ck_name.jx " Run " ck_count.jx
            do mx = 1 to isfline.0
              say isfline.mx
            end  /* done with history text */
          end
      end
    end
end
rc=isfcalls("off")
```

Print to SYSOUT

Using ISFEXEC, access the ST panel. Then, prior to printing, set SYSOUT-related special variables to control the attributes of the output SYSOUT file (class, copies, dest, and forms). Using ISFACT, issue the XSC action character against the desired row (row 1) to print all data sets represented by that row. XSC prints to SYSOUT and closes the print file after printing.

```
/* REXX */
rc=isfcalls('ON')
     /* Access the ST panel */
Address SDSF "ISFEXEC ST"
lrc=rc
call msgrtn
if lrc<>0 then
  exit 20
     /* Assign the special variables that correspond to */
     /* the attributes of the print file.  Unassigned   */
     /* variables will use defaults.                    */
isfprtclass="U"
isfprtcopies="2"
isfprtdest="ken"
isfprtformdef="ffff"
isfprtforms="8888"
isfprtpagedef="pppp"
isfprtprmode="pmode"
     /* Issue an XSC action against the row to be printed */
do ix=1 to JNAME.0
  if JNAME.ix = "RJONES" then
    do
      Address SDSF "ISFACT ST TOKEN('"TOKEN.ix"') PARM(NP XSC)"
      lrc=rc
      call msgrtn
      if lrc<>0 then
        exit 20
    end
end
exit
     /* Subroutine to list error messages */
msgrtn: procedure expose isfmsg isfmsg2.
     /* The isfmsg variable contains a short message */
if isfmsg<>"" then
  Say "isfmsg is:" isfmsg
     /* The isfmsg2 stem contains additional descriptive */
     /* error messages                                   */
do ix=1 to isfmsg2.0
  Say "isfmsg2."ix "is:" isfmsg2.ix
end
return
```

List action characters

Set the ISFACTIONS special variable to ON, which causes the action characters to be returned in the ISFRESP variables. Then access the ST panel, and list the valid action characters for that panel.

```
/* REXX */
rc=isfcalls('ON')
    /* Set isfactions special variable to */
    /* the equivalent of SET ACTION ON    */
isfactions="ON"
    /* Invoke the ST panel */
Address SDSF "ISFEXEC ST"
if rc<>0 then
  Exit rc
    /* List each of the valid action characters */
    /* for the panel.                           */
```

```
Say "Actions valid on the panel are:"
do ix=1 to isfresp.0
  Say " " isfresp.ix
end
rc=isfcalls('OFF')
```

Issue system commands using ISFSLASH

```
/* REXX */
rc=isfcalls('ON')
mycmd.0=3
mycmd.1="$DSPL"
mycmd.2="$D JOBQ,JM=S*"
mycmd.3="$D I"
Address SDSF ISFSLASH "("mycmd.") (WAIT)"
/* List any error messages */
  Say "isfmsg is:" isfmsg
   Say "isfmsg2.0 is:" isfmsg2.0
   if datatype(isfmsg2.0) = "NUM" then
     do ix=1 to isfmsg2.0
       Say "isfmsg2."ix "is:" isfmsg2.ix
     end
  rc=isfcalls('OFF')
```

Work with the last 24 hours of SYSLOG

Use special variables and the REXX DATE and TIME functions to specify the member to process, the date format, date range, and the limit for the number of records in the stem variable ISFLINE. Then use the ISFLOG command to read the SYSLOG to ISFLINE.

```
/* REXX */
rc=isfcalls('ON')
isfsysid="sy2"              /* Member to process */
isfdate="mmddyyyy /"        /* Date format for special variables */
currday=date("C")
currday=currday-1           /* yesterday */
isflogstartdate=date("U",currday,"C")   /* yesterday in mm/dd/yy */
isflogstarttime=time("N")   /* current time */
isflogstopdate=date("U")    /* current date in mm/dd/yy */
isflogstoptime=time("N")    /* current time */
isflinelim=10000
Address SDSF "ISFLOG READ TYPE(SYSLOG)"
do ix=1 to isfmsg2.0
  say isfmsg2.ix
end
do ix=1 to isfline.0        /* Process the returned variables */
  say isfline.ix
end
rc=isfcalls('OFF')
```

Work with the current day of the system log

Use the ISFLOG command to read the system log for the current day to the ISFLINE stem variable. This example is for the SYSLOG. To work with the OPERLOG, you would specify TYPE(OPERLOG) with the ISFLOG command.

```
/* REXX */
rc=isfcalls('ON')
isflinelim=100000
Address SDSF "ISFLOG READ TYPE(SYSLOG)"
do ix=1 to isfmsg2.0
  say isfmsg2.ix
end
do ix=1 to isfline.0    /* Process the returned variables */
  say isfline.ix
end
rc=isfcalls('OFF')
```

Find a message in the system log

Use the ISFLOG command to read the system log. Use the ISFFIND and ISFSCROLLTYPE special variables to find message $HASP100.

```
/* REXX */

rc=isfcalls('ON')
isfsysid="sy1"              /* Member to process */
isfdate="mmddyyyy /"        /* Date format for special variables */
currday=date("C")
currday=currday-2           /* yesterday */
isflogstartdate=date("U",currday,"C")   /* yesterday in mm/dd/yy */
isflogstarttime=time("N")   /* current time */
isflogstopdate=date("U")    /* current date in mm/dd/yy */
isflogstoptime=time("N")    /* current time */

isffind = '$HASP100'
isffindlim = 9999999
isfscrolltype = 'FINDNEXT'
isflinelim = 1

do until isfnextlinetoken=''
   Address SDSF "ISFLOG READ TYPE(SYSLOG)"

   lrc=rc
   if lrc>4 then
     do
       call msgrtn
       exit 20
     end
   do ix=1 to isfline.0        /* Process the returned variables */
     say isfline.ix
   end

   /*********************************************/
   /* Continue reading SYSLOG where we left off */
   /*********************************************/
   isfstartlinetoken = isfnextlinetoken
end
rc=isfcalls("off")

exit

     /************************************/
     /* Subroutine to list error messages */
     /************************************/
msgrtn: procedure expose isfmsg isfmsg2.

     /*********************************************/
```

```
     /* The isfmsg variable contains a short message */
     /************************************************/
if isfmsg <> "" then
  Say "isfmsg is:" isfmsg

     /************************************************/
     /* The isfmsg2 stem contains additional descriptive */
     /* error messages                                   */
     /************************************************/
do ix=1 to isfmsg2.0
  Say "isfmsg2."ix "is:" isfmsg2.ix
end

return
```

Work with the last 24 hours of OPERLOG

This example shows reading the last 24 hours of OPERLOG. Use special variables and the REXX DATE and TIME functions to specify the member to process, the date format, date range, and the limit for the number of records in the stem variable ISFLINE. Then use the ISFLOG command to read the SYSLOG to ISFLINE. Print a subset of messages which were either highlighted, have descriptor code 12, or colored in red when they were issued.

```
/* REXX */

rc=isfcalls('ON')

isfsysid="sy2"              /* Member to process */
isfdate="mmddyyyy /"        /* Date format for special variables */
currday=date("C")
currday=currday-1           /* yesterday */
isflogstartdate=date("U",currday,"C")   /* yesterday in mm/dd/yy */
isflogstarttime=time("N")   /* current time */
isflogstopdate=date("U")    /* current date in mm/dd/yy */
isflogstoptime=time("N")    /* current time */
isflinelim=1000

do until isfnextlinetoken=''
   Address SDSF "ISFLOG READ TYPE(OPERLOG)"
   do ix=1 to isfmsg2.0
     say isfmsg2.ix
   end
   do ix=1 to isfline.0        /* Process the returned variables */
     desccodematch = 0
     do jx=1 to words(isfdesccode.ix)
        if word(isfdesccode.ix,jx)='12' then desccodematch=1
     end

     if isfhighlight.ix = 'h' |,   /* if hilighted */
        isfcolor.ix = 'r'     |,   /* if red       */
        desccodematch = 1    then
          say isfline.ix

   end
   /********************************************/
   /* Continue reading OPERLOG where we left off */
   /********************************************/
   isfstartlinetoken = isfnextlinetoken
end
rc=isfcalls("off")
```

Issue the WHO command

Issue the WHO command and echo back the response.

```
/* REXX */
rc=isfcalls('ON')
     /* Issue the WHO command */
Address SDSF "ISFEXEC WHO"
     /* The responses are returned in the isfresp stem */
do ix=1 to isfresp.0
  Say "isfresp."ix "is:" isfresp.ix
end
rc=isfcalls('OFF')
exit
```

System REXX and SDSF

If you invoke SDSF's REXX using System REXX, you need to be aware of the following:

- You must set up the ISFJESNAME variable to identify the JES2 subsystem, or the ISFJES3NAME variable to identify the JES3 subsystem.
- You must be authorized to invoke SDSF functions from REXX, as described in "Security and REXX."

For more information on System REXX (SYSREXX), see *z/OS MVS System Commands*.

Security and REXX

Using SDSF function from a REXX exec is protected just as using SDSF interactively is protected, with the same SAF resources and ISFPARMS parameters. Where special REXX variables correspond to SDSF commands, the authorization for those special variables is the same as for the associated command. In some cases, using a special variable when you are not authorized to the associated command will cause the exec to fail and the invocation of SDSF to end.

Determining which group in ISFPARMS a user is assigned to

To control which group in ISFPARMS a user is assigned to, you can use either SAF or ISFPARMS. Using SAF is the recommended approach, as it is more dynamic and allows you to assign users to the same group regardless of the environment from which they invoke SDSF (interactive, batch, REXX or Java).

The WHO command displays the group to which you are assigned.

Using SAF

To determine group membership, SDSF checks the SAF resource GROUP.*group-name*.*server-name* in the SDSF class. This is explained in detail in "Using SAF to control group membership" on page 36.

Using ISFPARMS

You can use parameters in the GROUP statement or ISFGRP macro to determine group membership. These allow you to control membership based on user ID, logon procedure, terminal name, or TSO authority. See "Group membership" on page 36 for more information.

When you use SDSF's REXX support, special values are assigned as follows:

Logon proc name
Set to `REXX`.

TSO authority
Set to JCL authority.

Terminal name
Derived from SAF or TSO based on the current environment.

Diagnosing errors in a REXX exec

To diagnose errors in a REXX exec:

- Examine the contents of the special variables that contain the SDSF messages, ISFMSG and ISFMSG2. ISFMSG2 is a stem variable.
- If the SDSF messages do not provide enough information to resolve the errors, try adding the VERBOSE option to the ISFEXEC and ISFACT host commands, then examining the contents of the ISFMSG2 stem variable. VERBOSE causes diagnostic messages to be added to the ISFMSG2 stem variable. The messages describe each row variable created by SDSF.
- For problems related to security, use the ISFSECTRACE special variable along with the contents of the ISFMSG2 or ISFULOG variables. For more information, refer to "Diagnosing security" on page 192.
- For problems associated with authorization to system commands, see the contents of the ISFULOG special variable, which includes SAF authorization messages. Note that SAF authorization messages will not be preceded by the system command. That is because SDSF checks the SAF resource for the command in advance and does not issue the command if the user is not authorized to it.
- If you need to call IBM for service, prepare documentation by printing the contents of these special variables:
  - ISFMSG and ISFMSG2
  - ISFDIAG. This variable is intended for use by IBM service personnel. It contains internal reason codes associated with a request.

  If IBM requests that you run a trace, include the following special variables in your exec prior to the ISFEXEC or ISFACT commands:

  ```
  isftrace="ON"
  isftrmask="ALL"
  ```

  You must be authorized to the TRACE command to use these variables.

If jobs that you expect to see are missing from a panel, or you are not authorized to function that you expect to be authorized to, the problem may be with the group in ISFPARMS that you are being assigned to. To see if you are being assigned to a different group when you use SDSF REXX than when you use SDSF interactively, issue the WHO command from a REXX exec and from the command line, and compare the values for group index. If you believe you are being assigned to the wrong group, contact your security administrator. Security and SDSF REXX is described in"Security and REXX" on page 448.

End of Programming Interface Information

Chapter 14. Using SDSF with the Java programming language

Programming Interface Information

This topic provides an overview of accessing SDSF function with the Java programming language, and describes how to protect the use of SDSF through Java.

Using SDSF with Java allows you to create Java applications that exploit SDSF function. It provides a more powerful alternative to using SDSF in batch, which is described in Chapter 12, "Using SDSF in batch," on page 369, and complements SDSF's support for REXX, which is described in Chapter 13, "Using SDSF with the REXX programming language," on page 377.

You must be authorized to use SDSF from Java and you must be authorized to the SDSF functions that you invoke from Java.

System programmers should define ISFPARMS group membership to ensure that SDSF users have the proper authorization when invoking SDSF with Java. For more information, see "Security and Java" on page 460.

Where to look for information

The principal source of information for using Java with SDSF is the Javadoc supplied with SDSF. To use the Javadoc:

1. Download the isfjcallDoc.jar file, in binary, to an empty directory on your workstation. By default, this file is installed into /usr/include/java_classes/isfjcallDoc.jar.
2. If you have the Java SDK installed, use this command:

   ```
   jar -xf isfjcallDoc.jar
   ```

   Otherwise, use another utility to unzip the file.
3. Navigate to the index.html file and open it with a Web browser. Once the index.html file is displayed, links allow you to navigate to specific classes or topics, such as:

   Overview
   Display an overview to using SDSF with Java

   Package
   Display a list of classes

   Tree Display a hierarchical view of classes

   Index Display an index to the Javadoc

See the following for further information.

- Using SDSF, including descriptions of panels, action characters, overtypeable columns and commands: refer to SDSF's online help. For a brief introduction, see Chapter 11, "Introduction to SDSF application services," on page 363.
- Columns on SDSF panels: to display a list of columns and other column attributes, use the COLSHELP command. The columns are also described in Chapter 4, "Columns on the SDSF panels," on page 133.

© Copyright IBM Corp. 1997, 2016

Simplifying systems management with SDSF Java

With the SDSF Java API, you can access SDSF panel data and function through a Java program.

Accessing panels and panel data: Each of the panels that you work with when using SDSF interactively (DA, O, PR and so on) has an associated Java interface that describes the returned data and the available methods. Panel data is represented by lists, with each element in a list corresponding to a row on the panel. You access column data within a list element by referencing column values by column name.

Processing system log and issuing commands: You can retrieve records from the system log (SYSLOG) and the sysplex-wide log (OPERLOG), and search for specific messages or events. You can also issue free-form system commands and receive their responses in a manner similar to using the SDSF slash (/) command.

Retrieving job output: You can retrieve records from the output data sets for a job and search for specific messages or return codes.

Taking action: You use methods to perform functions similar to action characters and overtypeable fields, for example, to cancel a job or change the print destination for job output.

Filtering data: For best performance, you should limit the data that a request returns to the minimum that is required. You do this with request settings, which allow you to specify things like:

- Filters of various kinds. The same filters that are available when you use SDSF interactively are available with request settings. They include filters by job name, owner and destination, like the PREFIX, OWNER and DEST commands, or any column, like the FILTER command.
- The list of columns to process. Specify columns by column name.
- Whether to include columns with delayed access. Because gathering the data for these columns can take a significant amount of time, they are not included unless you request them explicitly.

Viewing results: You can access messages and return codes that describe the completion of a request through a results object. SDSF messages and system messages, if any, issued in response to commands are contained in lists, with each element corresponding to a message. Return codes from SDSF functions are available both in the results object and as return codes on most methods.

Controlling access: Standard SDSF authorization checking occurs for all requests and for attempts to modify the row represented by a returned object.

Enabling your application to use SDSF Java

Your application must make the SDSF Java classes and libraries accessible to it. To do this, add the SDSF JAR file to the CLASSPATH and modify your application LIBPATH. The syntax for doing this varies based on how your application is invoked.

CLASSPATH: The SDSF JAR file (**isfjcall.jar**) must be included on the CLASSPATH. The CLASSPATH can be included on the Java command (using the -cp keyword) that invokes your application, or through the CLASSPATH

environment variable. For example, to invoke an application from the z/OS Unix System Services (z/OS Unix) shell, you might have the following statement:

```
export CLASSPATH=/usr/include/java_classes/isfjcall.jar:$CLASSPATH
```

LIBPATH: The LIBPATH references a path containing the SDSF native library. There is one library for 31-bit Java and one for 64-bit Java. You must point to the appropriate library based on the version of Java you are running.

This example assumes SDSF has been installed in the default directories and 31-bit Java is being used:

```
export LIBPATH=/usr/lib/java_runtime:$LIBPATH
```

If you are using 64-bit Java, the LIBPATH would be similar to the following:

```
export LIBPATH=/usr/lib/java_runtime64:$LIBPATH
```

Note that the LIBPATH references a path and not a specific file, whereas the CLASSPATH references a specific JAR file.

JAVA LEVEL: SDSF requires any of the following Java levels or higher:

- IBM 31-bit SDK for z/OS, Java Technology Edition, V6
- IBM 64-bit SDK for z/OS, Java Technology Edition, V6

To access Java, update your PATH environment variable to point to the level of Java you need (either 31-bit or 64-bit). Assuming Java has been installed in the default path, you would use a command similar to the following for 31-bit Java:

```
export PATH=/usr/lpp/java/J6.0/bin:$PATH
```

If you are using 64-bit Java, the PATH would be similar to the following:

```
export PATH=/usr/lpp/java/J6.0_64/bin:$PATH
```

Installation verification

You can use the ISFAbout class to verify that SDSF Java has been configured correctly. It produces a report that includes the service levels of the SDSF Java classes and other information about the runtime environment. A successful run of ISFAbout shows that your classpath and libpath are acceptable to SDSF and that SDSF can be used to retrieve data.

To run ISFAbout, use a command similar to the following:

```
java -cp classpath -jar /usr/include/java_classes/isfjcall.jar
```

Alternatively, you can invoke ISFAbout with this command:

```
java -cp classpath com.ibm.zos.sdsf.core.ISFAbout
```

ISFAbout is controlled through arguments. By default, a report is written to stdout. You can use arguments to write the report to a file. The arguments are as follows:

-f:filename
Names a path to which the report will be written. If this is not specified, the report is written to stdout.

-append
Indicates that the report will be appended to the file. If this is not specified, the file is replaced.

-m:modnames
Names a list of SDSF module names, separated by commas, for which module level information is desired. These names will be provided by IBM service personnel when diagnosing problems.

-help or -?
Requests the usage text to be displayed.

For example, to write a report describing the SDSF Java environment to a file called /tmp/about.txt (replacing it), you could use a command similar to the following:

```
java -cp classpath -jar /usr/include/java_classes/isfjcall.jar -f:/tmp/about.txt
```

Writing a Java application

A basic SDSF Java application might do the following:

1. Create a runner that corresponds to the panel you want to work with. A runner is a Java class that provides access to SDSF and contains a results object describing completion of the request. Runners are described in "Using runners and request settings" on page 456.
2. Create request settings and associate it with the runner to limit the results that are returned. (This is optional but recommended.) Request settings are described in "Using runners and request settings" on page 456.
3. Invoke SDSF to create a list of objects and check the results object for SDSF completion messages.
4. Process the returned object list and obtain column values for each row.
5. Invoke methods on a row object to retrieve additional information or modify the object.

You should always test the return codes from SDSF functions. These are available in the results object and as return codes on most methods. SDSF and system messages describing the completion of a request are also contained in the results object.

Example

The code snippet below requests job-related data from the Status (ST) panel. The settings object is used to restrict the returned data to a subset of jobs with the indicated job name prefix (in this case, all job names) and owner (IBMUSER).

```
// Create optional settings object
ISFRequestSettings settings = new ISFRequestSettings();
settings.addISFPrefix("**");  // Set job name prefix
settings.addISFOwner("ibmuser");  // Set job owner

// Get a runner used to access SDSF ST panel
ISFStatusRunner runner = new ISFStatusRunner(settings);

List<ISFStatus> statObjList = null;

try {
  statObjList = runner.exec();
} catch (ISFException e) {
  // Process exception here
} finally {
  // Print SDSF messages related to request
  results.printMessageList(System.err);

}// List job properties
```

```
if (statObjList != null) {
   for (ISFStatus statObj : statObjList) {
     System.out.println(statObjList.toVerboseString());
   }
}
```

Working with objects

SDSF creates objects which represent rows on the panel being requested. The column values for the row are contained in the object. To limit the size of the object, it is good practice to use the addISFCols setting to request only the columns that are needed.

SDSF action characters are implemented through methods driven on the object. Overtyping columns is implemented through the requestPropertyChange method which allows one or more column values to be changed at the same time.

Obtaining column values

Request column values by column name using the getValue method. The value can be returned as a formatted string or as a byte array for processing by the application.

Column names are different than the column titles that are displayed when you use SDSF interactively. Use the SDSF COLSHELP command to list the column names recognized by the getValue method. Column names are not case sensitive.

Some classes include convenience methods for obtaining common values such as job name. The fixed field (the first column on a panel when you use SDSF interactively) can also be obtained using the getFixedField method.

The following code snippet shows how to obtain column values using a previously created ISFStatus statObj object.

```
// Get job name and owner
String jobname = statObj.getValue("jname");
String owner = statObj.getValue("ownerid");

// Get fixed field (jobname)
String fixedField = statObj.getFixedField();
```

Actions and overtypes

The available methods for an object are defined by the interface for the object. The method names are similar to the descriptions for action characters that you can display with the SET ACTION LONG command when using SDSF interactively.

The following snippet shows how to cancel a job and list the command responses on the console.

```
// Cancel job without a dump
statObj.cancel();

// List the command responses
results.printResponseList(System.out);
```

You can change column values, in a manner similar to overtyping a column, with the requestPropertyChange method. This method takes an array of column names to change and a corresponding array of values with the new value for each column. The following code snippet shows how to change the class of a job to class A.

```
// Build column name array
String propName = { "jclass" };

// Build column value array
String propValue = { "a" };

// Change the job class
statObj.requestPropertyChange(propName, propValue);

// Print response list
results.printResponseList(System.out);
```

See “Samples” on page 458 for more examples of working with objects.

Browsing data

To browse job output from the job-related panels (DA, H and so on) you can:

- Use an external utility. With this approach, you first allocate the output data sets with the browseAllocate method.
- Use SDSF's browse. With this approach, you use the browse or browseJCL methods.

You can also browse the output of a check on the CK panel, or the system log on the SYSLOG or OPERLOG panels.

SDSF provides a variety of samples for browsing and searching data. Refer to “Samples” on page 458.

Using runners and request settings

A runner is a Java class that provides access to SDSF in a means similar to using SDSF commands to access panels. To access SDSF, you create an instance of a runner for the desired panel and then use methods in the runner class to obtain the requested data. For functions that are not panel-related, such as issuing system commands, you use a special runner.

You can optionally provide request settings that are associated with the runner. You create an instance of the ISFRequestRunner class and add the desired settings to it. The settings correspond to SDSF settings such as job name prefix, job owner, and destination name filters. In addition, you can provide sort criteria for the returned data, as well as more complex filtering using all the capabilities of the SDSF FILTER command.

The request settings object contains all possible SDSF settings, although not all of them apply to the request being processed. SDSF ignores settings that are not appropriate for the function being performed, so you do not need to remove them.

The runner provides a constructor that is used to associate the request settings with the runner. However, you can always associate a settings object after the runner is created. Note that the settings take effect the next time SDSF is invoked. You can also remove settings after the runner is created, in which case SDSF uses the default settings when processing the request.

You can use the same runner for the duration of your application and modify the request settings between each request. Note that when invoking methods on previously obtained objects (for example, invoking the cancel method on a job) SDSF uses the request settings to verify that the object still exists. As a result, use

caution when changing the request settings after a row object has been obtained since the new settings may prevent SDSF from re-deriving the object.

After a request has been processed, the runner contains a reference to the ISFRequestResults object that describes the completion of the request. This object contains SDSF messages, system responses or return codes that were generated by SDSF. You should check the return codes to ensure your request has been processed successfully.

Determining which runner to use

You select the runner based on what rows, columns or other SDSF capabilities your application needs. For example, if you need information about active jobs, you would use the ISFActiveRunner because it provides access to the SDSF DA panel.

Similarly, if you need to enter MVS system commands, you would use the ISFRunner class because it enables use of the SDSF slash command.

The relationship between the SDSF panel commands and the runners is shown in the table belowTable 170. Use this chart to determine the runner to create based on the data that is required.

Table 170. SDSF Commands and Runners

Panel or Command	Runner	Description
APF	ISFApfRunner	APF data sets
AS	ISFAsmRunner	Address space memory
CK	ISFHealthCheckRunner	Checks for IBM Health Checker for z/OS
DA	ISFActiveRunner	Active jobs
DYNX	ISFDynxRunner	Dynamic exits
ENC	ISFEnclaveRunner	WLM enclaves
ENQ	ISFEnqueueRunner	Enqueues
H	ISFHeldOutputRunner	Output groups for jobs on held queues
I	ISFInputRunner	Jobs on the input queue or executing
INIT	ISFInitiatorRunner	JES and WLM initiators
JC	ISFJobClassRunner	JES job classes
J0	ISFJob0Runner	JES3 Job 0
LI	ISFLineRunner	JES lines
LNK	ISFLnkLstRunner	Link list data sets
LPA	ISFLpaRunner	Link pack area data sets
MAS / JP	ISFJESPlexRunner	Members of a JES2 MAS or JES3 JESPLEX
NC	ISFNetworkConnectionRunner	JES network connections
NO	ISFNodeRunner	JES nodes
NS	ISFNetworkServerRunner	JES network servers
O	ISFOutputRunner	Output groups for jobs on nonheld queues
PAG	ISFPageRunner	Page data sets
PARM	ISFParmlibRunner	PARMLIB data sets
PR	ISFPrinterRunner	JES printers

Table 170. SDSF Commands and Runners (continued)

Panel or Command	Runner	Description
PS	ISFProcessRunner	z/OS Unix processes
PUN	ISFPunchRunner	JES punches
QUERY	ISFRunner	QUERY command
RDR	ISFReaderRunner	JES readers
RES	ISFWLMResourceRunner	WLM resources
RM	ISFResourceMonitorRunner	JES resources
SE	ISFSchedulingEnvironmentRunner	WLM scheduling environments
SO	ISFSpoolOffloadRuner	JES spool offloaders
SP	ISFSpoolRunner	JES spool volumes
SR	ISFSystemRequestRunner	z/OS system requests
ST	ISFStatusRunner	Jobs on any queue
SYM	ISFSystemSymbolRunner	System symbols
SYS	ISFSystemRunner	System information
WHO	ISFRunner	WHO command (user and environment)
/	ISFRunner	Slash command (system commands)

Samples

SDSF provides several sample classes to show how to use SDSF Java. The samples are installed by default under the /usr/lpp/sdsf/java/samples path. The available samples are:

Sample	Class Name	Description
Get list of jobs	ISFGetJobsSample	Access the ST panel and display the properties of selected jobs
Change job priority	ISFChangeJobPrioritySample	Change the priority of jobs
Browse a check	ISFBrowseHealthCheckSample	Browse a check for IBM Health Checker for z/OS
Browse a job data set	ISFBrowseJobDataSetSample	Browse a selected job data set
Browse job output	ISFBrowseStatusJobSample	Browse a job's output
	ISFBrowseSample	Allocate the spool data sets for a job and browse them
Browse and search the system log	ISFSearchSyslogSample	Read the last day of SYSLOG and search for one or more strings
	ISFSearchSyslogSample2	Browse and search the SYSLOG, specifying the lines
	ISFSearchOperlogSample	Browse the OPERLOG
Browse	ISFLineResultsSample	Browse, use methods in ISFLineResults
Issue MVS commands	ISFSlashCommandSample	Issue one or more system commands
Issue WHO command	ISFWhoCommandSample	Issue the SDSF WHO command to obtain user attributes
List exception health checks and their output	ISFHealthCheckSample	Find all exception health checks and list the check output

Running the samples

Invoke samples using the main method. See the class descriptions in the Javadoc for any arguments that are needed. Compiled versions of the classes are available in the SDSF JAR file (**isfjcall.jar**) so you invoke the samples by adding the JAR file to your classpath.

Troubleshooting

Check the list below for help if you encounter a problem using the SDSF Java API.

Problem	Solution
Not all columns returned for an object	Some columns are classified as "delayed" access, which means the data can be expensive to gather. These columns are not returned unless the **delayed** option is added to the request settings. Use the SDSF COLSHELP command to determine which columns are delayed.
Objects not returned	Be sure the request settings reflect the correct prefix and owner for a job. SDSF uses these settings when determining which objects to return.
Object not found or row token invalid	When you invoke a method on an object, such as cancel, the object must be valid. A job may be invalid, for example, if it has been purged and thus cannot be found. Examine the SDSF messages to determine why the request failed.
Too many objects returned	It is possible to generate requests that return an excessive number of objects. This may result in failures related to insufficient storage, or performance problems. Be sure to refine the request settings to return the fewest number of objects needed to satisfy a request. You should also limit the number of column values returned for each object.
Object no longer valid	A returned object contains a row token that SDSF uses to find the object on subsequent requests. The format of the token may vary across SDSF releases or maintenance levels. Therefore, it is expected that the object will be used on the same level of SDSF that gathered it.
Request failed with a non-zero return code	Be sure to examine the SDSF messages that describe any errors found by SDSF. To do this, use the getRunner().getRequestResults().getMessageList() method.
SDSF Java classes not found	The SDSF Java classes are packaged in a JAR file that by default is installed in /usr/include/java_classes/isfjcall.jar. Be sure this JAR file is in your application CLASSPATH.
Unsatisfied link error	The SDSF Java classes require that the SDSF DLL is included in your application LIBPATH. There are two versions of the DLL, based on whether your are running the 31-bit or 64-bit version of Java. By default, the DLLs are installed in /usr/lib/java_runtime (for 31-bit Java), and /usr/lib/java_runtime64 (for 64-bit Java).

Problem	Solution
Unable to modify an object property	You may not be authorized to modify the property. Even though you may be able to overtype the column interactively, the modify fails using SDSF Java. Verify that you are in the expected SDSF group. Use the who method of ISFRunner. Note that unless you are using SAF for security, your authority level may be different when using SDSF Java than when running interactively.
Method return code 16 (not authorized to SDSF)	Verify your authorization to use SDSF. Message ISF024I may have been issued to the system console.

Tracing

If you need to report a problem to IBM, the SDSF Java classes can produce trace records using the facilities of the java.util.logging package. To enable tracing you must modify your logging.properties file or point to your own copy of the file when invoking your SDSF Java application.

If you are using file-based logging, you can add the following statement to your logging.properties file to enable SDSF Java tracing:

```
com.ibm.zos.sdsf.level = ALL
```

You can reference your modified logging.properties file using the following system property when invoking your application:

```
-Djava.util.logging.config.file=logging.properties
```

In addition, IBM service personnel may request that an SDSF trace be obtained. This causes the SDSF host code to create trace records that can be used to diagnose problems. You can enable trace by using the addISFTrace method in the ISFRequestSettings class or by using the following system property when invoking your application:

```
-Dcom.ibm.zos.sdsf.core.ISFRequestSettings.sdsfTrace=true
```

SDSF trace records are recorded to a SYSOUT file associated with the process that is running your application. The ddname for the sysout file is named ISFTRACE.

Security and Java

Using SDSF function from a Java program is protected just as using SDSF interactively, or from a REXX exec, is protected, with the same SAF resources and ISFPARMS parameters. For example, when a Java method corresponds to an SDSF action character, the authorization for that method is the same as for the action character. See “Protecting runners” on page 461 and “Protecting methods” on page 461 for more information.

Determining which group in ISFPARMS a user is assigned to

To control which group in ISFPARMS a user is assigned to, you can use either SAF or ISFPARMS. Using SAF is the recommended approach, as it is more dynamic and allows you to assign users to the same group regardless of the environment from which they invoke SDSF (interactive, batch, REXX or Java).

The WHO command displays the group to which you are assigned.

Using SAF

To determine group membership, SDSF checks the SAF resource GROUP.*group-name.server-name* in the SDSF class. This is explained in detail in "Using SAF to control group membership" on page 36.

Using ISFPARMS

You can use parameters in the GROUP statement or ISFGRP macro to determine group membership. These allow you to control membership based on user ID, logon procedure, terminal name, or TSO authority. See "Group membership" on page 36 for more information.

When you use SDSF's Java support, this special value is assigned:

Logon proc name
Set to `EXTERNAL`.

Protecting runners

You protect the runners in the same way that you protect the associated SDSF commands. For a discussion of how the runners relate to SDSF commands, see Table 170 on page 457. For information on protecting the runners if you are using SAF for security, see "Authorized SDSF commands" on page 235. If you are using ISFPARMS for security, see AUTH parameter.

Protecting methods

You protect the Java methods in the same way that you protect the corresponding action characters and overtypeable fields. The relationship of methods in each class to action characters is described in the topics that follow. For information about the SAF resources that you use to protect action characters, see "Action characters" on page 211. For information about the SAF resources that you use to protect overtyping fields with the requestPropertyChange method, see "Overtypeable fields" on page 255. If you are using ISFPARMS for security, see "Action characters and overtypeable fields for each command level" on page 73.

ISFActive (DA panel)

Table 171. ISFActive Methods for Action Characters

Method	Action Character	Description
browse	S	Browse
browseAllocate	SA	Allocate spool data sets
browseJCL	SJ	Browse JCL
cancel	C, CA, CD, CDA	Cancel a job without a dump
cancelPrint	CP, CDP	Cancel a job and delete all held data sets (JES3 only)
display	D, DL	Display job information in the log
displayDDNames	DSD	Display job information in the log with DD names of all spool data sets that contain data (JES3 only)
displayEstimates	DE	Display job information in the log with line, page, record, and card counts (JES3 only)
displayExtended	DX	Display job information in the log with extended information (JES3 only)
displaySpoolHold	DSH	Display job information in the log with DD names of spool data sets in spool hold status that contain data (JES3 only)

Table 171. ISFActive Methods for Action Characters (continued)

Method	Action Character	Description
displaySpoolPartition	DSP	Display job information in the log with the spool partition name (JES3 only)
hold	H	Hold a job
list	L, LL	List the output status of the job in the log
listBDT	LB	List q=bdt output status of the job in the log (JES3 only)
listHold	LH	List q=hold output status of the job in the log (JES3 only)
listTCP	LT	List q=tcp output status of the job in the log (JES3 only)
print	XS, XSC	Print a job to SYSOUT
printDataset	XD, XDC	Print a job to a data set
printFile	XF, XFC	Print a job to a file
purge	P, PP	Purge a job
quiesce	RQ	Quiesce a job
release	A	Release a job
restart	E, EC	Restart a job
restartStep	ES	Restart a job after the current step completes (JES2 only)
restartStepHold	ESH	Restart and hold the job the current step completes (JES2 only)
resume	R	Resume a job
spin	W	Spin a job
sysCancel	K, KD	Cancel a job using the system CANCEL command
sysForce	Z	Cancel a job using the system FORCE command
sysStop	Y	Stop a job using the system STOP command (RMF environment only)

ISFApf (APF panel)

Table 172. ISFApf Methods for Action Characters

Method	Action Character	Description
display	D	Display the data sets in the APF list
displayAll	DA	Display the data sets in the APF list

ISFDynx (DYNX panel)

Table 173. ISFDynx Methods for Action Characters

Method	Action Character	Description
display	D	Display a dynamic exit
displayAll	DA	Display all dynamic exits
displayAllImp	DAI	Display all implicitly defined exits
displayDiag	DD	Display dynamic exit with diagnostic information
displayInstallation	DI	Display exits defined with type installation

Table 173. ISFDynx Methods for Action Characters (continued)

Method	Action Character	Description
displayNotProgram	DNP	Display exits not defined with type program
displayProgram	DP	Display exits defined with type program

ISFEnclave (ENC panel)

Table 174. ISFEnclave Methods for Action Characters

Method	Action Character	Description
quiesce	RQ	Quiesce an enclave
resume	R	Resume an enclave

ISFENQ (ENQ panel)

Table 175. ISFENQ Methods for Action Characters

Method	Action Character	Description
display	D	Display enqueue information

ISFHealthCheck (CK panel)

Table 176. ISFHealthCheck Methods for Action Characters

Method	Action Character	Description
activate	A	Activate a check
browse	S	Browse the check message buffer
deactivate	H	Deactivate a check
delete	P, PF	Delete a check
display	D, DL	Display a check
displayDiag	DD	Display a check with diagnostics
displayPolicies	DP, DPO	Display check policies
displayStatus	DS	Display check status
list	L	List history
print	XS, XSC	Print a check to SYSOUT
printDataset	XD, XDC	Print a check to a data set
printFile	XF, XFC	Print a check to a file
refresh	E	Refresh a check
removeCategories	U	Remove all categories for a check
run	R	Run a check

ISFHealthCheckArchive (CKH panel)

Table 177. ISFHealthCheckArchive Methods for Action Characters

Method	Action Character	Description
browse	S	Browse a check message buffer
print	XS, XSC	Print a check to SYSOUT
printDataset	XD, XDC	Print a check to a data set

Table 177. ISFHealthCheckArchive Methods for Action Characters (continued)

Method	Action Character	Description
printFile	XF, XFC	Print a check to a file

ISFHeldOutput (H panel)

Table 178. ISFHeldOutput Methods for Action Characters

Method	Action Character	Description
browse	S	Browse
browseAllocate	SA	Allocate spool data sets
browseJCL	SJ	Browse JCL
cancel	C	Cancel an output group
hold	H	Hold an output group
list	L, LL	List an output group to the log
outputRelease	O, OK	Output release an output group
print	XS, XSC	Print to SYSOUT
printDataset	XD, XDC	Print to a data set
printFile	XF, XFC	Print to a file
purge	P	Purge output
release	A	Release an output group

ISFInitiator (INIT panel)

Table 179. ISFInitiator Methods for Action Characters

Method	Action Character	Description
display	D, DL	Display initiator information in the log
halt	Z	Halt an initiator
start	S	Start an initiator
stop	P	Stop an initiator

ISFInput (I panel)

Table 180. ISFInput Methods for Action Characters

Method	Action Characters	Description
browse	S	Browse
browseAllocate	SA	Allocate spool data sets
browseJCL	SJ	Browse JCL
cancel	C, CA, CD, CDA	Cancel a job
cancelPrint	CP, CDP	Cancel a job with print (JES3 only)
display	D, DL	Display job properties in the log
displayDDNames	DSD	Display DD names of spool data sets (JES3 only)
displayEstimates	DE	Display estimated lines, pages and records for a job (JES3 only)
displayExtended	DX	Display extended information for a job, such as scheduling environment and service class

Table 180. ISFInput Methods for Action Characters (continued)

Method	Action Characters	Description
displayMains	DM	Display a list of mains on which the job is eligible to run
displayMDSAlloc	DMA	Display the MDS allocation queue (JES3 only)
displayMDSError	DME	Display the MDS error queue (JES3 only)
displayMDSRestart	DMR	Display the MDS restart queue (JES3 only)
displayMDSSysSel	DMSS	Display the MDS system select queue (JES3 only)
displayMDSSysVer	DMSV	Display the MDS system verify queue (JES3 only)
displaySpoolHold	DSH	Display DD names of spool data sets in spool hold status (JES3 only)
displaySpoolPartition	DSP	Display the spool partition assigned for a job (JES3 only)
displayUnavailVol	DMU	Display unavailable volumes (JES3 only)
hold	H	Hold a job
list	L, LL	List a job
listBDT	LB	List output on the BDT queue (JES3 only)
listHold	LH	List output on the hold queue (JES3 only)
listTCP	LT	List output on the TCP queue (JES3 only)
print	XS, XSC	Print a job to SYSOUT
printDataset	XD, XDC	Print a job to a data set
printFile	XF, XFC	Print a job to a file
purge	P, PP	Purge a job
release	A	Release a job
restart	E, EC	Restart a job
restartStep	ES	Restart a job after current step completes (JES2 only)
restartStepHold	ESH	Restart and hold the job after the current step completes (JES2 only)
spin	W	Spin job and message logs
start	J	Start a job

ISFJESplex (MAS and JP panels)

Table 181. ISFJESplex Methods for Action Characters

Method	Action Character	Description
display	D, DL	Display a member in the log
flush	F	Flush jobs currently running on the main (JES3 only)
monitor	J	Displays the current status of JES2 monitor subtasks
monitorDetails	JD	Display JES monitor details in the log (JES2 only)
monitorHistory	JH	Display JES2 resource history in the log
monitorStart	SM	Start the JES monitor (JES3 only)
monitorState	JJ	Display the JES2 state in the log
monitorStatus	JS	Display the current JES status in the log
monitorStop	ZM	Stop the JES monitor

Table 181. ISFJESplex Methods for Action Characters (continued)

Method	Action Character	Description
reset	ER	Reset a member (JES2 only)
restart	E	Restart a member (JES2 only)
start	S	Start a member
startScheduling	SX	Start scheduling jobs for the member
stop	P	Stop a member
stopAbend	PA	Stop a member by abending it (JES2 only)
stopQuick	PQ	Stop a member, ignoring cross system activity (JES2 only)
stopScheduling	PX	Stop scheduling jobs for the member (JES2 only)
stopTerminate	PT	Stop the member, ignoring active programs (JES2 only)
varyOffline	VF	Vary a member offline and stop scheduling jobs (JES3 only)
varyOnline	V	Vary a member online and start scheduling jobs (JES3 only)

ISFJobClass (JC panel)

Table 182. ISFJobClass Methods for Action Characters

Method	Action Character	Description
display	D	Display a job class in the log
displayClass	DC	Display the status of a job class in the log (JES3 only)
displayGroup	DG	Display the status of a group in the log (JES3 only)

ISFJobDataSet (JDS panel)

Table 183. ISFJobDataSet Methods for Action Characters

Method	Action Character	Description
browse	S	Browse
browseAllocate	SA	Allocate spool data sets
browseJCL	SJ	Browse JCL
cancel	C	Cancel a data set
hold	H	Hold a data set
print	XS, XSC	Print a data set to SYSOUT
printDataset	XD, XDC	Print a data set to a data set
printFile	XF, XFC	Print a data set to a file
purge	P	Purge a data set
release	O	Release a data set
spin	W	Spin a data set

ISFJob0 (J0 panel)

Table 184. ISFJob0 Methods for Action Characters

Method	Action Character	Description
browseAllocate	SA	Allocate spool data sets
cancel	C	Cancel a data set
display	D	Display a data set
hold	H	Hold a data set
print	XS, XSC	Print a data set to SYSOUT
printDataset	XD, XDC	Print a data set to a data set
printFile	XF, XFC	Print a data set to a file
purge	P	Purge a data set
release	O	Release a data set

ISFLine (LI panel)

Table 185. ISFLine Methods for Action Characters

Method	Action Character	Description
cancel	C	Cancel a transmitter or receiver
display	D (all forms)	Display a line, transmitter or receiver in the log
fail	L (all forms)	Fail a line (JES3 only)
interrupt	I	Interrupt a line
quiesce	Q	Quiesce a line
restart	E	Restart a line, transmitter or receiver
start	S (all forms except SN)	Start a line, transmitter or receiver
startNetworking	SN	Start communication on a line (JES2 only)
stop	P	Stop a line, transmitter or receiver
vary	V (all forms)	Vary a line online or offline (JES3 only)

ISFLnkLst (LNK panel)

Table 186. ISFLnkLst Methods for Action Characters

Method	Action Character	Description
display	D	Display the data sets in the LnkLst
displayNames	DN	Display the data set names in the LnkLst

ISFNetworkConnection (NC panel)

Table 187. ISFNetworkConnection Methods for Action Characters

Method	Action Character	Description
display	D (all forms)	Display a network connection in the log
restart	E	Restart a device (JES2 only)
start	S	Start a transmitter or receiver (JES2 only)
startNetworking	SN	Start network communication
stop	P	Stop a transmitter or receiver (JES2 only)

ISFNetworkServer (NS panel)

Table 188. ISFNetworkServer Methods for Action Characters

Method	Action Character	Description
callTCP	X	Call the network server DSP (JES3 only)
cancel	C	Cancel a network server (JES3 only)
display	D and DL	Display a network server in the log
displayAppl	DA	Display a application (JES2 only)
displaySocket	DS	Display a socket (JES2 only)
fail	L and LD	Fail a device (JES3 only)
restart	E	Restart a device
start	S	Start a device (JES2 only)
stop	P	Stop a device (JES2 only)
sysCancel	K and KD	Cancel a network server address space
sysForce	Z	Force a network server address space
sysStop	Y	Stop the network server address space

ISFNode (NO panel)

Table 189. ISFNode Methods for Action Characters

Method	Action Character	Description
display	D	Display information about a node in the log
displayConnections	DC	Display information about node connections in the log (JES2 only)
displayPaths	DP	Display information about paths in the log (JES2 only)
startNetworking	SN	Start node communication on a line (JES2 only)

ISFOutput (O panel)

Table 190. ISFOutput Methods for Action Characters

Method	Action Character	Description
browse	S	Browse
browseAllocate	SA	Allocate spool data sets
browseJCL	SJ	Browse JCL
cancel	C	Cancel an output group
hold	H	Hold an output group
list	L, LL	List an output group to the log
print	XS, XSC	Print an output group to SYSOUT
printDataset	XD, XDC	Print an output group to a data set
printFile	XF, XFC	Print an output group to a file
purge	P	Purge output
release	A	Release an output group

ISFPage (PAG panel)

Table 191. ISFPage Methods for Action Characters

Method	Action Character	Description
display	D	Display the page data sets
displayCommon	DC	Display common page data sets
displayPageDel	DD	Display page deletes
displayLocal	DL	Display local page data sets
displayPLPA	DP	Display PLPA page data sets
displaySCM	DS	Display storage class memory

ISFParmlib (PARM panel)

Table 192. ISFParmlib Methods for Action Characters

Method	Action Character	Description
display	D	Display the parmlib data sets
displayErrors	DE	Display errors

ISFPrinter (PR panel)

Table 193. ISFPrinter Methods for Action Characters

Method	Action Character	Description
backSpace	B (all forms)	Backspace a printer
call	X	Call a writer (JES3 only)
cancel	C (all forms)	Cancel a job on the printer or writer
display	D, DL	Display information about the printer in the log
fail	L, LD	Fail a writer (JES3 only)
forceFSS	K	Force termination of the FSS
forwardSpace	F (all forms)	Forward space a printer
halt	Z	Halt a printer
interrupt	I	Interrupt a printer
repeat	N	Repeat a printer
restart	E	Restart a printer or writer
start	S	Start a printer or writer
stop	P	Stop a printer
vary	V, VF	Vary a writer (JES3 only)

ISFProcess (PS panel)

Table 194. ISFProcess Methods for Action Characters

Method	Action Character	Description
cancel	C	Cancel a process
(display) ()	D	Display a process in the log
kill	K	Kill a process
terminate	T	Terminate a process

ISFPunch (PUN panel)

Table 195. ISFPunch Methods for Action Characters

Method	Action Character	Description
backSpace	B (all forms)	Backspace a punch
call	X (all forms)	Call a punch (JES3 only)
cancel	C (all forms)	Cancel a job on the punch
display	D, DL	Display information about the punch in the log
fail	L (all forms)	Fail the punch (JES3 only)
forwardSpace	F (all forms)	Forward space a punch
halt	Z	Halt a punch (JES2 only)
interrupt	I	Interrupt a punch (JES2 only)
repeat	N	Repeat a punch (JES2 only)
restart	E (all forms)	Restart a punch
start	S (all forms)	Start a punch
stop	P	Stop a punch (JES2 only)
vary	V (all forms)	Vary a punch online or offline (JES3 only)

ISFReader (RDR panel)

Table 196. ISFReader Methods for Action Characters

Method	Action Character	Description
call	X (all forms)	Invoke a reader (JES3 only)
cancel	C (all forms?)	Cancel a job on the reader
display	D, DL	Display information about the reader in the log
fail	L (all forms)	Fail a reader (JES3 only)
halt	Z	Halt a reader (JES2 only)
start	S (all forms)	Start a reader
stop	P	Stop a reader (JES2 only)
vary	V (all forms)	Vary a reader online or offline (JES3 only)

ISFRequestSettings

Some methods in the ISFRequestSettings class correspond to SDSF commands that require authorization. For more information, see "Authorized SDSF commands" on page 235 (SAF) or AUTH parameter (ISFPARMS).

Table 197. ISFRequestSettings Methods for Commands that Require Authorization

Method	Command	Description
addISFDest	DEST	Filter by destination
addISFJESName	JESNAME parameter on SDSF command	Set the JES2 subsystem name to be processed
addISFJES3Name	JES3NAME parameter on SDSF command	Set the JES3 subsystem name to be processed
addISFOwner	OWNER	Filter by job owner
addISFPrefix	PREFIX	Filter by job name
addISFServer	SERVER parameter on SDSF command	Set the SDSF server name to be used

Table 197. ISFRequestSettings Methods for Commands that Require Authorization (continued)

Method	Command	Description
addISFSysId	SYSID	Set the system ID used to select the system log
addISFSysName	SYSNAME	Set the system name pattern to process
addISFTrace	TRACE	Set the SDSF trace mask option

ISFResourceMonitor (RM panel)

Table 198. ISFResourceMonitor Methods for Action Characters

Method	Action Character	Description
display	D	Display information about the resource in the log

ISFSchedulingEnvironment (SE panel)

Table 199. ISFSchedulingEnvironment Methods for Action Characters

Method	Action Character	Description
display	D	Display information about the scheduling environment in the log

ISFSpool (SP panel)

Table 200. ISFSpool Methods for Action Characters

Method	Action Character	Description
display	D, DL	Display a spool volume or partition
halt	Z	Halt a spool volume, deallocating it after active work completes its current phase of processing
hold	H	Hold a spool data set and hold further scheduling for jobs with data on the data set (JES3 only)
holdCancel	HC	Hold a spool data set and cancel all jobs using the data set (JES3 only)
holdStop	HP	Hold a spool data set and hold further scheduling for jobs with data on theh data set
jobqueue	J	Display information about all jobs using the spool volume in the log
purge	P, PC	Drain a spool volume
release	A	Release a spool data set and all jobs that have data on spool for scheduling (JES3 only)
start	S	Start a spool volume, adding or reactivating it to the spool configuration
use	U	Resume allocating space on the spool data set (JES3 only)

ISFSpoolOffload (SO panel)

Table 201. ISFSpoolOffload Methods for Action Characters

Method	Action Character	Description
cancel	C	Cancel a transmitter or receiver

Table 201. ISFSpoolOffload Methods for Action Characters (continued)

Method	Action Character	Description
display	D	Display an offloader, transmitter or receiver in the log
restart	E	Restart a transmitter
start	S	Start a transmitter or receiver
startReceive	SR	Start an offloader to receive jobs or SYSOUT
startTransmit	ST	Start an offloader to transmit jobs or SYSOUT
stop	P	Drain an offloader, transmitter or receiver in the log

ISFStatus (ST panel)

Table 202. ISFStatus Methods for Action Characters

Method	Action Characters	Description
browse	S	Browse
browseAllocate	SA	Allocate spool data sets
browseJCL	SJ	Browse JCL
cancel	C, CA, CD, CDA	Cancel a job
cancelPrint	CP, CDP	Cancel a job with print (JES3 only)
display	D, DL	Display job properties in the log
displayDDNames	DSD	Display DD names of spool data sets (JES3 only)
displayEstimates	DE	Display estimated lines, pages and records for a job (JES3 only)
displayExtended	DX	Display extended information for a job, such as scheduling environment and service class
displayMains	DM	Display a list of mains on which the job is eligible to run
displayMDSAlloc	DMA	Display the MDS allocation queue (JES3 only)
displayMDSError	DME	Display the MDS error queue (JES3 only)
displayMDSRestart	DMR	Display the MDS restart queue (JES3 only)
displayMDSSysSel	DMSS	Display the MDS system select queue (JES3 only)
displayMDSSysVer	DMSV	Display the MDS system verify queue (JES3 only)
displaySpoolHold	DSH	Display DD names of spool data sets in spool hold status (JES3 only)
displaySpoolPartition	DSP	Display spool partition assigned for the job (JES3 only)
displayUnavailVol	DMU	Display unavailable volumes (JES3 only)
hold	H	Hold a job
list	L, LL	List a job
listBDT	LB	List output on the BDT queue (JES3 only)
listHold	LH	List output on the hold queue (JES3 only)
listTCP	LT	List output on the TCP queue (JES3 only)
outputRelease	O	Release held output for printing
print	XS, XSC	Print a job to SYSOUT
printDataset	XD, XDC	Print a job to a data set
printFile	XF, XFC	Print a job to a file
purge	P, PP	Purge a job

Table 202. ISFStatus Methods for Action Characters (continued)

Method	Action Characters	Description
purgeOutput	PO	Purge output for a job (JES2 only)
release	A	Release a job
restart	E, EC	Restart a job
restartStep	ES	Restart a job after current step completes (JES2 only)
restartStepHold	ESH	Restart and hold the job the current step completes (JES2 only)
spin	W	Spin job and message logs
start	J	Start a job

ISFSystem (SYS panel)

Table 203. ISFSystem Methods for Action Characters

Method	Action Character	Description
display	D	Display IPL information
displayAll	DAA	Display all address spaces
displayAlloc	DALO	Display allocation options
displayConsoles	DC	Display consoles
displayList	DAL	Display address space list
displayLE	DCEE	Display language environment options
displayDumps	DD	Display dump information
displayEMCS	DEM	Display EMCS consoles
displayGRS	DG	Display GRS information
displayIOS	DI	Display IOS information
displayIQP	DIQP	Display IQP options
displayLLA	DLL	Display LLA information
displayLogger	DLO	Display system logger information
displayConfig	DM	Display configuration information
displayLogrec	DLR	Display LOGREC information
displayMPF	DMP	Display MPF information
displayOMVS	DO	Display OMVS options
displayPCIEDev	DPCD	Display PCIE device information
displayPCIE	DPCI	Display PCIE options
displayProd	DP	Display product registration
displaySMF	DSF	Display SMF information
displaySlip	DSL	Display Slip information
displaySMS	DSM	Display SMS information
displaySymbols	DSY	Display symbol information
displayTime	DT	Display time information
displayTrace	DTR	Display trace information
displayTSOptions	DTO	Display TSO options
displayTSUsers	DTS	Display TSO address spaces

Table 203. ISFSystem Methods for Action Characters (continued)

Method	Action Character	Description
displayWLM	DW	Display WLM information
displaySysplex	DX	Display sysplex information

ISFSystemSymbol (SYM panel)

Table 204. ISFSystemSymbol Methods for Action Characters

Method	Action Character	Description
display	D	Display symbol information

ISFSystemRequest (SR panel)

Table 205. ISFSystemRequest Methods for Action Characters

Method	Action Character	Description
autoReplyIgnore	AI	Ignore auto reply text
display	D	Display a message in the log
remove	C	Remove an action message
reply	R	Reply to a message

ISFWLMResource (RES panel)

Table 206. ISFWLMResource Methods for Action Characters

Method	Action Character	Description
display	D	Display information about the resource in the log

End of Programming Interface Information

Chapter 15. SDSF messages and codes

This topic explains the messages and abend codes that SDSF issues to the terminal or console.

Displaying message help

There is a help panel for each SDSF message. To display the help for a message you can:

- Use the SEARCH command, for example SEARCH ISF024I. Use the SEARCH command when running SDSF under ISPF. Type it on an SDSF command line, not on the command line of a help panel.
- Select the option for message help from a help menu. Most SDSF help menus include an option for message help.

You can also search online documents, using:

- LookAt, for messages with message numbers. LookAt is on the Web at http://www.ibm.com/systems/z/os/zos/bkserv/lookat/.
- The BOOK command. When the cursor is in the SDSF message area, BOOK uses the message text as a search string. See the online help for more information.

User authorization

You might see a message that you are not authorized to perform a certain task. If you should be authorized, do the following:

1. Issue the WHO command. This displays your user ID, TSO logon procedure name, terminal ID, group index, and group name of the authorization group you have been assigned to based on ISFGRP macros or GROUP statements in ISFPARMS. (The index indicates the group by a count of groups. For example, an index of 3 indicates the group defined by the third GROUP statement in ISFPARMS.)
2. Check or ask the system programmer to check your authorization group against the ISFGRP, ISFNTBL, and ISFFLD macros in ISFPARMS. The macros are described in Chapter 2, "Using ISFPARMS for customization and security," on page 15.
3. If the programmer has used the System Authorization Facility (SAF) for security authorization and has activated the resource class to perform the required checking, SDSF ignores ISFPARMS information.
4. If SAF rejects the security check, do the following:
   a. Issue the TSO command `PROFILE WTPMSG`.
   b. Try the SDSF request that failed.
   c. Note the text of the ICH408I message that appears. This message identifies the profile (by name and class) that caused the authorization failure. Report the complete text of this message when asking for authorization.

SDSF messages

This section explains the SDSF messages. The messages are in alphabetic order.

© Copyright IBM Corp. 1997, 2016

Write-to-operator messages appear at the bottom of the log panels. For information on those messages, see “Messages with ISF message numbers” on page 508.

Messages issued in response to SDSF's checks for IBM Health Checker for z/OS are described in “Messages for IBM Health Checker for z/OS” on page 547.

The entry for each message includes a brief description of the meaning of the message and a suggested response.

ACTIVE MODIFY INVALID

Explanation: An attempt to issue an action character or to modify a field for an active job, user, started task, printer or node was made. However, the action character or field modification is invalid for the active job, user, started task, or printer or node.

User response: Remove the action character or modification from the panel by restoring or blanking the field, or enter the RESET command.

AFD CURSOR row,column

Explanation: A job that invokes SDSF with program name ISFAFD has encountered an error in working with an SDSF panel. The cursor is positioned at *row,column*, where *row* is the number of rows from the top of the display, and *column* is the number of characters from the left of the panel. The possible values for *row* and *column* are 1-9999.

AFD ERROR error-number

Explanation: An error has been encountered in a job that invokes SDSF with program name ISFAFD.

User response: Use the error number to resolve the error. The error numbers are:

001 A comment has not been closed. Comments should be enclosed in `/* */`, for example: `/* This is a comment */`

002 An action character or overtype has been entered on a non-tabular panel, such as a print panel. Action characters and overtypes are valid only on tabular panels.

003 A record has exceeded the maximum length of 9999 bytes. Trailing commas are treated as a continuation character.

004 There is an error in the input syntax. Correct the syntax.

005 Input could not be processed because there are no rows on the panel. This may be because all rows have been blanked out by filters such as FILTER, PREFIX, DEST, and OWNER.

006 An attempt was made to enter an action character, but the NP column is not conditioned for input. The NP column is not conditioned for input on the OD panel. On other tabular panels, the problem may be that there are no rows because all rows have been filtered out by filters such as FILTER, PREFIX, DEST, and OWNER.

007 The specified column could not be found. Either it is not a valid column for the panel, or the column name is an abbreviation that does not uniquely identify a column on the panel. If the column name is an abbreviation, specify the full column name.

008 An attempt has been made to overtype a column that is not overtypeable. If the column is a valid overtypable column for the panel, it may be that the user is not authorized for that column either through ISFPARMS or SAF.

009 Brackets with no column or value, that is <>, were entered on a tabular panel. This syntax is valid only on non-tabular panels such as the print panels.

010 An overtype with no column name, that is <=*value*> was entered on a tabular panel. This syntax is valid only on non-tabular panels such as the print panels.

011 An attempt has been made to overtype the fixed field. The fixed field is not overtypeable.

012 The input could not be processed because there were no rows on the screen. This may be because all rows have been filtered out by filters such as FILTER, PREFIX, DEST, and OWNER.

013 There is an error in the input syntax. Correct the syntax..

ALLOC ERROR return-code error-code information-code

Explanation: Dynamic allocation of the print file failed. SDSF was unable to allocate or create a print file in response to a PRINT command, to a print action character (X), or to the processing of an open print data set panel.

An accompanying message that describes the error can also appear.

For information on dynamic allocation error codes, see the appropriate manual concerning system macros and facilities, or job management.

User response: Use the codes in the message text to determine the source of the error.

ALLOCATION ERROR - error-code

Explanation: An error has occurred during the dynamic allocation of a SYSOUT data set.

User response: For information on dynamic allocation error codes, see the appropriate manual concerning system macros and facilities, or job management.

APPL NOT AVAILABLE

Explanation: An action or overtype requires a SNA application to be associated with the object. However, no SNA application is associated with the object

User response: Remove the action character or modification from the panel by restoring or blanking the field, or type the RESET command.

ARR CRITERIA DISCARDED

Explanation: SDSF detected that the arrange criteria that had been saved from a previous session is invalid. The arrange criteria were deleted from your ISPF profile.

User response: Use the Arrange pop-up or the ARRANGE command to rearrange columns.

ARRANGE CRITERIA OBSOLETE

Explanation: One or more of the columns saved from a previous arrange command has been removed from the ISFPARMS definition for this panel. A column might have been removed because of security changes, release migration, or customization of the field lists.

User response: Look at the INVALID COLUMN message displayed in the message line to see the number of obsolete columns.

ARRANGE PENDING

Explanation: You selected a column or block of columns but did not enter the destination for it.

User response: Scroll the list to the desired column and mark the destination by typing a or b next to it.

AUTHORIZED DEST REQUIRED

Explanation: During SDSF initialization or DEST command processing, SDSF did not find any authorized destination names. You are not authorized to access all destinations, therefore, a valid destination list, specified by IDEST in ISFPARMS, is required. This message also appears in response to a destination query command (DEST ?) if no destination names are authorized.

User response: Enter the DEST command specifying one or more authorized destinations. Notify the SDSF or security administrator regarding the ISF005I messages issued during session initialization.

AUTHORIZED DESTINATION REQUIRED. PRESS THE HELP KEY FOR MORE INFORMATION

Explanation: This message corresponds to the current AUTHORIZED DEST REQUIRED message, and is issued when you display the Destination pop-up.

User response: Press PF1 for complete information, and contact the system programmer.

****** AUTO UPDATE - number SECONDS ******

Explanation: SDSF is running in automatic update mode. The interval between updates is given in seconds. (See the online help for more information on automatic update mode.)

User response: None.

BLOCK COMMAND INCOMPLETE

Explanation: You entered a block command but did not close it (the beginning of a block has been marked with //, but the end has not been marked with //). SDSF does not process pending actions until you close the block.

User response: Close the open block, or use the RESET command to cancel all pending actions.

BLOCK COMMAND INVALID

Explanation: You entered data both on the first and last rows of the block you want to repeat. Only the first or last row of the block can contain data.

User response: Blank out the changes on either the first or last row of the block, or use the RESET command to cancel all pending actions.

BLOCK INPUT REQUIRED

Explanation: You entered a block command but did not specify the action character or overtype. The first row of the block is made current to allow you to enter the action character or overtype to be repeated throughout the block.

User response: Specify the action character or overtype on either on the first or last row of the block or use the RESET command to cancel all pending actions.

BLOCK IS INCOMPLETE

Explanation: You marked the beginning of a block with //, but the end has not been marked with //.

User response: Mark the end of the block with //.

BOOKMANAGER IS REQUIRED

Explanation: The command or pull-down choice requires BookManager READ/MVS.

User response: Blank out the command or pull-down choice.

BOOKMGR SELECT RC=return-code

Explanation: The BOOK command has been issued but SDSF was unable to invokeBookManager. The message text contains the decimal return code from the ISPF select service used to invoke the BOOKMGR command.

User response: Ensure that BookManager is installed and available to your SDSF session, and then retry the BOOK command.

BOTTOM OF DATA REACHED

Explanation: A FIND command reached the bottom of the data without finding the requested character string.

User response: Use the Repeat-Find PF key, or enter an F on the command line, to resume the search at the top of the data.

BRIF ERROR RC=return-code

Explanation: An unexpected error occurred during invocation of the ISPF browse service. The message contains the decimal *return-code* from ISPF. SDSF terminates the browse request.

User response: Refer to *z/OS ISPF Services Guide*.

BROWSE NOT AVAILABLE

Explanation: The SB action character was entered to browse a data set using ISPF, but either SDSF is not running under ISPF or the ISPF level is insufficient. Instead, SDSF does the browse.

User response: Reenter the SB action character when running under the required level of ISPF.

CANNOT MOVE FIXED FIELD

Explanation: You have attempted to move the fixed field with the ARRANGE command. ARRANGE can be used to move columns after the fixed field, but the fixed field itself cannot be moved.

User response: None

number CHARS 'string'

Explanation: In response to a FIND ALL command on the ODS panel or the logs, a number of occurrences of a character string have been found. If SDSF finds more than 999,999 occurrences, *number* is displayed as `999999+`. The cursor is positioned on the character string.

User response: None.

CHARS 'string' FOUND

Explanation: In response to a FIND command, a character string has been found. The cursor is positioned on the character string.

User response: None.

number CHARS 'string' FOUND

Explanation: In response to a FIND ALL command a number of occurrences of a character string has been found. If SDSF finds more than 9,999 occurrences, *number* is displayed as `9999+`. The cursor is positioned on the character string.

User response: None.

CHECK NO LONGER VALID

Explanation: An attempt was made to browse a check. However, the instance of the check has changed since the CK panel was displayed, probably because the check has run.

User response: Press Enter to refresh the CK panel, then browse the check again.

CHECKPOINT OUT OF DATE

Explanation: A checkpoint version has been obtained, but the data might not be current. This can indicate that JES2 is down or not responding. The panel is built using the old data.

User response: Retry the request. If the problem persists, contact your system programmer to determine the cause of the out-of-date data.

CHECKPOINT READ ERROR

Explanation: An error occurred when SDSF attempted to read from the checkpoint data set in order to determine a user's authority to issue a command.

User response: Retry the command. If the problem persists, contact the system programmer.

CHOICE NOT AVAILABLE ON THIS PANEL

Explanation: The pull-down choice is not available on the current SDSF panel.

User response: Use the HELP PF key for information on the pull-down choice.

CKPT OBT ERR return-code-reason-code

Explanation: An error has occurred obtaining a checkpoint version. In the message text, *return-code* is the hexadecimal SSI return code from SSOBRETN and *reason-code* is the hexadecimal reason code from field SSJIRETN. The version is not obtained.

User response: Contact your system programmer to determine the reason for the failure. The return and reason codes are documented in macro IAZSSJI.

CKPT REL ERR return-code-reason-code

Explanation: An error has occurred releasing a checkpoint version. In the message text, *return-code* is the hexadecimal SSI return code from SSOBRETN and *reason-code* is the hexadecimal reason code from file SSJIRETN. The version is not released.

User response: Contact your system programmer to determine the reason for the failure. The return and reason codes are documented in macro IAZSSJI.

CMD NOT ISSUED – NO CONS

Explanation: The function that was attempted requires an EMCS console to issue a system command, and an EMCS console was not available. The command was not issued.

User response: None required.

count CMDS NOT ISSUED

Explanation: A block of action characters was discarded at the request of the user. *count* is the number of action characters that were discarded. No commands were issued.

User response: None.

COLUMN NOT FOUND

Explanation: You specified a column that does not exist for the panel. The cursor is positioned under the column name.

User response: Correct the column name and reenter the command.

COLUMN NOT UNIQUE

Explanation: The column name matches more than one column on the current panel. The cursor is positioned under the column name.

User response: Reenter the column name.

COLUMN TRUNCATED

Explanation: The column width specified with the Arrange function for one or more columns is shorter than the title for the column. The column will be truncated to the specified width.

User response: None required.

COMM NO LONGER AVAIL

Explanation: The user is no longer communicating with the local SDSF server. SDSF will show only data for the system the user is logged on to.

User response: The system may have issued a previous message describing the error. To restore communications, correct any errors and reaccess SDSF.

COMMAND ISSUED

Explanation: SDSF has issued the requested MVS or JES system command.

User response: None.

COMMAND NOT APPLICABLE

Explanation: The command does not apply to the current panel and so is not allowed. It may be valid only on tabular panels.

User response: Access a panel to which the command applies and try the command again. For more information, see "Where used" in the online help for the command.

COMMAND NOT AUTHORIZED

Explanation: You entered an SDSF command that you are not authorized to issue. Refer to “User authorization” on page 475 for more information.

User response: Delete the command.

COMMAND NOT ISSUED

Explanation: An action character was discarded at the request of the user. No command was issued.

User response: None.

COMMAND NOT VALID

Explanation: The command is not valid on the command line of the pop-up.

User response: Correct or erase the command.

COMMAND TRUNCATED

Explanation: You have overtyped more fields than can be processed in a single JES request. All fields up to the JES limit are processed.

User response: Refresh the SDSF displays and overtype the fields that were not updated.

command-count COMMANDS ISSUED

Explanation: A block command has successfully executed and *command-count* commands have been issued.

User response: None.

CONS ACT ERR returncode-reasoncode

Explanation: An attempt to activate an extended console has failed. The message text contains the hexadecimal return code and reason code from the MCSOPER macro. Message ISF032I is also written to the ULOG display.

User response: Use the return code and reason code to determine the cause of the error. Issue the ULOG command to activate the console.

CONS ACT ERR – IN USE

Explanation: An attempt to activate an extended console has failed because the console name is in use. The MCSOPER macro return code is 4 and reason code is 0.

User response: None required. Use the SET CONSOLE command to specify a different console.

CONS DEACT ERR returncode-reasoncode

Explanation: An attempt to deactivate an extended console has failed. The message text contains the hexadecimal return code and reason code from the MCSOPER macro.

User response: Use the return code and reason code to determine the cause of the error.

CONSOLE console-name SHARED

Explanation: An attempt has been made to activate an extended console but the console is in use. SDSF shares the console by issuing commands using its console ID. However, responses are not returned to the SDSF session issuing the commands.

If the console is in use by another SDSF session (such as through split screen), any command responses caused by the shared session is returned to that session.

Message ISF031I is written to the ULOG display.

User response: None

CONVERSION COMPLETE

Explanation: SDSF parameters in ISFPARMS have been assembled through the conversion utility and converted to ISFPARMS in statement format.

User response: You can edit the statements from the pop-up. To activate the ISFPARMS, or check their syntax, use the MODIFY command.

DATA ACCESS ERROR

Explanation: An error has occurred retrieving data to build an SDSF panel. Communications with the server may have been lost. Additional messages may be issued describing the error. The current request will be ended.

User response: See accompanying messages, if any, for more information about the problem. Retry the request.

DATA NOT AVAIL system-name

Explanation: A sysplex request for data has been processed, but the data from *system-name* cannot be gathered. The plus (+) character is shown if more than one system is not responding.

User response: None if the system is at the z/OS V1R12 or lower level. Otherwise, ensure the SDSF server is started and configured to process XCF requests. Verify that the SDSFAUX address space is started on all systems for which data is to be gathered.

DATA NOT SAVED

Explanation: A user entered the SE action character to edit a data set using ISPF, and either entered the SAVE command or made changes to the data during the ISPF session. The changes were not saved upon exit since permanent changes cannot be made.

User response: None.

DATA SET ALLOCATED

Explanation: In response to a browse action, a data set has been allocated.

User response: None.

DATA SET DISPLAYED

Explanation: SDSF is displaying the requested SYSOUT data set on the Output Data Set panel.

User response: None.

****** DATA SET NOT CATALOGED DSNAME= data-set-name**

Explanation: The required data set is not cataloged. This message accompanies the message `ALLOC ERROR`*return-code error-code information-code*, or `LOCATE ERROR`*return-code*, and explains why allocation of the print file failed.

User response: None.

DATA SET NOT ELIGIBLE

Explanation: The data set is not eligible for the operation. The data set is not changed. This condition can occur if the output group is in operator or system hold or is currently being processed by the SSI.

User response: Ensure that the output group is not in operator or system hold.

DATA SET NOT FOUND

Explanation: A data set entered on an SDSF panel could not be located.

User response: Either allocate the data set or change the name of the data set on the SDSF panel.

******* DATA SET NOT ON VOLUME DSNAME= data-set-name**

Explanation: The required data set is not on the specified volume. This message accompanies the message `ALLOC ERROR`*return-code error-code information-code*, or `OBTAIN ERROR`*return-code*, and explains why allocation of the print file failed.

User response: None.

****** DATA SET OPEN DSNAME = data-set-name**

Explanation: The data set *data-set-name* is open. This message accompanies the message `ALLOC ERROR`*return-code error-code information-code*, and explains why dynamic allocation of the print file failed.

User response: None.

****** DATA SET UNAVAILABLE DSNAME= data-set-name**

Explanation: The required data set is unavailable. This message accompanies the message `ALLOC ERROR`*return-code error-code information-code*, and explains why dynamic allocation of the print file failed.

User response: None.

DATA TRUNCATED FOR EDIT

Explanation: A request has been made to edit a data set using the SE action character, but the job contains a data set that exceeds the maximum record length supported by edit. The edit request is processed, but the data is truncated to the 255 character maximum.

User response: Use the S or SB action characters to display the entire record.

DEALLOCATION ERROR - error-code

Explanation: An error has occurred during the dynamic deallocation of a SYSOUT data set.

User response: For information on dynamic allocation error codes, see the appropriate manual concerning system macros and utilities or job management.

DEST ALREADY EXISTS

Explanation: The DEST command was issued to add a destination that already exists in the current destination list.

User response: Use DEST ? or SET DISPLAY to display the current destinations and correct the command.

DEST NOT FOUND

Explanation: The DEST command was issued to delete a destination that is not in the current destination list. The destination not in the list has the cursor positioned under it.

User response: Use DEST ? or SET DISPLAY to display the current destinations and correct the command.

DETAIL NOT AVAIL

Explanation: A request to retrieve the enclave detail information has failed because the information is not available. The enclave may no longer be valid.

User response: None required.

DISPLAY RESET

Explanation: The logical screen size changed, causing SDSF to rebuild the display. SDSF ignored and cleared any action characters or commands you had entered but had not yet executed.

User response: None.

DSORG NOT PS OR PO

Explanation: In a PRINT ODSN command, the specified data set was not sequential, (DSORG=PS) or partitioned (DSORG=PO).

User response: Reissue the PRINT ODSN command specifying an acceptable data set name. When the data set is allocated, a data set organization of sequential or partitioned must be specified.

EDIF ERROR RC=return-code

Explanation: An unexpected error occurred during invocation of the ISPF edit service. The message contains the decimal *return-code* from ISPF. SDSF terminates the edit request.

User response: Refer to *z/OS ISPF Services Guide*.

EDIT NOT AVAILABLE

Explanation: The SE action character was entered to edit a data set using ISPF, but SDSF is not running under ISPF. Instead, SDSF does a browse.

User response: Reenter the SE action character when SDSF is running under the required level of ISPF.

ENC IMPLICITLY QUIESCED

Explanation: An attempt was made to quiesce an enclave that is already implicitly quiesced because one or more address spaces associated with it is quiesced.

User response: None required.

END OF DATA ON MENU

Explanation: SDSF could not read a requested help panel from the SDSF help panel data set.

User response: The system programmer should check any changes that have been made to the SDSF help panel data set. If the problem cannot be found, the system programmer might want to replace the installed SDSF help panel data set with the original help panel data set on the SDSF distribution tape.

ENGLISH HELP NOT AVAILABLE

Explanation: You selected the English language but the English help panels are not available.

User response: Erase the selection or see your system programmer about the installation.

ENTER REQUIRED FIELD

Explanation: Data is missing for a required field. The cursor is positioned at the field in error.

User response: Enter the requested data.

ERROR IN ASSEMBLING PARAMETERS. RETURN CODE return-code

Explanation: SDSF parameters being assembled through the conversion utility caused assembly errors.

User response: Use the return code from the assembler to help identify the problem. The conversion utility pop-up lets you edit the ISFPARMS source data set (PF4) or browse the assembler listing (PF5).

ERROR PROCESSING DATA

Explanation: SDSF could not successfully process the spool control blocks of one of the jobs on the panel.

User response: The user or system programmer could use one of the filter commands to identify which job is causing the problem.

For example, the user's panel shows these jobs: `ABLEJOB ABLEBJOB ANDJOB BJOB BBBJOB CJOB`

The user issues PREFIX A*, and the panel shows these jobs: `ABLEJOB ABLEBJOB ANDJOB`

The error message still appears on the panel, so the problem is with one of the three jobs shown. The user then issues a second PREFIX command, PREFIX ABLE*. The panel then shows: `ABLEJOB ABLEBJOB`

The error message no longer appears on the panel. The user knows that the problem is not with ABLEJOB or ABLEBJOB; the problem must be with ANDJOB.

ERROR PROCESSING LINE line-number: text-of-line

Explanation: The conversion exec has encountered an error in the indicated line.

User response: Follow your local procedure for reporting a problem to IBM

service FAILED WITH RC=return-code REASON=ispf-message-text

Explanation: An ISPF or TSO service, *service*, failed with the indicated return code, and text of an ISPF message if it is available.

User response: Use the return code and the message text, if any, to understand and resolve the problem. If the problem persists, follow your local procedure for reporting a problem to IBM

FIELD INVALID

Explanation: Invalid information was typed in a field.

User response: Correct what was typed in the field or type `RESET` on the command line.

FIELD NOT NUMERIC

Explanation: A numeric field was overtyped with non-numeric data, or there are blanks in the numeric field. The cursor is positioned at the field in error.

User response: Enter the field using numeric data. Within a tabular panel, use the RESET command to clear any overtyped data.

FILE SIZE NOT AVAILABLE

Explanation: A request has been made to view a data set, but the file size (in bytes) is not available from JES. The file size is required by SDSF to allocate the temporary file used by GDDM

FILTER CRIT DISCARDED

Explanation: SDSF detected that the filter criteria that had been saved from a previous session are invalid. The filter criteria were deleted from your ISPF profile.

User response: Use the Filter pop-up or FILTER command to define filters.

FILTER CRITERIA OBSOLETE

Explanation: One or more of the columns saved from a previous session has been removed from the ISFPARMS definition for this panel. A column might have been removed because of security changes, release migration, customization of the field lists in ISFPARMS, or other customization of function such as symbol support. The obsolete filter criteria are deleted.

SDSF filtered the columns using the remaining columns. Look at the INVALID COLUMN message displayed in the message line to see the number of obsolete columns.

User response: No action is required.

FILTER NOT FOUND

Explanation: An attempt was made to delete a filter that does not exist.

User response: No action is required. If the command to delete the filter was entered incorrectly, correct the command.

FILTER VALUE TRUNCATED

Explanation: A filter value entered with a previous command exceeds the 25-character length of the value field on the Filter pop-up. The value is truncated to fit the field.

User response: None required. To change the value, type the changes on the pop-up.

FILTERING IS ON|OFF

Explanation: In response to a query of the filters, the current state of filtering is displayed.

User response: If a filter is displayed on the command line, pressing Enter issues the command and makes the filter active.

GDDM ERROR severity-msgnumber

Explanation: An error occurred during execution of a GDDM service. *severity* is the severity code, in decimal, of the message; *msgnumber* is the GDDM message number in decimal.

The request to view a data set is ended. Other explanatory messages might have been issued by GDDM.

User response: Correct the error described by the GDDM message text and retry the view request.

GDDM LEVEL ERR gddm-level

Explanation: The view function was requested, but the installed level of GDDM cannot be used by SDSF. *gddm-level* is the level of GDDM currently being accessed by SDSF. SDSF requires GDDM Version 2 Release 2 or a later release.

User response: The system programmer should ensure that the correct level of GDDM is available to the SDSF session either through a STEPLIB or the system LINKLST.

GDDM NOT AVAILABLE

Explanation: SDSF was unable to load the GDDM interface module, ADMASPT, in response to a view request to compose a page-mode data set. The view function is not available because GDDM services cannot be used.

User response: The system programmer should ensure the GDDM load modules are available to the SDSF session either through a STEPLIB or the system LINKLST.

GET ERROR RC=return-code

Explanation: The GET request for the spool data for a job failed. The job's SYSOUT is not displayed. This may occur if the job was purged or if the SYSOUT data was selected from the Display Active Users (DA) panel and the job was swapped out.

User response: Try displaying the SYSOUT later. If the job was active and swapped out, the SYSOUT will be accessible. If the job was purged, the SYSOUT will not be found. For a description of the return codes, refer to *z/OS DFSMS Macro Instructions for Data Sets*.

HC NOT ACTIVE sysname | count SYSTEMS

Explanation: Checks could not be displayed because z/OS is not running. If a single system reports that z/OS is not running, the system name, *sysname*, is displayed. If more than one system reports that z/OS is not running, the number of systems, *count*, is shown.

User response: For information on starting z/OS, the system programmer should refer to *IBM Health Checker for z/OS User's Guide*.

HELP MENU ERROR= member-name

Explanation: SDSF could not find the requested help panel.

User response: The system programmer should check any changes that have been made to the SDSF help panel data set. If the problem cannot be found, the system programmer might want to replace the installed SDSF help panel data set with the original help panel data set supplied by IBM

HEX STRING INVALID

Explanation: The FIND command with a hexadecimal string has been issued on a panel other than the logs or ODS panels.

User response: Correct the command and reissue it.

INACTIVE MODIFY INVALID

Explanation: An attempt to issue an action character or to modify a field for an inactive job, user, started task, printer or node was made. However, the action character or field modification is invalid for the inactive job, user, started task, or printer or node.

User response: Remove the action character or modification from the panel by restoring or blanking the field, or enter the RESET command.

INCONSISTENT PARAMETERS

Explanation: The FIND command has been issued with parameters that conflict.

User response: Correct the command and reissue it.

****** INCORRECT UNIT NAME SUPPLIED**

Explanation: The dynamic allocation of a tape drive failed with a X'021C' return code. This return code specifies that an incorrect unit name has been supplied. The valid units that are supported are: 3480, 3400-3, 3400-5, 3400-6, and 3400-9.

User response: Specify a cataloged data set name that is on a supported tape unit.

INPUT FILE ALLOC FAILED

Explanation: An error occurred trying to allocate the input file to be composed. Additional messages describing the reason for the allocation failure is issued by the system. The file cannot be viewed using GDDM

INPUT INVALID WITH BLOCK

Explanation: An action character or overtype was entered within an open block. Data to be repeated can only be entered on the first or last row of the block. The display is positioned to the row containing the data within the block.

User response: Blank out the data on the row or enter the RESET command to cancel all pending actions.

INPUT INVALID WITHIN BLOCK

Explanation: You entered one or more characters within a block on the pop-up.

User response: Erase the character.

INT CONSOLE NOT ALLOWED

Explanation: An attempt was made to issue a system command using console ID 0 (INTERNAL), but an EMCS console is required by values specified in ISFPARMS.

User response: Reissue the command using an EMCS console. If you are issuing a command using `i/`, remove the `i`.

INVALID CALL TYPE

Explanation: During initialization, SDSF found an error processing ISFPARMS. The error is in the ISFNTBL macro or NTBL statement named in the IDEST parameter of the ISFGRP macro or GROUP statement for the user.

User response: The system programmer should check the ISFNTBL macro or NTBL statement named in the IDEST parameter of the ISFGRP macro or GROUP statement that was used to place the user in a user group.

The system programmer might also want to put the installation-defined names last in the ISFNTBL macro or NTBL statement, as the installation-defined names can be the most likely to cause an error. When SDSF encounters an error in the destination names during initialization, it continues initialization with the destination names that were successfully processed before the error.

INVALID CLASS class ENTERED

Explanation: An invalid class was entered with the ST, I, or O command. The class is ignored. Valid class names are:

ST command:
A-Z, 0-9, +, !, $, *,), -, ?, #, @. = and /

I command:
A-Z, 0-9, !, $, *, #, and @

JC command:
A-Z, 0-9, $ and #

O command:
A-Z, 0-9, and @

User response: Retry the command with a valid class.

INVALID CLASS NAME

Explanation: This field was updated with an invalid class name. Valid class names consist of the characters A-Z and 0-9.

User response: Type either a valid class name or a blank in the field, or type `RESET` in the command line.

INVALID COLUMN: column-info

Explanation: Column criteria for this panel were saved from a previous SDSF session, but one or more of the columns have been removed from this panel. SDSF ignores the criteria and deletes it from your SDSF profile. *column-info* is either a number of columns, or, for SORT, a list of columns. This message is issued as explanatory information with the ARRANGE, FILTER, or SORT CRITERIA OBSOLETE message.

User response: No action is required. You can establish new arrange, filter, or sort criteria.

INVALID COMMAND

Explanation: A command or action character was entered that is not recognized by SDSF, was entered in an unsupported environment, or was entered on a panel or row for which it is invalid. The command or action character might have been entered with an invalid parameter.

User response: Correct the command or action character and retry the request. See the SDSF publications or online help for a list of valid SDSF commands and action characters. For system commands, see the appropriate MVS and JES manuals. For the AFD command, see Chapter 12, "Using SDSF in batch," on page 369.

INVALID DESTINATION NAME

Explanation: The specified destination name is invalid for this system. If the destination name is an installation-defined destination name, this message might be issued because JES is not active. When JES is not active, the installation-defined destination names are not available to SDSF.

User response: Enter a valid destination name.

INVALID DSN - LENGTH

Explanation: A data set name has been entered that is longer than 44 characters.

User response: Correct the data set name being entered.

INVALID DSN - QUOTES

Explanation: A data set name has been entered with unmatched quotes.

User response: Correct the data set name being entered.

INVALID HEX STRING

Explanation: Invalid hexadecimal data has been entered either by overtyping a field or with a FIND command. The invalid data contains non-hexadecimal characters or has an uneven number of digits.

User response: Correct the hexadecimal string.

INVALID LEFT BOUNDARY

Explanation: The value entered for the starting column with a FIND command is greater than the logical record size or is greater than the length of the field.

User response: Correct the FIND command and reissue it.

INVALID RETURN CODE

Explanation: An invalid return code has been received after a call to an internal SDSF subroutine. The table being displayed might be incomplete.

User response: Retry the command, and if the problem persists, contact IBM

INVALID SAVED DEST

Explanation: A saved destination name from a previous SDSF session is no longer valid. This could occur if an enhanced destination name was retrieved from an SDSF session that was running on a system prior to MVS/ESA SP-JES2 4.2.0. Use DEST ? or SET DISPLAY ON to view the current destination list.

User response: None. SDSF is initialized using any remaining saved values.

INVALID SCROLL AMOUNT

Explanation: The amount specified in the SCROLL field of the panel, or in a scroll command, is invalid.

User response: Enter one of the following valid scroll amounts:

Page to scroll one panel.

Half to scroll half of one panel.

number to scroll a specific number of lines or columns. *number* can be up to four digits.

Max to scroll to the end of the data.

Csr to scroll to the position of the cursor.

Data to scroll one line or column less than one page. This is valid only under ISPF.

If the message is accompanied by an audible alarm, it was issued by ISPF. Pressing the PF key assigned to HELP signals ISPF to display the valid scroll entries on line 3 of the display.

INVALID SELECTION

Explanation: The input is not valid for this panel.

User response: Enter a valid command or menu option.

INVALID SYNTAX

Explanation: The command entered on the command line has too many parameters, has unmatched quotes, or is an invalid range.

User response: Use the appropriate manual or online help to find the syntax of the command.

INVALID UNIT

Explanation: Either an invalid device number was entered on the PR, PUN, RDR or LI panel, or both a volume serial and a generic unit have been specified on the open print data set panel.

For the PR or PUN panel, the unit device number must consist of all hexadecimal digits. Leading zeros are required.

For the LI panel, the unit device number must be either all hexadecimal digits or `SNA`. Leading zeros are required.

The device number can be preceded with a slash (/).

For the open print data set panel, only one of the fields (volume serial or unit) can be specified.

User response: Enter a valid device number or specify only one of the print panel fields.

INVALID UPDATE VALUE

Explanation: The user has entered an invalid update value for an overtypeable field. Invalid values include: a semicolon, a comma when not enclosed in parentheses, or a left parenthesis if it is the first update character in a field that does not allow multiple values to be entered.

User response: Enter a valid name.

INVALID VALUE

Explanation: A value has been entered that is unrecognized or not allowed on the current panel.

User response: Change the input to an allowable value.

I/O ERROR ON INDEX

Explanation: An I/O error occurred in reading the SDSF SYSLOG index. It is normal for this message to appear the first time SDSF is used.

User response: The system programmer should check the accompanying system messages for more information on the I/O error.

ISFTRACE DD MISSING

Explanation: A TRACE command has been entered, but the ISFTRACE file is not allocated. The TRACE command is not processed.

User response: Allocate the ISFTRACE file and reissue the TRACE command.

ISPF REQUIRED

Explanation: The command was issued when SDSF was not operating under ISPF. Some commands are valid only when SDSF was accessed through ISPF.

User response: Access SDSF through ISPF and reissue the command.

JAPANESE HELP NOT AVAILABLE

Explanation: The Japanese Help/Tutorial feature is not installed.

User response: See your system programmer.

JCT NOT AVAILABLE

Explanation: Either the object has no job control table (JCT) or an error occurred trying to process the JCT for the object.

User response: Delete the command or type `RESET` on the command line.

jesx NOT ACTIVE

Explanation: The JES subsystem *jesx* is not active and one of the following has happened:

- You attempted to enter a command, select a pull-down choice, or process a pop-up that requires JES.
- SDSF attempted to obtain a checkpoint version. The checkpoint is not obtained.

User response: Exit SDSF and retry the request when *jesx* is active.

JES REQUIRED

Explanation: You issued a command, selected a pull-down choice or attempted to process a pop-up that requires JES. JES is not currently active.

User response: Contact the system programmer. When JES is active again, exit SDSF and reaccess it to make all SDSF functions available.

JES REQUIRED FOR MAS

Explanation: The RES panel was accessed with the default parameter of MAS, either with the command or pull-down choice, but SDSF cannot determine which members are in the MAS. SDSF requires JES2 to determine the members in the MAS, and JES2 is unavailable. As a result, the panel shows all systems in the sysplex.

User response: None required.

JES 1.7 REQUIRED

Explanation: The function that was attempted requires z/OS V1R7 JES2. For action characters and overtypeable columns, both the user's system and the object's system must be at z/OS V1R7 JES2.

User response: Delete the action character or overtype.

JES2 ENVIRONMENT ONLY

Explanation: A command or option was entered that requires SDSF to be processing a JES2 subsystem, but SDSF is processing a JES3 subsystem. The command is rejected.

User response: None required.

JES3 ENVIRONMENT ONLY

Explanation: A command or option was entered that requires SDSF to be processing a JES3 subsystem, but SDSF is processing a JES2 subsystem. The command is rejected.

User response: None required.

JES2 REQUIRED FOR MAS

Explanation: A command included the MAS option when SDSF was processing a JES3 subsystem. The MAS option requires the JES2 environment. The option is internally converted to ALL.

User response: None required.

JOB IS PROTECTED

Explanation: The P action character has been used against a protected job. The job has not been canceled.

User response: Use the PP action character to cancel a protected job.

JOB NO LONGER VALID

Explanation: A command that was issued for a job failed, either because:

- The job has been purged
- The output group is no longer available. This could be because the characteristics have changed.

User response: If the output group is no longer available but the data sets still exist, re-access the panel again and try again.

JPN HELP NOT AVAILABLE

Explanation: The Japanese Help/Tutorial feature is not installed.

User response: See your system programmer.

number LINES PRINTED

Explanation: In response to a PRINT command or print action character (X), *number* lines have been printed. When you enter multiple X action characters, *number* is the lines in the last printed data set.

User response: None.

LINE NOT AVAILABLE

Explanation: An action or overtype requires a line device to be associated with the object. However, no line device is associated with the object

User response: Remove the action character or modification from the panel by restoring or blanking the field, or type the RESET command.

LOCATE ERROR return-code

Explanation: An attempt was made to open a print data set. A LOCATE request for the specified data set failed with return code *return-code*. The system can also issue an explanatory message.

User response: Ensure that the data set being

processed is an existing data set.

LOG BROWSE ERR returncode-reasoncode

Explanation: An error occurred in trying to browse the log stream displayed on the OPERLOG panel. The message text contains the hexadecimal return and reason codes from the IXGBRWSE macro.

User response: Try issuing the LOG command again or scrolling up or down with a scroll amount of MAX. If the problem persists, use the return and reason codes to determine the cause of the error.

LOG CONN ERR returncode-reasoncode

Explanation: An error occurred in trying to connect to the log stream when displaying the OPERLOG panel. The message text contains the hexadecimal return and reason codes from the IXGCONN macro.

User response: Use the return and reason codes to determine the cause of the error.

LOG DISC ERR returncode-reasoncode

Explanation: An error occurred in trying disconnect from the log stream displayed on the OPERLOG panel. The message text contains the hexadecimal return and reason codes from the IXGCONN macro.

User response: Use the return and reason codes to determine the cause of the error.

LOG FUNCTION INOPERATIVE

Explanation: The SDSF SYSLOG panel is not available due to an SDSF initialization error.

User response: The system programmer should check the accompanying write-to-operator message for more information.

LOGIC ERROR 1

Explanation: SDSF could not process the command as it was entered.

User response: Delete the command or enter the correct command.

LOGIC ERROR 2

Explanation: SDSF could not process the command as it was entered.

User response: Delete the command or enter the correct command.

LOGIC ERROR 3

Explanation: An internal error has occurred processing action characters or overtypes. Some actions since the last enter might have been lost.

User response: Press Enter to refresh the display and retry the actions or overtypes. If the problem persists, contact IBM for assistance.

LOGIC ERROR IN INDEX

Explanation: The SYSLOG index was reformatted and reverified to recover from a logic error. The logic error could have been caused by a TSO user pressing the PA1 key to terminate SDSF during initialization.

User response: This message is informational. However, if this error occurs consistently, contact the system programmer.

LOGLIM yyyy.ddd hh:mm:ss

Explanation: The OPERLOG is being filtered and the limit for the number of hours to search has been reached. *yyyy.ddd hh:mm:ss* is the date and time of the record being processed when the limit was reached. Processing is ended for the current request.

SDSF might have been reading forward or backward in the OPERLOG. If SDSF detected more than one limit in processing a single request, the message is issued for the last record that was processed.

User response: Enter the LOGLIM command to change the limit for the operlog display. You can also enter the LOCATE command (by date and time) the NEXT and PREV commands, or SCROLL UP or DOWN MAX commands to scroll to a new position in the OPERLOG.

LRECL TOO LARGE FOR GDDM

Explanation: An attempt was made to view a file using the V action character. However, GDDM could not be invoked because the input record length of the file exceeded the maximum that can be processed by GDDM. See the GDDM documentation for the maximum record lengths acceptable to GDDM.

User response: The view request is terminated. The file can be browsed using SDSF, but not viewed using GDDM.

MEMBER NAME MISSING

Explanation: A member name was not specified on an SDSF panel, but the data set being used is partitioned.

User response: Specify a member name for the data set, or use a different data set name.

MEMBER NAME NOT ALLOWED

Explanation: A member name was specified on a command or panel, but the data set being used is sequential.

User response: Delete the member name for the data set, or use a different data set name.

MEMBER NOT FOUND

Explanation: A member of a PDS was specified on an SDSF panel, but the PDS does not contain a member with that name.

User response: Correct the member name.

MENU READ LOOP

Explanation: A loop has occurred processing the SDSF help panels under TSO.

User response: Contact IBM for assistance.

MERGE ERROR returncode-reasoncode

Explanation: An error occurred issuing an SJF merge request. In the message text, *returncode* is the decimal return code from the SJF merge service and *reasoncode* is the decimal reason code.

User response: Attempt to reissue the modify request. If the error persists, contact your system programmer for assistance.

MIGRAT ALLOC FAILURE

Explanation: In response to a PRINT ODSN command, the required print data set was migrated and could not be allocated.

User response: Recall the print data set and reissue the PRINT ODSN command.

MOD NOT ALLOWED FOR PDS

Explanation: An attempt has been made to allocate a print data set with MOD, but the data set is partitioned. SDSF does not support MOD for this case.

User response: Change the disposition to OLD or NEW or specify a sequential data set.

MODULE NOT FOUND

Explanation: A QUERY MODULE command was issued for a module but the module could not be located.

User response: The module named on the QUERY MODULE command must be an SDSF module that is accessible or currently loaded by SDSF.

MODIFY ISSUED-number DS

Explanation: A request to modify the output descriptors has been scheduled. *number* is a count of the number of data sets in the output group at the time the request was issued (leading zeros suppressed). A SWB modify request applies to all the data sets in the group.

User response: None.

MUTUALLY EXCLUSIVE UPD

Explanation: The use of an action character or overtype was incompatible with the concurrent use of another overtype. For example, you cannot use the P action character on the H display while simultaneously overtyping the class field. Purge and the class change are mutually exclusive.

User response: Either restore or delete the field, or type `RESET` on the command line.

NO sysid SYSLOG FOUND

Explanation: SDSF is unable to locate any SYSLOG data sets for the SYSID being processed.

User response: Use the SYSID command to change the SYSID, for example `SYSID IP01`.

NO CHARS 'string' FOUND

Explanation: The FIND command could not find the character string *string*.

User response: None.

NO COMMAND PROVIDED

Explanation: Command text was not entered with the command on the system command extension pop-up or the / command.

User response: None.

NO DATA IN DATA SETS

Explanation: The data sets for the job that has been selected are all empty data sets. There is no data to browse.

User response: None.

NO DATA SETS ALLOCATED

Explanation: An allocation failure has occurred for each data set in the job to be displayed. Since no data sets were allocated, they cannot be browsed.

Additional messages describing the specific allocation failures might have been issued by the system.

User response: Use the system messages to determine

the reason for the allocation failure and retry the request.

NO DATA SETS AUTHORIZED

Explanation: An attempt was made to display a job but there is no data set the user is authorized to view.

User response: If you have been denied access in error, see "User authorization" on page 475 for more information.

NO DATA SETS OPENED

Explanation: An open failure occurred for each data set in the job to be displayed. Since no data sets were opened, they cannot be browsed.

Additional messages can be issued by the system describing the error.

User response: Determine the reason for the open failure using the error codes in the message.

NO DISPLAYABLE DATA

Explanation: A user has attempted to display a job's SYSOUT data, but the job has no data that can be displayed by that user.

User response: Delete the command or type `RESET` on the command line.

NO FILTERS AVAILABLE

Explanation: An attempt was made to turn filtering on when there are no available filters.

User response: None required. To filter the panel, type a filter command or type FILTER ? to enter a filter on the Filter pop-up.

NO HELP AVAILABLE

Explanation: SDSF could not show a help panel under TSO because it was unable to allocate or open the SDSF help panel data set.

User response: Check that the SDSFMENU data set is allocated to the SDSF help panel library. Check the MENUS and MENUVOL parameters in ISFPARMS to see that they are coded correctly.

NO OPERLOG FOUND

Explanation: You entered a LOG command to display the OPERLOG panel, but no log stream is available to display.

User response: To display the SYSLOG panel, which contains messages for a single system, type `LOG S`.

NO PREFIX 'string' FOUND

Explanation: The character string *string* was not found in response to a FIND command.

User response: None.

NO PREVIOUS INPUT

Explanation: You entered a repeat command, but no modification has yet been done to repeat.

User response: Enter an action character or overtype a field prior to using the repeat command.

NO RESPONSE FROM RMF

Explanation: SDSF has passed the timeout limit awaiting a response from RMF to display the DA panel.

User response: Retry the request. To bypass the error, use the SYSNAME command or pull-down choice to limit the number of systems being processed.

NO RESPONSE RECEIVED

Explanation: The delay interval for a command response or sysplex data had been reached. The command response or data on the SDSF panel is not shown. Sysplex data not shown may include WTORs on the Log panel, when you have used the SYSID command to request the log for a system other than the one you are logged on to.

User response: To see the command response, issue the ULOG command to view the user log. To increase the delay interval, use the SET DELAY command.

To increase the delay interval for sysplex data, use the SET TIMEOUT command.

You might also try limiting the amount of sysplex data being returned, with one or more of the following:

- Parameters on the panel command, for example, `PR 1` to see only printer 1.
- The SYSNAME command or pull-down choice, to restrict the systems to be included.
- The DEST command or pull-down choice, to restrict the destinations to be included.
- The SELECT command, to temporarily restrict the panel based on the fixed field, for example, `SELECT PRT33` to see only printer PRT33.

Note that the Filter function does not have the effect of limiting the data returned

If the problem cannot be corrected with these methods, the operator or system programmer should ensure that one or more SDSF servers has not been stopped by issuing the `F server,D,C` command. The system programmer should also review the WebSphere MQ configuration for possible communications problems.

Refer to “WebSphere MQ considerations” on page 344.

NO SUFFIX 'string' FOUND

Explanation: The character string *string* was not found in response to a FIND command.

User response: None.

NO SYSLOG DATA

Explanation: No SYSLOG data has been found in any SYSLOG data set.

User response: Verify that the SYSID command was issued for the proper SYSLOG data set, or that the saved SYSID value is correct.

NO WORD 'string' FOUND

Explanation: The character string *string* was not found in response to a FIND command.

User response: None.

NOT ALL SYMBOLS SHOWN

Explanation: The number of symbols exceeds the number of symbols that can be shown by the pop-up.

User response: Follow your local procedure for reporting a problem to IBM.

NOT ALLOWED WITH OUTDESC

Explanation: A value for forms, process mode, PAGEDEF, or FORMDEF has been entered along with an Output Descriptor Name. Those fields cannot be specified when Output Descriptor Name is used.

User response: Delete the value for forms, process mode, PAGEDEF, or FORMDEF if an Output Descriptor Name is to be used. Alternatively, delete the Output Descriptor Name.

NOT AUTH TO LOGSTREAM

Explanation: You are not authorized to the log stream. Access to the log stream is required for this function.

User response: Contact your security administrator for authorization to the log stream.

NOT AUTH TO OPERLOG

Explanation: You entered a LOG command to display the OPERLOG panel, but are not authorized to the log stream that is displayed on the OPERLOG panel.

User response: To display the SYSLOG panel, which contains messages for a single system, type `LOG S`.

NOT AUTHORIZED BY EXIT

Explanation: You attempted to issue a command that you are not authorized by the SDSF user exit to issue.

User response: Delete the command.

If you have been denied authorization in error, the system programmer should check the SDSF user exit module, ISFUSER.

NOT AUTHORIZED FOR CHECK

Explanation: You are not authorized to issue the command for the check.

User response: Delete the command.

If you have been denied authorization in error, see “User authorization” on page 475 for more information.

NOT AUTHORIZED FOR CHOICE

Explanation: You are not authorized for the pull-down choice.

User response: Select another choice or press PF3 to close the pull-down. If your authorization has changed during the current SDSF session and the change is not yet reflected in the pull-down, either type the SDSF command associated with the choice or exit and reenter SDSF.

If you have been denied authorization in error, see “User authorization” on page 475 for more information.

NOT AUTHORIZED FOR CLASS

Explanation: The user is not authorized to issue commands against the class.

User response: Delete the command.

If you have been denied authorization in error, see “User authorization” on page 475 for more information.

NOT AUTHORIZED FOR CMD

Explanation: You attempted to issue an action character, overtype a field, or issue an MVS or JES command that you are not authorized to issue.

User response: Delete the action character, overtyped information, or MVS or JES command.

If you have been denied authorization in error, see “User authorization” on page 475 for more information.

NOT AUTHORIZED FOR CONS

Explanation: You attempted to activate an extended console but are not authorized to the console. The console is not activated, and the message responses is not available to the ULOG panel or with the slash command.

User response: Contact your security administrator to grant you access to the extended console.

If you have been denied authorization in error, see "User authorization" on page 475 for more information.

NOT AUTHORIZED FOR DEV

Explanation: The user is not authorized to issue commands against the device.

User response: Delete the command.

If you have been denied authorization in error, see "User authorization" on page 475 for more information.

NOT AUTHORIZED FOR DEST

Explanation: You are not authorized for a requested destination name.

User response: Delete the destination name.

If you have been denied authorization in error, see "User authorization" on page 475 for more information.

NOT AUTHORIZED FOR ENC

Explanation: The user is not authorized to issue commands for the enclave.

User response: Delete the command.

NOT AUTHORIZED FOR FUNCTION

Explanation: You are not authorized for the function provided by a pop-up.

User response: Cancel the pop-up.

If you have been denied authorization in error, see "User authorization" on page 475 for more information.

NOT AUTHORIZED FOR INIT

Explanation: You are not authorized to issue commands to the initiator.

User response: Delete the command.

If you have been denied authorization in error, see "User authorization" on page 475 for more information.

NOT AUTHORIZED FOR JOB

Explanation: You are not authorized to issue commands against the job.

User response: Delete the command.

If you have been denied authorization in error, see "User authorization" on page 475 for more information.

NOT AUTHORIZED FOR NODE

Explanation: The user is not authorized to issue commands against the node.

User response: Delete the command.

If you have been denied authorization in error, see "User authorization" on page 475 for more information.

NOT AUTHORIZED FOR PROC

Explanation: You are not authorized to issue commands to the z/OS UNIX process.

User response: Delete the command.

If you have been denied authorization in error, see "User authorization" on page 475 for more information.

NOT AUTHORIZED FOR PRT

Explanation: You are not authorized to issue commands to the printer.

User response: Delete the command.

If you have been denied authorization in error, see "User authorization" on page 475 for more information.

NOT AUTHORIZED FOR RES

Explanation: You are not authorized to issue commands to the system resource.

User response: Delete the command.

If you have been denied authorization in error, see "User authorization" on page 475 for more information.

NOT AUTHORIZED FOR SE

Explanation: You are not authorized to issue commands to the WLM scheduling environment.

User response: Delete the command.

If you have been denied authorization in error, see "User authorization" on page 475 for more information.

NOT AUTHORIZED FOR SYS

Explanation: You are not authorized to issue commands for the member of the MAS.

User response: Delete the command.

If you have been denied authorization in error, see "User authorization" on page 475 for more information.

NOT AUTHORIZED TO DATA

Explanation: The server has rejected a request for sysplex data due to an authorization failure. The data is not displayed.

User response: Exit SDSF and then reaccess it.

NOT PAGE MODE DATA

Explanation: A view request was entered for a data set that is not page mode. SDSF considers a data set to be page mode only if it is identified as being page mode by JES. SDSF converts the view request to browse. The data set is not be composed by the view utility, but is displayed on the ODS panel.

User response: None.

NOT VALID FOR TYPE

Explanation: The action character is not a valid action against that object type.

User response: Enter the correct action character.

NOT VALID WHEN REXX

Explanation: An SDSF command was issued or a command operand was used that is not valid in the REXX environment.

User response: Delete the command or operand.

Refer to Chapter 13, "Using SDSF with the REXX programming language," on page 377 for more information.

"O" ACTION REQUIRED

Explanation: The field modification the user has attempted requires the O action character.

User response: Issue the O action character.

OBTAIN ERROR return-code

Explanation: An attempt was made to open a print data set. An OBTAIN request failed with return code *return-code*.

The system can also issue an explanatory message.

User response: Ensure that the data set being processed exists either on the volume pointed to by the catalog or specified on the request.

For a description of the return code, refer to *z/OS DFSMSdfp Advanced Services*.

OFFSET NOT ZERO

Explanation: The number specified after the destination name in an ISFNTBL macro is not 1. The number must be 1 in ISFNTBL macros that are named in the IDEST parameter.

User response: The system programmer should check the ISFNTBL macros named in the IDEST parameter of the ISFGRP macro.

OPERLOG NOT ACTIVE

Explanation: You entered the LOG O command but OPERLOG is not active on the system to which you are logged on. The OPERLOG panel is displayed, but may not contain messages from the system to which you are logged on.

User response: To see messages from the system to which you are logged on, type `LOG` or `LOG S`.

OPTION LOCALLY DISABLED

Explanation: The command or option has been disabled by the installation.

User response: If the command or option should be allowed, contact your system programmer to review the SDSF configuration options.

OPTS=mask REC-CNT=record-count DSNAME=data-set-name

Explanation: This message is issued to the message line in response to a TRACE command. *mask* is the event mask used for tracing; *record-count* indicates the number of records written to the trace data set; *data-set-name* is the name of the trace data set.

User response: None.

****** OS CVOL ERROR**

Explanation: This message accompanies the `ALLOC ERROR`*return-code error-code information code* message.

User response: None.

OUTADD ERROR return-code-reason-code

Explanation: An error occurred creating an output descriptor for the PRINT command. *return-code* is the decimal return code from the OUTADD macro, and *reason-code* is the hexadecimal reason code. The PRINT request is not executed.

User response: Use the return and reason codes to diagnose the error.

OUTPUT DESC NOT AVAIL return-code

Explanation: An error occurred trying to obtain the output descriptors for at least one data set being displayed on the JDS panel. The output descriptor fields begin with the PageDef column in the default field list (PageDef, FormDef, Title, Name, and so on) in the default field list. See "Job Data Set panel (JDS)" on page 151.

In the message text, *return-code* is a reason code describing the source of the error, as follows:

01 — SJF service error

02 — SWBIT block validation error

03 — SWBIT job or data set key validation error
04 — SWBIT read I/O error.

The output descriptors for the data set are not shown. If the reason code is 01, message ISF027I is also issued to further identify the data set and error that occurred.

User response: Contact your system programmer to determine the cause of the error.

OVERTYPE VALUE TOO LONG

Explanation: The value typed on an overtype extension pop-up is longer than the maximum width for the field.

User response: Correct the value.

---

number PAGES PRINTED

Explanation: In response to a PRINT command, *number* pages were printed.

User response: None.

PARM INVALID

Explanation: You entered a command with an invalid parameter, invalid printer name, or the parameter is not allowed in the current environment. The cursor is positioned under the parameter in error.

User response: Correct the invalid parameter.

PARM NOT ACCEPTABLE

Explanation: The command that was entered is not valid in the current environment. It may have been rejected because of a setting in the SDSF configuration parameters, ISFPARMS.

User response: Correct the invalid parameter.

PARTIAL DATA SHOWN

Explanation: While generating the PR panel, SDSF detected that printers were being added dynamically. SDSF was unable to build a complete printer list because the list exceeded a table retry limit. The printer list is incomplete.

User response: Refresh the PR panel after dynamic addition of printers is complete.

POINT ERROR RC=return-code

Explanation: The POINT request for the spool data for a job failed. The job's SYSOUT is not displayed. This may occur if the job was purged or if the SYSOUT data was selected from the Display Active Users (DA) panel and the job was swapped out.

User response: Try displaying the SYSOUT later. If the job was active and swapped out, the SYSOUT will be accessible. If the job was purged, the SYSOUT will not be found. For a description of the return codes, refer to *z/OS DFSMS Macro Instructions for Data Sets*.

---

number PREFIX string

Explanation: In response to a FIND command, a number of occurrences of a character string have been found. If SDSF finds more than 999999 occurrences, *number* is 999999+. The cursor is positioned on the character string.

User response: None.

PREFIX INVALID

Explanation: The PREFIX parameter was used with the FIND command on a panel other than the SYSLOG or ODS panel. The cursor is positioned on the character string.

User response: None.

PRINT ABEND abend-code

Explanation: An abend occurred during an SDSF print request. *abend-code* is the abend completion code in hexadecimal. The print operation is terminated and the print file is closed.

User response: Use the abend code to determine the reason for the abend. Additional explanatory messages might have been issued by the system to further describe the abend.

PRINT ALREADY OPEN

Explanation: An attempt has been made to open a previously opened print file.

User response: If a different print file is to be used, issue a PRINT CLOSE command to close the current file.

If the current print file is to be used, use the PRINT command or print action character (X) to print to the file.

PRINT CLOSED number LINE

Explanation: In response to a PRINT CLOSE command or a print action character, *number* lines were printed before the print file was closed.

User response: None.

PRINT ENDED — LOOP COND

Explanation: An attempt was made to print an open print data set. The data set was not printed. This error occurs if you are trying to print an active print file or trying to print the active SDSF trace data set.

User response: Data sets other than the open print

data set belonging to the user's TSO session can be printed individually from the JDS panel. Issue a PRINT CLOSE or TRACE OFF command before printing.

PRINT FILE ERROR

Explanation: The *ddname* you specified for printing cannot be found.

User response: Allocate a *ddname* and retry the request.

PRINT NOT OPENED

Explanation: A command requiring an open print data set was issued, but the print data set has not been opened.

User response: Issue either the PRINT OPEN or PRINT ODSN command to retry the request. For information on printing, see the online help.

PRINT OPEN ERROR

Explanation: The PRINT OPEN command or print action character failed.

User response: See the online help to diagnose the cause of error.

PRINT OPENED

Explanation: The print file has been successfully opened.

User response: None.

PRINT SCREEN UNAVAILABLE

Explanation: Another print job was in progress when you requested the print screen panel.

User response: Retry the command.

****** PRIVATE CATALOG ERROR**

Explanation: This message accompanies the `ALLOC ERROR`*return-code error-code information-code* or `LOCATE ERROR`*return-code* message, and explains why the allocation of the print file failed.

User response: Ensure that the data set used in the PRINT ODSN command is an existing data set.

PROFILE DESCRIPTIONS CREATED.

Explanation: The first step of the ISFPARMS-to-RACF conversion is complete. Profile descriptions have been created for the ISFPARMS.

User response: Review the profile descriptions for completeness and appropriateness. In particular, look for lines marked `CHANGE`. These lines need to be edited.

PROFILE DESCRIPTIONS DATA SET MUST BE ALLOCATED.

Explanation: The menu option that has been selected requires the profile description data set, but the data set has not been allocated. The data set is named on the conversion utility profile pop-up, which you display with option 1 of the conversion utility menu.

User response: Choose another menu option, or allocate the profile description data set. It must be a sequential file with record length of at least 80.

PROMPT NOT AVAILABLE

Explanation: The Prompt function is not available. It may have been disabled by the installation.

User response: None required. You can type the desired value in the field.

RACF COMMANDS CREATED.

Explanation: Creation of the RACF commands from profile descriptions is complete.

User response: Review the RACF commands for completeness and appropriateness. In particular, look for lines marked `CHANGE`. These lines need to be edited.

RACF COMMANDS DATA SET MUST BE ALLOCATED.

Explanation: The menu option that has been selected requires the RACF commands data set, but the data set has not been allocated. The data set is specified in the SDSF Security Assist profile.

User response: Choose another menu option, or allocate the RACF commands data set. It must be a sequential file with record length of at least 133.

number RECORDS SEARCHED

Explanation: A FIND command searched *number* SYSLOG or output data set records without finding the requested character string. The FIND ended before FINDLIM was reached.

User response: Use the Repeat-Find PF key or enter an F in the command input area to resume the search, or reset FINDLIM if authorized.

RESPONSE NOT RECEIVED

Explanation: The timeout interval has been reached before one ore more SDSF servers responded with data. The data on the SDSF panel is incomplete.

User response: To increase the timeout interval, use the SET TIMEOUT command or pull-down choice.

You might also try limiting the amount of sysplex data being returned, with one or more of the following:

- Parameters on the panel command, for example, `PR 1` to see only printer 1.
- The SYSNAME command or pull-down choice, to restrict the systems to be included.
- The DEST command or pull-down choice, to restrict the destinations to be included.
- The SELECT command, to temporarily restrict the panel based on the fixed field, for example, `SELECT PRT33` to see only printer PRT33.

Note that the Filter function does not have the effect of limiting the data returned

If the problem cannot be corrected with these methods, the operator or system programmer should ensure that one or more SDSF servers has not been stopped by issuing the `F server,D,C` command. The system programmer should also review the WebSphere MQ configuration for possible communications problems. Refer to "WebSphere MQ considerations" on page 344 for more information.

---

number RESPONSES NOT SHOWN

Explanation: An action character or slash command has been entered that results in messages being displayed on the screen, and the number of message responses received exceeds the screen depth. *number* message responses could not be shown.

User response: Enter the ULOG or LOG commands to view all of the message responses.

RMF EXIT NOT INSTALLED

Explanation: The SDSF-supplied RMF data reduction exit is not installed on all systems in the sysplex. RMF is installed and active, but the SDSF exit is not in the RMF steplib or accessible to it.

User response: Ensure that the exit is installed. Refer to "RMF considerations" on page 344 for information.

RMF LOCAL ERR returncode-reasoncode

Explanation: An error occurred during invocation of the RMF ERBSMFI Application Interface. *returncode-reasoncode* is the decimal return and reason code from the interface.

User response: Use the return code and reason code, along with the appropriate RMF documentation, to determine the cause of the error.

RMF NOT ENABLED

Explanation: An attempt was made to access the DA panel with RMF as the source of the data. RMF is not enabled on your system.

User response: None required. The DA panel is displayed with information derived from MVS control blocks rather than RMF. To request that DA use the MVS control blocks rather than RMF, and prevent display of this message, the installation can use the installation exit point of ISFUSER. For more information on the installation exit routines, refer to Chapter 9, "Using installation exit routines," on page 329.

RMF PLEX ERR returncode-reasoncode

Explanation: An error occurred during invocation of the RMF ERB2XDGS Application Interface. *returncode-reasoncode* is the decimal return and reason code from the interface.

User response: Use the return code and reason code, along with the appropriate RMF documentation, to determine the cause of the error.

You can bypass the problem by typing `SYSNAME` with no operands to see data for the local system.

RMF REQUIRED

Explanation: An attempt was made to access the DA panel when SDSF is processing JES3, and either RMF is not installed or is disabled. The command is rejected.

User response: None required.

RMF SYSPLEX NOT ACTIVE

Explanation: The RMF server is not active. Sysplex data cannot be obtained for the DA display.

User response: You can bypass the problem by typing `SYSNAME` with no operands to see data for the local system.

For information about the RMF server, see your system programmer.

SAPI ERROR returncode - reasoncode

Explanation: A problem was encountered related to the SYSOUT application programming interface (SAPI). The return code *returncode* is from the SSOBRETN feld and the reason code *reasoncode* is from the SSS2REAS field.

User response: For a description of the return code and reason code, see *z/OS MVS Using the Subsystem Interface*.

SCREEN DEFINITION ERROR

Explanation: Incorrect or invalid screen dimensions have been specified for SDSF running in batch. The dimensions are ignored.

Possible causes of this error are:
- Dimensions out of bounds
- Non-numeric dimensions
- Syntax error specifying parameter.

User response: Correct the screen dimensions and resubmit the SDSF job.

SCREEN IMAGE PRINTED

Explanation: The contents of the screen have been printed in response to an SDSF PRINT SCREEN command.

User response: None.

SDSF ABEND abend-code

Explanation: A recoverable abend occurred. *abend-code* is the abend completion code in hexadecimal. SDSF continues; some functions may not be available.

User response: Use the abend code and the dump to diagnose the problem.

SERVER NAME server-name TOO LONG

Explanation: The server name *server-name* specified on the SERVER parameter is longer than 8 characters.

User response: Correct *server-name*.

SERVER NOT COMPATIBLE

Explanation: The SDSF client attempted to connect to an SDSF server, but the level of the server is not compatible with the level of the client.

User response: Ensure the client is connecting to the correct server. To see the name of the server, issue the WHO command.

Refer to "Accessing the server" on page 109 for details on how SDSF selects a server for connection.

SERVER server-name NOTAVAIL

Explanation: SDSF was invoked using the SERVER keyword, but the named server is not available. SDSF continues execution using the parameters from the ISFPARMS in assembler macro format.

User response: Ensure that the named server is running and that the ISFPARMS statements have been activated.

SET COMMAND COMPLETE

Explanation: The user issued the SET command and it has been completed successfully.

User response: None.

SET SCREEN FAILED function code

Explanation: SDSF has received an error from the ISPF dialog manager. *function* is a number indicating the ISPF dialog function that failed. The numbers and the functions they represent are:

01 — VDEFINE
02 — VGET
03 — DISPLAY
04 — VPUT
05 — VCOPY
06 — ADDPOP
07 — VREPLACE

code is the return code from the failing function. Refer to *z/OS ISPF Dialog Developer's Guide and Reference* or *z/OS ISPF Services Guide* for the meaning of the return code.

User response: The system programmer should correct the error with the ISPF function.

SOCKET NOT AVAILABLE

Explanation: An action or overtype requires a socket to be associated with the object. However, no socket is associated with the object

User response: Remove the action character or modification from the panel by restoring or blanking the field, or type the RESET command.

SORT COLUMN NOT FOUND

Explanation: A SORT command was entered specifying a column name that does not exist for this panel. The cursor is positioned under the column name that was not recognized.

User response: Correct the column name and reenter the command.

SORT COLUMN NOT UNIQUE

Explanation: A SORT command was entered using an abbreviated column name that does not uniquely identify one column in the panel. The cursor is positioned under the column name in error.

User response: Reenter the command specifying a unique abbreviation or a full column name.

SORT COLUMN REPEATED

Explanation: In a sort request, a column was specified more than once.

User response: Correct the sort request so that no column is specified more than once.

SORT CRITERIA OBSOLETE

Explanation: During the current SDSF session, this is the first display of this panel. This first display uses sort criteria saved from a previous session. One or more of the saved criteria specify a column name that has been removed from the ISFPARMS definition of this panel. A column might have been removed because of security changes, release migration, or customization

of the installation supplied field lists.

The obsolete criteria are deleted. If there are any valid sort criteria, the panel is sorted using only the valid criteria.

An additional message, INVALID COLUMN, is displayed in the message line and indicates the column name that no longer exists.

User response: No action is required. A new SORT command can be issued to establish new sort criteria. See the additional message in the message line for more information.

SORT ORDER NOT A OR D

Explanation: A SORT command was entered, but the sort order specified is not A (for ascending sort) or D (for descending sort). The cursor is positioned under the operand in error.

User response: Correct the command and reenter it.

SPOOL DATA ERROR

Explanation: The spool data for a job became invalid while the job's SYSOUT data was being displayed. This might occur if the job was purged or if the SYSOUT data was selected from the DA panel and the job was swapped out.

User response: Try displaying the SYSOUT later. If the job was active and swapped out, the SYSOUT is accessible. If the job was purged, the SYSOUT will not be found.

SPOOL RECORD ERROR

Explanation: A spool record in a SYSLOG data set became invalid. This usually means that the SYSLOG data that was being displayed just finished printing.

User response: Leave the SYSLOG panel (with the `END` command or PF key) and then reaccess it (with `LOG`).

SRVCLASS NAME INVALID

Explanation: The value entered for a service class was rejected by the WLM programmable service IWMERES.

User response: Refer to *z/OS MVS Programming: Workload Management Services* for more information about service classes.

SSI 82 ERR returncode - reasoncode

Explanation: A problem was encountered retrieving data from SSI 82. The return code is from the SSOBRETN field and the reason code is from the SSJPRETN field.

User response: For a description of the return and reason code, see *z/OS MVS Using the Subsystem Interface*.

SSI RETURN CODE return-code

Explanation: A subsystem interface (SSI) return code of *return-code* was issued when a user tried to requeue an output group from the H panel or the JDS panel or tried to overtype a field on the OD panel.

User response: The system programmer should see one of the following return codes:

4 — Subsystem does not support this function

8 — Subsystem exists but is not up

12 — Subsystem does not exist

16 — Function not completed

20 — Logical error.

SSOB RETURN CODE return-code

Explanation: An SSOB return code of *return-code* was issued when a user tried to requeue an output group from the H panel or the JDS panel.

User response: The system programmer should see one of the following return codes:

4 — No more data sets to select

8 — Job not found

12 — Invalid search arguments

16 — Unable to process now

20 — Duplicate job names

24 — Invalid combination of job name and job ID

28 — Invalid destination specified.

STEP NAME NOT AVAILABLE

Explanation: The user is trying to reset the performance group number for a started task and the step name is unavailable.

User response: None.

SUBS RETURN CODE return-code

Explanation: SDSF hThanks, Billas issued a return code of *return-code*.

User response: The system programmer should refer to the return code for a description of the error. The return codes are:

4 — Bad option passed

8 — Not in an authorized state

12 — Different JES system

16 — Requested address space identifier not valid

20 — Requested address space identifier not a TSO user

24 — JES not active

28 — Bad job key

32 — SRB abend

36 — Parameter invalid

40 — User swapped out

48 — Abend processing parameter

52 — Bad data set key

56 — Bad member-track-track-record (MTTR).

If SUBS RETURN CODE 56 appears randomly on the log, and disappears when the user presses Enter, and if the system has a high paging rate, the message might indicate a timing exposure. Press Enter when the message appears.

60 — Buffer full

64 — GETMAIN failed

68 — User canceled

72 — Attention key pressed

76 — Cross-memory not active

80 — Bad application copy error

84 — Application copy level error

88 — Application copy update error

92 — Application copy no longer available

96 — ECSA application copy no longer available

100 — Invalid spool data set name call

104 — Buffer size invalid

108 — Dynamic printer addition overflow

112 — JQE no longer valid

116 — SJB/SDB invalid.

120 — Checkpoint version error

124 — Subsystem not defined

128 — Invalid buffer header

132 — Unable to obtain printer data

---

number SUFFIX 'string'

Explanation: In response to a FIND ALL command, *number* occurrences of a character string have been found. If SDSF finds more than 999,999 occurrences, *number* is 999999+. The cursor is positioned on the character string.

User response: None.

SUFFIX INVALID

Explanation: The SUFFIX parameter was used with the FIND command on a panel other than the logs or ODS panels.

User response: Correct the command and reissue it.

SWB ERROR nnnn-rea1-rea2

Explanation: An error occurred issuing a SWB modify request. In the message text, *nnnn* is the decimal return code from the SWB modify request. *rea1* and *rea2* are the decimal reason codes.

User response: Attempt to reissue the modify request. If the error persists, contact your system programmer for assistance.

---

field-name SYNTAX ERROR

Explanation: An output descriptor has been overtyped, but SJF has detected a syntax error in the input for the *field-name* keyword. The variable *field-name* is the name of the output descriptor and might not necessarily be the same as the field title shown on the display.

User response: Correct the overtype.

SYSOUT NOT FOUND

Explanation: An attempt to work with SYSOUT was rejected by the subsystem interface (SSI).

User response: Try the request again.

SYSOUT REQUEUED

Explanation: In response to your request, SYSOUT has been requeued or purged.

User response: None.

---

number SYSOUT REQUEUED | PURGED

Explanation: In response to your request, *number* SYSOUT data sets have been requeued or purged.

User response: None.

SYSPLEX DA NOT AVAIL

Explanation: You requested a sysplex-wide DA display, but either the RMF ERB2XDGS interface could not be loaded, or the installation has disabled the use of RMF for the DA display.

User response: No action is required. For information about the RMF server, see your system programmer.

SYSTEM BUSY, RETRY

Explanation: SDSF was unable to gather the data for a panel because a required system was busy.

User response: Refresh the panel by pressing Enter. If the problem persists, follow your local procedure for contacting IBM for service.

SYSTEM MESSAGES NOTAVAIL

Explanation: An error occurred initializing the Consoles query environment. WTORs and AMRF queue entries will not be displayed on the SR panel or the LOG panel.

User response: See your system programmer. SDSF may have previously issued a message describing the error.

SYSTEM NOT CONNECTED

Explanation: A command has been issued for a member of the MAS, but the command must be routed to the system and the system is not accessible.

User response: Retry the command when the system is connected.

TEMP FILE ALLOC FAILED

Explanation: An error occurred attempting to allocate the temporary file required by the GDDM view utility. The request to view a data set is ended.

User response: See the accompanying explanatory system message describing the error.

TEMP FILE OPEN FAILED reason-code

Explanation: An error occurred in the attempt to open the temporary file required by the GDDM view utility. The request to view a data set is ended. *reason-code* is one of the following:

01 — SDSF was unable to open the temporary file DCB. Accompanying messages can further describe the error.

02 — The block size of the temporary file exceeded the capacity of the DASD device on which it is allocated.

User response: Determine the reason for the failure and retry the view request. If *reason-code* is 02, the system programmer should change the unit name for the temporary file (defined by the VIO keyword in the ISFGRP macro of ISFPARMS) to a device capable of holding a copy of the page-mode data to be composed.

TOO FEW PARMS

Explanation: There were not enough parameters specified on the command. SDSF does not process the command.

User response: Correct the command and retry the request.

TOO MANY COLUMNS SELECTED

Explanation: You have selected too many columns or blocks on the pop-up.

User response: Correct the selection. For ARRANGE, you can select one column.

TOO MANY DEST NAMES

Explanation: More than four destination names were specified in an ISFNTBL macro or NTBL statement that is named in the IDEST parameter of the user's ISFGRP macro or GROUP statement.

No more than four destination names can be specified in an ISFNTBL macro or NTBL statement that is named in the IDEST parameter of the ISFGRP macro or GROUP statement.

User response: The system programmer should correct ISFPARMS. The user should correct or delete the DEST command so the maximum number is not exceeded.

TOO MANY FILTERS

Explanation: An attempt was made to enter more filters than are allowed. The maximum number of filters is 25.

User response: Delete the command. You can remove a filter with `FILTER -column`. Under ISPF, you can use `FILTER ?` to display the pop-up, which allows you to modify filters, or delete them by blanking them out.

TOO MANY PARAMETERS

Explanation: Too many parameters were specified with a command.

User response: Correct or delete the command.

TOO MANY PARMS

Explanation: Too many parameters were specified with a command.

User response: Correct or delete the command.

TOO MANY COLUMNS SELECTED

Explanation: You have selected too many columns or blocks on the pop-up.

User response: Correct the selection. For ARRANGE, you can select one column.

*** TOP OF DATA REACHED ***

Explanation: A FIND PREV or FIND FIRST command reached the top of the data without finding the requested character string.

User response: Use the Repeat-Find PF key or enter an F in the command input area to resume the search at the bottom of the data.

TRACE DCB ALREADY CLOSED

Explanation: A TRACE OFF command was entered, but the ISFTRACE file has already been closed. The TRACE OFF command is ignored.

User response: None.

TRACE DCB ALREADY OPENED

Explanation: A TRACE ON command was entered, but the ISFTRACE file has already been opened. The TRACE ON command is ignored.

User response: None.

TRACE DCB CLOSED

Explanation: In response to a TRACE OFF command, the ISFTRACE file has been closed.

User response: None.

TRACE DCB OPENED

Explanation: In response to a TRACE ON command, the ISFTRACE file has been opened.

User response: None.

TRACE NOT AVAILABLE

Explanation: SDSF is operating in split-screen mode, and the trace facility is not available in the session in which the message was issued. The trace facility is available in the other session.

User response: To use the trace facility, swap sessions.

TRACE OFF - ABEND abend-code

Explanation: An I/O error has caused SDSF to turn tracing off. A system abend with an abend code of *abend-code* has occurred but has been handled by SDSF.

User response: To continue tracing, allocate a new trace data set. For more information on the abend, see the appropriate system codes manual.

TRACE OFF - PERM I/O ERR

Explanation: An I/O error has caused SDSF to turn tracing off.

User response: To continue tracing, allocate a new trace data set.

TRACING IS ON|OFF

Explanation: In response to a TRACE command, the status of tracing is shown to be on or off.

User response: None.

TYPE A COLUMN NAME

Explanation: You left a field requiring a column name blank.

User response: Type a valid column name in the field.

TYPE A NUMBER IN THIS FIELD

Explanation: You typed data that was not numeric in a numeric field, or there are blanks in the numeric field. The cursor is positioned on the field in error.

User response: Enter numeric data in the field.

TYPE A OR D FOR SORT ORDER

Explanation: You typed something other than an A, D, or a blank on the Sort pop-up. The valid values are A (for ascending) or D (for descending). If the character is blank, the order is ascending.

User response: Type an A or D or blank out the character.

TYPE LINES OR TIMES AND DATES

Explanation: You pressed Enter on a Print pop-up but didn't specify either lines or times and dates to print.

User response: Type values for either lines or times and dates.

ULOG CLOSED

Explanation: A ULOG CLOSE command was issued and the user log has been successfully closed. All message responses have been deleted from the user log and the extended console has been deactivated.

User response: None.

UNABLE TO FIND ORIGINAL

Explanation: The user attempted an action on a foreign, independent enclave, but the corresponding original enclave could not be found. The original enclave may have terminated before the action was attempted.

User response: None.

UNABLE TO FIND OWNER

Explanation: The user attempted an action on a dependent enclave, but the owning address space could not be found. The owning address space may have ended before the action was attempted, or may be running on a system that does not support the Enclave Reset function.

User response: None.

UNBALANCED PARENTHESIS

Explanation: In attempting to overtype a field, the user has omitted a required parenthesis.

User response: Enter the required parenthesis.

UNBALANCED QUOTES

Explanation: An ending quotation mark is either missing or you have an extra quote at the end.

User response: Correct the quote marks or enter a new string.

UPDATE LENGTH TOO LONG

Explanation: The update interval entered with the & command is longer than three digits.

User response: Retry the & command with an interval of 999 or less.

UPDATE NOT AUTHORIZED

Explanation: You have attempted to issue the & command to enter automatic update mode, but are not authorized to do so.

User response: Delete the & command.

If you have been denied authorization in error, see "User authorization" on page 475 for more information.

UPDATE TIME TOO SMALL

Explanation: The user has issued the & command to enter automatic update mode, but the update interval specified was less than the installation-defined minimum.

User response: Retry the & command with a larger interval.

USE EQ,NE WITH PATTERNS

Explanation: You specified an operator with less than or greater than and the value contained pattern matching.

User response: Change the operator to EQ or NE, or remove the pattern matching.

USE EQ OR NE WHEN THE FILTER VALUE INCLUDES PATTERN MATCHING

Explanation: You specified an operator with less than or greater than and the value contained pattern matching.

User response: Change the operator to EQ or NE, or remove the pattern matching.

VALUE NOT AUTHORIZED

Explanation: The value that was specified in an overtypeable field was rejected by SAF security. The value is ignored.

User response: None required. You can overtype the field with a different value. If the value should be allowed, contact your security administrator.

VALUE TOO LONG

Explanation: An attempt was made to add a value that was selected from a list to existing text. The resulting combination was too long for the field. As a result, the existing text was not changed.

User response: None required. You might change or delete the existing text and then try selecting a value from the list again.

****** VOLUME NOT MOUNTED**

Explanation: This message accompanies message ALLOC ERROR *return-code error-code information-code* or OBTAIN ERROR *return-code* and explains why allocation of the print file failed.

User response: Ensure that the PRINT ODSN command is issued using a valid existing data set.

WIDTH CANNOT EXCEED maximum

Explanation: The column width specified with the Arrange function is longer than the maximum allowed, which is *maximum*.

User response: Change the width to a number that is valid.

number WORD 'string'

Explanation: In response to a FIND ALL command, *number* occurrences of a character string have been found. If SDSF finds more than 999,999 occurrences, *number* is 999999+. The cursor is positioned on the character string.

User response: None.

WORD INVALID

Explanation: The WORD parameter was used with the FIND command on a panel other than the logs or ODS panels.

User response: None.

Messages with HSF message numbers

This section describes messages issued with HSF message numbers.

A letter following the message number indicates the severity of the message:

I Information.

W Warning.

E Error.

HSF0001I Server initializing

Explanation: The SDSFAUX server is initializing. This message is issued when the SDSFAUX server starts the SDSFAUX address space.

The SDSFAUX address space provides data collection services used by various SDSF commands and displays.

User response: No response is required.

HSF0002I Server initialization complete.

Explanation: SDSFAUX server initialization is complete. This message indicates that the SDSFAUX server has finished initializing and is ready to accept requests from SDSF users.

The SDSFAUX address space provides data collection services used by various SDSF commands and displays.

User response: No response is required.

HSF0003E Connect failed. RC=*return-code* **RSN=***reason*

Explanation: The connection request to the SDSFAUX server has failed for the indicated return and reason codes.

The SDSFAUX services are unavailable to the caller.

User response: Verify that the SDSFAUX server is active and that the caller has the required security access.

HSF0004E Cross-system resource *group* **version mismatch with** *member*

Explanation: The SDSFAUX server has detected an unsupported version of SDSF on the specified member and has stopped its XCF data collection agent.

SDSFAUX cannot share XCF resources with an unsupported release of SDSF.

User response: Update to a supported release of SDSF on the member listed.

HSF0005E SDSFAUX server is already active on this system.

Explanation: An attempt has been made to start the SDSFAUX server, which was already active on the system.

The SDSFAUX server attempting to start will stop.

There must only be one SDSFAUX server active at any one time.

User response: Before you restart the SDSFAUX or SDSF server, stop the current instance and ensure SDSFAUX is inactive.

HSF0006E Operating system level not supported.

Explanation: An attempt has been made to start the SDSFAUX server on a system that is running an unsupported version of the operating system.

The SDSFAUX server will stop.

User response: Upgrade to a supported release of the operating system.

HSF0007I Joined data-sharing group *name* **as** *member***.**

Explanation: The SDSFAUX server has successfully joined the indicated XCF group. The server will use this XCF group to perform cross-system data gathering requests.

User response: No response is required.

HSF0009E Incorrect execution key.

Explanation: The SDSFAUX server cannot start because the execution key of the HSFSRV00 program did not match the IBM value of 4.

The SDSFAUX server will not start.

User response: Verify that all required maintenance has been applied for SDSF and confirm that there are no modifications to the SCHED*xx* PARMLIB members that override the IBM PPT entry for HSFSRV00.

HSF0010I **Module** *name* **loaded successfully at address** *hex***.**

Explanation: The SDSFAUX server successfully loaded the indicated module at the specified address.

This message appears only in the HSFLOG output.

User response: No response is required.

HSF0011I **Queue recovery for** *jobname* **ASCB(** *ascb* **) TCB(** *tcb* **) RB(** *rb* **)**

Explanation: The SDSFAUX server has attempted to recover a pending request for the indicated unit of work. The requestor's ASCB, TCB and RB addresses are listed.

This message is issued when there are problems with the task that owns the request queue in the SDSFAUX server. Typically there was an abend or server error when there were active requests.

This message appears only in the HSFLOG output.

User response: The requesting unit of work will be resumed with an appropriate return and reason code.

HSF0020I **Command entered:** *command*

Explanation: The SDSFAUX server has received the specified operator command.

User response: No response is required.

HSF0025E **Unknown operation**

Explanation: The SDSFAUX server has received an unknown operator command. Only DISPLAY and MODIFY operations are supported.

User response: Issue a supported operator command.

HSF0026I **Command accepted:** *text*

Explanation: The SDSFAUX server has accepted the specified operator command.

User response: No response is required.

HSF0027E **Invalid command :** *text*

Explanation: The SDSFAUX server has rejected the specified operator command because it is unrecognized or contains invalid syntax.

User response: Examine related messages and correct the operator command.

HSF0028W **RMF data collection failed ERBSMFI RC=***rc* **RSN=***rsn*

Explanation: The SDSF data collection task received a non-zero return code and reason code from the RMF interface program ERBSMFI. Any SDSF commands that depend on the data collected by this RMF interface program will not be able to show any results.

User response: Ensure that RMF Monitor I has been started and that the ERBSMFI program is available to SDSFAUX.

HSF0030W **Critical error in data collection for** *name*

Explanation: The named task has encountered a non-recoverable error during data collection. Any SDSF commands that depend on the data collected by this task will not be able to show any results.

User response: Look for any other earlier error messages issued by this task to determine the root cause of the problem.

HSF0031I **Keyword** *keyword* **updated with new value** *value*

Explanation: The SDSFAUX server has refreshed the specified keyword with the new value.

User response: No response is required.

HSF0032W **Internal resource shortage** *type* **:** *percent*

Explanation: The SDSFAUX server has detected an internal resource shortage of the specified type. The percentage of the maximum limit for the resource type is listed.

Known types:

- PRV-STOR : Private storage below 16Mb
- EPRV-STOR : Private storage above 16Mb

User response: Examine the resource type to see if there is an underlying issue that is causing the shortage.

HSF0033I **Internal resource shortage relieved for** *type*

Explanation: The SDSFAUX server internal resource shortage of the indicated type has been relieved.

Known types:

- PRV-STOR : Private storage below 16Mb
- EPRV-STOR : Private storage above 16Mb

User response: No response is required.

HSF0034I **Task** *name* **terminated RC=** *rc*

Explanation: The SDSFAUX server task has terminated with the specified return code.

This message is written to the HSFLOG output.

User response: No response is required.

HSF0035W **SAF Class SDSF not active RC=** *rc* **RSN=** *rsn*

Explanation: The SDSF SAF class is required for the SDSFAUX server to protect its services. A RACROUTE REQUEST=STAT service for the class has responded with the specified return and reason code.

All protected services will return a SAF "No Decision" return code.

User response: Activate the SDSF SAF class and define the required profiles to protect the SDSFAUX services.

For more information see Chapter 5, “Using SAF for security,” on page 191.

HSF0036I **Task** *name* **initialization complete**

Explanation: The SDSFAUX server task successfully initialized.

This message is written to the HSFLOG output.

User response: No response is required.

HSF0037W **SAF Class SDSF not RACLISTed**

Explanation: The SDSF SAF class is not RACLISTed. The SDSFAUX server uses RACROUTE REQUEST=FASTAUTH to verify access to its services, and therefore, must have the SDSF class RACLISTed.

All protected services will return a SAF "No Decision" return code.

User response: RACLIST the SDSF class so that the SDSFAUX server can use the RACROUTE REQUEST=FASTAUTH service.

For more information see Chapter 6, “SDSF and RACF,” on page 205.

HSF0040I **ENF listener** *name* **installed for event** *code*

Explanation: The SDSFAUX server has successfully installed the specified module as an ENF listener for the event code.

This message appears only in the HSFLOG output.

User response: No response is required.

HSF0041I **ENF listener** *name* **delete for event** *code* **RC=** *rc*

Explanation: The SDSFAUX server has attempted to delete the specified module from the ENF listeners for the event code.

This message appears only in the HSFLOG output.

User response: If the return code is non-zero, contact IBM Software Support.

HSF0042E **ENF listener install for** *name* **event code** *num* **failed RC=** *rc*

Explanation: The SDSFAUX server has attempted to install the specified module as an ENF listener for the event code, and the operation has failed with the indicated return code.

User response: Contact IBM Software Support.

HSF0044E **Command** *name* **install failed RC=** *RC* **RSN=** *rsn*

Explanation: The SDSFAUX server has attempted to install the specified command and the operation has failed with the indicated return and reason code.

The command and its associated data gathering service will be unavailable.

User response: Contact IBM Software Support.

HSF0045I **Command** *name* **installed successfully**

Explanation: The SDSFAUX server has successfully installed the specified command.

This command and its associated data gathering service will be available.

This message appears only in the HSFLOG output.

User response: No response is required.

HSF0047I **Left data-sharing group** *name*

Explanation: The SDSFAUX server has left its data-sharing group.

All cross-system services for this SDSFAUX server are now marked unavailable.

This message appears only in the HSFLOG output.

User response: No response is required.

HSF0048I **No active users**

Explanation: During shutdown, the SDSFAUX server determined that there are no connected users. Shutdown will proceed without delay.

User response: No response is required.

HSF0049E Required SDSF server not active

Explanation: During startup the SDSFAUX server has determined that the SDSF server is not active.

The SDSFAUX server will stop.

User response: The SDSFAUX server is typically started automatically by the SDSF server. Restart the SDSF server.

HSF0050I Sectoken \ *userid lvl* access to *name* class profile *res*

Explanation: This message appears in the HSFTRACE output when the SDSFAUX security trace is active.

The userid has requested the indicated level of access to the SAF class profile.

The result of this access request will be described by a subsequent HSF0061I message that uses the same sectoken value.

User response: No response is required.

HSF0051I SDSFAUX RESPONSE IN PROGRESS / RESPONSE COMPLETE *Sysname JES Version Status*

Explanation: This message is produced in response to the SDSFAUX DISPLAY JES operator command.

The "RESPONSE IN PROGRESS" message will be followed by a list of the systems, JES subsystems and versions that are known by the SDSFAUX server.

After all responses are sent, the "RESPONSE COMPLETE" message is issued.

User response: No response is required.

HSF0052I SDSFAUX RESPONSE IN PROGRESS / RESPONSE COMPLETE *Jobname ASID TCB Connect UCON*

Explanation: This message is produced in response to the SDSFAUX DISPLAY USER operator command.

A "RESPONSE IN PROGRESS" message will be followed by a list of the active SDSFAUX users and their connect date stamps.

After all responses are sent, the "RESPONSE COMPLETE" message is issued.

User response: No response is required.

HSF0053I SDSFAUX RESPONSE IN PROGRESS / RESPONSE COMPLETE Task*TCB RXTA Flag Samples CPU*

Explanation: This message is produced in response to the SDSFAUX DISPLAY TASK operator command.

A "RESPONSE IN PROGRESS" message will be followed by a list of the active SDSFAUX tasks and their resource consumption.

After all responses are sent, the "RESPONSE COMPLETE" message is issued..

User response: No response is required.

HSF0054I SDSFAUX RESPONSE IN PROGRESS / RESPONSE COMPLETE *Name Active Get Free Lost RXBP*

Explanation: This message is produced in response to the SDSFAUX DISPLAY BPOOL operator command.

A "RESPONSE IN PROGRESS" message will be followed by a list of the SDSFAUX buffer pools.

After all responses are sent, the "RESPONSE COMPLETE" message is issued.

User response: No response is required.

HSF0056I SDSFAUX RESPONSE IN PROGRESS / RESPONSE COMPLETE *Name EPA Invoke Normal Return Abend*

Explanation: This message is produced in response to the SDSFAUX DISPLAY EXIT operator command.

A "RESPONSE IN PROGRESS" message will be followed by a list of the system exits installed by SDSFAUX.

After all responses are sent, the "RESPONSE COMPLETE" message is issued.

User response: No response is required.

HSF0057I SDSFAUX RESPONSE IN PROGRESS / RESPONSE COMPLETE Name *Jobname TCB CPU-SRB CPU-TCB*

Explanation: This message is produced in response to the SDSFAUX DISPLAY ZIIP operator command.

A "RESPONSE IN PROGRESS" message will be followed by a list of the zIIP offload environments managed by SDSFAUX.

After all responses are sent, the "RESPONSE COMPLETE" message is issued.

User response: No response is required.

HSF0061I Sectoken *token* **SAF RC=** *safrc* **RACF RC=** *rc* **RACF RSN=** *rsn*

Explanation: This message appears in the HSFTRACE output when the SDSFAUX security trace is active.

This trace message qualifies an earlier HSF0050I message with the same internal sectoken value. The HSF0050I message will describe the access request details.

The message specifies the SAF return code and the RACF return and reason codes from the RACROUTE REQUEST=FASTAUTH service.

User response: No response is required.

HSF0062I Server shutdown waiting for users to disconnect

Explanation: During shutdown, the SDSFAUX server will wait for connected users to gracefully disconnect before shutdown proceeds.

The SDSFAUX server lists any connected users in a HSF0052I message.

The SDSFAUX waits 60 seconds for users to disconnect and then shuts down.

User response: No response is required.

HSF0064E Service *name* **failed RC=** *rc* **RSN=** *rsn*

Explanation: The named service failed with the specified return and reason code.

This is a generic message that is used to present non-zero return codes from both internal SDSF services and other external programs and interfaces.

User response: When the service name is clear, refer to the return and reason codes in the appropriate manual for the owning software product.

If the cause is unclear, contact IBM Software Support.

HSF0067E CSVDYLPA add for module *name* **failed RC=** *rc* **RSN=** *rsn* **DIAG=** *code*

Explanation: The SDSFAUX server failed to dynamically add the specified module into LPA.

After this error, the SDSFAUX server issues a user abend and stops.

User response: Refer to the return and reason codes for the CSVDYLPA service in *z/OS MVS Programming: Authorized Assembler Services Reference LLA-SDU*.

If the cause is unclear, contact IBM Software Support.

HSF0072I Server shutdown proceeding

Explanation: During shutdown processing, SDSF has determined that no users are connected or that the time allowed for users to disconnect has been exceeded.

Shutdown processing continues and any user who is still connected will receive an error response when they resume processing.

User response: No response is required.

HSF00741I CSVDYLPA delete for *type* **module** *name* **RC=** *rc* **RSN=** *rsn*

Explanation: The SDSFAUX server attempted to delete the specified module from LPA and it completed with the indicated return and reason code.

User response: If the return code is non-zero, refer to the return and reason code descriptions for the CSVDYLPA service in *z/OS MVS Programming: Authorized Assembler Services Reference LLA-SDU*.

If the cause is unclear, contact IBM Software Support.

HSF0078W RMF Monitor I not active – some data may not be available

Explanation: The SDSF server has detected that RMF Monitor I is not active. Any SDSF commands that depend on the data collected by RMF will not be able to show any results.

User response: Ensure that RMF Monitor I has been started and that the ERBSMFI program is available to SDSFAUX.

HSF0080I Event : *text*

Explanation: The SDSFAUX server is logging the occurrence of a specific event in the HSFLOG output for diagnostic purposes.

User response: No response is required.

Messages with ISF message numbers

This section describes messages issued by SDSF with ISF message numbers.

A letter following the message number indicates the severity of the message:

I Information.

W Warning. The command will be processed, or the ISFPARMS will be activated. For ISFPARMS, SDSF has found an inconsistency and may have changed a value for a parameter.

E Error. A command will not be processed, or the ISFPARMS will not be activated.

ISF001I ERROR PROCESSING SYSLOG *jobid*

Explanation: An error occurred while reading the spool control blocks for the SYSLOG job *jobid*. The SYSLOG data set has an invalid control block structure. This might be because SDSF cannot access a SYSLOG data set, either because the dataset contains an error or because the data set is not a spin data set.

User response: To determine whether the problem is with the SYSLOG data set, use SDSF to locate the data set and attempt to browse it. If SDSF cannot browse the data set, the data set contains an error and should be purged from the spool.

ISF002I MASTER SYSLOG INDEX FULL

Explanation: The SYSLOG index is full because the block size of the HASPINDX data set is too small.

User response: Either increase the block size of the HASPINDX data set, or delete some SYSLOG data sets from spool. For more information on the HASPINDX data set, refer to "SYSLOG" on page 349.

ISF003I NEEDED SPOOL VOLUME NOT OPEN

Explanation: A spool volume required by SDSF cannot be opened. This might be due to an invalid spool pointer, an allocation failure, or an unavailable spool volume. Additional messages might have been issued by the system to describe the error.

User response: If an allocation error occurred, use the system messages to determine the cause of the error.

If the spool volume is not available, contact the system operator to mount the spool volume.

ISF004I I/O ERROR ON SPOOL READ

Explanation: An I/O error occurred while attempting to read a spool record. This can be caused by a logic error in SDSF or might be due to a control block error in a job being processed.

User response: If the problem is with a job that is being processed, the operator can use a filter command to identify which job is causing the problem.

For example, the operator brings up an SDSF panel with these jobs:

```
ABLEJOB
ABLEBJOB
ANDJOB
BJOB
BBBJOB
CJOB
```

The operator issues `PREFIX A*`, and the panel shows these jobs:

```
ABLEJOB
ABLEBJOB
ANDJOB
```

The error message still appears on the panel, so the problem is with one of the three jobs shown. The operator then issues a second PREFIX command, `PREFIX ABLE*`. The panel then shows:

```
ABLEJOB
ABLEBJOB
```

The error message no longer appears on the panel. The operator knows that the problem is not with ABLEJOB or ABLEBJOB; the problem must be with ANDJOB.

When the job that is causing the error has been identified, try processing the job without SDSF. If JES2 cannot process the job successfully, the error might be with the job and not SDSF.

ISF005I INVALID IDEST FOR *userid entry reason*

Explanation: During initialization for *userid*, SDSF found an error processing *entry* in the ISFNTBL macro named in the IDEST parameter of the ISFGRP macro. The ISFGRP macro is in the ISFPARMS module.

The values for *reason* are:

`INVALID CALL`
means that a logic error exists in SDSF. Follow your local procedure for calling IBM. Have the following documentation of the problem ready:

- A description of the panel being used and the operation being performed when the message was received

- A record of the message

INVALID DEST
means that the destination name is invalid for this system. If the name is an installation-defined name, the error could be caused by the JES system not being active during the installation of SDSF.

NAME NOT AUTH
At SDSF initialization, SDSF found the user was not authorized to access one or more destination names specified in the ISFNTBL macro for the IDEST parameter in the user's ISFGRP macro. If both the IDEST and DEST parameters are coded, the destination names in the IDEST ISFNTBL macro must also be in the DEST ISFNTBL macro in order for the user to be authorized.

If this is not the problem, a logic error might exist in SDSF. Follow your local procedure for calling IBM and have the following documentation of the problem ready:

- A description of the panel being used and the operation being performed when the message was received
- A record of the message

nnnn **NOT SPECIFIED**
During SDSF initialization or DEST command processing, SDSF did not find any authorized destination names. The user is not authorized to access all destinations, therefore, a valid authorized destination list is required. *nnnn* is the number of destinations.

This message also appears in response to a destination query command (DEST ?) if no destination names are authorized.

The system programmer or security administrator should either add an IDEST parameter to the user's ISFGRP macro, or authorize the user to access the ISFOPER.ANYDEST.*jesx* resource. If these conditions are not met, the user's destination filter is set to blanks or the character string QQQQ, and no jobs appear on the panels.

OFFSET NOT ZERO
means that the number specified after the destination name in the ISFNTBL macro is not 1. This number must be 1 in ISFNTBL macros that are named in the IDEST parameter.

TOO MANY DESTS
means that more than four destination names were specified. No more than four destination names can be specified in ISFNTBL macros that are named in the IDEST parameter.

User response: The system programmer should check the ISFNTBL macros named in the IDEST parameter of the user's ISFGRP macro. The ISFGRP macro is described in "Group authorization parameters (GROUP or ISFGRP)" on page 36.

The system programmer might also want to put the installation-defined names last in the ISFNTBL macros, as the installation-defined names can be the most likely to cause an error. When SDSF encounters an error in the destination names during initialization, it continues initialization with the destination names that were successfully processed before the error.

ISF006I ERROR PROCESSING INITIAL CHECKPOINT REQUEST FOR SUBSYSTEM *subsystem-name*, **CODE=***error-code*, **REASON=***reason-code*

Explanation: An error occurred during SDSF initialization attempting to obtain checkpoint data from *subsystem-name*. The *error-code* contains the reason for the failure and is listed below. If the error occurred processing a checkpoint version, *reason-code* indicates the return code (SSJIRETN) from the checkpoint version obtain request.

User response: Use the return and reason codes to diagnose the error.

4 — Bad option passed

8 — Not in an authorized state

12 — Different JES system

16 — Requested address space identifier not valid

20 — Requested address space identifier not a TSO user

24 — JES not active

28 — Bad job key

32 — SRB abend

36 — Parameter invalid

40 — User swapped out

48 — Abend processing parameter

52 — Bad data set key

56 — Bad member-track-track-record (MTTR)

60 — Buffer full

64 — GETMAIN failed

68 — User canceled

72 — Attention key pressed

76 — Cross-memory not active

80 — Bad application copy error

84 — Application copy level error

88 — Application copy update error

92 — Application copy no longer available

96 — ECSA application copy no longer available

100 — Invalid spool data set name call

104 — Buffer size invalid

108 — Dynamic printer definition overflow

112 — JQE no longer valid

116 — SJB/SDB invalid.

120 — Checkpoint version error

124 — Subsystem not defined

ISF007I **CAN NOT FIND MASTER CHECKPOINT RECORD**

Explanation: The master checkpoint record failed a validity check. This could be because:

- Maintenance was applied to JES2, and SDSF was not reassembled.
- During the assembly of SDSF, a SYS1.HASPSRC data set used in the SYSLIB concatenation does not match the JES2 system being processed.
- SMP/E was used to install SDSF, and maintenance was applied to JES2 and not accepted, and the SMPMTS data set is not the first data set in the SYSLIB concatenation.

User response: The system programmer should determine which of the problems listed above is causing the message to be issued, and correct the problem.

ISF008I **DYNAMIC ALLOCATION ERROR RC=***return-code* **EC=***error-code* **IC=***information-code* **DDN=***ddname* **VOL=***volume-serial* **DSN=***data-set-name* *****

Explanation: An error has occurred during the dynamic allocation of a data set.

User response: For information on dynamic allocation return, error, and information codes, see the appropriate manual concerning system macros and facilities, or job management.

ISF009I **SDSF TRACE I/O ERROR**

Explanation: An error occurred while writing a record to the trace output data set. Trace is no longer available for this SDSF session.

User response: Allocate a new trace output data set.

ISF011I **OPEN ERROR** *ddname*

Explanation: An error occurred trying to open the indicated *ddname*. The *ddname* can be:

HASPINDX
The SYSLOG index data set

SDSFMENU
The SDSF help panel data set

User response: Verify the *ddname* is allocated to the proper data set.

ISF012I **SDSF ABEND USER|SYSTEM** *abend-code* **AT** *address* **IN MODULE** *module-name* **OFFSET** *offset*

Explanation: SDSF has abended with the user or system abend code *abend-code*. User abend codes are in decimal; system abend codes are in hexadecimal.

If the abend address is not in module *module-name*, UNKNOWN is displayed for *address*.

User response: The system programmer should see "SDSF user abend codes" on page 551 for information on the user abend codes, or the appropriate system codes manual for information on the system abend codes.

ISF013I **R***x***-R***y* *rega_rega regb_regb regc_regc regd_regd*

Explanation: The registers listed here are displayed in conjunction with ISF012I. R*x*-R*y* indicates the range of registers and *rega_rega regb_regb regc_regc regd_regd* is the contents of those registers.

User response: None.

ISF014I **TEA=***tea* **BEA=***bea* **IN MODULE** *module-name* **OFFSET** *offset*

Explanation: This message is displayed in conjunction with ISF012I. TEA is the translation exception address and BEA is the breaking event address. If they cannot be displayed, the message shows N/A.

User response: None.

ISF015I **ISF015I SDSF COMMAND ATTEMPTED|EXECUTED** *command userid logon-proc terminal-name*

Explanation: The message contains the first 42 characters of the command being processed. If the text exceeds 42 characters, the text contains a trailing + sign.

User response: The operator should respond according to the installation's procedures.

Note: If the command attempted or executed is the REPLY command, the command field of this message contains "REPLY *nn* TEXT of REPLY IS SUPPRESSED". The text of the REPLY command is suppressed to prevent confidential data from being logged.

ISF019I **OUTPUT REQUEUE|RELEASE|PURGE ATTEMPTED|SUCCESSFUL JOBNAME=***jobname* **JOBID=***jobid* **CLASS=***class* **DEST=***dest userid logon-proc terminal-name*

Explanation: A user *userid* running with logon procedure *logon-proc* on terminal *terminal-name* has requested that the indicated job (*jobname* and *jobid*) be requeued to the class *class* and destination *dest*, or released to the output queue to the class *class* and destination *dest*, or purged. If the message indicates the requeue was attempted rather than successful, the user was not authorized to make the request.

User response: None.

ISF020E **SDSF LEVEL ERROR FOR MODULE** *module*, **SDSF ASSEMBLED FOR** *level* **BUT JES2 IS AT** *level jes2-level*

Explanation: SDSF has determined that the assembly level *level* of module *module* does not match the JES2 execution level *jes2-level*. SDSF initialization is terminated.

User response: The system programmer should verify that SDSF has been installed using the proper levels of the JES2 MACLIBS.

ISF023I **I/O ERROR** *text*

Explanation: An I/O error occurred while SDSF was creating the temporary file used as input for the GDDM view utility. In the message, *text* describes the type of error.

All records up to the record causing the error are passed to the view utility. Other records are ignored. Because only partial data is passed to the view utility, formatting errors can occur.

User response: Ensure that the data set being viewed contains the correct data streams for the view utility.

ISF024I **USER** *user-id* **NOT AUTHORIZED TO SDSF,** *reason*

Explanation: An unauthorized user, *user-id*, has attempted to use SDSF.

User response: Contact the system programmer or the Help Desk to find out if the user should be authorized to use SDSF.

A user is not authorized to use SDSF for one of these reasons:

- `COMMAND OPTION ERROR`. A failure occurred in parsing the parameters passed to SDSF. Initialization failed. If this problem persists, contact IBM support.
- `DENIED BY EXIT`. An initialization exit routine has denied authority.
- `INVALID BCP LEVEL`. SDSF was invoked under an unsupported level of the BCP. Initialization failed. Be sure the appropriate level of SDSF is being used with the level of operating system that you are running.
- `NO GROUP ASSIGNMENT`. The user does not fall into any group of users defined by ISFPARMS. For more information, see "Group authorization parameters (GROUP or ISFGRP)" on page 36.
- `PRODUCT NOT ENABLED`. SDSF has attempted to register its invocation on a z/OS system, and the registration has failed. If SDSF should be enabled for execution, check the IFAPRD*xx* member of your parmlib concatenation for an entry for SDSF.
- `REXX INIT FAILED`. Initialization of the REXX environment failed.
- `SERVER NOT AVAILABLE`. The SDSF server is required for ISFPARMS but is not active. The server is required for ISFPARMS when the user is not authorized to revert to an ISFPARMS defined with assembler macros. For more information, see Chapter 3, "Using the SDSF server," on page 107.
- `STORAGE NOT AVAIL`. The amount of storage available was insufficient to complete the request.
- `UNEXPECTED INIT FAIL`. SDSF has encountered an unrecoverable error during execution. Follow your local procedure for reporting a problem to IBM.

ISF025I **SDSF SYSLOG INDEX BEING FORMATTED. DO NOT HIT ATTENTION!**

Explanation: The SDSF SYSLOG index (HASPINDX data set) is being reformatted due to an I/O error on the index, a logic error in the index, or a configuration change on the JES2 spool system. **Do not press the Attention key.**

When formatting SYSLOG indexes on different JES2 levels, a reformat occurs the first time a user accesses the different level JES.

User response: None.

ISF026I **SDSF SYSLOG INDEX IN USE. DO NOT HIT ATTENTION!**

Explanation: The SDSF SYSLOG index (HASPINDX data set) was in use when a user pressed the Attention key. If the Attention key is pressed again, a logic error could occur in the index.

User response: None.

ISF027I **ERROR OCCURRED PROCESSING OUTPUT DESCRIPTORS FOR** *jobname*, *procstep*, *stepname*, *ddname*, **RC=***return-code reason-code*

Explanation: An error occurred retrieving the output descriptors for job *jobname*, procedure step *procstep*, step *stepname*, and ddname *ddname*. The scheduler JCL facility (SJF) SWBTUREQ service failed with return-code *return-code* and reason-code *reason-code*.

The output descriptors for the indicated data set are not shown on the JDS panel. The message OUTPUT DESC NOT AVAIL is issued in the SDSF message area.

User response: The meanings of the return and reason codes are documented in the SJF macro IEFSJTRC. Use the SDSF TRACE command to trace the SJF service

calls to obtain additional information about the problem.

ISF028E **ISFGRP INDEX** *return-code* **HAS AN INVALID ISFNTBL SPECIFICATION for** *listname***.**

Explanation: During SDSF initialization, an include or exclude list was being processed for a non-destination list. However, an ISFNTBL TYPE=DEST macro was used to specify the list. In the message text, *return-code* is the index number of the ISFGRP macro being processed, and *listname* is the name of the ISFGRP list that was being processed. (The index indicates the group by a count of groups. For example, an index of 3 indicates the group defined by the third GROUP statement in ISFPARMS.)

Initialization is terminated with a U0016 abend after the remaining include and exclude lists are processed.

User response: Correct the ISFNTBL macro pointed to by the indicated ISFGRP statement.

ISF029I **SWB MODIFY ATTEMPTED|EXECUTED** *data-set-name userid logon-proc terminal-name*

Explanation: A user *userid* running with logon procedure *logon-proc* on terminal *terminal-name* has requested that output descriptors for data set *data-set-name* be modified.

If the message indicates ATTEMPTED, the user was not authorized to make the request. If the message indicates EXECUTED, the request has been scheduled for execution.

User response: None.

ISF030E **SDSF TERMINATING DUE TO PROGRAM AUTHORIZATION FAILURE, REASON=***reason-code*

Explanation: SDSF has been invoked but it cannot obtain authorized state. SDSF execution is terminated. The decimal *reason-code* describes the error as follows:

4 — Unrecognized SDSF SVC option code

8 — SDSF SVC not called from a program request block (PRB)

12 — SDSF SVC not called from an SDSF module

16 — SDSF SVC not called from a module residing in an authorized library

20 — SDSF SVC invoked by a module with an invalid prefix

24 — SDSF SVC was invoked by a module with an active ESTAE

28 — SDSF SVC was invoked by a module called through XCTL

32 — SDSF SVC was called from a non-reentrant module

36 — SDSF SVC not called from within an SDSF module

40 — SDSF SVC was invoked by a caller with an unacceptable mode

User response: Use the reason code to determine the cause of the error. Ensure that SDSF is invoked from an authorized library and in the proper environment.

ISF031I **CONSOLE** *console-name (migration-id)* **ACTIVATED** *(share-status)*

Explanation: A user log has been started using console *console-name*. If a migration identifier has been assigned, *migration-id* contains the ID being used. If the console is being shared, the *share-status* is `(SHARED)`.

User response: None.

ISF032I **CONSOLE** *console-name* **ACTIVATE FAILED, RETURN CODE** *return-code***, REASON CODE** *reason-code*

Explanation: An attempt to activate an extended console has failed. The message text contains the hexadecimal *return-code* and *reason-code* from the MCSOPER macro.

User response: Use the return and reason codes to determine the cause of the error.

ISF033I *console-name* **MESSAGE RETRIEVAL FAILED, MCSOPMSG RETURN CODE** *return-code***, REASON CODE** *reason-code*

Explanation: An attempt to retrieve a message from the extended console *console-name* failed. The message text contains the hexadecimal *return-code* and *reason-code* from the MCSOPMSG macro. Some messages might have been discarded by consoles.

User response: Use the return and reason codes to determine the cause of the error. You can reset the console by issuing a ULOG CLOSE command, followed by a ULOG command.

ISF034I **ULOG IS EMPTY**

Explanation: An attempt has been made to access the user log, but it contains no records.

User response: If the ULOG is inactive, issue the ULOG command to activate it.

ISF035I **SDSF SDUMP FAILED, RETURN CODE=***return-code* **REASON=***reason-code*

Explanation: SDSF failed to take an SDUMP. SDUMP returns the return code and the reason code.

User response: Use the return and reason codes to determine the cause of the error. For more information, refer to *z/OS MVS Programming: Authorized Assembler Services Reference LLA-SDU*.

ISF036I NO RECORDS TO DISPLAY

Explanation: A LOG command has been entered to display the OPERLOG panel, but there are no log records to display.

User response: To display the SYSLOG panel, which contains messages for a single system, type `LOG S`.

ISF037I SDUMP NOT TAKEN, SUPPRESSED BY DAE

Explanation: SDSF attempted to take an SDUMP, but it has been suppressed by the Dump Analysis and Elimination (DAE) component.

User response: None.

ISF039I ERROR PROCESSING ISPF *service* **RC=***return-code***:** *message-text*

Explanation: An error has been encountered in using the ISPF service *service*. The return code from the service and the text of the ISPF message are displayed.

User response: Use the return code and message text to understand and resolve the problem. If the problem persists, follow your local procedure for reporting a problem to IBM.

ISF040I INVALID MDB DISCARDED FOR BLOCKID *blockid*

Explanation: SDSF encountered an invalid message data block (MDB) in the log stream when displaying the OPERLOG panel. The MDB is discarded. The ID of the block in which the MDB was found is *blockid*.

User response: None.

Routing code: ERLOG

ISF041I CONSOLE *console-name* **IS IN USE**

Explanation: SDSF needed to activate an extended console and the default console name was already in use. As a result, SDSF activated a console with a unique name generated by modifying the default name.

User response: None.

ISF042I CONSOLE *console-name* **IS IN USE**

Explanation: SDSF attempted to activate an extended console but the console name was in use. The console was not activated. The console will be shared by SDSF if sharing has not been disabled.

User response: Use the SET CONMOD ON command to allow SDSF to retry the activation using a modified console name, or change the console name with the SET CONSOLE command.

For more information, refer to "Extended console name" on page 343.

ISF050I USER=*user* **GROUP=***group* **PROC=***proc* **TERMINAL=***terminal*

Explanation: Tracing of messages related to security has been requested, or the user has been assigned to a group in ISFPARMS. The message identifies the user by user ID, group in ISFPARMS, logon procedure and terminal.

User response: None required.

ISF051I SAF *authorization* **SAFRC=***saf-rc* **ACCESS=***access* **CLASS=***class* **RESOURCE=***resource* **RECVR=***userid*

Explanation: A SAF check has been performed.

authorization
describes the decision by SAF.

saf-rc is a return code from SAF, or N/A, when the pre-SAF exit is being used.

access is the access mode that was requested.

class is the SAF class.

resource is the SAF resource.

userid is the user's ID. RECVR= is included only if it is specified by this SAF check.

User response: None required. For more information on SAF resources used by SDSF, refer to Chapter 7, "Protecting SDSF functions," on page 211.

ISF052I ISFUSER *exit-type authorization* **EXITRC=***exit-rc* **SAFRC=***saf-rc* **ACCESS=***access* **CLASS=***class***RESOURCE=***resource* **RECVR=***userid*

Explanation: A SAF check has been performed.

exit-type
is the type of exit.

authorization
describes the security decision.

exit-rc is a return code from the exit.

saf-rc is a return code from SAF, or N/A, when the pre-SAF exit is being used.

access is the access mode that was requested.

class is the SAF class.

resource is the SAF resource.

userid is the user's ID. RECVR= is included only if it is specified by this SAF check.

User response: None required. For more information on SAF resources used by SDSF, refer to Chapter 7, "Protecting SDSF functions," on page 211. For more information on user exit routines, refer to Chapter 9, "Using installation exit routines," on page 329.

ISF053I **COMMAND=***command authorization*

Explanation: A check of ISFPARMS security for an SDSF command has been performed.

command
is the command.

authorization
describes the security decision.

User response: None required. For more information, refer to the AUTH parameter in "Group function parameters reference" on page 40.

ISF054I **DEST=** *destination authorization*

Explanation: A check of ISFPARMS security for a destination has been performed.

destination
is the destination.

authorization
describes the security decision.

User response: None required. For more information, refer to the DEST parameter in "Group function parameters reference" on page 40.

ISF055I **ACTION=***action-character authorization* **USERLEVEL=***user-level* **REQLEVEL=***required-level jobname jobid* **RSN=***reason*

Explanation: A check of ISFPARMS security for an action character has been performed.

action-character
is the action character.

authorization
describes the security decision.

user-level
is the user's command level.

required-level
is the required command level.

jobname is the job name, if applicable.

jobid is the job ID, if applicable.

reason is the reason that authorization was denied. It is included only if authorization is denied. The reasons are:

RSN=01 Job no longer valid
Either the job has been purged or the output group is no longer available.

RSN=02 CMDAUTH ALL was not specified
The action requires a value of ALL for CMDAUTH in ISFPARMS.

RSN=03 Not authorized for INIT command
The user is not authorized to the INIT command.

RSN=04 Destination not specified
A destination that is required was not specified.

RSN=05 Not a JES command
The command that was issued must be a JES command but was not.

RSN=06 Not authorized for command
The user is not authorized for the command.

RSN=07 Job name not in include list
An include list is defined with Ixxx parameters in ISFPARMS.

RSN=08 Job name in exclude list
An exclude list is defined with Xxxx parameters in ISFPARMS.

RSN=09 Command authority insufficient
The user does not have the required command authority.

User response: None required. For more information, refer to the CMDLEV parameter in "Group function parameters reference" on page 40.

ISF056I **ISFUSER=***exit-type authorization* **ACTION=***action-character* **EXITRC=***exit-rc jobname jobid*

Explanation: An exit has made a security check for an action character.

exit-type
is the type of exit.

authorization
describes the security decision.

action-character
is the action character.

exit-rc is the return code from the exit.

jobname is the job name, if applicable.

jobid is the job ID, if applicable.

User response: None required. For more information, refer to Chapter 9, "Using installation exit routines," on page 329.

ISF057I **GROUP=***group authorization* **USERAUTH=***user-authorization* **REQAUTH=***req-authorization* **RSN=***reason*

Explanation: A security check has been made for a group in ISFPARMS.

group is the name of the group.

authorization
describes the security decision.

user-authorization
is the list of user authority (OPER, ACCT, JCL, MOUNT).

req-authorization
is the authority that is required by the group.

reason is the reason authorization was denied. It is included only if authorization was denied. The reasons are:

RSN=01 User has insufficient authority
The user does not have the required authority.

RSN=02 User ID is not in include list (IUID)
The include list is defined with the IUID parameter in ISFPARMS.

RSN=03 user ID is in exclude list (XUID)
The exclude list is defined with the XUID parameter in ISFPARMS.

RSN=04 logon proc is not in include list (ILPROC)
The include list is defined with the ILPROC parameter in ISFPARMS.

RSN=05 logon proc is in exclude list (XLPROC)
The exclude list is defined with the XLPROC parameter in ISFPARMS.

RSN=06 terminal is not in include list (ITNAME)
The include list is defined with the ITNAME parameter in ISFPARMS.

RSN=07 terminal is in exclude list (XTNAME)
The exclude list is defined with the XTNAME parameter in ISFPARMS.

User response: None required. For more information, refer to "Group function parameters reference" on page 40.

ISF058I **COLUMN** *column authorization* **USERLEVEL=***user-level* **REQLEVEL=***required-level*

Explanation: A security check has been made for an overtypeable column.

column is the column title, or, for REXX, the column name.

authorization
describes the security decision.

user-level
is the user's authority, specified by the CMDLEV parameter in ISFPARMS.

required-level
is the required authority.

User response: None required. For more information, refer to the CMDLEV parameter in "Group function parameters reference" on page 40.

ISF059I **SAF ACCESS** *auth* **SAFRC=***(rc, rrc, rrs)* **ACCESS=***access* **CLASS=***class* **RESOURCE=***resource*

Explanation: A security check was performed by the SDSFAUX address space on behalf of the user.

auth describes the security decision.

rc,rrc,rrs
is the SAF return code, RACF return code, and RACF reason code.

access is the access level requested.

resource is the resource name being checked.

User response: No response is required.

ISF101E **SDSF INTERNAL ERROR OCCURRED IN MODULE** *module***, REASON CODE** *reason-code***. ADDITIONAL INFORMATION:** *additional-information*

Explanation: An error occurred in SDSF or in a system service required by SDSF.

User response: Use the reason code and additional information (if any) to determine the cause of the error.

The reason codes are:

101 The execution environment was not recognized.

104 The SVT for the server failed a validity check.

105 A call to the IFAEDREG service failed.

106 A call to the IFAEDDRG service failed.

110 The system symbol service ASASYMBM failed.

111 The output area provided for the system symbol service ASASYMBM is too small.

120 A ENFREQ listen request has failed.

121 A ENFREQ delete listen request has failed.

124 The console query service CNZQUERY has failed.

130 The level was invalid for the name/token service.

131 The persist indicator was invalid for the name/token service.

132 A name/token service call has terminated with an error.

142 The IXCARM register service has failed.

143 The IXCARM ready service has failed.

144 The IXCARM deregister service has failed.

160 The SAF encryption service has failed.

161 The encryption key is invalid.

176 An error occurred during the AXSET service.

178 An error occurred establishing an ESTAE.

179 An error occurred deleting an ESTAE.

180 An error occurred during the ATTACH service.

182 An error occurred attempting to ENQ a resource.

184 An error occurred attempting to DEQ a resource.

185 The CIB contained an unexpected command verb.

186 An error occurred during execution the QEDIT service.

187 An error occurred creating a resource termination manager.

188 An error occurred deleting a resource termination manager.

189 An error occurred obtaining the current task token.

190 An error occurred obtaining the job step task token.

192 An error occurred attempting to issue an ETDES service.

197 An error occurred invoking the DEVTYPE service.

211 TCB address not found in task management table.

301 A required REQ address was not provided.

302 An unexpected request was sent to a routine.

303 A request level is not supported by the current version.

511 An invalid parameter value was detected by a routine.

512 An invalid function code was detected by a routine.

513 A service was invoked in an invalid environment, such as a client request in the server environment.

514 A required storage area does not exist.

515 A storage area is not accessible or is in the wrong key.

516 An unexpected condition was detected which indicates a logic error.

517 A mutually exclusive value was detected which indicates a logic error.

519 An invalid sub-type code was detected by a routine.

520 A required module was not loaded or available.

530 An error occurred during execution of the STIMERM service.

531 An error occurred during execution of the STIMER service.

532 An error occurred during execution of the TTIMER service.

533 A failure occurred during termination of a server subtask.

555 An error occurred in setting the CIB count using QEDIT.

557 The LX system token contains an invalid LX value.

558 Unable to reserve a system LX.

559 Unable to create an entry table.

560 Unable to connect an entry table.

561 The ALESERV extract service has failed.

562 The ALESERV add service has failed.

563 The ALESERV delete service has failed.

564 The ALESERV search service has failed.

576 Unable to insert a node in a linked list.

577 An error occurred during processing of a DETACH macro.

578 Unable to delete a node from a linked list.

583 Unexpected token passed to a parse action routine.

584 Unrecognized parse token.

585 Invalid display type key.

586 A buffer is too small.

587 A required buffer is not provided or the buffer length is zero.

601 A default CSCA was not found on the CSCA chain.

602 A local server was not found in the server group.

603 No servers were found in the server group.

604 A communications protocol was not specified for a server in a server group.

605 A communications protocol type was invalid.

606 The request queue name was not provided.

607 An index into the server status table was invalid.

608 A request requires the server status table but it is not defined.

609 The server status table is not marked active.

610 Unable to build the server status table.

611 An error occurred receiving a message.

612 The associated data retrieval routine for a request was not assigned.

613 Field offsets within the request were not assigned.

614 The transmission length for a request is zero.

615 The transmission length for a request is greater than the total length of the request.

616 The request origin is invalid in the current context. The request may have been forwarded but is not trusted.

617 The request is rejected because the request has already been marked as failed.

618 The request queue name is invalid, possibly because it is too long.

619 A server status value is incorrect.

620 A server status value is not expected in the current state.

621 A server request is not expected with the current server status.

622 The platform code for a queue manager is unacceptable.

623 The req fixed length is zero or greater than the total req length.

624 An invalid action character was detected.

625 An unsupported field was overtyped.

626 A base64 encoding has failed.

627 A data compression request has failed.

628 A data masking request has failed.

650 A JSON parse has failed.

ISF102E **I/O ERROR DETECTED BY** *module* **ON** *I/O request* **FOR DDNAME** *ddname*, **RETURN CODE** *return-code*, **REASON CODE** *reason-code*, *additional-information*.

Explanation: An error occurred in an input or output function requested by SDSF.

User response: The additional information (if any) may include system messages for the requested I/O function. See the appropriate system messages manual for more information.

ISF103E **MEMBER** *member-name* **NOT FOUND, DDNAME** *ddname*.

Explanation: A member name specified as input to the server could not be found.

User response: Correct the member name and retry the request.

ISF104E **ALLOCATION OF LOGICAL PARMLIB FAILED, RETURN CODE** *return-code*, **REASON** *reason-code*

Explanation: An error occurred attempting to allocate the logical parmlib using the IEFPRMLB service.

User response: Use the return and reason codes from the service to determine the cause of the error.

ISF105E **DEALLOCATION OF LOGICAL PARMLIB FAILED, RETURN CODE** *return-code*, **REASON** *reason-code*

Explanation: An error occurred attempting to deallocate the logical parmlib using the IEFPRMLB service.

User response: Use the return and reason codes from the service to determine the cause of the error.

ISF106W **SDUMP ERROR OCCURRED IN MODULE** *module*, **RETURN CODE** *return-code*, **REASON CODE** *reason-code*.

Explanation: An error in taking an SDUMP occurred in module *module* with the indicated return and reason codes.

User response: Use the return and reason codes to determine the cause of the error.

ISF108E **DCB SYNAD INFORMATION** *synad-text*.

Explanation: An I/O error has occurred on an input or output function requested by SDSF. The DCB SYNAD information returned as a result of the error is listed in *synad-text*.

User response: Use the text to determine the cause of the error.

ISF109E **DYNAMIC ALLOCATION OF DDNAME** *ddname* **FAILED, RETURN CODE** *return-code*, **REASON** *reason-code*, **INFO CODE** *information-code*.

Explanation: SDSF attempted to allocate ddname *ddname*, but the allocation failed.

User response: For information on dynamic allocation error codes, see the appropriate manual concerning system macros and facilities, or job management.

ISF110I **LOGGING TO DDNAME** *ddname* **SUSPENDED, MESSAGES WILL BE DIRECTED TO THE HARDCOPY LOG.**

Explanation: SDSF encountered an error using *ddname* as the server log. All server messages that are written to the log will be directed to the hardcopy log.

User response: None required. If you want server messages to be written to the server log, stop and start the server, being sure you have a server log allocated. If you do not want logging, allocate the server log to a dummy data set.

ISF111E **DYNAMIC ALLOCATION OF** *dataset-name* **FAILED, RETURN CODE** *return-code*, **REASON** *reason-code*, **INFO CODE** *information-code*

Explanation: SDSF attempted to allocate data set *dataset-name*, but the allocation failed.

User response: For information on dynamic allocation error codes, see the appropriate manual concerning system macros and facilities, or job management.

ISF112I **SDSF ABEND** *ab-code* **REASON** *code*
SERVER *server-name*
MODULE *x* **OFFSET** *y* **LEVEL** *z*
PSW *psw* **CAB** *cab*
TEA *tea*
BEA *bea* **MODULE** *x* **OFFSET** *y*
contents-of-registers

Explanation: SDSF has abended with the user or system abend code *ab-code*. User abend codes are in decimal; system abend codes are in hexadecimal. Variable *tea* is the translation exception address; *bea* is the breaking event address. The contents of registers, *contents-of-registers*, are displayed two registers per line, in the format *access-register/ general-purpose-register*.

User response: The system programmer should refer to "SDSF user abend codes" on page 551 for information on the user abend codes, or the appropriate system codes manual for information on the system abend codes.

ISF115E **SECURITY ERROR DETECTED BY***module-name* **ON OPEN FOR DDNAME** *ddname resource-name*

Explanation: An error occurred in an OPEN operation. In response to a SAF check from JES, SAF denied access to a SYSOUT data set.

User response: See your security administrator.

ISF116E **UNABLE TO LOCATE REQUESTED** *jes-type* **SUBSYSTEM NAMED** *subsystem-name*.

Explanation: SDSF is attempting to process the JES2 or JES3 subsystem *subsystem-name* but it is not defined to the system. SDSF initialization is terminated with a U0080 abend.

User response: Ensure that the subsystem has been specified correctly on the OPTIONS statement in ISFPRM*xx*, the JESNAME or JES3NAME command invocation options, or the isfjesname and isfjes3name REXX special variables.

ISF120E **REQUEST FAILED, MODULE** *module-name* **WAS UNABLE TO OBTAIN** *number* **BYTES OF STORAGE FOR** *area-name*.

Explanation: A request to obtain storage by SDSF *module-name* for *area-name* failed because the indicated bytes of storage were not available.

User response: The request is not processed. If possible, increase the region size used to invoke SDSF.

In the REXX environment, use special variables or other filter options to limit the number of REXX variables needed to satisfy a request. For more information, type REXXHELP (ISPF only).

ISF121I **MODULE ISFSM64 WAS UNABLE TO OBTAIN** *number* **BYTES OF STORAGE (** *nnn* **SEGMENTS). CHECK MEMLIMIT VALUE.**

Explanation: SDSF attempted to obtain storage that is above the bar (above the 2-gigabyte line) but the amount of storage was not available. The value for MEMLIMIT for the user ID may be too low. This message is issued only once per session.

System action: SDSF attempts to obtain storage below the bar.

User response: Contact your system programmer. If SDSF could not obtain the required storage below the bar, the request is not processed and an additional message is issued.

ISF130E **UNABLE TO ADD** *check-name* **HEALTH CHECK, HZSADDCK RETURN CODE** *return-code* **REASON CODE** *reason-code***.**

Explanation: SDSF is attempting to add the check *check-name* to IBM Health Checker for z/OS. The HZSADDCK service has failed with the indicated return and reason codes. The check is not added. .

User response: Use the return and reason codes to diagnose the error. They are described in *IBM Health Checker for z/OS User's Guide*.

ISF137I **SDSF SDUMP NOT TAKEN, SUPPRESSED BY DAE.**

Explanation: SDSF attempted to take an SDUMP, but it has been suppressed by the Dump Analysis and Elimination (DAE) component.

User response: None.

ISF138E **POINT FAILED READING** *dataset-name***, RETURN CODE** *return-code***, RPLFDBK** *feedback-code***, RPLRBAR** *rba***.**

Explanation: A POINT request failed in an attempt to read *dataset-name* with the indicated return code, RPL feedback and relative block address. SDSF is unable to read the file.

When SYSLOG is being processed, *dataset-name* may be a logical data set name of the form *sysname*.SYSLOG.SYSTEM, where *sysname* is the MVS system name for the SYSLOG being processed. SDSF uses the current value of the SYSID command to derive the system name.

In a JES3 environment, a value of `FF04FFFFFFFFFFFF` for *rba* might indicate the SYSLOG data set is empty. This is to be expected if the SYSLOG is on a JES3 local system and no records have been written to it. In that case, you can issue the command SYSID * to specify the global system. The global SYSLOG is processed regardless of which system you are logged on to.

User response: Use the return code and feedback to diagnose the error. If the SYSLOG was being processed, verify that the value of SYSID is correct for the SYSLOG you want to process.

ISF139E **GET FAILED READING** *dataset-name***, RETURN CODE** *return-code***, RPLFDBK** *feedback-code***.**

Explanation: A GET request failed in an attempt to read *dataset-name* with the indicated return code and RPL feedback. SDSF is unable to read the file.

User response: Use the return code and feedback to diagnose the error.

ISF140E **I/O ERROR OCCURRED READING** *control-block-name,* **SPOOL ADDRESS** *spool-address***, RETURN CODE** *return-code***, REASON** *reason-code,* **INFO CODE** *info-code***.**

Explanation: An error occurred attempting to read *control-block-name* from spool using the JES spool I/O SSI. In the message text, the return code is the IEFSSREQ return code, the reason code is from SSOBRETN and info code is from SSJIRETN.

User response: Use the return and reason codes to diagnose the error or follow your local procedures to contact IBM for support.

ISF141E **SPOOL I/O INTERFACE ERROR, RETURN CODE** *return-code***, REASON** *reason-code,* **INFO CODE***info-code***.**

Explanation: An error occurring during the invocation of the JES spool I/O SSI. In the message text, the return code is the IEFSSREQ return code, the reason code is from SSOBRETN and info code is from SSJIRETN.

User response: Use the return and reason codes to diagnose the error or follow your local procedures to contact IBM for support.

ISF142E **DEVICE NAME CONVERSION ERROR OCCURRED FOR DEVICE ID** *device-id***, RETURN CODE** *return-code***, REASON** *reason-code***, INFO CODE** *info-code***.**

Explanation: An error occurred during the invocation of the JES device name conversion SSI. In the message text, the device id is the JES internal device being converted, the return code is from IEFSSREQ, the reason code is from SSOBRETN, and the info code is from SSJIRETN.

User response: Use the return and reason codes to diagnose the error, and then follow your local procedures for contacting IBM for support.

ISF143I **CHECKPOINT ENVIRONMENT CHANGED, PREVIOUS LEVEL WAS** *previous-level,* **CURRENT LEVEL IS** *current-level.*

Explanation: SDSF has detected that the JES checkpoint has changed in an incompatible manner after SDSF has initialized. This can be the result of a JES $ACTIVATE command. In the message text, the previous and current levels of the checkpoint are displayed. SDSF continues, using the new checkpoint format.

User response: Reaccess SDSF so that the new level of the checkpoint can be processed.

ISF144E **UNABLE TO OBTAIN HEALTH CHECKER CHECK INFORMATION ON SYSTEM** *system*, **HZSQUERY CHECKINFO RETURN CODE** *return-code*, **REASON** *reason-code*.

Explanation: An attempt to gather IBM Health Checker for z/OS data was unsuccessful because the HZSQUERY CHECKINFO service failed.

User response: See *IBM Health Checker for z/OS User's Guide* and use the return and reason codes from the HZSQUERY CHECKINFO service to diagnose the error. If the error persists, follow your local procedures for calling IBM for service.

ISF145E **REXX REQUEST SERVICE** *service-name* **FAILED, RETURN CODE** *return-code*, **REASON** *reason-code*.

Explanation: An invocation of the REXX service *service-name* failed with the indicated return and reason code.

User response: The request is not processed. Use the return and reason codes from the service to diagnose the error.

ISF146I **REXX VARIABLE** *variable-name* **SET, RETURN CODE** *return-code*, **VALUE IS** '*value*'.

Explanation: The indicated REXX variable has been assigned the indicated value. The return code corresponds to the SHVRET field returned by the IRXEXCOM service. This message is issued only in verbose mode.

User response: None.

ISF147I **REXX VARIABLE** *variable-name* **FETCHED, RETURN CODE** *return-code*, **VALUE IS** '*value*'.

Explanation: The indicated REXX variable has been obtained and contains the indicated value. The return code corresponds to the SHVRET field returned by the IRXEXCOM service. This message is issued only in verbose mode.

User response: None.

ISF148E **UNABLE TO OBTAIN SUBSYSTEM INFORMATION FOR SUBSYSTEM** *subsystem-name*, **RETURN CODE** *return-code*, **REASON CODE** *reason-code*.

Explanation: SDSF has attempted to obtain information about *subsystem-name* using the subsystem version information (SSVI) subsystem interface call but the SSI has failed. In the message text, *return-code* is the return code from IEFSSREQ and *reason-code* is the reason code in SSOBRETN.

User response: Use the return and reason codes to diagnose the error or follow your local procedures to contact IBM for support.

ISF149E **UNABLE TO OBTAIN** *ssi-request* **DATA FOR SUBSYSTEM** *subsystem-name*, **RETURN CODE** *return-code*, **SSOBRETN** *ssob-return-code*, **REASON CODE** *reason-code*.

Explanation: A subsystem request directed to *subsystem-name* failed for *ssi-request* data with the referenced SSI return code and SSOB return code. The reason code is for the specific SSI function being performed. The SDSF function that required the SSI data cannot be performed.

User response: Use the request type and return codes to diagnose the error.

ISF150E **COMMUNICATIONS ERROR OCCURRED PROCESSING** *service-name,* **RETURN CODE** *return-code,* **REASON CODE** *reason-code*. **ADDITIONAL INFORMATION:** *additional information*

Explanation: A error occurred while processing the indicated communications service. The required communication is not completed.

User response: If the service name begins with MQ, a WebSphere MQ service has failed. Use the WebSphere MQ service return and reason codes, and the additional information to determine the cause of the error.

ISF151E **MESSAGE REJECTED FROM UNSUPPORTED PLATFORM, PLATFORM CODE** *code*, **PLATFORM NAME** *name*

Explanation: An error occurred in communications between SDSF servers. A message was received from a platform that is not supported. The message is ignored.

User response: If the message has been received in error, follow your local procedures for contacting IBM support.

ISF152E **MESSAGE REJECTED FROM USER** *user-identity* **DUE TO UNEXPECTED FORMAT NAME** *format-name*.

Explanation: A server request has been rejected due to an incorrect format name. The format is not recognized. The server does not process the request.

User response: None required. If the message has been received in error, follow your local procedures for contacting IBM support.

ISF153E **MESSAGE REJECTED FROM USER** *user-identity* **DUE TO INCORRECT APPLICATION IDENTITY.**

Explanation: A server request has been rejected due to invalid data in the application identity data section of the message context. The request is not processed

User response: If the message is issued in error, follow your local procedures for contacting IBM for support.

ISF154E **REQUEST REJECTED, TARGET JES UNACCEPTABLE FOR REQUESTOR.**

Explanation: A request for data has been processed by the server, but the target JES is not in the same MAS as the requestor. The request is rejected.

User response: Ensure that the server group definition references only those JES subsystems in the same MAS as the client. If the problem persists, follow your local procedures for contacting IBM support.

ISF155E **REQUEST REJECTED, TARGET SYSPLEX UNACCEPTABLE FOR REQUESTOR.**

Explanation: A request for data has been processed by the server, but the target sysplex is not in the same sysplex as the requestor. The request is rejected.

User response: Ensure that the server group definition references only those systems in the same sysplex as the client. If the problem persists, follow your local procedures for contacting IBM support.

ISF156I **UNABLE TO OBTAIN SYSPLEX INFORMATION, IXCQUERY** *function-name* **FAILED, RETURN CODE** *return-code*, **REASON CODE** *reason-code*.

Explanation: An error occurred using the IXCQUERY service to gather sysplex information. The sysplex information is not shown.

User response: Use the return and reason codes to diagnose the error.

ISF160E **IXCSEND TO SERVER** *server-name* **FAILED, RETURN CODE** *return-code*, **REASON CODE** *reason-code*.

Explanation: The IXCSEND service has failed sending a message to *server-name* with the indicated return and reason code. The request is not processed.

User response: Use the return and reason codes to diagnose the problem. Refer to *z/OS MVS Programming: Sysplex Services Reference*. If the error persists, follow your local procedures for contacting IBM support.

ISF161E **IXCSEND FROM SERVER** *server-name* **FAILED, RETURN CODE** *return-code*, **REASON CODE** *reason-code*.

Explanation: The IXCSEND service has failed receiving a message to *server-name* with the indicated return and reason code. The request is not processed.

User response: Use the return and reason codes to diagnose the problem. Refer to *z/OS MVS Programming: Sysplex Services Reference*. If the error persists, follow your local procedures for contacting IBM support.

ISF162E **START SERVER** *server-name* **FAILED, IXCSRVR RETURN CODE** *return-code*, **REASON CODE** *reason-code*.

Explanation: The IXCSRVR start service has failed processing *server-name* with the indicated return and reason code. The request is not processed.

User response: Use the return and reason codes to diagnose the problem. Refer to *z/OS MVS Programming: Sysplex Services Reference*. If the error persists, follow your local procedures for contacting IBM support.

ISF163E **STOP SERVER** *server-name* **FAILED, IXCSRVR RETURN CODE** *return-code*, **REASON CODE** *reason-code*.

Explanation: The IXCSRVR stop service has failed processing *server-name* with the indicated return and reason code. The request is not processed.

User response: Use the return and reason codes to diagnose the problem. Refer to *z/OS MVS Programming: Sysplex Services Reference*. If the error persists, follow your local procedures for contacting IBM support.

ISF166E **SEND FAILED, BPX4QSN RETURN CODE** *return-code*, **REASON CODE** *reason-code*, **msgtype** *message-type*, **length** *length*.

Explanation: An error occurred in sending a message using the BPX4QSN service with the indicated return and reason codes. The message type used when sending the message was *message-type*. The size of the message being sent is indicated by *length*. The message is not sent.

User response: Use the return and reason codes to diagnose the error.

For return code 121 reason code xxxx030B, the size of the USS interprocess communication (IPC) message queue may be too small for SDSF to put a message on the queue. The message size needed by SDSF varies based on the type of request and the size of the response. Determine the maximum size of the queue by issuing the D OMVS,O operator command and inspecting the value of the IPCMSGQBYTES option. Use the length of the message being sent from the

message text to increase the size of the queue as necessary.

Refer to *z/OS UNIX System Services Messages and Codes*.

ISF167E **RECEIVE FAILED, BPX4QRC RETURN CODE** *return-code*, **REASON CODE** *reason-code*, **msgtype** *message-type*.

Explanation: An error occurred in receiving a message using the BPX4QRC service with the indicated return and reason codes. The message type used when sending the message was *message-type*. The message is not sent.

User response: Use the return and reason codes to diagnose the error. Refer to *z/OS UNIX System Services Messages and Codes* .

ISF170I **SERVER** *server-name* **ARM REGISTRATION COMPLETE FOR ELEMENT TYPE** *element-type*, **ELEMENT NAME** *element-name*.

Explanation: The server has successfully registered with ARM with the indicated element type and name.

User response: None required.

ISF171E **SERVER** *server-name* **ARM REGISTRATION FAILED FOR ELEMENT TYPE** *element-type*, **ELEMENT NAME** *element-name*, **RETURN CODE** *return-code*, **REASON CODE** *reason-code*.

Explanation: The server has attempted to register with ARM with the indicated element name and type. However, the registration has failed with the listed return and reason codes from the IXCARM macro.

User response: Use the return and reason codes to understand the problem. Refer to *z/OS Security Server RACF Security Administrator's Guide*.

ISF172E **SERVER server-name ARM DEREGISTRATION FAILED, RETURN CODE** *return-code*, **REASON CODE** *reason-code*.

Explanation: The server has attempted to deregister from ARM, but the IXCARM service has failed with the indicated return and reason codes.

User response: Use the return and reason codes to understand the problem. See *z/OS Security Server RACF Security Administrator's Guide*.

ISF174E *xxxx* **UNABLE TO LOAD MODULE** *module*, **RETURN CODE** *return-code*, **REASON CODE** *reason-code*.

Explanation: SDSF was unable to load the indicated module.

User response: See the return and reason codes for information about the problem. If the codes indicate that the load module was not found, the libraries containing the SDSF load modules may not have been correctly installed.

ISF175W *xxxx* **UNABLE TO DELETE MODULE** *module*, **RETURN CODE** *return-code*, **REASON CODE** *reason-code*.

Explanation: SDSF was unable to delete the indicated module.

User response: See the return and reason codes for information about the problem.

ISF180I **TASK** *task-id* **IS BEING RESTARTED DUE TO ABEND.**

Explanation: In response to an abend, the task indicated by *task-id* is being restarted.

User response: None required.

ISF181I **TASK (***task-name, taskid***) CANNOT BE RESTARTED DUE TO ABEND.**

Explanation: The indicated task has abended and cannot be restarted. If the task is required for SDSF server execution, the server will be terminated.

User response: Correct the problems indicated by the abend, or follow your local procedures for contacting IBM support

---

ISF182I **TASK (***task-name, taskid***) HAS BEEN RESTARTED.**

Explanation: The indicated task has been successfully restarted.

User response: None required.

ISF190E **UNABLE TO CREATE DATASPACE** *dataspace-name*, **DSPSERV RETURN CODE** *return-code*, **REASON CODE** *reason-code*.

Explanation: A failure has occurred trying to create the named data space. WTORs will not be displayed on the SR panel or on the Log panel.

User response: Follow your local procedures for reporting a problem to IBM.

ISF191E **UNABLE TO DELETE DATASPACE** *dataspace-name* **(***dataspace-generated-name***), DSPSERV RETURN CODE** *return-code*, **REASON CODE** *reason-code*.

Explanation: A failure has occurred trying to delete the named data space.

User response: Follow your local procedures for reporting a problem to IBM.

ISF300E **MODIFY COMMAND IGNORED DUE TO ERRORS.**

Explanation: The text of an operator MODIFY command *command* was not recognized.

User response: Correct the command and retry the request.

ISF301E *value* **WAS EXPECTED IN COMMAND POSITION** *position* **BEFORE** *keyword***.**

Explanation: A value, *value*, was missing in the indicated position in the command.

User response: Correct the command and retry the request.

ISF302E *value* **WAS SEEN IN COMMAND POSITION** *position* **WHERE ONE OF THE FOLLOWING WAS EXPECTED:** *valid-values***.**

Explanation: An invalid value, *value*, was found at the indicated position in the command.

User response: Correct the command using one of the listed valid values.

ISF303E **MODIFY COMMAND TEXT MISSING, COMMAND IGNORED.**

Explanation: The MODIFY command was entered without required command text. The command is ignored.

User response: Correct the command and retry the request.

ISF304I **MODIFY** *parameter* **COMMAND ACCEPTED.**

Explanation: The indicated parameter of the MODIFY command was accepted for processing.

User response: None required.

ISF305E **ABEND** *abend-code* **OCCURRED PROCESSING MODIFY COMMAND.**

Explanation: An abend occurred in processing the MODIFY command. The command is not executed.

User response: Use the abend code to diagnose the problem.

ISF306E **MODIFY** *command* **COMMAND IGNORED DUE TO AUTHORIZATION FAILURE.**

Explanation: A MODIFY command could not be processed because SAF checking has determined that the user is not authorized to issue the command.

User response: If you have been denied access in error, refer to "User authorization" on page 475 for more information.

ISF307E **MODULE** *module* **NOT FOUND.**

Explanation: A MODIFY D,MODULE command was issued for a module, but the module could not be located.

User response: Verify that the module name was entered correctly. The module must be accessible or currently loaded by SDSF.

ISF308E **"***value***" WAS SEEN IN COMMAND POSITION** *position* **BUT NOT EXPECTED.**

Explanation: An invalid value, *value*, was found at the indicated position in the command. The command is not processed.

User response: Correct the command.

ISF310I *server-name* **COMMUNICATIONS**
ID SERVER STATUS
SYSTEM JESN MEMREQSPROC
*requests-processed***BER**
id server status
system jesn member
QMGR: *qmgr* **REQUESTQ:** *server-q*
QMGR: *qmgr* **CLIENTQ:** *client-q*
CLUSTER/CLUSTERNL: *cluster-name*

Explanation: Information about communication between SDSF servers is displayed in response to an operator command:

id an identifier associated with the server

server name of the server

status status of the server

system system on that the server is processing

jesn JES2 subsystem for which the server gathers data

member
member of the MAS for the JES2 subsystem

requests-processed
number of requests processed

qmgr name of the WebSphere MQ queue manager

server-q
name of the server request queue (shown only for the local server). The server request queue is used by the local server to get requests from the remote servers.

client-q
name of the client request queue. The client request queue is used by the client to send messages to the local server, and by the local server to send messages to the remote servers.

cluster-name
name of the WebSphere MQ cluster or cluster name-list

User response: None required.

ISF311I **SERVER COMMUNICATIONS NOT ACTIVE.**

Explanation: A command to display information about server communication was issued, but communication between SDSF servers is not active.

User response: None required. For information about enabling communication between SDSF servers, refer to "Using the server for sysplex data" on page 110.

ISF312I *server-name* **DISPLAY**
SERVER STATUS: *status* **DEFAULT:** *status*
COMMUNICATIONS: *status*
PARMS: *member/dataset-name*
XCF COMMUNICATIONS: *xcf-status*

Explanation: In response to an operator command, information about the status of server communications is displayed. The server status codes are:

CloseQ request queue being closed

Connected
connect to queue manager complete

Connecting
connect to queue manager in progress

CreateModelQ
create of model queue in progress

CreatedModelQ
model queue create complete

DeleteClientQ
delete of client queue in progress

DeletedClientQ
delete of client queue complete

DeleteModelQ
delete of model queue in progress

DisableClientQ
client queue being disabled

Disconnecting
disconnect from queue manager in progress

EnableClientQ
client queue being enabled

EnabledClientQ
client queue enable complete

Failed prior initialization failed

Inactive
communications not active

OpenReqQ
request queue open in progress

OpenedReqQ
request queue open complete

OpenClientQ
client queue open in progress

OpenedClientQ
client queue open complete

SetSignal
event signal being set

Signalled
event signal complete

Starting
communications being started

Stopping
communications being stopped

TaskInit
task initialization in progress

TaskTerm
task termination in progress

TestComm
test communication in progress

Values for XCF application server status, *xcf-status*, are:

Configured
SDSF can exploit XCF for sysplex requests

Not Configured
the server is not configured to use XCF for sysplex requests

User response: None required.

ISF313I *server-name* **MODULE DISPLAY**
NAME: *name* **EPADDR:** *entry-address*
FMID: *module-fmid* **LEVEL:** *apar-level*
COMPDATE: *date* **COMPTIME:** *time*

Explanation: The service-level information for module *name*, including its compile date and time is displayed in response to a MODIFY D,MODULE command.

User response: None.

ISF314I **ACCESS DENIED TO** *class-name*/*resource-name* **LEVEL** *access-level* **DUE TO SAF NO DECISION.**

Explanation: An attempt to access the resource *resource-name* protected by SAF class *class-name* with a requested access level of *access-level* has been denied. The SAF authorization check has resulted in a no-decision (indeterminate) result and SDSF has consequently failed the request.

User response: In the JES3 environment, all resources must be protected through SAF. It may be necessary to define profiles so that the named resources can be accessed.

ISF315I *server-name* **XCF COMMUNICATIONS APPLICATION SERVER NAME:** *name* **TASKS ACTIVE:** *nnn* **IDLE:** *nnn* **SENDS:** *count* **RECEIVES:** *count*

Explanation: In response to a display communications command, XCF communications data is displayed. *name* is the application server name being used by SDSF for XCF communications. A task is active if it is actively processing a request. An idle task is waiting for work. The send and receive counts indicate the number of messages sent or received by the server. The count is scaled using the K, M, G, T, and P characters or all asterisks if the count exceeds the space available.

User response: None.

ISF401I **SERVER** *server-name* **COMMUNICATIONS INITIALIZATION IN PROGRESS.**

Explanation: The communications between SDSF servers is being initialized.

User response: None required.

ISF402I **SERVER** *server-name* **COMMUNICATIONS READY.**

Explanation: Initialization of communications for the indicated SDSF server has completed successfully. The server is ready to begin communications with other SDSF servers.

User response: None required.

ISF403E **SERVER** *server-name* **COMMUNICATIONS INITIALIZATION FAILED, COMMUNICATIONS NOT AVAILABLE.**

Explanation: Communications for the indicated SDSF server did not initialize successfully. The server is not ready to begin communications with other SDSF servers.

User response: See associated messages for an explanation of the error.

ISF404I **SERVER** *server-name* **COMMUNICATIONS STOPPED.**

Explanation: Communications for the indicated server was stopped. Communications is no longer available.

User response: Correct your server group definition in ISFPARMS and refresh them.

ISF405I **SERVER** *server-name* **COMMUNICATIONS IN USE, SERVERGROUP DEFINITION UNCHANGED.**

Explanation: An attempt was made to modify the server group in ISFPARMS after the ISFPARMS were already being processed by the SDSF server. The request is ignored.

User response: None required. You cannot change the properties of a server group defined in ISFPARMS after the server has begun processing the ISFPARMS. To change the properties of the server group, first stop the server with the STOP command.

ISF406I **SERVER** *server-name* **COMMUNICATIONS WAITING FOR CONNECTION.**

Explanation: Communications for the indicated server are waiting for a connection. The server cannot communicate with other servers in the group, and data from that server will not be included on the SDSF panels. It may be that WebSphere MQ is not active.

User response: See accompanying messages for more information. If WebSphere MQ is not active, start it.

ISF407I **SERVER** *server-name* **COMMUNICATIONS WAITING FOR ACCESS TO REQUEST QUEUE.**

Explanation: During communications initialization, the server detected that the request queue name was in use. The server requires exclusive control of the request queue. Initialization will wait until the queue name is available. If the server has been recycled, there might be a delay until the queue manager marks the queue as being available.

The server will periodically try the failing request until the queue name is accessed.

User response: See accompanying messages for more information. Verify that the queue name is not in use by any other application.

ISF408I **SERVER** *server-name* **DEFINING OBJECT** *object-name* **ON QUEUE MANAGER** *queue-manager-name***.**

Explanation: SDSF is attempting to define an object using the named queue manager.

User response: None required.

ISF409E **SERVER** *server-name* **UNABLE TO DEFINE OBJECT** *object-name* **ON QUEUE MANAGER** *queue-manager-name***.**

Explanation: SDSF was unable to define the indicated object on the named WebSphere MQ queue manager.

User response: See additional messages for more information.

ISF410I **SERVER** *server-name* **HAS DEFINED OBJECT** *object-name* **ON QUEUE MANAGER** *queue-manager-name***.**

Explanation: SDSF defined the indicated object on the named WebSphere MQ queue manager.

User response: None required.

ISF411I **RESPONSE FROM** *queue-manager: response-text***.**

Explanation: The SDSF server has invoked the WebSphere MQ system command interface to perform an administrative request, such as creating a queue. The queue manager has responded with the indicated text.

User response: None required.

ISF412I **COMMUNICATIONS WITH SERVER** *server-name* **SYSTEM** *system-name* **STOPPED.**

Explanation: Communications has been stopped with the indicated server in the server group. Requests will no longer be forwarded to the server for processing.

User response: Use the start communications command to resume processing for the server.

ISF413E **SERVER ID** *server-id* **NOT PROCESSED, SERVER NOT FOUND IN SERVERGROUP.**

Explanation: A command was entered to modify a server in the server group, but the server ID was not recognized. The command is not processed.

User response: Retry the command with the correct server ID. To display the server ID, use the server operator command `F server-name,DISPLAY,C`.

ISF414E **SERVER** *server-name* **SYSTEM** *system-name* **NOT PROCESSED, SERVER NOT FOUND IN SERVERGROUP.**

Explanation: A command was entered to modify a server in the server group, but the server and system name patterns did not match any server. The command is not processed.

User response: Retry the command with the correct server ID. To display the server and system names, use the server operator command:

`F server-name,DISPLAY,C`.

ISF415I **SERVER** *server-name* **SYSTEM** *system-name* **STARTED, CURRENT STATUS IS** *status-text.*

Explanation: A server with the indicated name has been started. The status of the server is *status-text*.

User response: None required.

ISF416I **SERVER** *server-name* **COMMUNICATIONS WILL BE RESTARTED.**

Explanation: Communications with *server-name* is being restarted. A restart may have been necessary because the connection was broken or was quiescing. Additional messages will be issued indicating when the restart is complete.

User response: None required.

ISF417I **SERVER** *server-name* **COMMUNICATIONS STOPPING.**

Explanation: Communications is ending for *server-name*. No additional sysplex requests will be processed.

User response: None required.

ISF418I **COMMAND TO** *queue-manager-name***:** *command-text*

Explanation: The indicated queue manager administrative command is being sent to the queue manager for processing.

User response: None required.

ISF420I **SERVER** *server-name* **DELETING OBJECT** *object-name* **ON QUEUE MANAGER** *queue-manager-name***.**

Explanation: The SDSF server is deleting the indicated WebSphere MQ object on queue-manager *queue-manager-name*, because QDELETE(YES) was specified on the COMM statement in ISFPARMS for the server. The object was originally created by the SDSF server.

User response: None required.

ISF421I **SERVER** *server-name* **HAS DELETED OBJECT** *object-name* **ON QUEUE MANAGER** *queue-manager-name***.**

Explanation: The SDSF server has deleted the indicated WebSphere MQ object on queue manager *queue-manager-name*. The object was originally created by the SDSF server.

User response: None required.

ISF422E **SERVER** *server-name* **UNABLE TO DELETE OBJECT** *object-name* **ON QUEUE MANAGER** *queue-manager-name***.**

Explanation: The indicated WebSphere MQ object was not deleted by the SDSF server because the object was in use by WebSphere MQ. The server attempted to delete the object because QDELETE(YES) was specified on the COMM statement of ISFPARMS.

User response: See additional messages in the server joblog for more information.

ISF423I **SERVER** *server-name* **COMMUNICATIONS WAITING FOR ACCESS TO CLIENT REQUEST QUEUE.**

Explanation: During communications initialization, the SDSF server detected that the client request queue was in use. The server requires exclusive control of the client request queue. Initialization will wait until the queue name is available. If the server has been recycled, there might be a delay until the queue manager marks the queue as being available.

The server will periodically try the failing request until the queue name is accessed.

User response: None required.

ISF424E **SERVER** *server-name* **UNABLE TO DEFINE OBJECT** *object-name* **ON QUEUE MANAGER** *queue-manager-name***, OBJECT ALREADY EXISTS.**

Explanation: The SDSF server was unable to create the indicated WebSphere for MQ object on the named queue manager because the object already exists.

User response: To have the object redefined by the server, specify QREPLACE(YES) on the COMM statement for the server in ISFPARMS.

ISF425I **SERVER** *server-name* **CLIENT QUEUE** *queue-name* **HAS A TARGET OF** *target-queue-name* **THAT DIFFERS FROM THE REQUEST QUEUE NAME OF** *request-queue-name***.**

Explanation: During communications initialization, the SDSF server has detected that the client request queue has been defined with a target queue that differs from the expected name. The client request queue should be a queue alias for the server request queue. Processing continues. However, the server may not receive messages sent to the client queue because the target does not match.

User response: To have the server redefine the client request queue, specify QREPLACE(YES) on the COMM statement of ISFPARMS for the server.

ISF426E **SERVER** *server-name* **CLIENT QUEUE** *queue-name* **CONFIGURED FOR CLUSTER** *cluster-name* **BUT QUEUE DEFINED FOR CLUSTER** *cluster-name-two***.**

Explanation: The SDSF server has detected an inconsistency in the definition of WebSphere MQ queue *queue-name*. The cluster name specified on the COMM statement of ISFPARMS does not match the cluster attribute for the queue. The cluster name specified for the SDSF server in ISFPARMS must match the name associated with the queue. Communications initialization failed.

User response: Check that the cluster name on the COMM statement is correct. To have the server redefine the queue, use the QREPLACE(YES) option of the COMM statement.

ISF427E **SERVER** *server-name* **CLIENT QUEUE** *queue-name* **CONFIGURED FOR CLUSTER NAMELIST** *comm-namelist-name* **BUT QUEUE DEFINED FOR CLUSTER NAMELIST** *queue-namelist-name***.**

Explanation: The SDSF server has detected an inconsistency in the definition of WebSphere MQ queue *queue-name*. The cluster namelist specified on the COMM statement of ISFPARMS does not match the cluster attribute for the queue. The cluster namelist specified for the SDSF server in ISFPARMS must match the namelist associated with the queue. Communications initialization failed.

User response: Check that the cluster namelist on the COMM statement is correct. To have the server redefine the queue, specify QREPLACE(YES) on the COMM statement in ISFPARMS.

ISF428I **SERVER** *server-name* **UNABLE TO DISABLE OBJECT***object-name***.**

Explanation: During server termination, a communications error prevented the server from disabling *object-name*. An object is disabled to prevent subsequent requests from being directed to it. Server communications continues.

Other servers in the server group may continue to send requests to this server. This may result in delays because the requests will timeout rather than being rejected immediately.

User response: Use any additional error messages issued by the server to determine the cause of the problem.

ISF429I **SERVER** *server-name* **NOT DEFINING OBJECT** *object-name*, **QUEUE DEFINITION PROHIBITED.**

Explanation: The server is not defining *object-name* because the QDEFINE initialization option has been specified as NO. Initialization continues. However, if *object-name* is required by the server but has not already been defined, initialization may fail.

User response: You can change the QDEFINE initialization option on the COMM statement of ISFPARMS. Refer to"COMM statement" on page 32 for more information.

ISF432E **SETTINGS DESCRIPTOR COLUMNS LENGTH** *length* **EXCEEDS MAXIMUM LENGTH OF** *maximum-length*.

Explanation: The columns list provided in the settings descriptor is too long and exceeds the maximum length. The columns list is ignored. An external call environment is used by the SDSF CIM provider.

User response: Follow your local procedures for contacting IBM for service.

ISF433I **SERVER** *server-name* **XCF CONNECTION ESTABLISHED AS SERVER** *xcf-application-server-name*.

Explanation: The SDSF server *server-name* has identified itself as *xcf-application-server-name* and is ready to process requests using XCF.

User response: None.

ISF434I **SERVER** *server-name* **CONNECTION WITH XCF STOPPING.**

Explanation: The SDSF server *server-name* is stopping communication with XCF.

User response: None.

ISF435I **SERVER** *server-name* **CONNECTION WITH XCF STOPPED.**

Explanation: The SDSF server *server-name* has stopped communication with XCF.

User response: None.

ISF436I **NO SYSTEMS SATISFY SYSTEM NAME FILTER. USE THE SYSNAME COMMAND TO CHANGE THE VALUE.**

Explanation: A request for sysplex data has been processed but the current SYSNAME value does not match any system in the sysplex. The request is not processed.

User response: Use the SYSNAME command to change the system names that will be processed.

ISF437I **DATA NOT AVAILABLE FROM SYSTEMS:** *system-name-list*.

Explanation: A sysplex request has been processed, but responses from the named systems have not been received within the timeout interval. The systems may be busy or unable to process the request.

User response: Review the timeout interval specified with the SET TIMEOUT command and retry the request.

ISF438I **XCF SERVER NAME** *server-name* **NOT PROCESSED SINCE SERVER** *xcf-application-name* **ALREADY ACTIVE.**

Explanation: A request to start XCF communications using *server-name* has not been processed because SDSF is already connected to the XCF application server *xcf-application-name*. *server-name* cannot be changed while the application server is active.

User response: Stop SDSF XCF communications and then retry the request.

ISF439I **SERVER** *server-name* **XCF CONNECTION ALREADY ESTABLISHED AS SERVER** *xcf-application-name*.

Explanation: SDSF server *server-name* has processed a request to start XCF communications, but the application server is already active as *xcf-application-name*.

User response: None.

ISF440I **XCF SERVER** *xcf-application-name* **CANNOT BE UNDEFINED SINCE IT IS ALREADY ACTIVE.**

Explanation: While processing a command to refresh ISFPRM*xx*, SDSF encountered a CONNECT statement that defines XCFSRVNM(NONE) to disable the use of XCF. However, the XCF application server is already active. The refresh is processed but there is no change to the XCF status.

User response: To undefine XCF, stop communications prior to the refresh or restart the server.

ISF441E DATA NOT AVAILABLE FROM ANY SYSTEM.

Explanation: A request for sysplex data has been made, but no systems have responded within the timeout interval. The systems may be busy or unable to process the request.

User response: Review the timeout interval specified with the SET TIMEOUT command and retry the request.

ISF442I SERVER *server-name* XCF COMMUNICATIONS READY.

Explanation: SDSF is ready to accept sysplex requests using XCF. *server-name* is the name of the SDSF server.

User response: None.

ISF450I SERVER *server-name* starting SDSFAUX

Explanation: SDSF server *server-name* has determined that SDSFAUX is not active and is starting it.

User response: No response is required.

ISF451I SERVER *server-name* stopping SDSFAUX

Explanation: During the shutdown of SDSF server *server-name*, SDSF has determined that SDSFAUX is active and is stopping it.

User response: No response is required.

ISF452E SDSFAUX COMMUNICATIONS FAILED, RETURN CODE *0xreturn-code*, REASON CODE *0xreason-code*, function *function-name*, *additional information*

Explanation: An internal SDSF request (*function-name)* has been sent to the SDSFAUX address space but has failed with the indicated return and reason code in hexadecimal. If available, additional information may be provided that describes the error.

The return code is as follows:

Return code (hexadecimal)	Description
00	Success
04	Warning
08	Error
0C	Environment error
10	Severe error
14	Fatal error

The *reason-code* is of the form xxxxrrrr where *xxxx* is an internal identifier for the module that has detected the error and *rrrr* is the reason code. The *reason-code* is as follows:

Reason code (hexadecimal)	Description	Response
xxxx040A	Results truncated	SDSF was unable to complete all data gathering requests because too much data was returned. Refine your request if possible and retry.
xxxx0410	Partial results	SDSF was unable to complete all data gathering requests because too much data was returned. Refine your request if possible and retry.
xxxx0412	RMF required	SDSF was unable to complete a data gathering request because RMF is required. Verify that RMF Monitor I is active.
xxxx0413	RMF not installed	SDSF was unable to complete a data gathering request because RMF is not installed. Verify that module ERBSMFI can be loaded.
xxxx0801	Not found	Ensure that the SDSFAUX address space has been started.

Reason code (hexadecimal)	Description	Response
xxxx0806	Access denied	For function connect, verify user is authorized to the ISF.CONNECT.*system* resource in the SDSF class. For other functions, enable security tracing using the SET SECTRACE command to determine the resource for which access is needed.
xxxx0813	SDSFAUX unavailable	A request could not be processed because SDSFAUX is not started. Ensure the SDSF server is active and refresh ISFPRMxx to restart SDSFAUX.
xxxx082F	Send to SDSFAUX failed	SDSF was unable to gather remote data because the send using XCF failed. Verify that all target systems are available.
xxxx0830	Receive by SDSFAUX failed	SDSF was unable to receive results from XCF possibly because too much data was returned or a timeout occurred. Refine your request and use the SET TIMEOUT command to increase the timeout.
xxxx0858	SDSFAUX shutdown in progress	SDSFAUX is shutting down. Retry your request after SDSFAUX restarts.
Other	Internal error	An internal error has occurred. Follow your local procedures for contacting IBM for support.

User response: Use the additional information to diagnose the error. If no information is provided or the error cannot be resolved, contact IBM Software Support.

ISF453I SDSFAUX is already active

Explanation: During initialization of the SDSF server or a refresh of ISFPRM*xx*, SDSF has determined that SDSFAUX is already active and does not need to be started.

Parameters related to SDSFAUX on the CONNECT statement such as AUXPROC, AUXNAME, and AUXSAF are ignored.

User response: If changes have been made to the CONNECT statement related to SDSFAUX, stop and start the SDSF server for the changes to take effect.

ISF488E SDSF NOT STARTED DUE TO ERRORS IN START PARAMETERS.

Explanation: One or more parameters on the EXEC statement for the SDSF server was not recognized.

User response: Correct the parameters and retry the request.

ISF491E *value* **WAS EXPECTED IN START PARAMETER POSITION** *position* **BEFORE** *string*.

Explanation: SDSF encountered an error in a parameter on the START command.

User response: Use the position and string values to identify the parameter in error. Retry the START command with a corrected parameter.

ISF492E *value* **WAS SEEN IN START PARAMETER POSITION** *position* **WHERE ONE OF THE FOLLOWING WAS EXPECTED:** *list-of-values*.

Explanation: SDSF encountered an error in a

parameter on the START command. The position of the error in the command string is indicated by *position*.

User response: Retry the START command using one of the valid values.

ISF493I **ABEND** *abend-code* **OCCURRED PROCESSING START PARAMETERS.**

Explanation: An abend occurred in processing the START command. The command is executed with any parameters that were processed prior to the abend.

User response: Use the abend code to diagnose the problem. You may want to use the MODIFY command to reset server options.

ISF515E **SDSF INITIALIZATION FAILED FOR SERVER** *server***.**

Explanation: Initialization of server *server* failed to complete. Messages describing the reason for the failure will have been issued prior to this one.

User response: Use the error messages issued by SDSF to determine the cause of the initialization failure.

ISF517E **SDSF SERVER WAS NOT STARTED DUE TO INVALID EXECUTION ENVIRONMENT, POSSIBLE MISSING PPT ENTRY.**

Explanation: The SDSF server could not start due to an incorrect execution environment. The server is not running in the correct protect key.

User response: Verify that a PPT entry has been defined in your SCHED*xx* member of the parmlib concatenation for program ISFHCTL.

ISF518E **SDSF SERVER** *server* **NOT STARTED, NOT ENABLED FOR EXECUTION**

Explanation: The SDSF server has attempted to register its invocation on a z/OS system, but the registration has failed. The server is not initialized.

User response: If SDSF should be enabled for execution, check the IFAPRD*xx* member of your parmlib concatenation for an entry for SDSF.

ISF527E **SDSF SERVER** *server* **NOT STARTED, START COMMAND MUST BE USED.**

Explanation: An attempt was made to start the SDSF server *server* through a batch job. The server must be started with the MVS START command.

User response: Issue the MVS START command to start the SDSF server.

ISF528E **SDSF SERVER** *server* **NOT STARTED, INVALID OPERATING SYSTEM LEVEL.**

Explanation: The SDSF server requires a higher level of the operating system than was found. The server was not started.

User response: None.

ISF538E **SDSF SERVER** *server* **ALREADY ACTIVE.**

Explanation: The START command was entered for an SDSF server that is already active. The command was ignored.

User response: None.

ISF540I **SERVER** *server-name* **ASSIGNED AS DEFAULT SERVER.**

Explanation: The indicated SDSF server has been made the default server. If no server is specified in the assembler ISFPARMS, users who do not explicitly state the server name on the SDSF command will connect to this server when accessing SDSF. Any server specified in ISFPARMS will be ignored.

User response: None required.

ISF541I **SERVER** *server-name* **UNASSIGNED AS DEFAULT SERVER.**

Explanation: The indicated SDSF server had been the default server but is no longer the default server. Either another server has been made the default server, or the server is terminating, or ISFPARMS has been refreshed with a different option on the CONNECT statement.

User response: None required.

ISF542I **SERVER** *server-name* **NOT ASSIGNED AS DEFAULT SERVER, SERVER** *server-default-name* **ALREADY ASSIGNED.**

Explanation: The indicated SDSF server, *server-name*, was not made the default server because a default server, *server-default-name*, already has been assigned.

User response: None required. To make the server the default, regardless of whether a default has already been assigned, change the DEFAULT option on the CONNECT statement in ISFPARMS to DEFAULT(YES).

ISF543I **SERVER** *server-name* **ALREADY ASSIGNED AS DEFAULT SERVER, ASSIGNMENT UNCHANGED.**

Explanation: Processing ISFPARMS has resulted in no change to the default SDSF server. The indicated server, *server-name*, is the default server.

User response: None required.

ISF544E *option* **REJECTED, NOT AUTHORIZED FOR USE.**

Explanation: The named REXX option was rejected because the user is not authorized to use it.

User response: None required.

ISF546I **OPTIONS NOT APPLICABLE TO THE INITIAL COMMAND IGNORED.**

Explanation: SDSF was invoked with initial command options, but the options are not applicable to the initial panel being invoked. The initial options are ignored.

User response: None required.

ISF701I **SDSF TRACE STARTED USING TRACE MASK** *trace-mask***.**

Explanation: In response to an operator command, the current trace mask is displayed.

User response: None required.

ISF702I **SERVER** *server-name* **DEBUG MODE IS ENABLED.**

Explanation: In response to an operator command, the current status for diagnostic mode is displayed.

User response: None required.

ISF703I **SERVER** *server-name* **DEBUG MODE IS DISABLED.**

Explanation: In response to an operator command, the current status for diagnostic mode is displayed.

User response: None required.

ISF709I **SDSF TRACE IS INACTIVE, TRACE MASK IS "***trace-mask***".**

Explanation: In response to an operator command, the current status for SDSF server trace is displayed.

User response: None required.

ISF710I **SDSF TRACE IS ACTIVE USING TRACE MASK "***trace-mask***".**

Explanation: In response to an operator command, the current status for SDSF server trace is displayed.

User response: None required.

ISF711I **SDSF TRACE STARTED USING TRACE MASK** *trace-mask.*

Explanation: In response to the TRACE command, tracing has been started with the indicated trace mask.

User response: None required.

ISF713E **SDSF TRACE INITIALIZATION FAILED, RETURN CODE** *return-code***, REASON CODE** *reason-code.*

Explanation: In response to the TRACE command, initialization of SDSF trace has failed with the indicated return and reason codes

User response: Use the indicated return and reason codes to diagnose the problem.

ISF714I **SDSF TRACE IS NOW INACTIVE.**

Explanation: In response to a TRACE OFF command, SDSF trace has become inactive.

User response: None required.

ISF715I **SDSF TRACE IS ALREADY ACTIVE USING TRACE MASK** *trace-mask*

Explanation: A TRACE ON command was entered, but SDSF trace is already active, with the indicated trace mask.

User response: None required.

ISF716E **SDSF TRACE DATA SET IS NOT ALLOCATED.**

Explanation: A TRACE ON command was entered, but the SDSF trace data set could not be dynamically allocated. SDSF trace is not started.

User response: Additional system messages may have been issued to the console. See them for additional information.

ISF717I **SDSF TRACE IS ALREADY INACTIVE.**

Explanation: A TRACE OFF command was entered, but SDSF trace is already inactive. The command is ignored.

User response: None required.

ISF718E **SDSF TRACE FAILED TO INACTIVATE.**

Explanation: A TRACE OFF command was entered, but SDSF trace was not turned off. Tracing continues.

User response: Retry the request.

ISF724I **SDSF LEVEL *fmid* INITIALIZATION COMPLETE FOR SERVER *server*.**

Explanation: The SDSF server was successfully initialized.

User response: None.

ISF725I **SDSF SHUTDOWN IN PROGRESS FOR SERVER *server*.**

Explanation: The SDSF server is being shut down.

User response: None.

ISF726I **SDSF PARAMETER PROCESSING STARTED.**

Explanation: The processing of the SDSF parameters has started.

User response: None.

ISF727I **SDSF PARAMETER PROCESSING STARTED IN TEST MODE.**

Explanation: The processing of the SDSF parameters has started in test mode. The syntax of the parameters will be checked, but the parameters will not be activated.

User response: None.

ISF728I **SDSF PARAMETERS HAVE BEEN ACTIVATED.**

Explanation: The processing of the SDSF parameters was successful and the parameters are now active.

User response: None.

ISF729I **NO ERRORS DETECTED IN SDSF PARAMETERS.**

Explanation: The processing of the SDSF parameters completed with no errors.

User response: None.

ISF731E **SDSF PARAMETERS NOT ACTIVATED DUE TO ERRORS.**

Explanation: Errors were found in the SDSF parameters. The parameters are not activated.

User response: Use the log file to review the parameters. Correct the errors and process the parameters again.

ISF732I **ERRORS DETECTED IN SDSF PARAMETERS.**

Explanation: Errors were found in the SDSF parameters.

User response: Use the log file to review the parameters. Correct the errors and process the parameters again.

ISF733E **UNABLE TO READ SDSF PARAMETERS DUE TO I/O ERROR.**

Explanation: An I/O error prevented SDSF from reading the SDSF parameters.

User response: See accompanying system messages for more information about the I/O error.

ISF734I **SDSF PARAMETERS HAVE BEEN ACTIVATED, WARNINGS WERE ISSUED.**

Explanation: SDSF ISFPARMS have been activated; however, during syntax checking of the ISFPARMS, SDSF issued warning messages.

User response: Check the server log for the warning messages. If you change the ISFPARMS, activate the changes with the MODIFY command.

ISF735E **SDSF PARAMETERS ARE NOT ACTIVE.**

Explanation: An error was detected in the SDSF parameters when the SDSF server was started. SDSF parameters are not activated.

User response: Use the log file to review the parameters. Correct the errors and activate the parameters with the MODIFY command.

ISF736I **SDSF SHUTDOWN PROCEEDING FOR SERVER *server-name*.**

Explanation: A STOP command has been issued to shut down an SDSF server. The server is waiting for completion of outstanding work.

User response: None required.

ISF737E **SDSF PARAMETERS NOT ACTIVATED DUE TO ABEND.**

Explanation: Due to an abend, SDSF parameters were not activated.

User response: Use the MODIFY command to active the parameters. The MODIFY command is described in “Server operator commands” on page 118.

ISF738I **ABEND** *abend-code* **DETECTED PROCESSING SDSF PARAMETERS.**

Explanation: While SDSF parameters were being processed in test mode, an abend was detected.

User response: Use the abend code to diagnose the problem.

ISF739I **SDSF PARAMETERS BEING READ FROM MEMBER** *member-name* **OF DATA SET** *dataset-name***.**

Explanation: The SDSF server is reading SDSF parameters from the indicated data set and member.

User response: None required.

ISF740E **VARIABLE** *variable-name* **DATA VALUE** *'value'* **IS TOO LONG.**

Explanation: The value for the named special variable exceeds the valid length.

User response: Special variables that are associated with SDSF commands cannot exceed the SDSF command length. Adjust the value of the special variable to the valid length.

ISF741E **ERROR PROCESSING COMMAND '***command***' ASSOCIATED WITH VARIABLE** *variable-name***, REASON:** *reason-text***.**

Explanation: The value of the special variable *variable-name* was rejected with the indicated reason text. The command is not processed.

User response: Ensure that the syntax of the special variable *variable-name* conforms to the syntax required by the SDSF command *command-name*. The syntax of the commands is described in the online help.

ISF742E **COLUMN** *column-name* **NAMED IN** *variable-name* **VARIABLE IGNORED, NOT FOUND IN CURRENT FIELD LIST.**

Explanation: The named column was not found in the current field list. A REXX variable will not be created with its value.

User response: Ensure the column name specified in *variable-name* are valid for the current field list. If the column is valid for the panel, but is found only on the alternate field list, use the ALTERNATE option on the SDSF host command used to invoke the panel. Refer to "Issuing commands with ISFEXEC" on page 380 for more information.

ISF743E **VARIABLE** *variable-name* **HAS A DATA VALUE EXCEEDING** *number* **BYTES AND IS TOO LONG.**

Explanation: The value of the special variable*variable-name* was rejected because the data value is too long. The associated command is not processed.

User response: Ensure that the syntax of the special variable *variable-name* conforms to the syntax required by the associated SDSF command. For the syntax of an SDSF command, see the online help.

ISF744E **UNABLE TO FETCH REXX VARIABLE** *variable-name***, IRXEXCOM SHVRET RETURN CODE** *return-code***.**

Explanation: SDSF was unable to read the value of *variable-name*. The IRXEXCOM service failed to fetch the variable with return code *return-code* for field SHVRET. The associated command will not be processed.

User response: Use the return code from the IRXEXCOM service as described in *z/OS TSO/E REXX Reference* to diagnose the error.

ISF745E **ERROR PROCESSING '***command***', REASON:** *reason-code***.**

Explanation: SDSF was unable to run command. The error is described by *reason-code.*

User response: Use the reason code to diagnose the error. For syntax errors, correct the command format or the operands specified on a special variable. For authorization errors, ensure the user has the appropriate authority to the command.

ISF746E **ACTION REQUEST REJECTED, ROW TOKEN INVALID.**

Explanation: A row token referenced on an ISFACT command has failed a validity check. The action is not performed.

User response: The row token is created by the ISFEXEC command and must be passed unmodified to SDSF on the ISFACT command. Some of the conditions causing the token to become invalid are:

- The token has been modified or contains an invalid character
- The token does not correspond to the display being modified. For example, the token was generated for a row on the H panel but is being used on the O panel.
- The token was generated on a different level of SDSF than the one currently being run.
- The token was generated for a different use ID than the one performing the action.

ISF747E ACTION REQUEST REJECTED, ROW NOT FOUND.

Explanation: A row token referencing a row that no longer exists was encountered during processing of an ISFACT command. The requested action is not performed.

User response: None.

ISF748E ACTION REQUEST REJECTED, ROW NOT UNIQUE.

Explanation: A row token that references a row that is not unique was encountered during processing of an ISFACT command. The requested action is not performed.

User response: Obtain a new row token by running the ISFEXEC command again and retrying the ISFACT request.

ISF749E ACTION REQUEST REJECTED, *column-name* IS NOT MODIFIABLE.

Explanation: An attempt to modify a column that could not be modified was encountered during processing of an ISFACT command. The requested modification was not performed.

User response: Verify that the named column can be modified. You must be authorized to modify the column. For a list of columns, issue the COLSHELP command from any SDSF command line under ISPF.

ISF750E ACTION REQUEST REJECTED, *column-name* NOT FOUND IN CURRENT FIELD LIST.

Explanation: A column that is not in the current field list was encountered during processing of an ISFACT command. The request was not performed.

User response: Ensure that you have included the necessary option on the ISFACT command:

- If the column is in the alternate field list, use ALTERNATE or ALTERNATE2 (when the panel is accessed from another panel with an action character)
- If the column is a delayed-access column, use DELAYED or DELAYED2.

To find which columns are available in your REXX exec, access the panel and display the contents of the ISFCOLS or ISFCOLS2 special variable.

To display a list of columns that identifies which are delayed access, type COLSHelp in SDSF's help (ISPF only).

The system programmer can specify the columns that are included in the primary and alternate field lists using ISFPARMS. Refer to "Variable field lists (FLD or ISFFLD)" on page 85 for more information.

ISF751E COLUMN *column-name* ACTION IGNORED, NO DATA PROVIDED.

Explanation: Data to modify a column was null or all blanks when processing an ISFACT command. The request is ignored.

User response: Ensure that the data to be used to modify a column is non-blank.

ISF752E COLUMN *column-name* ACTION REJECTED, DATA LENGTH *data-length* EXCEEDS THE MAXIMUM OF *maximum-length*.

Explanation: On an ISFACT command, the data to modify column *column-name* is too long. The request is rejected.

User response: Ensure that the length of the data to be modified does not exceed the maximum width for the field.

ISF753E ACTION REQUEST REJECTED, COMMAND *command* NOT ACCEPTABLE.

Explanation: A command, *command*, that is not acceptable to ISFACT was encountered while processing the ISFACT command.

User response: Ensure that the command used on ISFACT is a command to access a tabular panel.

ISF754I COMMAND '*command*' GENERATED FROM ASSOCIATED VARIABLE *variable-name*.

Explanation: The SDSF command *command* was run based on the data contained in the REXX special variable *variable-name*.

User response: None.

ISF755E HOST COMMAND NOT PROVIDED.

Explanation: The REXX SDSF host command environment was invoked but no command was provided.

User response: Ensure that a command is passed to the SDSF host command environment.

ISF756I NO ACTIONS PERFORMED, ROW NOT MODIFIED.

Explanation: No actions were provided or accepted for the row. The row has not been modified.

User response: None.

ISF757I **VARIABLE** *variable-name* **BEING PROCESSED WITH VALUE '***value***'.**

Explanation: The indicated special variable has been retrieved and contains the indicated value.

User response: None.

ISF758E **ERROR PROCESSING DATA ASSOCIATED WITH VARIABLE** *variable-name*, **REASON:** *reason-text.*

Explanation: An error occurred processing the data associated with the indicated variable. The reason is given by *reason-text*.

User response: The function is not performed.

ISF759E **PRINT ERROR OCCURRED:** *error-text*.

Explanation: In the processing of a print request, an error occurred. The error is described by *error-text*.

User response: None.

ISF760I **HOST COMMAND BEING PROCESSED:** *command*.

Explanation: SDSF has been invoked to process the REXX host command *command*.

User response: None.

ISF761E **COLUMN** *column-name* **ACTION REJECTED, DATA VALUE '***value***' UNACCEPTABLE.**

Explanation: An action for a row was rejected because the modified data was unacceptable for the column. For example, the overtype extension character (+) was specified, and that is not valid in the REXX environment.

User response: Correct the data to be used to modify the column.

ISF762I **COLUMN** *column-name* **ACTION REJECTED, VALUE '***value***' EXCEEDS THE MAXIMUM NUMBER OF VALUES OF** *max-values*.

Explanation: The number of values being used to modify the indicated column exceeds the maximum number of related values allowed for that column. The request is rejected.

User response: Correct the data so that the number of related values does not exceed the maximum number of values for the column. For more information, see the online help for overtyping columns on that panel.

ISF763E **COLUMN** *column-name* **ACTION REJECTED, DATA VALUE '***value***' INVALID, REASON:** *reason text*.

Explanation: An action taken against a row was rejected because the modified data failed a syntax check for the column. The reason is indicated by *reason-text*. For example, a syntax error can occur if the column is defined for numeric data but an attempt was made to modify it with non-numeric data.

User response: Correct the data to be used to modify the column.

ISF764I **COMMAND '***command***' GENERATED FROM ASSOCIATED VARIABLE** *variable-name*, **STATUS:** *status*.

Explanation: The SDSF command *command* was run based on the data contained in the REXX special variable *variable-name* with any completion status indicated in the status text.

User response: None.

ISF765I **VARIABLE** *variable-name* **NOT DEFINED, DEFAULT VALUE *'value'* BEING USED.**

Explanation: The named REXX variable was not found so the indicated value was applied as a default.

User response: None.

ISF766I **REQUEST COMPLETED, STATUS:** *completion-status*.

Explanation: An SDSF request has completed with the indicated status. The completion status is the text from the SDSF message area and also corresponds to the REXX special variable ISFMSG.

User response: None.

ISF767I **REQUEST COMPLETED, STATUS:** *completion-status*.

Explanation: An SDSF request has completed with no additional status. The REXX special variable ISFMSG contains no data.

User response: None.

ISF769I **SYSTEM COMMAND ISSUED, COMMAND TEXT:** *command-text*.

Explanation: A system command was issued with the ISFEXEC command *command* or the ISFSLASH command. The text of the command is shown in *command-text*.

User response: None.

ISF770W **REQUEST LIMIT** *limit* **FROM VARIABLE** *variable-name* **REACHED.**

Explanation: The limit for the number of requests, *limit*, set by special variable *variable-name*, has been reached.

User response: If necessary, change the limit.

ISF771E **VARIABLE** *variable-name* **NOT ACCESSIBLE, PROCESSING TERMINATED.**

Explanation: Variable *variable-name* does not exist or could not be fetched. Processing is stopped.

User response: Verify that the variable name is correct and exists.

ISF772I **VARIABLE** *variable-name* **IGNORED, DOES NOT CONTAIN DATA.**

Explanation: Variable *variable-name* does not contain any data and is skipped.

User response: Verify that the variable name is correct.

ISF775E **VARIABLE** *variable-name* **NOT ACCEPTABLE, DOES NOT CONTAIN DATA.**

Explanation: Variable *variable-name* has been fetched, but does not contain data. A value for this variable is required.

User response: Verify that the value for the variable is present.

ISF776I **PROCESSING STARTED FOR ACTION** *action-count* **OF** *total-count***.**

Explanation: When processing actions or commands, SDSF started processing the action that is number *action-count* out of the total number, *total-count.*

User response: None required.

ISF777E **STOP TIME AND DATE INCONSISTENT WITH START TIME AND DATE.**

Explanation: A date range is not acceptable because the ending time and date is prior to the starting time and date.

User response: Correct the time and date range.

ISF778I **STOP REQUEST BEING PROCESSED.**

Explanation: SDSF is processing a stop request and will end.

User response: None required.

ISF779E **PARSING ERROR OCCURRED WHILE PROCESSING JSON REQUEST, RETURN CODE=***return-code***, REASON=***reason***.**

Explanation: A parsing error occurred while parsing a JSON document as described by *return-code* and *reason*. The document may not be well formed or may contain a syntax error. The document is not processed. The return-code is an internal code that can be used by IBM to diagnose the error.

User response: Correct the document and retry the request.

ISF780E **JSON PROPERTY** *property-name* **NOT RECOGNIZED OR NOT IN CORRECT CONTEXT.**

Explanation: A JSON document was being processed, and *property-name* was not recognized as a valid property, or the property is not a valid subproperty of an object. The document is not processed.

User response: Correct the document and retry the request.

ISF781E **JSON OBJECT NESTING LEVEL EXCEEDED.**

Explanation: A JSON document was being processed and too many levels of subproperties were found. The document was not processed.

User response: Correct the document and retry the request.

ISF782W **NO ROWS SATISFY REQUEST.**

Explanation: A request was received but constraints resulted in no rows being generated for the response.

User response: None.

ISF783E **ERROR OCCURRED GENERATING JSON DOCUMENT FOR REQUEST.**

Explanation: An unrecoverable error occurred in generating a document for a JSON response.

User response: Refer to additional messages that further describe the error.

ISF784E **VARIABLE** *variable* **REQUIRES SPECIFICATION OF VARIABLE** *variable***.**

Explanation: A variable was specified that requires another variable that is missing. The request may fail or be processed as if neither variable were specified.

User response: Correct the error and retry the request.

ISF785E **VARIABLE** *variable1* **VALUE '***value***' MUST NOT BE LESS THAN VARIABLE** *variable12* **VALUE '***value***'.**

Explanation: The value in *variable1* is less than the value in *variable12*. This is not allowed.

User response: Correct the error and retry the request.

ISF786E **VARIABLE ISFFIND VALUE '***string***' WITH LENGTH** *length* **IS TOO LONG FOR SPECIFIED COLUMN RANGE** *start-column* **TO** *end-column***.**

Explanation: The string specified in the ISFFIND variable is too long to fit within the specified column range.

User response: Correct the error and retry the request.

ISF787E **VARIABLE** *variable* **VALUE '***value***' EXCEEDS THE RECORD LENGTH OF THE DATA.**

Explanation: The value of variable variable is greater than the record length of the data that is being browsed. The request cannot be processed.

User response: Correct the error and retry the request.

ISF788E **VARIABLE** *variable* **IS IGNORED, IT CONTAINS A TOKEN THAT IS NOT VALID.**

Explanation: The value of variable *variable* is a token that is not valid. The request is processed as if the variable were not specified.

User response: Ensure that the token was not modified before you attempted to use it.

ISF789E **VARIABLE** *variable* **IS IGNORED, IT CONTAINS A TOKEN THAT IS NOT VALID IN THIS CONTEXT.**

Explanation: The value of variable *variable* is a token that is not valid for this request. The request is processed as if the variable were not specified.

User response: Ensure that the token was not modified before you attempted to use it. The variable that contains the token may not have been cleared before it was set. To clear variables, use the ISFRESET function.

ISF790E **THE VALUE SPECIFIED FOR VARIABLE** *variable* **IS NOT VALID ON THE** *panel* **PANEL.**

Explanation: The value of variable *variable* is a token that is not valid for the current panel. The request cannot be processed.

User response: Correct the value that is in error. For the value that is in error, see the previous ISF757I message. For information about the valid values, use the SEARCH command or the REXXH command.

ISF791E **VARIABLE** *variable* **IS IGNORED, THE TOKEN REPRESENTS A RECORD THAT NO LONGER EXISTS.**

Explanation: The record represented by the token in variable *variable* does not exist. The request is specified as if the variable were not specified.

User response: None required.

ISF792E **DATA NOT AVAILABLE, NOT AUTHORIZED TO COMMAND** *command***.**

Explanation: A request for data could not be satisfied. The request requires a command that you are not authorized to use.

User response: For authorization to the command, contact your security administrator.

ISF793E **DATA NOT AVAILABLE, HEALTH CHECKER NOT ACTIVE ON SYSTEM** *system-name***.**

Explanation: A request for data could not be satisfied because IBM Health Checker for z/OS is not active on the indicated system.

User response: Contact your system programmer to activate IBM Health Checker for z/OS.

ISF794W **MAXIMUM RESPONSE SIZE REACHED, ROWS** *row-1* **THROUGH** *row-2* **NOT PROCESSED.**

Explanation: The size of the response exceeds the maximum allowed. Rows *row-1* through *row-2* are skipped. They are not included in the response.

User response: Use filters to limit the number of rows being selected, then try the request again.

ISF800E **UNEXPECTED END OF FILE ENCOUNTERED PROCESSING STATEMENT NUMBER** *number***.**

Explanation: While processing a continuation statement, the end of file was reached.

User response: Use the log file to review the parameters. Correct the errors and process the parameters again.

ISF801E **STATEMENT NUMBER** *number* **IS TOO LONG.**

Explanation: SDSF parameter statement number *number* is longer than the maximum allowed length of 32756 characters.

User response: Use the log file to review the parameters. Ensure that a statement is not continued incorrectly. Correct the statement in error and process the parameters again.

ISF802E **INPUT FILE IS EMPTY.**

Explanation: The input file for processing SDSF parameters contained no parameters.

User response: Correct the input file and retry the request.

ISF803E **COMMENT NOT CLOSED ON LINE NUMBER** *number***.**

Explanation: A comment opened on line number *number* was not closed. Comments must be complete on a single line.

User response: Use the log file to locate the line and close the comment.

ISF804E **PROCESSING ENDED DUE TO I/O ERROR.**

Explanation: Processing of SDSF parameters ended due to an input or output error. Either SDSF or the system may have issued additional messages describing the error.

User response: Use the messages to determine the cause of the I/O error.

ISF805I **PREVIOUSLY PROCESSED** *statement-type* **AT STATEMENT** *statement-number* **BEING REPLACED.**

Explanation: A statement of the same type has already been processed and will be replaced by the later statement. The statement being replaced is *statement-number*.

User response: None required. However, you should check your ISFPARMS to remove duplicate statements.

ISF806E *parameter* **VALUE** *value* **IS IN ERROR, INVALID SYNTAX SPECIFIED.**

Explanation: The value indicated by *value* in the parameter indicated by *parameter* contains invalid syntax.

User response: Correct the syntax.

ISF807E *parameter* **VALUE** *value* **IS TOO LONG, MAXIMUM LENGTH ALLOWED IS** *maximum***.**

Explanation: The value indicated by *value* in the parameter indicated by *parameter* is longer than the maximum allowed length, indicated by *maximum.*

User response: Correct the length of the value.

ISF808E *parameter* **VALUE** *value* **IS NOT NUMERIC.**

Explanation: The value indicated by *value* in the parameter indicated by *parameter* is not numeric. It must be numeric.

User response: Correct the value.

ISF809E *parameter* **VALUE** *value* **IS TOO SMALL, MINIMUM VALUE ALLOWED IS** *minimum***.**

Explanation: The value indicated by *value* in the parameter indicated by *parameter* is smaller than the minimum allowed value, indicated by *minimum*.

User response: Correct the value.

ISF810E *parameter* **VALUE** *value* **IS TOO LARGE, MAXIMUM VALUE ALLOWED IS** *maximum***.**

Explanation: The value indicated by *value* in the parameter indicated by *parameter* is larger than the maximum allowed value, indicated by *maximum*

User response: Correct the value.

ISF811E *parameter* **VALUE** *value* **IS INVALID.**

Explanation: The value indicated by *value* in the parameter indicated by *parameter* is invalid.

User response: Correct the value.

ISF812E *parameter* **VALUE** *value* **IS AN INVALID SYSOUT CLASS.**

Explanation: The value indicated by *value* in the parameter indicated by *parameter* is not a valid SYSOUT class. Valid classes are A-Z and 0-9.

User response: Correct the value.

ISF813E *parameter* **VALUE** *value* **CONTAINS INVALID HEXADECIMAL DIGITS.**

Explanation: The value indicated by *value* in the parameter indicated by *parameter* contains characters that are not valid hexadecimal digits. Valid hexadecimal digits are 0-9 and A-F.

User response: Correct the value.

ISF814E *parameter* **VALUE** *value* **IS TOO SHORT, MINIMUM LENGTH ALLOWED IS** *minimum***.**

Explanation: The value indicated by *value* in the parameter indicated by *parameter* is shorter than the minimum allowed length, indicated by *minimum*.

User response: Correct the value.

ISF815E *parameter* **VALUE** *values* **MUST HAVE DIFFERENT CHARACTERS FOR EACH VALUE.**

Explanation: The values indicated by *values* are not unique. Each value specified on this parameter must be unique.

User response: Correct the values so that each is unique.

ISF816E *first-parameter* **IS MUTUALLY EXCLUSIVE WITH** *second-parameter***.**

Explanation: The parameters indicated by *first-parameter* and *second-parameter* cannot be used together.

User response: Delete one of the parameters.

ISF817I **GROUP INDEX** *group-index-number* **ASSIGNED TO GROUP** *group-name***.**

Explanation: The index number indicated by *group-index-number* is assigned to the group indicated by *group-name*. The name, *group-name*, is a name assigned by you with the NAME parameter, or, if NAME is omitted, it is a name assigned by SDSF.

User response: None required.

ISF818I **GROUP** *group-name* **REPLACES STATEMENT** *statement-type*, **GROUP INDEX IS** *index-number* **.**

Explanation: A group named *group-name* has been encountered more than once; the latest occurrence replaces the previous occurrence. The index number assigned to the group is indicated by *index-number.* (The index indicates the group by a count of groups. For example, an index of 3 indicates the group defined by the third GROUP statement in ISFPARMS.)

User response: None required. You should check your parameters to remove duplicate group statements.

ISF819I *statement-type* **NAMED** *name* **REPLACES STATEMENT** *number.*

Explanation: The statement named *name* has been encountered more than once. The latest occurrence replaces the previous occurrence.

User response: None required. You should check your parameters to remove duplicate statements.

ISF820I *statement* **NAMED** *name* **FOR** *display1* **DISPLAY CONFLICTS WITH PRIOR DEFINITION FOR** *display2***.**

Explanation: An FLD statement with the name *name*, for the indicated SDSF display, conflicts with an FLD statement for another display that has already been encountered.

User response: None required. You should check your parameters to remove duplicate statements.

ISF821E *string* **WAS EXPECTED BEFORE** *string* **ON LINE** *line-number* **COLUMN** *column-number***.**

Explanation: A syntax error has been encountered at the indicated line and column.

User response: Correct the statement.

ISF822E *value* **WAS SEEN ON LINE** *line-number* **COLUMN** *column-number* **WHERE ONE OF THE FOLLOWING WAS EXPECTED:** *valid-values***.**

Explanation: An invalid value, *value*, was found at the indicated line and column. The valid values are shown in *valid-values*.

User response: Correct the statement using one of the listed values.

ISF823I **INPUT SKIPPED UP TO THE NEXT** *value***.**

Explanation: A syntax error has occurred on a previously identified statement. SDSF is skipping to the indicated *value* to continue processing.

User response: Correct the statement in error.

ISF824E *error-string* **ON LINE** *line-number* **COLUMN** *column-number* **SHOULD BE DELETED.**

Explanation: The character string *error-string* located on the indicated line and column is in error and should be deleted.

User response: Delete or correct the string in error.

ISF825I *string* **IS INSERTED BEFORE THE ERROR POINT.**

Explanation: In response to previous syntax errors, SDSF has inserted a character string, *string* before the error in order to continue processing.

User response: Correct the error.

ISF826E *statement* **OFFSET OF** *offset* **IS TOO LONG FOR USE WITH STRING** *string*, **MAXIMUM COMBINED OFFSET AND STRING LENGTH IS** *maximum*.

Explanation: In the indicated statement, the offset *offset*, when used with the string *string*, results in an invalid value for that statement. The maximum for the combination of the offset and string length is *maximum.*

User response: Correct the string or offset.

ISF828E *first-statement* **STATEMENT REQUIRED PRIOR TO THIS** *second-statement*.

Explanation: You must include a statement of the type indicated by *first-statement* before the statement indicated by *second-statement*.

User response: Reorder or add statements to achieve the required order.

ISF829E *first-value* **AND** *second-value* **MUST HAVE DIFFERENT VALUES.**

Explanation: The values indicated by *first-value* and *second-value* are the same. They must be different.

User response: Change one or both of the values so that they are different.

ISF830E *parameter* **VALUE IS TOO SHORT, VALUE MUST BE** *required-length* **BYTES BUT IS ONLY** *actual-length*.

Explanation: The value specified for the indicated parameter is too short. The message indicates the required length of the value (*required-length*) and the length of the value that was actually specified (*actual-length*).

User response: Correct the value to be the required number of bytes.

ISF831E *parameter* **VALUE IS TOO LONG, VALUE MUST BE** *required-length* **BYTES BUT IS** *actual-length*.

Explanation: The value specified for the indicated parameter is too long. The message indicates the required length of the value (*required-length*) and the length of the value that was actually specified (*actual-length*).

User response: Correct the value to be the required number of bytes.

ISF832I *statement* **NAMED** *name* **CONFLICTS WITH PREVIOUS DEFINITION FOR** *statement*.

Explanation: The statement with the name *name* conflicts with another statement of a different type that has already been encountered.

User response: None required. You should review your statements to remove the conflict.

ISF833E **COLUMN** *column* **IS NOT VALID FOR THE** *display* **DISPLAY.**

Explanation: The indicated column has been specified with an FLDENT statement for a display on which it is not valid.

User response: Remove the FLDENT statement for that display, or change the display with which the FLDENT statement is associated.

ISF834E *string* **WAS EXPECTED BEFORE** *string* **IN STATEMENT** *statement-number*.

Explanation: A syntax error has been encountered at the indicated statement.

User response: Correct the statement.

ISF835E *value* **WAS SEEN IN STATEMENT** *statement* **WHERE ONE OF THE FOLLOWING WAS EXPECTED:** *valid-values*.

Explanation: An invalid value, *value*, was found at the indicated statement. The valid values are shown in *valid-values*.

User response: Correct the statement using one of the listed values.

ISF836E *parameter* **VALUE** *string* **IS IN ERROR, INVALID DATA SET NAME SYNTAX.**

Explanation: The indicated parameter specifies a data set name containing invalid syntax.

User response: Correct the data set name and retry the request.

ISF837E *parameter* **VALUE CONTAINS** *number* **CHARACTERS, BUT IT MUST BE EVEN.**

Explanation: The value specified on the indicated parameter is an odd number of characters; the value

must be an even number of characters.

User response: Correct the value to contain an even number of characters.

ISF838E *secondary-statement-type* **NAMED** *secondary-statement-name* **REFERENCED BY** *primary-statement-type primary-statemet-name* **NOT FOUND.**

Explanation: A statement indicated by *primary-statement-type primary-statement-name* references a statement, *secondary-statement-type secondary-statement-name* that could not be found.

User response: Correct the parameters so that the group definition and the name of the referenced statement agree.

ISF839I *statement-type* **NAMED** *name* **IS NOT REFERENCED BY ANY OTHER STATEMENT.**

Explanation: The indicated statement is valid only when referred to by another statement. It was encountered, but no other statement referred to it.

User response: None required. However, if the statement is to be used, you must correct the parameters so that the statement name is referred to in a parameter in a group definition.

ISF840I *statement* **NAMED** *name* **CONTAINS NO ENTRIES.**

Explanation: The indicated statement contains no column or list entries. It is ignored.

User response: Delete or complete the statement.

ISF841E **GROUP** *group-name* **REFERENCES** *statementname* **WHICH IS AN INVALID TYPE FOR** *group-keyword***.**

Explanation: The indicated group statement references a statement that is the wrong type.

User response: Correct one or both statements.

ISF842E *group-statement* **IN GROUP** *group-name* **IS FOR DISPLAY TYPE** *type* **BUT REFERENCES** *statement* **NAMED** *name* **FOR DISPLAY TYPE** *type***.**

Explanation: The indicated group statement references a statement that is for the wrong SDSF display.

User response: Correct one or both statements.

ISF843E *value* **VALUE REQUIRED FOR THIS** *statement* **STATEMENT.**

Explanation: The indicated statement is missing a required value.

User response: Complete the statement by adding the missing value.

ISF844W *statement* **VALUE** *value* **EXCEEDS THE MAXIMUM ALLOWED, CHANGED TO** *new-value***.**

Explanation: The indicated value in the indicated statement was greater than the maximum allowed; SDSF has changed the value to *new-value*.

User response: Correct the value to be less than or equal to the maximum allowed.

ISF845W *statement* **VALUE** *value* **TOO LONG FOR COLUMN WIDTH, TRUNCATED TO** *number* **CHARACTERS.**

Explanation: The indicated value in the statement type indicated by *statement* is too long for the width of the column. It is truncated to fit the column.

User response: None required. To avoid truncation of the value, correct it to fit the column width, or lengthen the column.

ISF846W **NO GROUPS HAVE BEEN DEFINED.**

Explanation: The ISFPARMS contained no GROUP statements. At least one GROUP statement is required.

User response: Add at least one GROUP statement to the ISFPARMS.

ISF847I **WHEN STATEMENT SELECTED FOR THIS SYSTEM.**

Explanation: The WHEN statement has been selected for this system. All statements that follow the WHEN statement will be processed for this system, until another WHEN statement is encountered.

User response: None required.

ISF848I **WHEN STATEMENT DOES NOT MATCH THIS SYSTEM, FOLLOWING STATEMENTS SKIPPED UNTIL NEXT WHEN.**

Explanation: The WHEN statement specified conditions that do not match the current system. Subsequent statements will be checked for syntax but not processed, until the next WHEN statement is found.

User response: None required.

ISF849I *statement-name* **STATEMENT NOT SELECTED DUE TO PREVIOUS WHEN STATEMENT.**

Explanation: Because it follows a WHEN statement that contained conditions that were not met, the statement is checked for syntax but not otherwise processed.

User response: None required.

ISF850E *primary-statement* **CONTAINS NO** *secondary-statement* **ENTRIES.**

Explanation: A statement, *primary-statement*, was encountered that requires other statements, *secondary-statement*, but no such statements followed it. The statement *primary-statement* is ignored.

User response: Either delete the statement *primary-statement*, or add the required statements indicated by *secondary-statement*.

ISF851E **LOCAL SERVER NOT DEFINED IN SERVER GROUP (SERVER NAME** *server-name***, SYSTEM NAME** *system-name***).**

Explanation: A server group was defined for the indicated server on the indicated system, but the server group did not include the local server. A server group must include the local server. The local server is the currently executing server that is running on this system.

User response: Add a SERVER statement for the local server to the server group definition.

ISF852I *statement-type* **STATEMENT IGNORED,** *statement-type* **IN USE.**

Explanation: An attempt was made to modify an initialization statement after the SDSF server was already active. The statement is ignored.

User response: To change a server group after the server group is in use, you can:

1. Make the change to ISFPARMS.
2. End server communications with the `MODIFY server-name,STOP,C,TERM` command.
3. Use the `MODIFY server-name,REFRESH` command to cause the new ISFPARMS to be processed.

ISF853E **INSUFFICIENT SERVERS DEFINED IN SERVER GROUP.**

Explanation: A SERVERGROUP statement was encountered, but there are not at least two SERVER statements following it. A server group must consist of at least two servers, including the local server. The server group is not defined.

User response: Correct the server group definition so that it includes at least two servers.

ISF854E **NUMBER OF SERVERS IN SERVER GROUP** *number* **EXCEEDS THE MAXIMUM OF** *maximum***.**

Explanation: A SERVERGROUP statement was encountered with more than the maximum number of allowed SERVER statements following it.

User response: Correct the server group definition so that it includes a valid number of servers.

ISF855E **SERVER** *server-name* **DUPLICATES DEFINITION OF SERVER** *server-name* **ON STATEMENT** *statement-number* **FOR SYSTEM** *system-name***, JESNAME** *jes-name***, MEMBER** *member-name***.**

Explanation: A duplicate definition has been detected in the server group for the indicated system, JES, and member. More than one server in the server group is defined as processing a system, JES and member. This is not allowed.

User response: Correct the server group definition in ISFPARMS.

ISF856I **PARAMETER** *parameter* **IS OBSOLETE AND IS IGNORED.**

Explanation: An obsolete parameter has been encountered. It will be ignored.

User response: None required. To avoid seeing this message in the future, delete the parameter from ISFPARMS.

ISF857E **COMMAND IS TOO LONG, MAXIMUM LENGTH ALLOWED IS** *maximum-length***.**

Explanation: The command or parameter being processed causes the resulting command to exceed the valid maximum length.

User response: Ensure that the total length of the command conforms to the valid length.

ISF858E **VALUE '***value***' IS NOT VALID, BEGINS WITH THE RESTRICTED CHARACTERS** *characters***.**

Explanation: The value of an option is not valid because it has a prefix that consists of the restricted characters, *characters*. The option is not processed.

User response: Ensure that the value does not start with restricted characters. For example, the value of the REXX prefix option cannot start with ISF.

ISF859E **COMMAND IS TOO SHORT, MINIMUM LENGTH ALLOWED IS** *minimum-length*.

Explanation: The command being processed is too short.

User response: Ensure that the command conforms to the valid length.

ISF860I *statement* **STATEMENT IGNORED, NOT SUPPORTED IN THIS RELEASE.**

Explanation: The indicated statement in ISFPARMS has been ignored during ISFPARMS processing because it is not supported in this release of SDSF.

User response: None required, though you may want to remove the statement from ISFPARMS or use the WHEN statement to provide conditional processing of the statement.

ISF861I **STATEMENT** *statement* **KEYWORD** *keyword* **IGNORED, NOT SUPPORTED IN THIS RELEASE.**

Explanation: The indicated keyword in ISFPARMS has been ignored during ISFPARMS processing because it is not supported in this release of SDSF.

User response: None required, though you may want to remove the keyword from ISFPARMS or use the WHEN statement to provide conditional processing of the statement that contains it.

ISF862I **KEYWORD** *keyword* **VALUE** *value* **IGNORED, NOT SUPPORTED IN THIS RELEASE.**

Explanation: The indicated value in ISFPARMS has been ignored during ISFPARMS processing because it is not supported in this release of SDSF.

User response: None required, though you may want to change the value in ISFPARMS or use the WHEN statement to provide conditional processing of the statement that contains it.

ISF863E *option* **IS REQUIRED WHEN** *keyword* **IS SPECIFIED.**

Explanation: The command keyword *keyword* requires that option *option* also be specified, but it was omitted. The command or statement is not processed.

User response: Correct the command.

ISF864E **PROPERTY** *property* **VALUE CANNOT BE AN ARRAY.**

Explanation: A JSON document was being processed and *property* was recognized but its value was an array. The property cannot define array values. The document was not processed.

User response: Correct the document and retry the request.

ISF865E **PROPERTY** *property* **VALUE CANNOT BE NUMERIC.**

Explanation: A JSON document was being processed and *property* was recognized but its value was numeric. The property cannot define numeric values. The document was not processed.

User response: Correct the document and retry the request.

ISF866E **PROPERTY** *property* **VALUE CANNOT BE BOOLEAN.**

Explanation: A JSON document was being processed and *property* was recognized but its value was Boolean. The property cannot define Boolean values. The document was not processed.

User response: Correct the document and retry the request.

ISF867E *value-name1* **VALUE** *value1* **IS INCONSISTENT WITH** *value-name2* **VALUE** *value2*.

Explanation: The named values have dependencies that are inconsistent. For example, a starting value is greater than an ending value. The document is not processed.

User response: Correct the document and retry the request.

ISF868E **PROPERTY** *property-name* **VALUE CANNOT BE A STRING.**

Explanation: In a JSON document, *property-name* was recognized. Its value was a string, but the property cannot define string values.

User response: Correct the document and retry the request.

ISF901E **BINARY CONVERSION ERROR OCCURRED IN ISSUING AN SDSF MESSAGE.**

Explanation: In issuing an SDSF message, SDSF encountered a binary conversion error.

User response: Follow your local procedure to call IBM for service.

ISF902E INSERT OF AN INVALID TYPE WAS ENCOUNTERED IN AN SDSF MESSAGE.

Explanation: In issuing an SDSF message, SDSF encountered a problem in inserting a value into a message.

User response: Follow your local procedure to call IBM for service.

ISF903E INVALID INSERT NUMBER WAS ENCOUNTERED IN AN SDSF MESSAGE.

Explanation: In issuing an SDSF message, SDSF encountered a problem in inserting a value into a message.

User response: Follow your local procedure to call IBM for service.

ISF904E SDSF MESSAGE TOO LONG.

Explanation: In issuing an SDSF message, SDSF encountered a message that exceeded the maximum allowed length.

User response: Follow your local procedure to call IBM for service.

ISF905E INCORRECT NUMBER OF INSERTS PASSED FOR AN SDSF MESSAGE.

Explanation: In issuing an SDSF message, SDSF encountered a problem with inserting values into the message.

User response: Follow your local procedure to call IBM for service.

ISF906E SDSF MESSAGE NOT ISSUED, SDSF MESSAGE TABLE NOT LOADED.

Explanation: SDSF could not issue a message because the message table containing the messages was not loaded.

User response: Follow your local procedure to call IBM for service.

ISF908E MESSAGE *message-number* **LINE** *line-number* **NOT FOUND IN MESSAGE TABLE.**

Explanation: SDSF could not issue a message because the message or a line in the multi-line message was not found in the message table.

User response:

Follow your local procedure to call IBM for service.

ISF922E SDSF CONFIGURATION ERROR.

Explanation: SDSF has been invoked incorrectly when running as an ISPF dialog.

User response: The system programmer should correct the invocation of SDSF. For an example of the statements needed to invoke SDSF from the ISPF main menu, refer to member ISF@PRI4 in data set ISF.SISFPLIB and "ISPF considerations" on page 347.

ISF999I DIAG: *diagnostic-data***.**

Explanation: SDSF has encountered an internal condition in which diagnostic data has been collected.

User response: Follow your local procedure for reporting a problem to IBM.

ISF2001E SDSF INVOCATION FAILED, RETURN CODE *return-code***.**

Explanation: The SDSF Java API attempted to perform an SDSF request, but the invocation failed with the indicated return code. The return codes are the standard SDSF return codes documented in the class description for ISFBase.

User response: To determine the source of the error, list the SDSF messages contained in the ISFRequestResults object used for the request.

ISF2002E COMMAND NOT PROVIDED.

Explanation: A method was invoked that requires a command to be provided but the command was missing.

User response: Supply a command as required by the method parameters.

ISF2003E PROPERTY NAME ARRAY DIMENSION DIFFERENT THAN VALUE ARRAY DIMENSION.

Explanation: The requestPropertyChange method was invoked to change the property of an object. However, the number of property names does not match the number of supplied property values.

User response: The property name array must correspond one-to-one with the values supplied in the property value array. Correct the arrays that are passed in to the method.

ISF2004E PROPERTY NAME MISSING IN ARRAY ELEMENT *element-number***.**

Explanation: The requestPropertyChange method was invoked to change the property of an object. However, the number of property names does not match the number of supplied property values.

User response: Correct the property name array.

ISF2005E RESULTS OBJECT NOT PROVIDED.

Explanation: SDSF was invoked to perform a function but the results object was not provided.

User response: Follow your local procedures for contacting IBM for support.

ISF2006E ROW TOKEN WAS NOT PROVIDED FOR OBJECT *object-name***.**

Explanation: An action was attempted against a row object, but the object does not contain a row token. The object name is the fixed field for the object. The action cannot be performed.

User response: Verify that the object was not modified in any way such that the action cannot be performed. Check that the nomodify request setting was not used when the object was originally retrieved.

ISF2007E ROW TOKEN WAS NOT PROVIDED FOR OBJECT *object-name* **IN REPEAT LIST ENTRY** *entry-number***.**

Explanation: An action was attempted against a row object using a repeat list, but the object does not contain a row token. The object name is the fixed field and the entry number is the position of the object in the repeat list.

User response: Verify that the object was not modified in any way such that the action cannot be performed. Check that the nomodify request setting was not used when the object was originally retrieved.

ISF2008E PROPERTY NAME ARRAY NOT PROVIDED.

Explanation: The requestPropertyChange method was invoked to change the property of an object. However, the property name array was not provided.

User response: Supply the property name array.

ISF2009E PROPERTY VALUE ARRAY NOT PROVIDED.

Explanation: The requestPropertyChange method was invoked to change the property of an object. However, the property value array was not provided.

User response: Supply the property value array.

ISF2010E PARAMETER *parameter-name* **MUST HAVE THE VALUE** *parameter-value***.**

Explanation: A method was invoked using *parameter-name*, but the required value was not provided.

User response: Verify the parameter values for the method are correct.

ISF2011E INCONSISTENT INDEXES IN SETTINGS, fromIndex, *from-index***, IS EQUAL TO toIndex,** *to-index***.**

Explanation: The request settings have been used to specify a range of rows to return. However, the range indexes are not consistent because the from-index is equal to the to-index.

User response: Correct the request settings and retry the request.

ISF2012E INCONSISTENT INDEXES IN SETTINGS, fromIndex, *from-index***, IS GREATER THAN toIndex,** *to-index***.**

Explanation: The request settings have been used to specify a range of rows to return. However, the range indexes are not consistent because the from-index is greater than the to-index.

User response: Correct the request settings and retry the request.

ISF2101E SDSF INTERNAL ERROR OCCURRED IN *class-name***#***method-name***, REASON=***reason-code***.**

Explanation: An internal error occurred in the indicated class and method.

User response: Follow your local procedures to contact IBM for support.

ISF2102E TRACE TABLE ENTRY TOO LARGE.

Explanation: An error occurred processing an internal trace entry.

User response: Follow your local procedures to contact IBM for support.

ISF2103E TRACE TABLE TOO LARGE.

Explanation: An error occurred processing the internal trace table.

User response: Follow your local procedures to contact IBM for support.

ISF2104E TRACE TABLE ENTRY TOO SMALL.

Explanation: An error occurred processing an internal trace entry.

User response: Follow your local procedures to contact IBM for support.

ISF2105E TRACE TABLE TOO SMALL.

Explanation: An error occurred processing the internal trace table.

User response: Follow your local procedures to contact IBM for support.

ISF2106E CANNOT CONVERT VALUE *value* **WITH RESULT** *result***.**

Explanation: An error occurred processing an internal trace entry.

User response: Follow your local procedures to contact IBM for support.

ISF2201W RESPONSE LIMIT IN EFFECT, *number* **OF** *total* **OBJECTS RETURNED.**

Explanation: A request limit was set for the current request. The number of objects returned is limited by the request limit in ISFRequestSettings.

User response: None.

ISF2202I PROCESSING STARTED...

Explanation: SDSF has started processing a request.

User response: None.

ISF2203I PROCESSING COMPLETED.

Explanation: SDSF has finished processing a request.

User response: None.

ISF2204E VALUE NOT ALLOWED FOR OPTION "*option***".**

Explanation: A value was specified for option *option*, but the option does not accept values.

User response: Remove the value from the option and retry the request.

ISF2205E VALUE REQUIRED FOR OPTION "*option***".**

Explanation: An option was specified without a value, but the option requires that a value be used.

User response: Add a value to the option and retry the request.

ISF2206I REPORT BEING WRITTEN TO *pathname***.**

Explanation: A report has been requested and is being written to the named path.

User response: None.

ISF2207E UNABLE TO OPEN REPORT FILE *pathname***, REASON=***reason-text***.**

Explanation: An error occurred attempting to open the report file using the named path. The report will be written to stdout.

User response: Ensure the path names a valid path for the report.

ISF2208E UNRECOGNIZED OPTION "*option***".**

Explanation: An unknown option was specified.

User response: Correct the option and try the request again.

ISF2209I PARAMETERS IGNORED.

Explanation: A request was processed that does not accept parameters, but parameters were specified. The parameters are ignored and processing continues.

User response: Remove the unsupported parameters.

ISF2210W RESPONSE LIMIT IN EFFECT, *number* **OBJECTS RETURNED.**

Explanation: A response limit was set for the current request. The number of objects returned is limited by the response limit in ISFRequestSettings.

User response: None required.

Messages for IBM Health Checker for z/OS

This section describes messages that are issued as output of SDSF's checks for IBM Health Checker for z/OS.

ISFH1001I SDSF server *server-name* **is using statements from member** *member-name* **of data set** *dataset-name***.**

Explanation: The SDSF server is active and using the indicated parmlib member from the named data set.

System action: None.

Operator response: None.

System programmer response: None.

Problem determination: None.

Source: z/OS SDSF Operation and Customization

Module: ISFHCPRM

Reference documentation: z/OS SDSF Operation and Customization

Automation: None.

ISFH1002I SDSF server *server-name* is not active, parmlib statements are not being used.

Explanation: The SDSF server is not active. The use of the SDSF parmlib member ISFPRMxx requires that the SDSF server be active.

IBM recommends that you use parmlib member ISFPRMxx rather than assembler macro ISFPARMS to configure SDSF. The statements in ISFPRMxx are easier to define and more dynamic than assembler macros. Some functions, such as sysplex support, are not available using the assembler macros.

System action: In a JES2 environment, SDSF uses the assembler macro ISFPARMS for configuration parameters. In a JES3 environment, SDSF assigns default values.

Operator response: None.

System programmer response: Consider migrating from the assembler macro ISFPARMS to parmlib member ISFPRMxx if you plan on changing any SDSF configuration values from their default settings.

Problem determination: None.

Source: z/OS SDSF Operation and Customization

Module: ISFHCPRM

Reference documentation: z/OS SDSF Operation and Customization

Automation: None.

ISFH1003I SDSF server *server-name* is active but parmlib statements are not being used. A possible syntax error in the statements may exist.

Explanation: The SDSF server is active but parmlib member ISFPRMxx is not being used to configure SDSF. This may be because the SDSF server detected a syntax error in the configuration statements.

System action: In a JES2 environment, SDSF uses the assembler macro ISFPARMS for configuration parameters. In a JES3 environment, SDSF assigns default values.

Operator response: None.

System programmer response: Examine the server initialization log for errors in ISFPRMxx statements. Correct any errors that prevent the statements from being activated and then use the SDSF server refresh command to reprocess the statements.

Problem determination: None.

Source: z/OS SDSF Operation and Customization

Module: ISFHCPRM

Reference documentation: z/OS SDSF Operation and Customization

Automation: None.

ISFH1004I SDSF is not using parmlib statements for its configuration parameters. However, no values have been customized.

Explanation: SDSF is not using parmlib member ISFPRMxx for its configuration parameters, and SDSF-supplied defaults are being used for all values.

System action: If this is a JES2 environment, SDSF is using the assembler macro based ISFPARMS. No values have been changed in ISFPARMS. If this is a JES3 environment, SDSF is using default values and is not using the assembler macro based ISFPARMS.

Operator response: None.

System programmer response: If you plan on changing any SDSF configuration values from their default settings, use parmlib member ISFPRMxx for your configuration changes.

You can use the sample members ISFPRM00 and ISFPRM01 in ISF.SISFJCL to assist you in defining your configuration.

Source: z/OS SDSF Operation and Customization

Module: ISFHCPRM

Reference documentation: z/OS SDSF Operation and Customization

Automation: None.

ISFH1005E SDSF is using assembler macro ISFPARMS for its configuration parameters.

Explanation: SDSF is using the assembler macro based ISFPARMS for its configuration parameters rather than parmlib member ISFPRMxx. ISFPARMS has been customized by the installation.

System action: None.

Operator response: None.

System programmer response: IBM recommends that you use parmlib member ISFPRMxx rather than assembler macro ISFPARMS to configure SDSF. The statements in ISFPRMxx are easier to define and more dynamic than assembler macros. Some functions, such as sysplex support, are not available using the assembler macros.

Consider migrating from the assembler macro ISFPARMS to parmlib member ISFPRMxx.

You can use the migration tool ISFACP, supplied with SDSF, to convert your existing ISFPARMS to the statement format required by parmlib member ISFPRMxx. You can also use the sample members ISFPRM00 and ISFPRM01 in ISF.SISFJCL to define your configuration.

After defining the configuration statements, refer to Chapter 3, "Using the SDSF server," on page 107 for the steps necessary to start the SDSF server and activate the configuration.

Source: z/OS SDSF Operation and Customization

Module: ISFHCPRM

Reference documentation: z/OS SDSF Operation and Customization

Automation: None.

ISFH1006I ISFPARMS module being analyzed has a service level of *service-level***, and a compile date and time of** *compile-date compile-time***.**

Explanation: ISFPARMS will be analyzed for installation customization changes. The service level, compile date, and compile time of the ISFPARMS module that has been found are listed.

This message is only issued when the check is running in verbose mode.

System action: Processing continues.

Operator response: None.

System programmer response: Use the details from the message to determine that the intended level of ISFPARMS has been found on your system.

Source: z/OS SDSF Operation and Customization

Module: ISFHCPRM

Reference documentation: z/OS SDSF Operation and Customization

Automation: None.

ISFH1007I ISFPARMS group structure has been customized. No further analysis of ISFPARMS will be performed.

Explanation: The groups in ISFPARMS have been customized. Either the number of groups has been changed, or the group names have been changed from the defaults supplied by SDSF.

No further analysis of ISFPARMS will be performed to determine if other customizations are present.

System action: No further checking is done to determine which group keywords vary from the SDSF defaults.

Operator response: None.

System programmer response: Assess whether the customization is still required. Consider migrating from the assembler macro ISFPARMS to parmlib member ISFPRMxx if the configuration parameter is required.

You can use the migration tool ISFACP, supplied with SDSF, to convert your existing ISFPARMS to the statement format required by parmlib member ISFPRMxx. You can also use the sample members ISFPRM00 and ISFPRM01 in ISF.SISFJCL to define your configuration.

Source: z/OS SDSF Operation and Customization

Module: ISFHCPRM

Reference documentation: z/OS SDSF Operation and Customization

Automation: None.

ISFH1008I This check is not applicable since SDSF is not enabled for execution on this system.

Explanation: The IFAEDSTA service has indicated that SDSF is not enabled for execution on this system.

System action: The check is disabled and no further checking will be done.

Operator response: None.

System programmer response: If SDSF should be enabled, verify that the statements in the IFAPRDxx member of parmlib are correct.

Problem determination: None.

Source: *z/OS MVS Initialization and Tuning Reference*

Module: ISFHCPRM

Automation: None.

Reference documentation: *z/OS MVS Initialization and Tuning Reference*

ISFH1009I Load of ISFPARMS failed with abend code *abend-code* **reason code** *reason-code***. Analysis of ISFPARMS will not be performed.**

Explanation: The load of the ISFPARMS module failed with the indicated abend and reason codes. In a JES3 environment in which the SDSF JES2 feature is not installed, ISFPARMS will not be present and this error can be ignored.

System action: No analysis of ISFPARMS can be done to determine if it has been customized.

Operator response: None.

System programmer response: Use the abend return

and reason codes to determine why ISFPARMS cannot be loaded.

Problem determination: None.

Source: z/OS MVS System Codes

Module: ISFHCPRM

Automation: None.

Reference documentation: z/OS SDSF Operation and Customization

---

ISFH1010R ISFPARMS Customization Report

Explanation: Header line for SDSF_ISFPARMS_IN_USE check.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Problem determination: None.

Source: None.

Module: ISFHCPRM

Automation: None.

Reference documentation: None.

ISFH1011R S Macro Name Parameter Changed Comments

Explanation: Header line for SDSF_ISFPARMS_IN_USE check.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Problem determination: None.

Source: None.

Module: ISFHCPRM

Automation: None.

Reference documentation: None.

ISFH1012R -- -------- -------- ---------- ------- ----------

Explanation: Header line for SDSF_ISFPARMS_IN_USE check.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Problem determination: None.

Source: None.

Module: ISFHCPRM

Automation: None.

Reference documentation: None.

ISFH1013R *status macro name parameter changed comments*

Explanation: Detail line for SDSF_ISFPARMS_IN_USE check.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Problem determination: None.

Source: None.

Module: ISFHCPRM

Automation: None.

Reference documentation: None.

ISFH1014R Total changes found: *change-count***.**

Explanation: Total line for SDSF_ISFPARMS_IN_USE check.

System action: Processing continues.

Operator response: None.

System programmer response: None.

Problem determination: None.

Source: None.

Module: ISFHCPRM

Automation: None.

Reference documentation: None.

ISFH1015I The class *class-name* **is active.**

Explanation: The indicated SAF class is active, as recommended.

System action: None.

Operator response: None.

System programmer response: None.

Problem determination: None.

Source: None.

Module: ISFHCPRM

Automation: None.

Reference documentation: None.

ISFH1016E The class *class-name* is not active.

Explanation: The indicated SAF class is not active.

System action: If this is a JES2 environment, SDSF will use ISFPARMS to make authorization decisions related to the class. If this is a JES3 environment, requests for authorization that are related to the class will be denied.

Operator response: None.

System programmer response: IBM recommends that the security administrator activate this class and define profiles in it to protect use of SDSF function. In the JES3 environment, use of SAF security is required. The class should be activated and defined with the appropriate profiles so SDSF can be used with JES3.

Problem determination: None.

Source: None.

Module: ISFHCSAF

Automation: None.

Reference documentation: None.

ISFH1017I RACROUTE *request-type* completed. SAF return code *saf-return-code*, return code *return-code*, reason code *reason-code*.

Explanation: The named RACROUTE request issued by the check has completed with the indicated return and reason codes. This message is only issued in debug mode.

System action: None.

Operator response: None.

System programmer response: None.

Problem determination: None.

Source: None.

Module: ISFHCSAF

Automation: None.

Reference documentation: None.

SDSF user abend codes

This section explains the codes that SDSF issues in the case of an abend. The entry for each abend code includes a brief description of the meaning of the code and a suggested response for the system programmer.

The SDSF abend codes are issued in the `SDSF ABEND USER` message described in Chapter 15, "SDSF messages and codes," on page 475 (ISF012I). System abend codes are in the `SDSF ABEND SYSTEM` message (also ISF012I). See the appropriate system codes manual for information on system abend codes.

Table 207. SDSF Abend Codes

Abend Code	Explanation
0003	SDSF could not find the updated index buffer. *System Programmer Response:* Allocate another ISF.HASPINDX data set.
0004	No index buffer is available. *System Programmer Response:* Follow your local procedure to call IBM for service.
0005	Two exclusive requests for the same index buffer have been made. *System Programmer Response:* Follow your local procedure to call IBM for service.
0010	SDSF was invoked in an inconsistent manner. *System Programmer Response:* Check that SDSF was not invoked using an incorrect entry point, such as a line mode invocation using an interactive entry point.
0011	The logical screen size was changed to less than the minimum width of 80 characters. *User Response:* Change the logical screen size to have a width of at least 80 characters.
0012	SDSF detected a non-supported terminal. The terminal has a line length of less than 80 characters. *User Response:* Use a terminal with a line length of at least 80 characters.

Table 207. SDSF Abend Codes (continued)

Abend Code	Explanation
0013	An error has occurred opening the DCB for the index, or a read to the job file control block (JFCB) has failed. *System Programmer Response:* Check for a JCL or hardware error. If you are running SDSF in batch, be sure you have allocated both ISFIN and ISFOUT.
0015	A system initialization error has occurred. *System Programmer Response:* See an accompanying write-to-operator message for more information.
0016	During SDSF initialization, an include or exclude list was being processed that specified an ISFNTBL TYPE=DEST macro. However, the list being processed is not for destinations. SDSF initialization is terminated after all include and exclude lists are processed. Message ISF028E is issued to further describe the error. *System Programmer Response:* Ensure that the ISFNTBL macro is coded correctly for the include or exclude list being processed.
0021	There is an SDSF logic error in ISFENDD. *System Programmer Response:* Follow your local procedure to call IBM for service.
0022	The SYSLOG index data set is full. *System Programmer Response:* Make the SYSLOG index data set larger, or purge some of the SYSLOG output data sets.
0024	SDSF has encountered either an unrecoverable SYSLOG index logic error, or an I/O error. *System Programmer Response:* Check for a possible I/O error. If you find no I/O error, follow your local procedure to call IBM for service.
0025	The SYSLOG index data set is full. *System Programmer Response:* Make the SYSLOG index data set larger, or purge some of the SYSLOG output data sets.

0026	SDSF has encountered either an unrecoverable SYSLOG index logic error, or an I/O error.
	
	System Programmer Response: Check for these possible causes of the error:
	• JES2 has been shut down and restarted without an IPL of MVS. This makes the LOG command inactive. To make the LOG command active again, you must issue these commands:
	`W START`
	`V SYSLOG,HARDCOPY,CMDS,ROUT=ALL`
	The first command starts the LOG task, and the second command causes the LOG to be written to the spool.
	• The HASPINDX data set is being shared between two systems. If you have two or more JES2 systems at different release levels, you must have a unique HASPINDX data set for each JES2 system on which you have SDSF.

0027	SDSF has encountered an unrecoverable SYSLOG spool data error.
	
	System Programmer Response: Check for these possible causes of the error:
	• JES2 has been shut down and restarted without an IPL of MVS. This makes the LOG command inactive. To make the LOG command active again, you must issue these commands:
	`W START`
	`V SYSLOG,HARDCOPY,CMDS,ROUT=ALL`
	The first command starts the LOG task, and the second command causes the LOG to be written to the spool.
	• The HASPINDX data set is being shared between two systems, but the JES2 systems are at different release or maintenance levels. You must have unique HASPINDX data sets for each JES2 system that is not at the same level.

Table 207. SDSF Abend Codes (continued)

Abend Code	Explanation
0028	An error was encountered while attempting to locate, retrieve, or process a SYSOUT data set record. *System Programmer Response:* Follow your local procedure to call IBM for service.
0031	An invalid function code was passed to the SDSF I/O interface routine. *System Programmer Response:* Follow your local procedure to call IBM for service.
0032	An unrecoverable error has occurred in an SDSF storage management routine. A storage request could not be satisfied. *System Programmer Response:* Follow your local procedure for reporting a problem to IBM.
0041	There is a logic error in the SDSF DA panel routine. *System Programmer Response:* Follow your local procedure to call IBM for service.
0053	A dynamic allocation error has occurred. *System Programmer Response:* See the associated write-to-operator message for more information.
0061	The initialization of SDSF under ISPF was unsuccessful. The support for ISPF might have been installed incorrectly, or SDSF might have been put into the TSO authorized command tables. SDSF cannot run from the TSO authorized command tables. *System Programmer Response:* Check the support for ISPF, and be sure that SDSF is not in the TSO authorized command tables.
0071	The terminal has become disconnected, or there is a logic error in the terminal or display routine. *System Programmer Response:* None, if terminal has been disconnected. Otherwise, follow your local procedure to call IBM for service.
0072	SDSF has abended because the Attention key was pressed. *User Response:* Reaccess SDSF.
0073	The menu data set is defective. *System Programmer Response:*If you have made changes to the menu data set, check the changes. If the problem cannot be found, you can replace the installed SDSF panel data set with the original panel data set on the SDSF distribution tape.
0080	A SDSF initialization failure has occurred processing the JES2 checkpoint. Message ISF006I contains the explanatory information. *System Programmer Response:* See the accompanying write-to-operator message for information.
0081	The level of JES2 that SDSF was assembled for does not match the level of JES2 that is being executed. *System Programmer Response:* Ensure that SDSF has been assembled for the proper set of JES2 macro libraries for the execution system. If the JES2 macro libraries were not correct, reassemble SDSF for the correct JES2 macro libraries. See the accompanying ISF020E message for more information on JES2 levels. Also, check the SDSF library concatenations and the library authorizations to be sure the correct level of SDSF is being used.
0082	The level of the SDSF JES2 feature is not compatible with the level of the SDSF base code. This error may occur if maintenance is required by both the SDSF base and feature FMIDs but has been applied to only one of the FMIDs. *System Programmer Response:* Ensure that a consistent level of the SDSF load modules is being used. Check the lnklst data sets for compatible versions of the SISFLOAD and SISFMOD1 data sets. If maintenance has been applied to either SISFLOAD or SISFMOD1, ensure that any co-requisite relationships have been preserved when applying PTFs.
0083	ISFPARMS was found to not be generated at the current level. *System Programmer Response:* ISFPARMS was assembled using an incorrect macro level or with macros that do not correspond to this release. Reassemble ISFPARMS using the correct macro level.

Table 207. SDSF Abend Codes (continued)

Abend Code	Explanation
0091	SDSF has detected an error return code during the execution of an ISPF service. SDSF execution has terminated. *System Programmer Response:* See the accompanying ISF039I message for more information.
0092	A failure occurred when SDSF invoked an ISPF dialog service. *System Programmer Response:* See the accompanying ISF039I message for more information.
0093	SDSF has detected an error return code during the execution of an ISPF service. SDSF execution has terminated. *System Programmer Response:* See the accompanying ISF039I message for more information.
0101	A SYSLOG master index record was not found. *System Programmer Response:* Follow your local procedure to call IBM for service.
0102	A SYSLOG entry has not been found in the SYSLOG master index record. *System Programmer Response:* Follow your local procedure to call IBM for service.
0103	An invalid index pointer was found in the SYSLOG master index record. *System Programmer Response:* Follow your local procedure to call IBM for service.
0104	SDSF has encountered an invalid pointer in the SYSLOG master index record. *System Programmer Response:* Follow your local procedure to call IBM for service.
0105	A logic error has been encountered during SAF processing. Expected parameters were not available; SAF processing is unable to continue. *System Programmer Response:* Follow your local procedure to call IBM for service.
0201	An unrecoverable error has occurred which causes the server to abend. The reason code indicates the cause for the error: **0001** Unable to obtain storage for the CAB **0002** Unable to obtain storage for the SAB **0003** Incorrect execution environment. The server is not running in the correct protect key. Verify that a PPT entry has been defined in the SCHED*xx* member of the parmlib concatenation for program ISFHCTL.
0222	SDSF abended in response to the ABEND command. *System Programmer Response:* The person who issued the ABEND command can print or display the dump that was requested.

Appendix A. SDSF problem management

This topic is a guide to resolving problems with SDSF. It includes hints for observing and identifying a problem and a reference for managing problems.

Observing and identifying a problem

The following are some questions you might ask yourself when you experience a problem with SDSF. They may help you to identify and resolve the problem, or to give needed information to IBM personnel at the IBM Support Center.

- Are you using new levels of JES, ISPF, or TSO? The problem may be in the relationship between SDSF and JES, ISPF, or TSO.
- Was any maintenance applied, or hardware change made, at the time the problem first appeared? The problem may be in the maintenance or hardware change.
- If maintenance has been applied recently, does SDSF run properly when it is removed? Again, the maintenance may have been improperly applied, or may itself have a problem.
- Are all users of SDSF affected by the problem, or just a few users?
- If it is a recurring problem, does it always show the same symptoms?

Gathering information about a problem

Use this section when you need to gather information about a problem with SDSF, either to analyze the problem yourself, or to describe the problem to the IBM Support Center.

Dumps

SDSF requests an SDUMP whenever an abend occurs. This dump will be written to the SYS1.DUMPxx data sets. If the Dump Analysis and Elimination (DAE) component is active, duplicate dumps will be suppressed

When sending module listings to IBM along with a dump, be sure that the module listings have the same date as the date of the modules in the dump.

Trace

The trace facility is used to create trace records containing key environmental data useful for servicing SDSF. Trace records can be written to either a SYSOUT file or a wraparound DASD data set from strategic points throughout the SDSF code.

The trace facility is intended to be used under the direction of IBM Service.

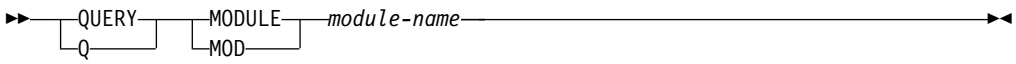
Module information

Use the commands under the direction of IBM Service to gather module level and compile information.

SDSF client

To gather information on the SDSF client, use the QUERY MODULE command. The syntax is as follows:

© Copyright IBM Corp. 1997, 2016

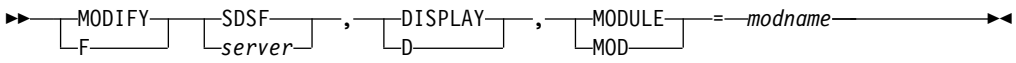


module-name
is the name of the SDSF module. The module must be in ISFVTBL or currently be loaded.

SDSF server

To gather information on the SDSF server, use the MODIFY command. The syntax is as follows:

Display Server Options



module-name
is the name of the SDSF module. The module must be in ISFVTBL or currently be loaded.

SDSF problem index

Problems with the Repeat-Find PF keys (PF5 and PF17)

If you use the Repeat-Find PF keys under ISPF and they don't invoke the Repeat Find function, the problem may be that the SDSF table library was not concatenated correctly with the ISPF table libraries. You may also see the ISPF message RFIND NOT ACTIVE to indicate this. The SDSF Repeat-Find key should be defined as IFIND.

Problems with the LOG, RETRIEVE and TUTOR commands

If you issue a LOG, RETRIEVE, or TUTOR command from ISPF and it does not invoke the SDSF LOG, RETRIEVE, or tutorial function, the problem may be that the SDSF table library was not concatenated correctly with the ISPF table libraries.

Users are experiencing authorization problems

If users are incorrectly being denied authorization to issue commands or access data sets, there are several possible explanations:

- The users are being placed in the wrong authorization group. Have the users issue the WHO command to display their authorization group index, and check the ISFGRP, ISFNTBL, and ISFFLD macros in ISFPARMS or the GROUP, NTBL, and FLD initialization statements to see that they are coded correctly. In ISFPARMS, be sure that you have used commas and continuation characters correctly with macros that occupy more than one line. (The index indicates the group by a count of groups. For example, an index of 3 indicates the group defined by the third GROUP statement in ISFPARMS.)

  Also, if the problem is with issuing MVS or JES2 commands from the command line, check the CMDAUTH parameter for that group. For users to issue MVS or JES2 commands from the command line, ALL must have been specified for CMDAUTH for their group. See the description of the CMDAUTH parameter in "Group authorization parameters (GROUP or ISFGRP)" on page 36.

- For SAF users, SAF resources were not authorized properly. See Chapter 7, “Protecting SDSF functions,” on page 211 for more information on authorizing users to use commands, action characters, overtypeable fields, and jobs using the SAF interface.
- The user exit module, ISFUSER, contains errors. Check any authorization code you have added to the user exit. For more information see Chapter 9, “Using installation exit routines,” on page 329.

If the authorization macros and the user exit appear to be coded correctly, follow your local procedures for calling the IBM Support Center. Have the following documentation ready:

- A description of the panel being used and the action being performed when the problem occurred
- A listing of the authorization parameters, and a listing of the user exit routine, if you have written one
- Output from SDSF trace with mask X'C000'.

SDSF has abended

If the abend message and code, along with the explanations in the documentation, don’t provide you with enough information to resolve the problem, follow your local procedure for calling IBM. Use the ABEND keyword to describe the problem and have the following documentation ready:

- A description of the panel being used and the operation being performed when the abend occurred.
- A record of any messages and abend codes issued. An error message at the system console includes such information as the name of the failing module and the contents of the registers.
- A dump. SDSF should have requested a dump be sent to a SYS1.DUMP data set.
- If the problem is related to the SYSLOG panel, a dump of the SDSF SYSLOG index and a listing of the SYSLOG messages.
- Output from SDSF trace with mask X'C000'.

Documentation is incorrect

Determine whether the problem directly affects your ability to use SDSF. If the problem does directly affect your ability to use SDSF, follow your local procedures for reporting a problem to IBM. Use the DOC keyword when calling IBM.

If the problem does not directly affect your ability to use SDSF, fill out and mail the Readers’ Comment Form that is at the back of this manual, or write to the address shown in the edition notice at the front of this manual.

An SDSF message is incorrect

Follow your local procedure for calling IBM. Have the following documentation ready, using the MSG keyword to describe the problem:

- A description of the panel being used and the operation being performed when the message was received
- A record of the incorrect message

A message was not found in the tutorial

If you get an ISPF message indicating that ISPF could not find an SDSF tutorial message, check to see that the SDSF message library, ISF.SISFMLIB, is concatenated to the ISPF message library.

Data on an SDSF panel is garbled or incorrect

Verify your ISFPARMS assembly condition code. Also, ensure that the SDSF panel library is concatenated correctly.

If the panel library is concatenated correctly, follow your local procedure for calling IBM. Have the following documentation of the problem ready:

- A printout of the screen. To print the screen, use the PRINT SCREEN command if you entered SDSF through TSO, or the ISPF PRINT-HI command if you entered SDSF through ISPF.

RMF exit is not installed

If you are trying to use the sysplex DA support and receive the message `RMF Exit Is Not Installed`, SDSF has not been properly defined to RMF.

SDSF supplies an exit module that must be accessible to the RMF started task. The exit module may reside in the linklst, lpa, or in a steplib defined to the RMF started task. The error message is issued when RMF attempts to load the exit routine but it is not found.

If you are running RMF and want to use the sysplex DA function in SDSF, modules in the SISFLOAD data set must be made accessible to the RMF started task on each system in the sysplex.

If you installed ISF.SISFLOAD in the link list or link pack area, no action is necessary. RMF will be able to load the SDSF modules it needs from the LNKLST or LPA.

If you are running SDSF in a TSO STEPLIB, you will need to add a steplib to the RMF started task procedure. Add the following statement to your RMF procedure JCL for each system in the sysplex:

```
//STEPLIB  DD  DSN=ISF.SISFLOAD,DISP=SHR
```

Communication problems within a server group

Possible causes of communications problems between SDSF servers or between a server are:

- Network connectivity problems. Follow your local procedures for diagnosing network problems.
- WebSphere MQ errors. When an WebSphere MQ error is detected, SDSF issues a message that may contain the return and reason code for the function being performed.
- Timeouts. These occur when the server does not respond in a specified time. This may be normal if the system is heavily loaded or if a large volume of data is requested. If the default timeout is too small, increase it with the TIMEOUT parameter in ISFPARMS. Users can increase the timeout with the SET TIMEOUT command. Users can also reduce timeouts by requesting less data. For example, when accessing the PR panel, they can request data only for specific printers by using parameters on the PR command.

It may be that the server has not responded because:

- The target JES is not in the same MAS as the user, or the target system is not in the same sysplex. The server group definition needs to be changed to correct this. See "Server group definition parameters (SERVERGROUP, SERVER, COMM)" on page 30 for details on defining a server group.
- The server is stopped. Ensure the server is active by issuing this command:

  ```
  F server,D,C
  ```

- There is a problem with the WebSphere MQ configuration. Ensure that WebSphere MQ channels are properly defined and started. See "WebSphere MQ considerations" on page 344 for information on configuring WebSphere MQ.

- Inconsistent levels of JES2. Ensure SDSF has been reassembled on the target system as described in Chapter 10, "Installation and configuration considerations," on page 339.

Appendix B. SAF equivalents for ISFPARMS

You can use this topic as a guide when providing SAF security for the function provided by ISFGRP parameters in ISFPARMS or GROUP statements. The tables list all of the parameters that can be coded on the ISFGRP macro or GROUP statement, except the parameters for the variable field lists.

The parameters and their values are shown with the corresponding SAF equivalent that can be used to protect the function. Some parameters are not related to security and must be specified in the ISFGRP macro or GROUP statement.

For details of the security interface used with SAF, refer to Chapter 5, "Using SAF for security," on page 191.

ACTION

SDSF initialization function. Not applicable to SAF.

ACTIONBAR

SDSF initialization function. Not applicable to SAF.

APPC

SDSF initialization function. Not applicable to SAF.

AUPDT

SDSF initialization function. Not applicable to SAF.

AUTH=ABEND

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.MAINT.ABEND	Gives user authority to issue the ABEND command.

AUTH=ACTION

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.FILTER.ACTION	Gives user authority to issue the ACTION command.

AUTH=ALL

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.**	Gives user authority to issue any SDSF command.

© Copyright IBM Corp. 1997, 2016

AUTH=ALLOPER

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.DSP.* ISFCMD.ODSP.* ISFCMD.FILTER.ACTION ISFCMD.FILTER.DEST ISFCMD.FILTER.FINDLIM ISFCMD.FILTER.OWNER ISFCMD.FILTER.PREFIX ISFCMD.FILTER.RSYS ISFCMD.FILTER.SYSID ISFCMD.FILTER.SYSNAME	Gives user authority to issue any SDSF operator command.

AUTH=ALLUSER

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.DSP.*	Gives user authority to issue any SDSF end user command.

AUTH=APF

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.APF.*system*	Gives user authority to issue the APF command.

AUTH=AS

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.AS.*system*	Gives user authority to issue the AS command.

AUTH=CK

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.HCHECKER.*system*	Gives user authority to issue the CK command.

AUTH=DA

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.DSP.ACTIVE.*jesx*	Gives user authority to issue the DA command.

AUTH=DEST

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.FILTER.DEST	Gives user authority to issue the DEST command. Destination names specified are also checked for authority.
SDSF	READ	ISFOPER.ANYDEST.*jesx*	Equivalent to DEST for the AUTH parameter, with no DEST parameter. Users authorized to the DEST command and to this resource can issue the DEST command using any destination name.
SDSF	READ	ISFAUTH.DEST.*destname*	Equivalent to DEST for the AUTH parameter, with a DEST parameter. In the SAF resource, *destname* is a destination name specified through the DEST parameter.

AUTH=DYNX

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.DYNX.*system*	Gives user authority to issue the DYNX command.

AUTH=ENC

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.ENCLAVE.*system*	Gives user authority to issue the ENC command.

AUTH=ENQ

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.ENQUEUE.*system*	Gives user authority to issue the ENQ command.

AUTH=FINDLIM

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.FILTER.FINDLIM	Gives user authority to issue the FINDLIM command.

AUTH=H

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.DSP.HELD.*jesx*	Gives user authority to issue the H command.

AUTH=I

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.DSP.INPUT.*jesx*	Gives user authority to issue the I command.

AUTH=INIT

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.INITIATOR.*jesx*	Gives user authority to issue the INIT command.

AUTH=INPUT

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.FILTER.INPUT	Gives user authority to issue the INPUT command.

AUTH=JC

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.JOBCLASS.*jesx*	Gives user authority to issue the JC command.

AUTH=J0

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.JOB0.*jesx*	Gives user authority to issue the J0 command.

AUTH=LI

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.LINE.*jesx*	Gives user authority to issue the LI command.

AUTH=LNK

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.LNK.*system*	Gives user authority to issue the LNK command.

AUTH=LOG

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.SYSLOG.*jesx*	Gives user authority to issue the LOG command.

AUTH=LPA

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.LPA.*system*	Gives user authority to issue the LPA command.

AUTH=MAS

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.MAS.*jesx*	Gives user authority to issue the MAS command.

AUTH=NC

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.NC.*jesx*	Gives user authority to issue the NC command.

AUTH=NO

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.NODE.*jesx*	Gives user authority to issue the NO command.

AUTH=NS

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.NS.*jesx*	Gives user authority to issue the NS command.

AUTH=O

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.DSP.OUTPUT.*jesx*	Gives user authority to issue the O command.

AUTH=PAG

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.PAGE.*system*	Gives user authority to issue the PAG command.

AUTH=PARM

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.PARMLIB.*system*	Gives user authority to issue the PARM command.

AUTH=PR

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.PRINTER.*jesx*	Gives user authority to issue the PR command.

AUTH=PREF

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.FILTER.PREFIX	Gives user authority to issue the PREFIX command.

AUTH=PS

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.PROCESS.*system*	Gives user authority to issue the PS command.

AUTH=PUN

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.PUNCH.*jesx*	Gives user authority to issue the PUN command.

AUTH=RDR

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.READER.*jesx*	Gives user authority to issue the RDR command.

AUTH=RES

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.RESOURCE.*system*	Gives user authority to issue the RES command.

AUTH=RM

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.RESMON.*jesx*	Gives user authority to issue the RM command.

AUTH=RSYS

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.FILTER.RSYS	Gives user authority to issue the RSYS command.

AUTH=SE

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.DSP.SCHENV.*system*	Gives user authority to issue the SE command.

AUTH=SO

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.SO.*jesx*	Gives user authority to issue the SO command.

AUTH=SP

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.SPOOL.*jesx*	Gives user authority to issue the SP command.

AUTH=SR

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.SR.*system*	Gives user authority to issue the SR command.

AUTH=ST

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.DSP.STATUS.*jesx*	Gives user authority to issue the ST command.

AUTH=SYS

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.SYSTEM.*system*	Gives user authority to issue the SYS command.

AUTH=SYSID

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.FILTER.SYSID	Gives user authority to issue the SYSID command.

AUTH=SYSNAME

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.FILTER.SYSNAME	Gives user authority to issue the SYSNAME command.

AUTH=TRACE

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.MAINT.TRACE	Gives user authority to issue the TRACE command.

AUTH=ULOG

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFCMD.ODSP.ULOG.*jesx*	Gives user authority to issue the ULOG command.

BROWSE

SDSF initialization function. Not applicable to SAF.

CMDAUTH=DEST

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFOPER.DEST.*jesx*	SDSF does further checking for authority to jobs and output based on destination (destination operator authority).
SDSF	ALTER	ISFAUTH.DEST.*destname*	Used with the above ISFOPER.DEST.*jesx* resource, is equivalent to DEST for CMDAUTH with a DEST parameter, when *destname* is a destination name specified through the DEST parameter.
WRITER	ALTER	*jesx*.LOCAL.*devicename* *jesx*.RJE.*devicename*	Authorizes user to specific LOCAL or RJE printers or punches based on *devicename*.

CMDAUTH=DISPLAY

Class	Access	SDSF Resource Name	Description
JESSPOOL	READ	*node.userid.jobname.jobid* *node.userid.jobname.jobid*.GROUP.*ogroupid*	Gives users authority to issue D and L action characters for any job or output group to which they have READ access.

CMDAUTH=GROUP

Class	Access	SDSF Resource Name	Description
JESSPOOL	ALTER	*node.userid.jobname.jobid* *node.userid.jobname.jobid*.GROUP.*ogroupid*	Equivalent to GROUP for CMDAUTH, when *jobname* is the group prefix. (With structured TSO user IDs, you can specify *userid* instead of *jobname*.)

CMDAUTH=INIT

Class	Access	SDSF Resource Name	Description
SDSF	CONTROL	ISFINIT.I(*xx*).*jesx*	Equivalent to INIT for CMDAUTH, when *xx* is the initiator identifier.

CMDAUTH=NOTIFY

No direct SAF equivalent. To provide comparable authority, see “Providing function comparable to NOTIFY authority” on page 248.

CMDAUTH=MSG

Logging of user access to resources is controlled by the security product.

CMDAUTH=USERID

Class	Access	SDSF Resource Name	Description
JESSPOOL	ALTER	*node.userid.jobname.jobid* *node.userid.jobname.jobid*.GROUP.*ogroupid*	Equivalent to USERID for CMDAUTH, when *userid* is the name of the job the user is trying to access. (Even when no profiles are defined in the JESSPOOL class, users are authorized to output that they own.)

CMDAUTH=ALL

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFOPER.SYSTEM	Gives user authority to issue the SDSF / command. When used in conjunction with OPERCMDS profiles, the system (MVS or JES2) checks authority to the MVS or JES2 commands issued.
SDSF	READ	ISFOPER.DEST.*jesx*	SDSF does further checking for authority to jobs and output based on destination (destination operator authority).
SDSF	ALTER	ISFAUTH.DEST.*destname*	Used with the above ISFOPER.DEST.*jesx* resource, is equivalent ALL for CMDAUTH, with no DEST parameter. Use generic profiles to give authority to all jobs and output.
WRITER	ALTER	*jesx*.LOCAL.*devicename* *jesx*.RJE.*devicename*	Use generic profiles to give authority to all printers and punches.
SDSF	ALTER	ISFINIT.I(*xx*).*jesx*	Use generic profiles to give authority to all initiators.

CMDLEV

Class	Access	SDSF Resource Name	Description
SDSF	UPDATE	ISFATTR.JOB.*field* ISFATTR.OUTPUT.*field* ISFATTR.OUTDESC.*field* ISFATTR.CHECK.*field* ISFATTR.ENCLAVE.*field* ISFATTR.JOBCL.*field* IISFATTR.LINE.*field* ISFATTR.MEMBER.*field* ISFATTR.NETOPTS.*field* ISFATTR.NODE.*field* ISFATTR.OFFLOAD.*field* SFATTR.PROPTS.*field* ISFATTR.RDR.*field* ISFATTR.RESMON.*field* ISFATTR.RESOURCE.*field* ISFATTR.SPOOL.*field* ISFATTR.MODIFY.*field* ISFATTR.SELECT.*field*	Authorizes use of overtypeable fields. See the SDSF resource name for each field in Table 101 on page 259.

Class	Access	SDSF Resource Name	Description
OPERCMDS		Depends on the generated MVS or JES2 command.	See the OPERCMDS resource names in Table 76 on page 228 and Table 103 on page 279.

CONFIRM

SDSF initialization function. Not applicable to SAF.

CPUFMT

SDSF initialization function. Not applicable to SAF.

CTITLE

SDSF initialization function. Not applicable to SAF.

CURSOR

SDSF initialization function. Not applicable to SAF.

DADFLT

SDSF initialization function. Not applicable to SAF.

DATE

SDSF initialization function. Not applicable to SAF.

DATESEP

SDSF initialization function. Not applicable to SAF.

DEST

Class	Access	SDSF Resource Name	Description
SDSF	ALTER	ISFAUTH.DEST.*destname*	Equivalent to DEST for the CMDAUTH parameter, with a DEST parameter. In the SAF resource, *destname* is a destination name specified through the DEST parameter.
SDSF	READ	ISFAUTH.DEST.*destname*.D*dsid*.*dsname*	Equivalent to DEST for the DSPAUTH parameter, with a DEST parameter. In the SAF resource, *destname* is a destination name specified through the DEST parameter.

DISPLAY

SDSF initialization function. Not applicable to SAF.

DSPAUTH=ADEST

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFOPER.DEST.*jesx*	SDSF does further checking for authority to jobs and output based on destination (destination operator authority).
SDSF	READ	ISFAUTH.DEST.*destname*. DATASET.*d*sname	Equivalent to ADEST for the DSPAUTH parameter, with a DEST parameter. In the SAF resource, *destname* is a destination name specified through the DEST parameter.

DSPAUTH=ALL

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFOPER.DEST.*jesx*	SDSF does further checking for authority to jobs and output based on destination (destination operator authority).
SDSF	READ	ISFAUTH.DEST.*destname*	Equivalent to ALL for the DSPAUTH parameter, with a DEST parameter. In the SAF resource, *destname* is a destination name specified through the DEST parameter.

DSPAUTH=AMDEST

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFOPER.DEST.*jesx*	SDSF does further checking for authority to jobs and output based on destination (destination operator authority).
SDSF	READ	ISFAUTH.DEST.*destname.* DATASET.JESMSGLG ISFAUTH.DEST.*destname.* DATASET.JESJCL ISFAUTH.DEST.*destname.* DATASET.JESYSMSG	Equivalent to AMDEST for the DSPAUTH parameter, with a DEST parameter, when JESMSGLG, JESJCL, JESYSMSG are data set names of JES2 message data sets and *destname* is a destination name specified through the DEST parameter.

DSPAUTH=AMSG

Class	Access	SDSF Resource Name	Description
JESSPOOL	READ	*node.userid.jobname.jobid.* D*ddsid.*JESMSGLG *node.userid.jobname.jobid.* D*ddsid.*JESJCL *node.userid.jobname.jobid.* D*ddsid.*JESYSMSG	Equivalent to AMSG for the DSPAUTH parameter, when JESMSGLG, JESJCL, JESYSMSG are data set names of JES2 message data sets. (You can define generic profiles for the above AMDEST resources to obtain equivalent function.)

DSPAUTH=GROUP

Class	Access	SDSF Resource Name	Description
JESSPOOL	READ	*node.userid.jobname.jobid*. D*dsid.dsname*	Equivalent to GROUP for the DSPAUTH parameter, when *jobname* is the group prefix. (With structured TSO user IDs, you can specify *userid* instead of *jobname*.)

DSPAUTH=GRPMSG

Class	Access	SDSF Resource Name	Description
JESSPOOL	READ	*node.userid.jobname.jobid*. D*dsid*.JESMSGLG *node.userid.jobname.jobid*. D*dsid*.JESJCL *node.userid.jobname.jobid*. D*dsid*.JESYSMSG	Equivalent to GRPMSG for the DSPAUTH parameter, when JESMSGLG, JESJCL, JESYSMSG are data set names of JES2 message data sets and *jobname* is the group prefix.

DSPAUTH=NOTIFY

No direct SAF equivalent. To provide comparable authority, see "Providing function comparable to NOTIFY authority" on page 248.

DSPAUTH=USERID

Class	Access	SDSF Resource Name	Description
JESSPOOL	READ	*node.userid.jobname.jobid*. D*dsid.dsname*	Equivalent to USERID for the DSPAUTH parameter, when *userid* is the name of the job the user is trying to access. (Even when no profiles are defined in the JESSPOOL class, users are authorized to output that they own.)

EMCSAUTH

SDSF initialization function. Not applicable to SAF.

EMCSREQ

SDSF initialization function. Not applicable to SAF.

GPLEN

SDSF initialization function. Not applicable to SAF.

GPREF

SDSF initialization function. Not applicable to SAF.

ICMD

Class	Access	SDSF Resource Name	Description
JESSPOOL	ALTER	*node.userid.jobname.jobid* *node.userid.jobname.jobid*. GROUP.*ogroupid*	Equivalent to the ICMD parameter, when *jobname* is a job name specified by the associated ISFNTBL macro or NTBL statement.

IDEST

Class	Access	SDSF Resource Name	Description
SDSF	READ	ISFOPER.ANYDEST.*jesx*	If users do not have an IDEST parameter with initial destinations specified, they must have READ access to this resource, or no jobs can appear on the panels.
SDSF	READ	ISFAUTH.DEST.*destname*	SDSF initialization function. Users must be authorized to the *destnames* that correspond to the initial destination values specified by their IDEST parameter. If not, no jobs can appear on the panels.

IDSP

Class	Access	SDSF Resource Name	Description
JESSPOOL	READ	*node.userid.jobname.jobid*. D*ddsid.dsname*	Equivalent to the IDSP parameter, when *jobname* is a job name specified by the associated ISFNTBL macro or NTBL statement.

IDSPD

Class	Access	SDSF Resource Name	Description
JESSPOOL	READ	*node.userid.jobname.jobid.* D*ddsid.*JESMSGLG *node.userid.jobname.jobid.* D*ddsid.*JESJCL *node.userid.jobname.jobid.* D*ddsid.*JESYSMSG	Equivalent to the IDSPD parameter, when JESMSGLG, JESJCL, JESYSMSG are data set names of JES message data sets.

ILOGCOL

SDSF initialization function. Not applicable to SAF.

INPUT

SDSF initialization function. Not applicable to SAF.

ISTATUS

Includes jobs on the SDSF panels based on job name. To provide equivalent function, see "Table build exit point" on page 337.

ISYS

SDSF initialization function. Not applicable to SAF.

LANG

SDSF initialization function. Not applicable to SAF.

LOG/LOGOPT

SDSF initialization function. Not applicable to SAF.

OWNER

SDSF initialization function. Not applicable to SAF.

PREFIX

SDSF initialization function. Not applicable to SAF.

RSYS

SDSF initialization function. Not applicable to SAF.

SYSID

SDSF initialization function. Not applicable to SAF.

UPCTAB

SDSF initialization function. Not applicable to SAF.

VALTAB

SDSF initialization function. Not applicable to SAF.

VIO

SDSF initialization function. Not applicable to SAF.

XCMD

Class	Access	SDSF Resource Name	Description
JESSPOOL	NONE	*node.userid.jobname.jobid* *node.userid.jobname.jobid*.GROUP.*ogroupid*	Equivalent to the XCMD parameter, when *jobname* is a job name specified by the associated ISFNTBL macro or NTBL statement and the access is NONE.

XDSP

Class	Access	SDSF Resource Name	Description
JESSPOOL	NONE	*node.userid.jobname.jobid.* D*ddsid.dsname*	Equivalent to the XDSP parameter, when *jobname* is a job name specified by the associated ISFNTBL macro or NTBL statement and the access is NONE.

XDSPD

Class	Access	SDSF Resource Name	Description
JESSPOOL	NONE	*node.userid.jobname.jobid.* D*ddsid.dsname*	Equivalent to the XDSPD parameter, when *jobname* is a job name specified by the associated ISFNTBL macro or NTBL statement and the access is NONE.
JESSPOOL	READ	*node.userid.jobname.jobid.* D*ddsid*.JESMSGLG *node.userid.jobname.jobid.* D*ddsid*.JESJCL *node.userid.jobname.jobid.* D*ddsid*.JESYSMSG	User must then be authorized to the message data sets for the job.

XSTATUS

Excludes jobs from the SDSF panels based on a job's name. To provide equivalent function, see “Table build exit point” on page 337.

Appendix C. SDSF resource names for SAF security

The following tables contain a list of all the resource names you need to use SAF security. See Chapter 5, "Using SAF for security," on page 191 for more information about using the SAF Security Interface.

Table 208. Security Classes, Resource Names, and What They Protect

Class	Resource Name	Protects
JESSPOOL	*nodeid.userid.jobname.jobid*	Jobs
JESSPOOL	*nodeid.userid.jobname.jobid.* GROUP.*ogroupid*	Output groups
JESSPOOL	*nodeid.userid.jobname.jobid.* D*dsid.dsname*	SYSIN/SYSOUT data sets
JESSPOOL	*nodeid*.+MASTER+.SYSLOG.SYSTEM. *sysname*	Access to the JES logical log, for displaying the SYSLOG
LOGSTRM	See "OPERLOG" on page 255.	Log stream used for OPERLOG
LOGSTRM	See "Checks on the CK and CKH panels" on page 238.	Log stream for check history (CKH panel)
OPERCMDS	See Chapter 7, "Protecting SDSF functions," on page 211.	MVS and JES generated commands
OPERCMDS	*server-name*.MODIFY.DEBUG	DEBUG parameter of MODIFY
OPERCMDS	*server-name*.MODIFY.DISPLAY	DISPLAY parameter of MODIFY
OPERCMDS	*server-name*.MODIFY.FOLDMSG	FOLDMSG parameter of MODIFY
OPERCMDS	*server-name*.MODIFY.LOGCLASS	LOGCLASS parameter of MODIFY
OPERCMDS	*server-name*.MODIFY.REFRESH	REFRESH parameter of MODIFY
OPERCMDS	*server-name*.MODIFY.START	START parameter of MODIFY
OPERCMDS	*server-name*.MODIFY.STOP	STOP parameter of MODIFY
OPERCMDS	*server-name*.MODIFY.TRACE	TRACE parameter of MODIFY
OPERCMDS	*server-name*.MODIFY.TRCLASS	TRCLASS parameter of MODIFY
SDSF	GROUP.*group-name.server-name*	Membership in groups defined in ISFPARMS
SDSF	ISFCMD.DSP.ACTIVE.*jesx*	DA panel command
SDSF	ISFCMD.DSP.HELD.*jesx*	H panel command
SDSF	ISFCMD.DSP.INPUT.*jesx*	I panel command
SDSF	ISFCMD.DSP.OUTPUT.*jesx*	O panel command
SDSF	ISFCMD.DSP.SCHENV.*system*	SE panel command
SDSF	ISFCMD.DSP.STATUS.*jesx*	ST panel command
SDSF	ISFCMD.ODSP.APF.*system*	APF panel command
SDSF	ISFCMD.ODSP.AS.*system*	AS panel command

© Copyright IBM Corp. 1997, 2016

Table 208. Security Classes, Resource Names, and What They Protect (continued)

Class	Resource Name	Protects
SDSF	ISFCMD.ODSP.DYNX.*system*	DYNX panel command
SDSF	ISFCMD.ODSP.ENCLAVE.*system*	ENC panel command
SDSF	ISFCMD.ODSP.ENQUEUE.*system*	ENQ panel command
SDSF	ISFCMD.ODSP.HCHECKER.*system*	CK panel command
SDSF	ISFCMD.ODSP.INITIATOR.*jesx*	INIT panel command
SDSF	ISFCMD.ODSP.JOBCLASS.jesx	JC panel command
SDSF	ISFCMD.ODSP.JOB0.*jesx*	J0 panel command
SDSF	ISFCMD.ODSP.LINE.*jesx*	LI panel command
SDSF	ISFCMD.ODSP.LNK.*system*	LNK panel command
SDSF	ISFCMD.ODSP.LPA.*system*	LPA panel command
SDSF	ISFCMD.ODSP.MAS.*jesx*	MAS panel command
SDSF	ISFCMD.ODSP.NC.*jesx*	NC panel command
SDSF	ISFCMD.ODSP.NODE.*jesx*	NO panel command
SDSF	ISFCMD.ODSP.NS.*jesx*	NS panel command
SDSF	ISFCMD.ODSP.PAGE.*system*	PAGE panel command
SDSF	ISFCMD.ODSP.PARMLIB.*system*	PARM panel command
SDSF	ISFCMD.ODSP.PRINTER.*jesx*	PR panel command
SDSF	ISFCMD.ODSP.PROCESS.*system*	PS panel command
SDSF	ISFCMD.ODSP.PUNCH.*jesx*	PUN panel command
SDSF	ISFCMD.ODSP.READER.*jesx*	RDR panel command
SDSF	ISFCMD.ODSP.RESMON.*jesx*	RM panel command
SDSF	ISFCMD.ODSP.RESOURCE.*system*	RES panel command
SDSF	ISFCMD.ODSP.SO.*jesx*	SO panel command
SDSF	ISFCMD.ODSP.SPOOL.*jesx*	SP panel command
SDSF	ISFCMD.ODSP.SR.*system*	SR panel command
SDSF	ISFCMD.DSP.SYMBOL.*system*	SYM panel command
SDSF	ISFCMD.ODSP.SYSTEM.*system*	SYS panel command
SDSF	ISFCMD.ODSP.SYSLOG.*jesx*	LOG panel command
SDSF	ISFCMD.ODSP.ULOG.*jesx*	ULOG panel command
SDSF	ISFCMD.FILTER.ACTION	ACTION command
SDSF	ISFCMD.FILTER.DEST	DEST command
SDSF	ISFCMD.FILTER.FINDLIM	FINDLIM command
SDSF	ISFCMD.FILTER.INPUT	INPUT command
SDSF	ISFCMD.FILTER.OWNER	OWNER command
SDSF	ISFCMD.FILTER.PREFIX	PREFIX command
SDSF	ISFCMD.FILTER.RSYS	RSYS command
SDSF	ISFCMD.FILTER.SYSID	SYSID command
SDSF	ISFCMD.FILTER.SYSNAME	SYSNAME command
SDSF	ISFCMD.MAINT.ABEND	ABEND command
SDSF	ISFCMD.MAINT.TRACE	TRACE command

Table 208. Security Classes, Resource Names, and What They Protect (continued)

Class	Resource Name	Protects
SDSF	ISFCMD.OPT.SERVER	SERVER parameter of SDSF command
SDSF	ISFDISP.DELAY.*owner.jobname*	JY action character on the DA panel
SDSF	ISFOPER.SYSTEM	Command line commands
SDSF	ISFOPER.DEST.*jesx*	Operator authority
SDSF	ISFAPF.*datasetname*	APF data sets
SDSF	ISFDYNX.*exitname*	DYNX data sets
SDSF	ISFENQ.*majorname.sysname*	Enqueues
SDSF	ISFLNK.*datasetname*	LnkLst data sets
SDSF	ISFLPA.*datasetname*	LPA data sets
SDSF	ISFPARM.*datasetname*	Parmlib data sets
SDSF	ISFPAG.*datasetname*	Page data sets
SDSF	ISFSYM.*symbolname.sysname*	System symbols
SDSF	ISFSYS.*sysplexname.systemname*	Systems
SDSF	ISFAUTH.DEST.*destname*	Operator destinations for command objects and destination names for the DEST command
SDSF	ISFAUTH.DEST.*destname*.DATASET.*dsname* ISFAUTH.DEST.DATASET.*dsname*	Operator destination to browse objects
SDSF	ISFOPER.ANYDEST.*jesx*	All destinations for the DEST command
SDSF	ISFENC.*subsys-type.subsys-name*	Enclaves
SDSF	ISFINIT.I(*xx*).*jesx*	Initiators
SDSF	ISFJDD.CF.*sysname*	Coupling facility on the JD panel
SDSF	ISFJDD.IP.*sysname*	TCP/IP server on the JD panel
SDSF	ISFJOBCL.*class.jesx*	Job classes
SDSF	ISFLINE.*device-name.jesx*	Lines
SDSF	ISFAPPL.*device-name.jesx* ISFSOCK.*device-name.jesx* ISFLINE*device-name.jesx*	Network connections
SDSF	ISFNODE.*node-name.jesx*	Nodes
SDSF	ISFNS.*device-name.jesx*	Network servers
SDSF	ISFPROC.*owner.jobname*	z/OS UNIX processes
SDSF	ISFSO.*device-name.jesx*	Offloaders
SDSF	ISFRDR.*device-name.jesx*	Readers
SDSF	ISFRES.*resource.system*	WLM resources
SDSF	ISFRM.*resource.jesx*	JES resources
SDSF	ISFSE.*sched-env.system*	Scheduling environments
SDSF	ISFSP.*volser.jesx*	Spool volumes

Table 208. Security Classes, Resource Names, and What They Protect (continued)

Class	Resource Name	Protects
SDSF	ISFSR.*msg-type.system.jobname*	System requests, where *message-type* is ACTION or REPLY
SDSF	SERVER.NOPARM	Fall-back to ISFPARMS in assembler format
WRITER	*jesx*.LOCAL.*devicename*	Local printers and punches, including those on other systems
WRITER	*jesx*.RJE.*devicename*	RJE devices
XFACILIT	HZS.*sysname.checkowner.checkname. action* where *action* is ACTIVATE, DEACTIVATE, DELETE, QUERY, REFRESH, RUN, UPDATE or MESSAGES	IBM Health Checker for z/OS

Table 209. SDSF Class Resource Names and Overtypeable Fields

SDSF Resource Name (UPDATE Authority Required)	Overtypeable Field	Panel
ISFATTR.CHECK.CATEGORY	CATEGORY	CK
ISFATTR.CHECK.DEBUG	DEBUG	CK
ISFATTR.CHECK.EINTERVAL	EINTERVAL	CK
ISFATTR.CHECK.INTERVAL	INTERVAL	CK
ISFATTR.CHECK.PARM	PARAMETERS	CK
ISFATTR.CHECK.SEVERITY	SEVERITY	CK
ISFATTR.CHECK.USERDATE	USERDATE	CK
ISFATTR.CHECK.VERBOSE	VERBOSE	CK
ISFATTR.CHECK.WTOTYPE	WTOTYPE	CK
ISFATTR.ENCLAVE.SRVCLASS	SRVCLASS	ENC
ISFATTR.INIT.ALLOC	ALLOC	INIT
ISFATTR.INIT.BARRIER	BARRIER	INIT
ISFATTR.INIT.DEFCNT	DEFCOUNT	INIT
ISFATTR.INIT.GROUP	GROUP	INIT
ISFATTR.INIT.MODE	MODE	INIT
ISFATTR.INIT.UNALLOC	UNALLOC	INIT
ISFATTR.JOB.CLASS	C	I ST
ISFATTR.JOB.EXECNODE	EXECNODE	I ST
ISFATTR.JOB.PGN	PGN	DA
ISFATTR.JOB.PRTDEST	PRTDEST	I ST
ISFATTR.JOB.PRTY	PRTY	I ST
ISFATTR.JOB.QUIESCE	QUIESCE	DA
ISFATTR.JOB.SCHENV	SCHEDULING-ENV	I ST
ISFATTR.JOB.SRVCLASS	SRVCLASS	DA
ISFATTR.JOB.SRVCLS	SRVCLASS	I ST

Table 209. SDSF Class Resource Names and Overtypeable Fields (continued)

SDSF Resource Name (UPDATE Authority Required)	Overtypeable Field	Panel
ISFATTR.JOB.SYSAFF	SAFF	I ST
ISFATTR.JOBCL.ACCT	ACCT	JC
ISFATTR.JOBCL.ACTIVE	ACTIVE	JC
ISFATTR.JOBCL.AUTH	AUTH	JC
ISFATTR.JOBCL.BLP	BLP	JC
ISFATTR.JOBCL.COMMAND	COMMAND	JC
ISFATTR.JOBCL.CONDPURG	CPR	JC
ISFATTR.JOBCL.COPY	CPY	JC
ISFATTR.JOBCL.GROUP	GROUP	JC
ISFATTR.JOBCL.HOLD	HOLD	JC
ISFATTR.JOBCL.IEFUJP	UJP	JC
ISFATTR.JOBCL.IEFUSO	USO	JC
ISFATTR.JOBCL.JCLIM	JCLIM	JC
ISFATTR.JOBCL.JESLOG	JESLOG	JC
ISFATTR.JOBCL.JLOG	LOG	JC
ISFATTR.JOBCL.JOBRC	JOBRC	JC
ISFATTR.JOBCL.JOURNAL	JRNL	JC
ISFATTR.JOBCL.MODE	MODE	JC
ISFATTR.JOBCL.MSGCLASS	MC	JC
ISFATTR.JOBCL.MSGLEVEL	MSGLV	JC
ISFATTR.JOBCL.ODISP	ODISP	JC
ISFATTR.JOBCL.OUTPUT	OUT	JC
ISFATTR.JOBCL.PARTNAME	PARTNAME	JC
ISFATTR.JOBCL.PGMRNAME	PGNM	JC
ISFATTR.JOBCL.PGN	PGN	JC
ISFATTR.JOBCL.PROCLIB	PL	JC
ISFATTR.JOBCL.QHELD	QHLD	JC
ISFATTR.JOBCL.REGION	REGION	JC
ISFATTR.JOBCL.RESTART	RST	JC
ISFATTR.JOBCL.SCAN	SCN	JC
ISFATTR.JOBCL.SCHENV	SCHEDULING-ENV	JC
ISFATTR.JOBCL.SDEPTH	SDEPTH	JC
ISFATTR.JOBCL.SWA	SWA	JC
ISFATTR.JOBCL.SYSSYM	SYSSYM	JC
ISFATTR.JOBCL.TDEPTH	TDEPTH	JC
ISFATTR.JOBCL.TIME	MAX-TIME	JC
ISFATTR.JOBCL.TYPE26	TP26	JC
ISFATTR.JOBCL.TYPE6	TP6	JC
ISFATTR.JOBCL.XBM	XBM	JC

Table 209. SDSF Class Resource Names and Overtypeable Fields (continued)

SDSF Resource Name (UPDATE Authority Required)	Overtypeable Field	Panel
ISFATTR.JOBGROUP.SCHENV	SCHEDULING-ENV	JG
ISFATTR.JOBGROUP.SYSAFF	SAFF	JG
ISFATTR.LINE.TRANSPARENCY	TRANSP	LI
ISFATTR.LINE.APPLID	APPLID	LI
ISFATTR.LINE.AUTODISC	ADISC	LI
ISFATTR.LINE.CODE	CODE	LI
ISFATTR.LINE.COMPRESS	COMP	LI
ISFATTR.LINE.DUPLEX	DUPLEX	LI
ISFATTR.LINE.INTERFACE	INTF	LI
ISFATTR.LINE.JRNUM	JRNUM	LI
ISFATTR.LINE.JTNUM	JTNUM	LI
ISFATTR.LINE.LINECCHR	LINECCHR	LI
ISFATTR.LINE.LOG	LOG	LI
ISFATTR.LINE.NODE	NODE	LI
ISFATTR.LINE.PASSWORD	PASSWORD	LI
ISFATTR.LINE.REST	REST	LI NC
ISFATTR.LINE.SPEED	SPEED	LI
ISFATTR.LINE.SRNUM	SRNUM	LI
ISFATTR.LINE.STNUM	STNUM	LI
ISFATTR.LOGON.PASSWORD	PASSWORD	NS
ISFATTR.MEMBER.CKPTHOLD	CKPTHOLD	MAS
ISFATTR.MEMBER.DORMANCY	DORMANCY	MAS
ISFATTR.MEMBER.SELMNAME	SELECTMODENAME	JP
ISFATTR.MEMBER.SPARTN	PARTNAME	JP
ISFATTR.MEMBER.SYNCTOL	SYNCTOL	MAS
ISFATTR.MODIFY.BURST	MBURST	SO
ISFATTR.MODIFY.CLASS	MCLASS	SO
ISFATTR.MODIFY.DEST	MDEST	SO
ISFATTR.MODIFY.FCB	MFCB	SO
ISFATTR.MODIFY.FLASH	MFLH	SO
ISFATTR.MODIFY.FORMS	MFORMS	SO
ISFATTR.MODIFY.HOLD	MHOLD	SO
ISFATTR.MODIFY.ODISP	MODSP	SO
ISFATTR.MODIFY.PRMODE	MPRMODE	SO
ISFATTR.MODIFY.SYSAFF	MSAFF	SO
ISFATTR.MODIFY.UCS	MUCS	SO
ISFATTR.MODIFY.WRITER	MWRITER	SO
ISFATTR.NETOPTS.APPL	APPL	NS
ISFATTR.NETOPTS.CONNECT	CONNECT	LI NC NO

Table 209. SDSF Class Resource Names and Overtypeable Fields (continued)

SDSF Resource Name (UPDATE Authority Required)	Overtypeable Field	Panel
ISFATTR.NETOPTS.CTIME	CONN-INT	LI NC NO
ISFATTR.NETOPTS.IPNAME	IPNAME	NC NS
ISFATTR.NETOPTS.LINE	LINE	NC
ISFATTR.NETOPTS.LOG	LOG	NS
ISFATTR.NETOPTS.LOGON	LOGON	NC
ISFATTR.NETOPTS.NETSRV	NETSRV	NC
ISFATTR.NETOPTS.NETSRV	SRVNAME	NC
ISFATTR.NETOPTS.NODE	ANODE	NC
ISFATTR.NETOPTS.PORT	PORT	NC NS
ISFATTR.NETOPTS.SECURE	SECURE	NC NO NS
ISFATTR.NETOPTS.SOCKET	SOCKET	NS
ISFATTR.NETOPTS.STACK	STACK	NS
ISFATTR.NODE.AUTHORITY	AUTHORITY	NO
ISFATTR.NODE.COMPACT	COMPACT	NC
ISFATTR.NODE.COMPACT	CP	NO
ISFATTR.NODE.DIRECT	DIRECT	NO
ISFATTR.NODE.ENDNODE	END	NO
ISFATTR.NODE.HOLD	HOLD	NO
ISFATTR.NODE.JRNUM	JRNUM	NO
ISFATTR.NODE.JTNUM	JTNUM	NO
ISFATTR.NODE.LINE	LINE	NC NO
ISFATTR.NODE.LOGMODE	LOGMODE	NC NO
ISFATTR.NODE.LOGON	LOGON	NO
ISFATTR.NODE.MAXRETR	MAXRETRIES	NO
ISFATTR.NODE.NETHOLD	NHOLD	NO
ISFATTR.NODE.NETSRV	NETSRV	NO
ISFATTR.NODE.NODENAME	NODENAME	NO
ISFATTR.NODE.PARTNAM	PARTNAME	NO
ISFATTR.NODE.PATH	PATH	NO
ISFATTR.NODE.PATHMGR	PMG	NO
ISFATTR.NODE.PENCRYPT	PEN	NO
ISFATTR.NODE.PRIVATE	PRV	NO
ISFATTR.NODE.PRTDEF	PRTDEF	NO
ISFATTR.NODE.PRTTSO	PRTTSO	NO
ISFATTR.NODE.PRTXWTR	PRTXWTR	NO
ISFATTR.NODE.PTYPE	PTYPE	NO
ISFATTR.NODE.PUNDEF	PUNDEF	NO
ISFATTR.NODE.PWCNTL	PWCNTL	NO
ISFATTR.NODE.RECEIVE	RECV	NO

Table 209. SDSF Class Resource Names and Overtypeable Fields (continued)

SDSF Resource Name (UPDATE Authority Required)	Overtypeable Field	Panel
ISFATTR.NODE.REST	REST	NO
ISFATTR.NODE.SENDP	SENDP	NO
ISFATTR.NODE.SENTREST	SENTRS	NO
ISFATTR.NODE.SRNUM	SRNUM	NO
ISFATTR.NODE.SSIGNON	SSIGNON	NO
ISFATTR.NODE.STNUM	STNUM	NO
ISFATTR.NODE.SUBNET	SUBNET	NO
ISFATTR.NODE.TRACE	TR	NO
ISFATTR.NODE.TRANSMIT	TRANS	NO
ISFATTR.NODE.VERIFYP	VERIFYP	NO
ISFATTR.OFFLOAD.ARCHIVE	ARCHIVE	SO
ISFATTR.OFFLOAD.CRTIME	CRTIME	SO
ISFATTR.OFFLOAD.DATASET	DSNAME	SO
ISFATTR.OFFLOAD.LABEL	LABEL	SO
ISFATTR.OFFLOAD.NOTIFY	NOTIFY	SO
ISFATTR.OFFLOAD.PROTECT	PROT	SO
ISFATTR.OFFLOAD.RETENT	RTPD	SO
ISFATTR.OFFLOAD.VALIDATE	VALIDATE	SO
ISFATTR.OFFLOAD.VOLS	VOLS	SO
ISFATTR.OUTDESC.ADDRESS	ADDRESS	JDS OD
ISFATTR.OUTDESC.AFPPARMS	AFPPARMS	JDS OD
ISFATTR.OUTDESC.BLDG	BUILDING	JDS OD
ISFATTR.OUTDESC.COLORMAP	COLORMAP	JDS OD
ISFATTR.OUTDESC.COMSETUP	COMSETUP	JDS OD
ISFATTR.OUTDESC.DEPT	DEPARTMENT	JDS OD
ISFATTR.OUTDESC.FORMDEF	FORMDEF	JDS OD
ISFATTR.OUTDESC.FORMLEN	FORMLEN	JDS OD
ISFATTR.OUTDESC.INTRAY	ITY	JDS
ISFATTR.OUTDESC.INTRAY	INTRAY	OD
ISFATTR.OUTDESC.IPDEST	IP DESTINATION	OD
ISFATTR.OUTDESC.NAME	NAME	JDS OD
ISFATTR.OUTDESC.NOTIFY	NOTIFY	JDS OD
ISFATTR.OUTDESC.OCOPYCNT	OCOPYCNT	JDS OD
ISFATTR.OUTDESC.OFFSETXB	OFFSETXB	JDS OD
ISFATTR.OUTDESC.OFFSETXF	OFFSETXF	JDS OD
ISFATTR.OUTDESC.OFFSETYB	OFFSETYB	JDS OD
ISFATTR.OUTDESC.OFFSETYF	OFFSETYF	JDS OD
ISFATTR.OUTDESC.OUTBIN	OUTBN	JDS
ISFATTR.OUTDESC.OUTBIN	OUTBIN	OD

Table 209. SDSF Class Resource Names and Overtypeable Fields (continued)

SDSF Resource Name (UPDATE Authority Required)	Overtypeable Field	Panel
ISFATTR.OUTDESC.OVERLAYB	OVERLAYB	JDS OD
ISFATTR.OUTDESC.OVERLAYF	OVERLAYF	JDS OD
ISFATTR.OUTDESC.PAGEDEF	PAGEDEF	JDS OD
ISFATTR.OUTDESC.PORTNO	PORTNO	OD
ISFATTR.OUTDESC.PORTNO	PORT	JDS
ISFATTR.OUTDESC.PRINTO	PRTOPTNS	OD
ISFATTR.OUTDESC.PRINTQ	PRTQUEUE	OD
ISFATTR.OUTDESC.RETAINF	RETAINF	OD
ISFATTR.OUTDESC.RETAINS	RETAINS	OD
ISFATTR.OUTDESC.RETRYL	RETRYL	OD
ISFATTR.OUTDESC.RETRYT	RETRYT	OD
ISFATTR.OUTDESC.ROOM	ROOM	JDS OD
ISFATTR.OUTDESC.TITLE	TITLE	JDS OD
ISFATTR.OUTDESC.USERDATA	USERDATA1	JDS
ISFATTR.OUTDESC.USERDATA	USERDATA	OD
ISFATTR.OUTDESC.USERLIB	USERLIB	JDS OD
ISFATTR.OUTPUT.BURST	BURST	JDS J0
ISFATTR.OUTPUT.BURST	BURST	H O
ISFATTR.OUTPUT.CHARS	CHARS	JDS J0
ISFATTR.OUTPUT.CLASS	C	H O JDS J0
ISFATTR.OUTPUT.COPYCNT	CC	JDS J0
ISFATTR.OUTPUT.COPYMOD	CPYMOD	JDS
ISFATTR.OUTPUT.DEST	DEST (secondary JES2)	H
ISFATTR.OUTPUT.DEST	DEST	H O JDS J0
ISFATTR.OUTPUT.FCB	FCB	JDS J0
ISFATTR.OUTPUT.FCB	FCB	H O
ISFATTR.OUTPUT.FLASH	FLASH	JDS J0
ISFATTR.OUTPUT.FLASH	FLASH	H O
ISFATTR.OUTPUT.FORMS	FORMS	H O JDS J0
ISFATTR.OUTPUT.ODISP	ODISP	H JDS O
ISFATTR.OUTPUT.PRMODE	PRMODE	H O JDS J0
ISFATTR.OUTPUT.PRTY	PRTY	H O
ISFATTR.OUTPUT.UCS	UCS	H O JDS J0
ISFATTR.OUTPUT.WRITER	WTR	H O JDS J0
ISFATTR.PROPTS.ASIS	ASIS	PR
ISFATTR.PROPTS.BPAGE	B	PR PUN
ISFATTR.PROPTS.CB	CB	PR
ISFATTR.PROPTS.CCTL	CCTL	PR PUN
ISFATTR.PROPTS.CHAR	CHAR1-4	PR

Table 209. SDSF Class Resource Names and Overtypeable Fields (continued)

SDSF Resource Name (UPDATE Authority Required)	Overtypeable Field	Panel
ISFATTR.PROPTS.CKPTLINE	CKPTLINE	PR PUN
ISFATTR.PROPTS.CKPTMODE	CKPTMODE	PR
ISFATTR.PROPTS.CKPTPAGE	CKPTPAGE	PR PUN
ISFATTR.PROPTS.CKPTSEC	CKPTSEC	PR
ISFATTR.PROPTS.CMPCT	CMPCT	PR PUN
ISFATTR.PROPTS.COMPACT	COMPACT	PR PUN
ISFATTR.PROPTS.COMPRESS	COMP	PR PUN
ISFATTR.PROPTS.COPIES	COPIES	PR PUN
ISFATTR.PROPTS.COPYMARK	COPYMARK	PR
ISFATTR.PROPTS.COPYMOD	CPYMOD	J0 PR
ISFATTR.PROPTS.CTRACE	CTR	LI NC NS
ISFATTR.PROPTS.DEVFCB	DFCB	PR
ISFATTR.PROPTS.DGRPY	DGRPY	PR PUN
ISFATTR.PROPTS.DYN	DYN	PR PUN
ISFATTR.PROPTS.FLUSH	FLS	PUN
ISFATTR.PROPTS.FSATRACE	FSATRACE	PR
ISFATTR.PROPTS.FSSNAME	FSSNAME	PR
ISFATTR.PROPTS.HONORTRC	HONORTRC	PR
ISFATTR.PROPTS.JTRACE	JTR	LI NC NS
ISFATTR.PROPTS.LRECL	LRECL	PUN
ISFATTR.PROPTS.MARK	M	PR
ISFATTR.PROPTS.NEWPAGE	NEWPAGE	PR
ISFATTR.PROPTS.NPRO	NPRO	PR
ISFATTR.PROPTS.OPACTLOG	OPLOG	PR PUN
ISFATTR.PROPTS.PAUSE	PAU	PR PUN
ISFATTR.PROPTS.PDEFAULT	PDEFAULT	PR
ISFATTR.PROPTS.PRESELCT	PSEL	PR
ISFATTR.PROPTS.RESTART	RESTART	LI
ISFATTR.PROPTS.RTIME	REST-INT	LI NS
ISFATTR.PROPTS.SELECT	SELECT	PR PUN
ISFATTR.PROPTS.SEP	SEP	PR PUN
ISFATTR.PROPTS.SEPCHARS	SEPCHAR	PR
ISFATTR.PROPTS.SEPDS	SEPDS	PR PUN RDR
ISFATTR.PROPTS.SETUP	SETUP	PR PUN
ISFATTR.PROPTS.SPACE	K	PR
ISFATTR.PROPTS.SUSPEND	SUS	PUN
ISFATTR.PROPTS.TRACE	TR	LI NC NS PR PUN
ISFATTR.PROPTS.TRANS	TRANS	PR
ISFATTR.PROPTS.TRKCELL	TRKCELL	PR

Table 209. SDSF Class Resource Names and Overtypeable Fields (continued)

SDSF Resource Name (UPDATE Authority Required)	Overtypeable Field	Panel
ISFATTR.PROPTS.UCSVERFY	UCSV	PR
ISFATTR.PROPTS.UNIT	UNIT	LI PR PUN SO
ISFATTR.PROPTS.VTRACE	VTR	LI NC NS
ISFATTR.PROPTS.WS	WORK- SELECTION	LI PR PUN SO
ISFATTR.PROPTS.MODE	MODE	PR
ISFATTR.RDR.AUTHORITY	AUTHORITY	RDR
ISFATTR.RDR.CLASS	C	RDR
ISFATTR.RDR.HOLD	HOLD	RDR
ISFATTR.RDR.MCLASS	MC	RDR
ISFATTR.RDR.PRIOINC	PI	RDR
ISFATTR.RDR.PRIOLIM	PL	RDR
ISFATTR.RDR.PRTDEST	PRTDEST	RDR
ISFATTR.RDR.PUNDEST	PUNDEST	RDR
ISFATTR.RDR.SYSAFF	SAFF1	RDR
ISFATTR.RDR.TRACE	TR	RDR
ISFATTR.RDR.UNIT	UNIT	RDR
ISFATTR.RDR.XEQDEST	XEQDEST	RDR
ISFATTR.RESMON.LIMIT	LIMIT	RM
ISFATTR.RESMON.WARNPCT	WARN%	RM
ISFATTR.RESOURCE.*system*	System	RES
ISFATTR.SELECT.BURST	SBURST	PR SO
ISFATTR.SELECT.CLASS	SCLASS	PR PUN
ISFATTR.SELECT.CLASS	SCLASS, SCLASS1-8	SO
ISFATTR.SELECT.DEST	SDEST1	PR PUN SO
ISFATTR.SELECT.DISP	SDISP	SO
ISFATTR.SELECT.FCB	SFCB	PR SO
ISFATTR.SELECT.FLASH	SFLH	PR SO
ISFATTR.SELECT.FORMS	SFORMS	PR PUN SO
ISFATTR.SELECT.HOLD	SHOLD	SO
ISFATTR.SELECT.JOBCLASS	CLASSES, CLASS1-8	INIT
ISFATTR.SELECT.JOBNAME	SJOBNAME	PR PUN SO
ISFATTR.SELECT.LIM	LINE-LIM-LO	PR PUN
ISFATTR.SELECT.LIM	LINE-LIM-HI	PR PUN
ISFATTR.SELECT.LIM	LINE-LIMIT	LI NC PR PUN SO
ISFATTR.SELECT.ODISP	SODSP	NC SO
ISFATTR.SELECT.OUTDISP	SODSP	LI
ISFATTR.SELECT.OWNER	SOWNER	PR PUN SO
ISFATTR.SELECT.PLIM	PAGE-LIM-LOW	PR
ISFATTR.SELECT.PLIM	PAGE-LIM-HI	PR

Table 209. SDSF Class Resource Names and Overtypeable Fields (continued)

SDSF Resource Name (UPDATE Authority Required)	Overtypeable Field	Panel
ISFATTR.SELECT.PLIM	PAGE-LIMIT	LI NC PR SO
ISFATTR.SELECT.PRMODE	SPRMODE1	PR PUN RDR
ISFATTR.SELECT.PRMODE	SPRMODE1	SO
ISFATTR.SELECT.RANGE	SRANGE	PUN SO
ISFATTR.SELECT.RANGE	SRANGE	PR
ISFATTR.SELECT.SCHENV	SSCHEDULING-ENV	SO
ISFATTR.SELECT.SRVCLS	SSRVCLASS	SO
ISFATTR.SELECT.SYSAFF	SSAFF	SO
ISFATTR.SELECT.UCS	SUCS	PR SO
ISFATTR.SELECT.VOL	SVOL1	PR
ISFATTR.SELECT.VOL	SVOL	PUN SO
ISFATTR.SELECT.WRITER	SWRITER	PR PUN SO
ISFATTR.SPOOL.MINPCT	MINPCT	SP
ISFATTR.SPOOL.OVFNAME	OVERFNAM	SP
ISFATTR.SPOOL.PARTNAME	PARTNAME	SP
ISFATTR.SPOOL.RESERVED	RES	SP
ISFATTR.SPOOL.SYSAFF	SAFF	SP

Appendix D. Accessibility

Accessible publications for this product are offered through IBM Knowledge Center (http://www.ibm.com/support/knowledgecenter/SSLTBW/welcome).

If you experience difficulty with the accessibility of any z/OS information, send a detailed message to the "Contact us" web page for z/OS (http://www.ibm.com/systems/z/os/zos/webqs.html) or use the following mailing address.

IBM Corporation
Attention: MHVRCFS Reader Comments
Department H6MA, Building 707
2455 South Road
Poughkeepsie, NY 12601-5400
United States

Accessibility features

Accessibility features help users who have physical disabilities such as restricted mobility or limited vision use software products successfully. The accessibility features in z/OS can help users do the following tasks:

- Run assistive technology such as screen readers and screen magnifier software.
- Operate specific or equivalent features by using the keyboard.
- Customize display attributes such as color, contrast, and font size.

Consult assistive technologies

Assistive technology products such as screen readers function with the user interfaces found in z/OS. Consult the product information for the specific assistive technology product that is used to access z/OS interfaces.

Keyboard navigation of the user interface

You can access z/OS user interfaces with TSO/E or ISPF. The following information describes how to use TSO/E and ISPF, including the use of keyboard shortcuts and function keys (PF keys). Each guide includes the default settings for the PF keys.

- *z/OS TSO/E Primer*
- *z/OS TSO/E User's Guide*
- *z/OS ISPF User's Guide Vol I*

Dotted decimal syntax diagrams

Syntax diagrams are provided in dotted decimal format for users who access IBM Knowledge Center with a screen reader. In dotted decimal format, each syntax element is written on a separate line. If two or more syntax elements are always present together (or always absent together), they can appear on the same line because they are considered a single compound syntax element.

Each line starts with a dotted decimal number; for example, 3 or 3.1 or 3.1.1. To hear these numbers correctly, make sure that the screen reader is set to read out

© Copyright IBM Corp. 1997, 2016

punctuation. All the syntax elements that have the same dotted decimal number (for example, all the syntax elements that have the number 3.1) are mutually exclusive alternatives. If you hear the lines 3.1 USERID and 3.1 SYSTEMID, your syntax can include either USERID or SYSTEMID, but not both.

The dotted decimal numbering level denotes the level of nesting. For example, if a syntax element with dotted decimal number 3 is followed by a series of syntax elements with dotted decimal number 3.1, all the syntax elements numbered 3.1 are subordinate to the syntax element numbered 3.

Certain words and symbols are used next to the dotted decimal numbers to add information about the syntax elements. Occasionally, these words and symbols might occur at the beginning of the element itself. For ease of identification, if the word or symbol is a part of the syntax element, it is preceded by the backslash (\) character. The * symbol is placed next to a dotted decimal number to indicate that the syntax element repeats. For example, syntax element *FILE with dotted decimal number 3 is given the format 3 * FILE. Format 3* FILE indicates that syntax element FILE repeats. Format 3* * FILE indicates that syntax element * FILE repeats.

Characters such as commas, which are used to separate a string of syntax elements, are shown in the syntax just before the items they separate. These characters can appear on the same line as each item, or on a separate line with the same dotted decimal number as the relevant items. The line can also show another symbol to provide information about the syntax elements. For example, the lines 5.1*, 5.1 LASTRUN, and 5.1 DELETE mean that if you use more than one of the LASTRUN and DELETE syntax elements, the elements must be separated by a comma. If no separator is given, assume that you use a blank to separate each syntax element.

If a syntax element is preceded by the % symbol, it indicates a reference that is defined elsewhere. The string that follows the % symbol is the name of a syntax fragment rather than a literal. For example, the line 2.1 %OP1 means that you must refer to separate syntax fragment OP1.

The following symbols are used next to the dotted decimal numbers.

? indicates an optional syntax element
The question mark (?) symbol indicates an optional syntax element. A dotted decimal number followed by the question mark symbol (?) indicates that all the syntax elements with a corresponding dotted decimal number, and any subordinate syntax elements, are optional. If there is only one syntax element with a dotted decimal number, the ? symbol is displayed on the same line as the syntax element, (for example 5? NOTIFY). If there is more than one syntax element with a dotted decimal number, the ? symbol is displayed on a line by itself, followed by the syntax elements that are optional. For example, if you hear the lines 5 ?, 5 NOTIFY, and 5 UPDATE, you know that the syntax elements NOTIFY and UPDATE are optional. That is, you can choose one or none of them. The ? symbol is equivalent to a bypass line in a railroad diagram.

! indicates a default syntax element
The exclamation mark (!) symbol indicates a default syntax element. A dotted decimal number followed by the ! symbol and a syntax element indicate that the syntax element is the default option for all syntax elements that share the same dotted decimal number. Only one of the syntax elements that share the dotted decimal number can specify the ! symbol. For example, if you hear the lines 2? FILE, 2.1! (KEEP), and 2.1 (DELETE), you know that (KEEP) is the

default option for the FILE keyword. In the example, if you include the FILE keyword, but do not specify an option, the default option KEEP is applied. A default option also applies to the next higher dotted decimal number. In this example, if the FILE keyword is omitted, the default FILE(KEEP) is used. However, if you hear the lines 2? FILE, 2.1, 2.1.1! (KEEP), and 2.1.1 (DELETE), the default option KEEP applies only to the next higher dotted decimal number, 2.1 (which does not have an associated keyword), and does not apply to 2? FILE. Nothing is used if the keyword FILE is omitted.

*** indicates an optional syntax element that is repeatable**

The asterisk or glyph (*) symbol indicates a syntax element that can be repeated zero or more times. A dotted decimal number followed by the * symbol indicates that this syntax element can be used zero or more times; that is, it is optional and can be repeated. For example, if you hear the line 5.1* data area, you know that you can include one data area, more than one data area, or no data area. If you hear the lines 3* , 3 HOST, 3 STATE, you know that you can include HOST, STATE, both together, or nothing.

Notes:

1. If a dotted decimal number has an asterisk (*) next to it and there is only one item with that dotted decimal number, you can repeat that same item more than once.
2. If a dotted decimal number has an asterisk next to it and several items have that dotted decimal number, you can use more than one item from the list, but you cannot use the items more than once each. In the previous example, you can write HOST STATE, but you cannot write HOST HOST.
3. The * symbol is equivalent to a loopback line in a railroad syntax diagram.

+ indicates a syntax element that must be included

The plus (+) symbol indicates a syntax element that must be included at least once. A dotted decimal number followed by the + symbol indicates that the syntax element must be included one or more times. That is, it must be included at least once and can be repeated. For example, if you hear the line 6.1+ data area, you must include at least one data area. If you hear the lines 2+, 2 HOST, and 2 STATE, you know that you must include HOST, STATE, or both. Similar to the * symbol, the + symbol can repeat a particular item if it is the only item with that dotted decimal number. The + symbol, like the * symbol, is equivalent to a loopback line in a railroad syntax diagram.

Notices

This information was developed for products and services offered in the U.S.A. or elsewhere.

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 character set (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.

© Copyright IBM Corp. 1997, 2016

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:

Site Counsel
IBM Corporation
2455 South Road
Poughkeepsie, NY 12601-5400
USA

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.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

COPYRIGHT LICENSE:

This information might contain 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.

Policy for unsupported hardware

Various z/OS elements, such as DFSMS, HCD, JES2, JES3, and MVS™, contain code that supports specific hardware servers or devices. In some cases, this device-related element support remains in the product even after the hardware devices pass their announced End of Service date. z/OS may continue to service element code; however, it will not provide service related to unsupported hardware devices. Software problems related to these devices will not be accepted

for service, and current service activity will cease if a problem is determined to be associated with out-of-support devices. In such cases, fixes will not be issued.

Minimum supported hardware

The minimum supported hardware for z/OS releases identified in z/OS announcements can subsequently change when service for particular servers or devices is withdrawn. Likewise, the levels of other software products supported on a particular release of z/OS are subject to the service support lifecycle of those products. Therefore, z/OS and its product publications (for example, panels, samples, messages, and product documentation) can include references to hardware and software that is no longer supported.

- For information about software support lifecycle, see: IBM Lifecycle Support for z/OS (http://www.ibm.com/software/support/systemsz/lifecycle/)
- For information about currently-supported IBM hardware, contact your IBM representative.

Programming interface information

This book is intended to help the customer customize SDSF and set up SDSF security.

This publication primarily documents information that is NOT intended to be used as Programming Interfaces of SDSF.

This publication also documents intended Programming Interfaces that allow the customer to write programs to obtain the services of SDSF. This information is identified where it occurs, either by an introductory statement to a chapter or section or by the following marking:

Programming Interface Information

End of Programming Interface Information

Trademarks

IBM, the IBM logo, and 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 at Copyright and Trademark information (http://www.ibm.com/legal/copytrade.shtml).

Index

Special characters

? query character
  defining with DCHAR parameter 24
/ command
  authority to enter MVS and JES commands 47, 252
* character, with SCHARS parameter 27
% character, with SCHARS parameter 27
&; (automatic reissue of commands)
  interval, specifying minimum 46
  restricting use of 37

A

abbreviations in column data 86
abend
  codes 551
  managing 557
ABEND command
  authorization to issue 235
ABENDCON default field title
  Dynamic Exits panel 138
ABENDNUM default field title
  Dynamic Exits panel 138
access authority, SAF 194
access to SDSF, controlling 46, 561
accessibility 589
  contact IBM 589
  features 589
ACCT default field title
  Held Output Queue panel 145
  Input Queue panel 148
  JC panel 150
  Output Queue panel 167
  Status panel 188
ACTDORM default field title
  Multi-Access Spool panel 159
ACTFILES default field title
  Process panel 174
ACTHOLD default field title
  Multi-Access Spool panel 159
action characters
  confirmation of 50
  reference to by command level 73
  specifying authorization to use 47, 50, 211
ACTION command
  authorization to issue 235
action messages
  protecting with SAF 304
ACTION parameter 37
ACTIONBAR parameter in ISFPARMS 45
ACTIVE default field title
  Dynamic Exits panel 138
ACTIVE, default field title
  Job Class panel 151
ADDRESS default field title
  Output Descriptors panel 165
Address Space Memory panel 133
  ASFLD2 parameter 46
  ASFLDS parameter 45
  variable field list of
    alternate, defining 46
    primary, defining 45
ADDRESS-LINE1 default field title
  Job Data Set panel 152
ADDRESS-LINE2 default field title
  Job Data Set panel 152
ADDRESS-LINE3 default field title
  Job Data Set panel 153
ADDRESS-LINE4 default field title
  Job Data Set panel 153
ADEST value 55
ADISC default field title
  Lines panel 155
ADMSYMBL 24
AFD command 371
AFPPARMS Default field title
  Job Data Set panel 153
  Output Descriptor panel 166
alias
  queue 116, 346
  queue manager 346
ALL value
  for ACTION parameter 44
  for CMDAUTH parameter 47
  for DSPAUTH parameter 55
Alloc default field title
  Initiator panel 146
allocating
  SDSF data sets 25
  SYSLOG index 26
AllocCount default field title
  Initiator panel 146
alternate field list for SDSF panels
  defining 85
AMDEST value 55, 60, 73
AMSG value 56, 60, 73
ANode default field title
  NC panel 160
ANODE default field title
  LI panel 156
APF default field title
  Link Pack Area panel 158
APF indicator default field title
  Link List panel 157
APPC parameter in ISFPARMS 45
Appl default field title
  NS panel 161
APPLID default field title
  Lines panel 156
ARCHIVE default field title
  Spool Offload panel 182
AReq default field title
  Reader panel 177
ARM registration of server 118
ARRINTV default field title
  Enclaves panel 140
ARRTIME default field title
  Enclaves panel 140
ASID default field title
  Address Space Memory panel 134
  Display Active Users Panel 136
  Enqueue panel 141
  Initiator panel 146
  Input Queue panel 148
  NS panel 162
  Process panel 174
ASIDX default field title
  Address Space Memory panel 133
  Display Active Users Panel 136
  Enqueue panel 141
  Initiator panel 146
  Input Queue panel 148
  Process panel 174
ASIS default field title
  Printer panel 173
assembling SDSF 351
assistive technologies 589
ASYS default field title
  Input Queue panel 147
  Status panel 188
AttStat default field title
  MAS and JESPLEX panels 159
auditing
  commands with MSG 50
  SAF requests 192
AUPDT parameter 37
AUTH default field title
  JC panel 150
AUTH parameter 46
  specifying with SAF 313, 561
AUTHORITY default field title
  NODE panel 162
  Reader panel 177
authorization
  using SAF to change 192
authorization group
  users placed in wrong 556
  with error messages 475
authorization parameters 15
  quick reference to 37
  using to define
    action characters that can be issued 50
    automatic update interval 37
    default address space types and positions displayed on Display Active Users panel 52
    destination names used with DEST command 53, 59
    display of SYSOUT and SYSIN data sets 55, 59
    fields overtypeable on SDSF panels 47, 50
    groups of users 36
    jobs displayed on SDSF panels 59, 65, 66

© Copyright IBM Corp. 1997, 2016

authorization parameters *(continued)*
using to define *(continued)*
jobs for which commands can be issued 47
MVS and JES commands that can be issued 47
MVS and JES2 commands that can be issued 50
SDSF commands allowed 46
SYSLOG displayed in multi-access spool environment 71
who can use SDSF 46
WTOR messages displayed on SYSLOG 44
authorization parameters in ISFPARMS
checking for user authorization 475
using to define
combining use with SAF security 192
Authorized Program Facility panel 135
APFFLD2 parameter 45
variable field list of
alternate, defining 45
Authorized Program Facilitypanel
APFFLDS parameter 45
variable field list of
primary, defining 45
authorizing users 15
problems with 556
to use SDSF 46
with an installation exit routine 329
AUTODELAY default field title
SR panel 190
Automatic Restart Manager (ARM)
registration of server 118
automatic update
interval, specifying minimum 37
restricting use of 37
AUTOREPLY default field title
SR panel 190
AUTOTEXT default field title
SR panel 190
AUTOTIME default field title
SR panel 190
Aux default field title
Address Space Memory panel 134
AUXPCT default field title
SYS panel 186
AVGMSU default field title
SYS panel 187

B

B abbreviation in column data 86
B default field title
Printer panel 173
BADASID default field title
SYS panel 186
Barrier default field title
Initiator panel 146
batch 369
batch job
displaying on Display Active Users panel 52
running SDSF as 375
BDTNAME default field title
NODE panel 163
BLKSIZE default field title
Authorized Program Facility panel 135
Link List panel 157
Link Pack Area panel 158
PARMLIB panel 169
SRCH panel 184
BLP default field title
JC panel 150
Reader panel 178
BOSS default field title
MAS and JESPLEX panels 159
browse check output with ISFBROWSE 441
browse job output with EXECIO 435, 437
browse job output with ISFBROWSE 435, 436, 438
browse the OPERLOG with ISFLOG 447
browse the SYSLOG with ISFLOG 445
browsing output
authority 55
custom property 92
ISFPARMS authority 50, 73
SAF authority 211, 247, 313
browsing output with ISFBROWSE 394
BUFSIZE default field title
NODE panel 164
BUILDING default field title
Job Data Set panel 152
Output Descriptors panel 164
BURST default field title
Held Output Queue panel 144
Job 0 panel 154
Job Data Set panel 152
Output Queue panel 167
Printer panel 170
BURSTPAGE default field title
Punch panel 176
BYTE-CNT default field title
Job 0 panel 155
Job Data Set panel 151

C

C default field title
Display Active Users panel 136
Held Output Queue panel 144
Initiator panel 146
Input Queue panel 147
Job 0 panel 154
Job Data Set panel 151
Output Queue panel 166
Printer panel 170
Reader panel 177
Status panel 188
CADSPCT default field title
SYS panel 186
CANCEL ($C) command
authorization with ISFPARMS 48, 73
authorization with SAF 194
canceling jobs
authorization with ISFPARMS 48
authorization with SAF 194
CARDS default field title
Held Output Queue panel 145
Input Queue panel 148
CARDS default field title *(continued)*
Output Queue panel 167
Status panel 189
carriage control
custom property 100
case of panels, specifying 101
CATDSN default field title
SYS panel 187
CATEGORY default field title
CK panel 142
CATTYPE default field title
SYS panel 187
CATVOL default field title
SYS panel 187
CC default field title
Job 0 panel 154
Job Data Set panel 152
CCTL default field title
Printer panel 173
Punch panel 175
CGS default field title
Printer panel 172
changing
class of output of jobs
authorization with ISFPARMS 48
authorization with SAF 194
CHAR1 default field title
Printer panel 172
CHAR2 default field title
Printer panel 172
CHAR3 default field title
Printer panel 172
CHAR4 default field title
Printer panel 172
CHARS default field title
Job 0 panel 154
Job Data Set panel 152
CK command
authorization to issue 235
CK panel 142
CK panel scope
controlling 96
CKH panel 141
CKHFLDS parameter 47
limiting 97, 331
variable field list of
alternate, defining 47
primary, defining 47
CKPTLINE default field title
Printer panel 171
CKPTMODE default field title
Printer panel 171
CKPTPAGE default field title
Printer panel 171
Punch panel 175
CKPTREC default field title
Printer panel 171
Punch panel 175
CKPTSEC default field title
Printer panel 171
CKTPLINE default field title
Punch panel 175
CL default field title
Punch panel 175
Class1-8 default field titles
Initiator panel 147

CLASSES default field title
Initiator panel 146
CLEAR default field title
Printer panel 173
CMDAUTH parameter 47
authorization to issue MVS/JES commands 48
specifying with SAF 313
CMDLEV parameter 50
specifying with SAF 312, 313
CMPCT default field title
Printer panel 173
CODE default field title
Lines panel 155
code pages 101
CODPAG parameter 103
COLORMAP default field title
Job Data Set panel 153
Output Descriptors panel 165
column headings 85
COMCHAR default field title
MAS and JESPLEX panels 159
COMM statement 30, 32
command authority exit point 333
COMMAND default field title
JC panel 150
MAS and JESPLEX panels 159
Process panel 174
Spool Volumes panel 183
commands
AFD 371
and EMCS 56
authorization to issue
problems with 556
with installation exit routines 329
with ISFPARMS 47, 50
with SAF security scheme 212, 235, 252, 256
authorized SDSF commands 46, 235
generated MVS/JES commands 194, 202
ISFACP 17
logging of 50
MVS and JES, issuing 47, 252
unauthorized use of, tracking 50
communications
problems with 558
starting between servers 128
stopping 129
COMP default field title
Lines panel 156
Printer panel 173
Punch panel 175
Comp% default field title
Spool Volumes panel 184
COMPAC default field title
Printer panel 173
Compact default field title 160
COMPACT default field title
NODE panel 162
Punch panel 175
COMSETUP default field title
Job Data Set panel 153
Output Descriptors panel 165
conditional access with RACF 209
conditional processing of ISFPARMS
WHEN statement 20, 22
CONFIRM parameter 50
confirmation of action characters, controlling 50
Conn-Int default field title
LI panel 157
NC panel 160
Connect default field title 160
CONNECT default field title
LI panel 157
NODE panel 164
CONNECT statement 34
ConnStat default field title
MAS and JESPLEX panels 159
console
name 343
CONSOLE class 194, 209
CONSOLE default field title
SR panel 190
console used with / command 48
contact
z/OS 589
CONTROL (K) command
authorization to issue 73
conversion assist for RACF 323
converting assembler macros to statements 17
converting ISFPARMS to RACF 323
COPIES default field title
Printer panel 173
Punch panel 176
COPYMARK default field title
Printer panel 171
COUNT default field title
CK panel 142
CPC default field title
SYS panel 187
CPR default field title
JC panel 150
CPU default field title
Process panel 174
CPU use
custom property 97
CPU-TIME default field title
Display Active Users Panel 136
Enclave panel 139
CPU% default field title
Display Active Users Panel 136
CPUCRIT default field title
Display Active Users panel 137
Enclaves panel 140
CPUFMT parameter 51
CPUNUM default field title
SYS panel 187
CPUPR default field title
SYS panel 186
CPY default field title
JC panel 150
CPYMOD default field title
Job 0 panel 154
Job Data Set panel 152
Printer panel 171
CPYMODFT default field title
Job Data Set panel 152
CRDATE default field title
Authorized Program Facility panel 135
Held Output Queue panel 144
CRDATE default field title *(continued)*
Link List panel 157
Link Pack Area panel 158
Output Queue panel 167
PARMLIB panel 169
SRCH panel 185
CRDATE-CRTIME default field title
Job Data Set panel 152, 155
CRTIME default field title
Spool Offload panel 182
CSA default field title
Address Space Memory panel 134
CSAPCT default field title
Address Space Memory panel 134
SYS panel 186
CSQZPARM, values for SDSF 344
CTCNODE default field title
NODE panel 164
CTIME default field title
NODE panel 164
CTITLE parameter 51
CTr default field title
NC panel 160
NS panel 161
CTR default field title
LI panel 156
CURSOR parameter 52
cursor placement, controlling 52
CUSTOM parameter 52
customizing parameters 52
customizing properties 91
customizing SDSF
writing an installation exit routine 329
CVTVERID default field title
SYS panel 187

D

DA command
authorization to issue 235
DA panel scope
controlling 97
DADFLT parameter 52
data sets
allocating SDSF 25
freeing dynamically allocated 29
DataSetName default field title
Spool Volumes panel 183
DATE default field title
SR panel 190
date format, controlling 53
DATE parameter 53
date separator, controlling 53
DATEE default field title
CK panel 143
CKH panel 141
Process panel 174
RM panel 179
DATEN default field title
CKH panel 142
DATESEP parameter 53
DATETIMEE default field title
Input Queue panel 149
Output Queue panel 168
DATETIMEN default field title
Output Queue panel 168

DATETIMER default field title
Input Queue panel 149
Output Queue panel 168
DCHAR parameter 24
DDNAME default field title
Job 0 panel 155
DEBUG default field title
CK panel 143
Default default field title
Initiator panel 146
Spool Volumes panel 183
default field title
JC panel 149
default SDSF server, defining 34
DefCount default field title
Initiator panel 146
DEFDATE default field title
CK panel 143
DEFINE command
example 116
DEFVOL default field title
Authorized Program Facility panel 135
DELAYRSN default field title
Input Queue panel 148
Status panel 189
DELDATE default field title
CK panel 143
DEPARTMENT default field title
Job Data Set panel 152
Output Descriptors panel 165
DESC default field title
SR panel 190
DESCRIPT default field title
SE panel 179
DEST command
destination names used on 53, 59
how to protect with SAF 239
DEST default field title
Held Output Queue panel 144
Job 0 panel 154
Job Data Set panel 151
Output Queue panel 166
Printer panel 170
DEST parameter 53
Specifying with SAF 53
DEST value on CMDAUTH parameter 49
DESTDEF
coding 339
destination names of jobs
SAF authorization 239
specifying usage with DEST command 53
used to control
authorization to issue commands 47
authorization to view SYSOUT and SYSIN for jobs 55, 59
jobs displayed on SDSF panels 59, 61, 73
destination operator authority 194, 202, 240, 314
DESTN default field title
Punch panel 176
DEVICE default field title
Held Output Queue panel 144
DEVICE default field title *(continued)*
Input Queue panel 147
Output Queue panel 166
Status panel 188
DevType default field title
Reader panel 177
DEVTYPE default field title
Printer panel 172
Punch panel 176
DFCB default field title
Printer panel 171
DFIELD2 parameter 54
DFIELDS parameter 54
DGRPY default field title
Printer panel 172
Punch panel 176
DIAG1 default field title
CK panel 142
CKH panel 141
DIAG2 default field title
CK panel 142
CKH panel 141
DIAGFROM default field title
CK panel 142
diagnostic information, modules 555
DIRECT default field title
NODE panel 163
DISCON default field title
LI panel 156
DISPLAY ($D) command
authorization to issue, specifying 48, 73
DISPLAY (D) command
authorization to issue 73
Display Active Users panel 135
address space types and positions on, specifying
in ISFPARMS 52
and RMF 344
authorization to display 46, 235
jobs displayed on 59, 66
by owner 65
variable field list of
alternate, defining 85
primary, defining 85
DISPLAY parameter in ISFPARMS 55
DISPLAY value on CMDAUTH parameter 49
displaying
message help 475
DLY default field title
Input Queue panel 148
Status panel 188
DORMANCY default field title
Multi-Access Spool panel 159
DP default field title
Display Active Users panel 136
Printer panel 170
DPRIO default field title
Punch panel 176
DSecLabel default field title 161
DSECLABEL default field title
LI panel 156
NODE panel 163
NS panel 162
Printer panel 172
Punch panel 176
DSECLABEL default field title *(continued)*
Reader panel 177
Spool Offload panel 182
DSENQSHR, default field title
Job Class panel 151
DSI parameter 25
DSID default field title
Job 0 panel 154
Job Data Set panel 151
DSNAME default field title
Job 0 panel 155
Job Data Set panel 152
Spool Offload panel 181
DSORG default field title
Authorized Program Facility panel 135
Link List panel 157
Link Pack Area panel 158
PARMLIB panel 169
SRCH panel 184
DSPAUTH parameter 55
Specifying with SAF 313
DSPName default field title
NS panel 161
DSPNAME default field title
Printer panel 172
Punch panel 176
Reader panel 177
DSYSID default field title
RM panel 179
DTracks default field title
Spool Volumes panel 183
DTrackU default field title
Spool Volumes panel 183
dump
of SDSF, taking 555
with abends 551
DUPJOB, default field title
Job Class panel 151
DUPLEX default field title
Lines panel 156
DYNAMIC default field title
Printer panel 172
Punch panel 176
Dynamic Exits panel 138
ASFLD2 parameter 56
DYNXFLDS parameter 56
variable field list of
alternate, defining 56
primary, defining 56
dynamically allocated data sets
enqueueing on 25
freeing 29

E

ECPU% default field title
Display Active Users panel 137
ECPU—Time default field title
Display Active Users panel 137
ECSA default field title
Address Space Memory panel 134
ECSAPCT default field title
Address Space Memory panel 134
SYS panel 186
EDT default field title
SYS panel 187

EINTERVAL default field title
CK panel 143
EMCS authority 56
EMCS console
name modification 94
sharing 95, 331, 333
EMCS required for system commands 56
EMCSAUTH parameter 56
EMCSREQ parameter 56
ENC command
authorization to issue 235
ENC panel scope
controlling 98
ENCFLD2 parameter 57
ENCFLDS parameter 57
Enclaves panel
authorization to display 46, 235
variable field list 139
variable field list of
alternate, defining 57
primary, defining 57
END default field title
NODE panel 162
end user group 312, 318
END-DATE default field title
Held Output Queue panel 145
Output Queue panel 167
Status panel 189
END-TIME default field title
Held Output Queue panel 145
Output Queue panel 167
Status panel 189
ENQTOKEN default field title
Enqueue panel 141
Enqueue panel 140
ENQFLDS parameter 57
variable field list of
primary, defining 57
enqueueing on dynamically allocated data sets 25
ESQA default field title
Address Space Memory panel 134
ESQAPCT default field title
Address Space Memory panel 134
SYS panel 186
ESYS default field title
Held Output Queue panel 145
Input Queue panel 148
Output Queue panel 167
Status panel 189
EXCOUNT default field title
CK panel 142
EXCP-Cnt default field title
Display Active Users Panel 136
EXECNAME default field title
CK panel 143
EXECNODE default field title
Input Queue panel 147
Status panel 188
exit point
command authority 333
display authority exit point 334
initialization 331
installation of 330
post-SAF 335
pre-SAF 335
exit point *(continued)*
SDSF termination 335
table build 337
EXITNAME default field title
CK panel 143
EXT default field title
Spool Volumes panel 183
extended console 253, 305
name 343
OPERPARM 306
EXTENT default field title
Authorized Program Facility panel 135
Link List panel 157
Link Pack Area panel 158
PARMLIB panel 169
SRCH panel 184

F

FAIL default field title
CK panel 142
FASTPATH default field title
Dynamic Exits panel 138
FCB default field title
Held Output Queue panel 144
Job 0 panel 154
Job Data Set panel 152
Output Queue panel 167
Printer panel 170
FCBLOAD default field title
Printer panel 173
field title
Reader panel 177
fields on SDSF panels
defining in ISFPARMS 85
defining with statements 85
overtyping
authorization with ISFPARMS 50
authorization with SAF 255
FILTER command
limit of lines searched on OPERLOG 26
filtering
bypassing 332
FIND command
limit of lines searched for, specifying in ISFPARMS 25
FINDLIM command
authorization to issue 235
FINDLIM parameter 25
FIXBPCT default field title
SYS panel 186
FIXED default field title
Address Space Memory panel 134
FIXEDB default field title
Address Space Memory panel 134
FIXPCT default field title
SYS panel 186
FLASH default field title
Held Output Queue panel 144
Job 0 panel 154
Job Data Set panel 152
Output Queue panel 167
Printer panel 170
FLASHC default field title
Job 0 panel 154
FLASHC default field title *(continued)*
Job Data Set panel 152
FLD statement 85
parameters 86
FLDENT statement
parameters 86
FLS default field title
Punch panel 175
FLUSHACT default field title
Input Queue panel 149
format of CPU busy, controlling 51
FORMDEF default field title
Job Data Set panel 152
Output Descriptors panel 164
FORMLEN default field title
Job Data Set panel 153
Output Descriptors panel 165
FORMS default field title
Held Output Queue panel 144
Job 0 panel 154
Job Data Set panel 152
Output Queue panel 166
Printer panel 170
Punch panel 176
FREEASID default field title
SYS panel 186
freeing dynamically allocated data sets 29
FSASYSNM default field title
Printer panel 172
FSATTRACE default field title
Printer panel 171
FSSNAME default field title
Printer panel 171
FSSPROC default field title
Printer panel 171

G

G abbreviation in column data 86
GB abbreviation in column data 86
GCP-Use% default field title
Display Active Users panel 137
GCP—Time default field title
Display Active Users panel 137
GDDM (Graphical Data Display Manager)
invoking with V action character 24
viewing page-mode output 247
generated MVS/JES commands 194, 202
GLOBAL default field title
CK panel 142
MAS and JESPLEX panels 159
global search characters, specifying 27
GPLEN parameter 57
GPREF parameter 58
Group default field title
Initiator panel 146
GROUP default field title
Printer panel 169
PUN panel 174
RDR panel 177
Group field
JC panel 149
group membership
setting up with GROUP statement 311

group prefix
defining 57, 58
using to control
jobs users can display 66
jobs users can issue commands for 47
jobs users can view messages only for 57
jobs users can view output for 57
GROUP statement
examples of 36
parameters 36, 37
purpose of 36
sample with SAF profiles 313, 315, 318
setting up group membership with 311
using SAF security 191
GROUP value
for CMDAUTH parameter 49
for DSPAUTH parameter 56
for PREFIX parameter 66
groups of users, defining 36
GRPMSG value 56, 60, 73
GRS default field title
SYS panel 187
GSDSF class 206

H

H command
adding generic character 93, 331
authorization to issue 235
HALT ($Z) command
authorization to issue, specifying 48, 73
HASPINDX data set
allocating 351
block size of, specifying 25
buffers, specifying 27
considerations 349
DD statement in TSO logon procedure 26
forcing use of 96
name of 25
name of, specifying 25
sharing 349
volume serial of, specifying 26
HBURST default field title
Printer panel 173
HCHARS default field title
Printer panel 173
HCPYMOD default field title
Printer panel 173
headings on SDSF panels, defining 85
Health Check history panel
variable field list of
alternate, defining 85
primary, defining 85
Health Check History panel 141
Health Checker panel 142
authorization to display 46, 235
CKFLD2 parameter 47
CKFLDS parameter 47
variable field list of
alternate, defining 47
primary, defining 47
Held Output Queue panel 144
authority to display 235
authority to display, specifying 46
jobs displayed on, controlling
by destination name 59
by job name 61, 73
by job name prefix 66
by owner 65
variable field list of 145
alternate, defining 85
primary, defining 85
help
for messages 475
search 475
help panel data set
DD statement in TSO logon procedure 27
specifying name and volume serial of 27
HFCB default field title
Printer panel 173
HFIELD2 parameter 58
HFLASH default field title
Printer panel 173
HFORMS default field title
Printer panel 173
Punch panel 176
HiCyl default field title
Spool Volumes panel 183
HiHead default field title
Spool Volumes panel 183
HITRK default field title
Spool Volumes panel 183
HiUsed default field title
Spool Volumes panel 184
HOLD ($H) command
authorization to issue 47, 48, 73
HOLD default field title
Multi-Access Spool panel 158
NODE panel 162
Reader panel 177
HOLD-CNT default field title
JC panel 149
holding
jobs
authorization for, specifying 48
output of jobs
authorization for, specifying 50
HOLDUNTIL default field title
Input Queue panel 149
HONORTRC default field title
Printer panel 172
HUCS default field title
Printer panel 173
HVCOM default field title
Address Space Memory panel 134
HVCOMHWM default field title
Address Space Memory panel 134
HVCOMNUM default field title
Address Space Memory panel 134
HWNAME default field title
SYS panel 187

I

I command
authorization to issue 235
IBM Health Checker for z/OS
CK panel 142
CKH panel 141
ICMD parameter 58
IDBACK parameter, value for 344
IDBLKS parameter 25
IDEST parameter 59
Specifying with SAF 59
IDFORE, value for 344
IDSP parameter 59
IDSPD parameter 55, 59, 73
IEANUC default field title
SYS panel 187
IEASYM default field title
SYS panel 187
IEASYS default field title
SYS panel 187
IFIELD2 parameter 60
IFIELDS parameter 60
ILOGCOL parameter 60
SYSLOG position, specifying 60
ILPROC parameter 39
IN value 52
IND default field title
Multi-Access Spool panel 159
INDEX parameter 25
index, to help and tutorial 475
INDXVOL parameter 26
INIT command
authorization to issue 235
INIT panel
default initiators 93
INIT value 52
initialization exit point 331
initialization parameters 15
sample of 18
initiator panel 146
Initiator panel
authority to display, specifying 47
authorization to display 46
variable field list of 146
alternate, defining 85
primary, defining 85
initiators
authority to control from Initiator panel, specifying 47
displaying
on Display Active Users panel 52
SAF authorization 194
INPUT command
authorization to issue 235
INPUT parameter 60
Input Queue panel 147
authority to display 46, 235
jobs displayed on, controlling
by destination name 59
by job name 61
by job name prefix 66
by owner 65
variable field list 147
variable field list of 147
alternate, defining 85
primary, defining 85
installation exit points 330
installation exit routines 329
installing SDSF
with JES2 secondary subsystem 339

INTAVG default field title
RM panel 179
INTERRUPT ($I) command
authorization to issue 48, 73, 194
INTERVAL default field title
CK panel 143
INTF default field title
Lines panel 156
INTFLD2 parameter 61
INTFLDS parameter 61
INTHIGH default field title
RM panel 179
INTLOW default field title
RM panel 179
INTRAY default field title
Output Descriptors panel 165
IODFDATE default field title
SYS panel 187
IODFDSN default field title
SYS panel 187
IOPRIOGRP default field title
Display Active Users panel 137
Enclaves panel 140
IP DESTINATION default field title
Output Descriptors panel 165
IPAddr default field title
NC panel 160
IPADDR default field title
LI panel 156
IPLDATE default field title
SYS panel 187
IPLDAYS default field title
SYS panel 187
IPLTYPE default field title
SYS panel 187
IPLUNIT default field title
SYS panel 186
IPLVOL default field title
SYS panel 186
IPName default field title
NC panel 160
NS panel 162
IPNAME default field title
LI panel 156
ISFACP command 17
ISFACR command 324
ISFACT command 388, 401
ISFACTIONS special variable 420
ISFAPPC special variable 420
ISFASMP
reassembling SDSF 351
ISFBROWSE command 394
ISFBROWSE, examples 395
ISFBROWSE, special variables 395
ISFCALLS 379
ISFCKLIM special variable 420
ISFCMDLIM special variable 420
ISFCMODE special variable 420
ISFCOLOR special variable 421
ISFCOLS special variable 420
ISFCOLUMNGROUPS special variable 421
ISFCONMOD special variable 420, 421
ISFCONS special variable 420
ISFDATE special variable 420
ISFDCOLS special variable 420
ISFDDNAME special variable 420
ISFDELAY special variable 420
ISFDESCODE special variable 422
ISFDEST special variable 420
ISFDIAG special variable 420
ISFDISPLAY special variable 420
ISFDISPLAYMODE special variable 420
ISFDSNAME special variable 420
ISFEXEC command 380
ISFFILTER special variable 420
ISFFILTERMODE special variable 420
ISFFIND special variable 423
ISFFINDENDCOL special variable 423
ISFFINDLIM special variable 423
ISFFINDSTARTCOL special variable 423
ISFFIRSTLINEDATE special variable 423
ISFFIRSTLINEDSID special variable 423
ISFFIRSTLINEJOBID special variable 423
ISFFIRSTLINERECNO special variable 423
ISFFIRSTLINETIME special variable 423
ISFFIRSTLINETOKEN special variable 423
ISFFLD macro 85
parameters 86
ISFGRP macro
examples of 36
parameters 36, 37
purpose of 36
setting up group membership with 311
using SAF security 191
ISFGRP macros
checking authorization 475
ISFHIGHLIGHT special variable 423
ISFINPUT special variable 420
ISFINTENSITY special variable 423
ISFISP 349
ISFISP2 349
ISFISUCL 351
ISFJES3NAME special variable 420
ISFJESNAME special variable 420
ISFLASTLINEDATE special variable 424
ISFLASTLINEDSID special variable 424
ISFLASTLINEJOBID special variable 424
ISFLASTLINERECNO special variable 424
ISFLASTLINETIME special variable 424
ISFLINE special variable 420
ISFLINELIM special variable 420
ISFLOG command 403
ISFLOGSTARTDATE special variable 420, 424
ISFLOGSTARTTIME special variable 420, 424
ISFLOGSTOPDATE special variable 420, 424
ISFLOGSTOPTIME special variable 420, 424
ISFLRECL special variable 420
ISFMSG special variable 420
ISFNTBL macro
examples of 89
format of 90
parameters 90
purpose of 89
ISFOWNER special variable 420
ISFPARMS
refreshing 127
ISFPARMS module 15
backup to SAF 192
conditional processing in 20, 22
converting to SAF security 311
converting to statements 17
NTBL statements in 15
running SDSF as a batch job 376
sample 18
sample with SAF resources 312
symbols in 22
WHEN statement 20
ISFPLIB data set
DD statement in TSO logon procedure 27
specifying name and volume serial of 27
ISFPMAC macro 23
ISFPREFIX special variable 420
ISFPRTBLKSIZE special variable 420
ISFPRTCLASS special variable 420
ISFPRTCOPIES special variable 420
ISFPRTDATACLAS special variable 420
ISFPRTDDNAME special variable 420
ISFPRTDEST special variable 420
ISFPRTDIRBLKS special variable 420
ISFPRTDISP special variable 420
ISFPRTDSNAME special variable 420
ISFPRTFCB special variable 420
ISFPRTFORMDEF special variable 420
ISFPRTFORMS special variable 420
ISFPRTLRECL special variable 420
ISFPRTMEMBER special variable 420
ISFPRTMGMTCLAS special variable 420
ISFPRTOUTDESNAME special variable 420
ISFPRTPAGEDEF special variable 420
ISFPRTPRIMARY special variable 420
ISFPRTPRTMODE special variable 420
ISFPRTRECFM special variable 420
ISFPRTSECONDARY special variable 420
ISFPRTSPACETYPE special variable 420
ISFPRTSTORCLAS special variable 420
ISFPRTUCS special variable 420
ISFPRTUNIT special variable 420
ISFPRTVOLSER special variable 420
ISFRCOLS special variable 420
ISFRECFM special variable 420
ISFRESET function 410, 420
ISFRESP special variable 420
ISFROWS special variable 420
ISFSCHARS special variable 420
ISFSCROLL special variable 427
ISFSCROLLTYPE special variable 427
ISFSECTRACE special variable 420
ISFSERVER special variable 420
ISFSLASH command 408
ISFSORT special variable 420
ISFSTARTLINETOKEN special variable 427
ISFSYSID special variable 420
ISFSYSNAME special variable 420
ISFTIMEOUT special variable 420
ISFTITLES special variable 420
ISFTLINE special variable 420

ISFTR macro 101
  syntax 102
  using for defining code pages 101
  using UPCTAB parameter with 101
  using VALTAB parameter with 101
ISFTRACE special variable 420
ISFTRMASK special variable 420
ISFUCOLS special variable 420
ISFULOG special variable 420
ISFUSER module, using to customize SDSF 329
ISPF (Interactive System Productivity Facility)
  adding SDSF to menu 347
  help panel data set
    DD statement in TSO logon procedure 27
    specifying name and volume serial of 27
  main menu 347
ISR@PRIM 347
ISTATUS parameter 61
ISYS default field title
  Held Output Queue panel 145
  Input Queue panel 148
  Output Queue panel 167
  Status panel 189
ISYS parameter 61
ITNAME parameter 40
ITY default field title
  Job Data Set panel 153
IUID parameter 36

J

J0 command
  authorization to issue 235
J0FLD2 parameter 62
J0FLDS parameter 62
Java
  information 451
  security 460
  using with SDSF 451
JC default field title
  Held Output Queue panel 145
  Output Queue panel 167
JCFLD2 parameter 62
JCFLDS parameter 62
JCLIM default field title
  JC panel 150
JDSFLD2 parameter 62
JDSFLDS parameter 62
JES commands, issuing
  authorization for, specifying
    problems with 556
    with a user exit routine 329
    with ISFPARMS parameters 47
    with SAF security 194
    with statements 47
JES2
  DESTDEF considerations 339
  secondary subsystems, installing SDSF with 339
  subsystem name, specifying 26
JES2 resources 179
JES2, processed by SDSF 349, 370
JES3
  considerations 340
  global 340
  subsystem name, specifying 26, 341
JES3, processed by SDSF 349, 370
JES3NAME parameter 26, 341
JESLevel default field title 161
JESLEVEL default field title
  Initiator panel 146
  LI panel 156
  NODE panel 163
  NS panel 162
  Printer panel 172
  Punch panel 176
  Reader panel 177
  RM panel 179
  Spool Offload panel 182
JESLOG, default field title
  Job Class panel 150
JESN default field title 161
  LI panel 156
  NODE panel 163
  NS panel 162
  Punch panel 176
  Reader panel 177
JESNAME default field title
  Initiator panel 146
  MAS and JESPLEX panels 159
  Printer panel 171
  RM panel 179
  Spool Offload panel 182
  SYS panel 186
JESNAME parameter 26
JESNODE default field title
  SYS panel 186
JESPLEX (JP) panel 158
JESPLEX panel
  authorization to display 235
JESSPOOL class 240, 245
JGSTATUS default field title
  Input Queue panel 148
JNAME default field title
  Dynamic Exits panel 138
  Output Queue panel 166
JNUM default field title
  Display Active Users panel 136
  Held Output Queue panel 144
  Initiator panel 146
  Input Queue panel 147
  LI panel 155
  Printer panel 169
  Punch panel 175
  Reader panel 177
  Spool Offload panel 180
  Status panel 188
job 0 panel 154
Job 0 panel
  variable field list for 154
Job Class panel
  fields 149
  variable field list of
    primary, defining 85
job Data Set panel
  variable field list for 151
Job Data Set panel 151
  variable field list of
    alternate, defining 85
    primary, defining 85
Job Status panel 188
JOB value 52
JOBACCT1 default field title
  Held Output Queue panel 145
  Input Queue panel 148
  Output Queue panel 167
  Status panel 189
JOBACCT2 default field title
  Held Output Queue panel 145
  Input Queue panel 148
  Output Queue panel 167
  Status panel 189
JOBACCT3 default field title
  Held Output Queue panel 145
  Input Queue panel 148
  Output Queue panel 167
  Status panel 189
JOBACCT4 default field title
  Held Output Queue panel 145
  Input Queue panel 148
  Output Queue panel 167
  Status panel 189
JOBACCT5 default field title
  Held Output Queue panel 145
  Input Queue panel 148
  Output Queue panel 167
  Status panel 189
JOBCORR default field title
  Display Active Users panel 137
  Held Output Queue panel 145
  Input Queue panel 148
  Output Queue panel 167
  Status panel 189
JOBGROUP default field title
  Input Queue panel 148
JOBGRPID default field title
  Input Queue panel 148
JobID default field title
  NC panel 160
JOBID default field title
  Address Space Memory panel 134
  Display Active Users panel 136
  Held Output Queue panel 144
  Initiator panel 146
  Input Queue panel 147
  LI panel 155
  Output Queue panel 166
  Printer panel 169
  Process panel 174
  Punch panel 175
  Reader panel 177
  Spool Offload panel 180
  SR panel 190
  Status panel 188
Jobname default field title
  NC panel 160
JOBNAME default field title
  Enqueue panel 141
  Initiator panel 146
  LI panel 155
  Printer panel 169
  Punch panel 175
  Reader panel 177
  Spool Offload panel 180
  SR panel 189
JOBNUM default field title
  SYS panel 186

JOBRC, default field title
Job Class panel 151
jobs
canceling
authorization with ISFPARMS 48
authorization with SAF 48
controlling
by issuing MVS/JES command for 47
by issuing MVS/JES2 command for 53
displayed on SDSF panels, controlling
by destination name 59
by job name 61, 73
displaying
SYSOUT and SYSIN for 55, 59
SAF authorization 194, 201, 245
JOBSET default field title
Input Queue panel 148
JP (JESPLEX) panel 158
JP command
authorization to issue 235
JP default field title
Held Output Queue panel 144
Output Queue panel 166
Printer panel 170
JPRIO default field title
Punch panel 176
JR default field title
LI panel 156
JRNL default field title
JC panel 150
JRNum default field title
NC panel 160
JRNUM default field title
NODE panel 163
JT default field title
LI panel 156
JTNum default field title
NC panel 160
JTNUM default field title
NODE panel 163
JTr default field title
NC panel 160
NS panel 161
JTR default field title
LI panel 157
JType default field title
NC panel 160
JTYPE default field title
Address Space Memory panel 134
LI panel 155

K

K default field title
Printer panel 171
keyboard
navigation 589
PF keys 589
shortcut keys 589

L

LABEL default field title
Spool Offload panel 182
LANG parameter in ISFPARMS 62
LAST-CHECKPOINT default field title
Multi-Access Spool panel 159
Last-GCon-Date-Tim default field title
MAS and JESPLEX panels 159
LATCHWAITPID default field title
Process panel 174
length of fields, specifying 85
LEVEL default field title
Enqueue panel 141
LgFree default field title
Spool Volumes panel 184
LI command
authorization to issue 235
LIMIT default field title
RM panel 179
limit of lines searched on OPERLOG 26
limit of lines searched with FIND command 25
Line default field title
NC panel 160
LINE default field title
NODE panel 163
Line-Lim-Hi default field title
LI panel 156
Line-Lim-Lo default field title
LI panel 156
Line-Limit default field title 161
LINE-LIMIT default field title
LI panel 155
Printer panel 171
Punch panel 175
Spool Offload panel 180
Line-Limit-Hi default field title 161
Line-Limit-Lo default field title 161
LINECCHR default field title
Lines panel 156
LINECNT parameter 26
LINEFLD2 parameter 63
LINEFLDS parameter 63
LINELIMH default field title
Printer panel 172
Punch panel 176
Spool Offload panel 182
LINELIML default field title
Printer panel 172
Punch panel 176
Spool Offload panel 182
lines
protecting with SAF 250
Lines panel
authorization to display 46, 235
variable field list 155
variable field list of
alternate, defining 85
primary, defining 85
lines per page of printed output, specifying 26
Link List panel 157
LNKFLD2 parameter 63
LNKFLDS parameter 63
variable field list of
alternate, defining 63
primary, defining 63
Link Pack Area panel 158
LPAFLD2 parameter 63
LPAFLDS parameter 63
Link Pack Area panel *(continued)*
variable field list of
alternate, defining 63
primary, defining 63
LIST ($L) command
authorization to issue, specifying 48, 73, 194
LNAME default field title
NODE panel 163
LOADDSN default field title
SYS panel 187
LOADPARM default field title
SYS panel 187
LOADUNIT default field title
SYS panel 187
LOCAL value 61
LOCALE default field title
CK panel 143
LoCyl default field title
Spool Volumes panel 183
LOG command
authorization to issue 235
Log default field title
NS panel 161
LOG default field title
JC panel 150
Lines panel 156
LOGDATE default field title
CK panel 143
logical log 302
logical SYSLOG 302
LOGLIM parameter 26
LogMode default field title 160
LOGMODE default field title
NODE panel 163
Logon default field title
NC panel 160
NODE panel 163
logon procedure
using to define a group of users 36
LOGSTREAM default field title
CK panel 143
LOGSTRM class 255
LoHead default field title
Spool Volumes panel 183
LOTRK default field title
Spool Volumes panel 183
LPAR default field title
SYS panel 186
LRECL default field title
Authorized Program Facility panel 135
Job Data Set panel 154
Link List panel 157
Link Pack Area panel 158
PARMLIB panel 169
Printer panel 173
Punch panel 176
SRCH panel 184

M

M abbreviation in column data 86
M default field title
Printer panel 171
MAJOR default field title
Enqueue panel 141

MargExc default field title
Spool Volumes panel 183
MargPct default field title
Spool Volumes panel 183
MAS (Multi-access spool) panel 158
MAS command
authorization to issue 235
MAS panel
authorization to display 235
MAX-RC default field title
Held Output Queue panel 145
Output Queue panel 167
Status panel 188
MAX-TIME default field title
JC panel 150
MAXASID default field title
SYS panel 186
MAXFILES default field title
Process panel 174
MAXLINES default field title
NODE panel 163
MAXRETR default field title
NODE panel 163
MB abbreviation in column data 86
MBURST default field title
Spool Offload panel 181
MC default field title
Held Output Queue panel 145
Input Queue panel 148
JC panel 150
Output Queue panel 167
Reader panel 177
Status panel 189
MCLASS default field title
Spool Offload panel 181
MDEST default field title
Spool Offload panel 181
MEMBER field
JC panel 149
membership in groups, SAF control of 252
MEMLIMIT default field title
Address Space Memory panel 134
Display Active Users panel 137
MENUS parameter 27
MENUVOL parameter 27
messages 475
help panels 475
problem with 557
MESSATE-TEXT default field title
SR panel 190
MFCB default field title
Spool Offload panel 181
MFLH default field title
Spool Offload panel 181
MFORMS default field title
Spool Offload panel 181
MHOLD default field title
Spool Offload panel 181
MigDSName default field title
Spool Volumes panel 184
MigPhase default field title
Spool Volumes panel 184
migrating
to statements for ISFPARMS 17
MigSys default field title
Spool Volumes panel 184
MigVol default field title
Spool Volumes panel 184
MinExc default field title
Spool Volumes panel 183
MinPct default field title
Spool Volumes panel 183
mixed case, specifying 101
MLA default field title
SYS panel 187
MOBJ default field title
Address Space Memory panel 134
MOBJHWM default field title
Address Space Memory panel 134
MOBJNUM default field title
Address Space Memory panel 134
Mode default field title
Initiator panel 146
MODE default field title
JC panel 149
Printer panel 171
model queue
defining 116, 346
MODEPA default field title
Dynamic Exits panel 138
MODIFIED default field title
CK panel 142
MODLOADPT default field title
Dynamic Exits panel 138
MODNAME default field title
CK panel 143
Dynamic Exits panel 138
MODSIZE default field title
Dynamic Exits panel 138
MODSP default field title
Spool Offload panel 181
module information, displaying 555
modules, SDSF
ISFPARMS 18
ISFUSER 329
mount messages
protecting with SAF 304
MPRMODE default field title
Spool Offload panel 181
MQSeries
cluster 33
configuration 344
considerations 344
help with 344
objects
defining manually 33
delete option 33
replace option 33
OS/390 libraries 344
queue prefix 33
queues 116, 346
queues, protecting 306
remote queues 346
sample DEFINE commands 116
security 300
storage estimates 345
MQSeries, APAR for large messages 344
MSAFF default field title
Spool Offload panel 181
MSG value 50
MsgDest default field title
MAS and JESPLEX panels 159
MSGLV default field title
JC panel 150
MsgPrefix default field title
MAS and JESPLEX panels 159
MSGTYPE default field title
SR panel 190
MSU default field title
SYS panel 187
MUCS default field title
Spool Offload panel 182
multi-access spool 158
multi-access spool (MAS) panel 158
Multi-Access Spool environment
specifying JES2 system ID in 71
MVS/JES commands
authorization for, specifying
problems with 556
with a user exit routine 329
with ISFPARMS parameters 47, 72
with SAF 202, 252
MVS/JES commands, console used to issue 48, 342
MVS/JES2 commands
authorization for, specifying
with ISFPARMS parameters 50
MWRITER default field title
Spool Offload panel 182

N

NAME default field title
CKH panel 142
Job Data Set panel 152
Output Descriptors panel 164
navigation
keyboard 589
NC command
authorization to issue 235
NC panel 160, 161
variable field list 160
NCFLD2 parameter 64
NCFLDS parameter 64
NETID default field title
SYS panel 187
netmail 67
NETPR default field title
NODE panel 164
NETPU default field title
NODE panel 164
Netsrv default field title
NC panel 160
NETSRV default field title
NODE panel 163
Network Connection panel
authorization to display 46
variable field list of
primary, defining 85
network connections
protecting with SAF 253
Network Server panel
authorization to display 46
variable field list of
primary, defining 85
network servers
protecting with SAF 254

NEWPAGE default field title
Printer panel 172
NIDBUF parameter 27
NO command
authorization to issue 235
NO panel
variable field list 162
NODE default field title
Held Output Queue panel 144
Input Queue panel 147
Job Data Set panel 152
LI panel 155
Output Queue panel 166
Printer panel 171
NODEFLD2 parameter 64
NODEFLDS parameter 64
NODENAME default field title
NODE panel 162
nodes
protecting with SAF 254
Nodes panel
authorization to display 46, 235
variable field list of
alternate, defining 85
primary, defining 85
NONE value 65, 66
Notices 593
NOTIFY default field title
Held Output Queue panel 145
Input Queue panel 148
Job Data Set panel 153
Output Descriptors panel 165
Output Queue panel 167
Spool Offload panel 181
Status panel 188
NOTIFY parameter of job card, using to control authorization 47, 56
NOTIFY value 49, 56, 312
NPRO default field title
Printer panel 171
NS command
authorization to issue 235
NS panel
variable field list 161
NSFLD2 parameter 65
NSFLDS parameter 64
NSNAME default field title
LI panel 156
NTBL statement
examples of 89
format of 90
parameters 90
purpose of 89
NTBLENT statement
examples of 89
purpose of 89
NUCLST default field title
SYS panel 187
NUMCAT default field title
CK panel 142
numeric values, scaling 86

O

O command
authorization to issue 235
O-GRP-N default field title
Held Output Queue panel 144
Job Data Set panel 152
Output Queue panel 166
Printer panel 171
OCLASS default field title
Punch panel 176
OCOPYCNT default field title
Job Data Set panel 154
OCOPYCNT Default field title
Output Descriptor panel 166
ODFLD2 parameter 65
ODFLDS parameter 65
ODISP default field title
Held Output Queue panel 144
JC panel 149
Output Queue panel 167
offloaders
protecting with SAF 296, 301
Offs default field title
Status panel 189
OffS default field title
Held Output Queue panel 145
Output Queue panel 167
OFFSETXB default field title
Job Data Set panel 153
Output Descriptors panel 165
OFFSETXF default field title
Job Data Set panel 153
Output Descriptors panel 165
OFFSETYB default field title
Job Data Set panel 153
Output Descriptors panel 165
OFFSETYF default field title
Job Data Set panel 153
Output Descriptors panel 165
OFIELD2 parameter 65
OFIELDS parameter 65
OGID default field title
Held Output Queue panel 144
OGID1 default field title
Output Queue panel 166
Printer panel 172
Punch panel 176
OGID2 default field title
Held Output Queue panel 144
Output Queue panel 166
Printer panel 172
Punch panel 176
OGNAME default field title
Punch panel 176
OHR default field title
Held Output Queue panel 144
Output Queue panel 167
OPACTLOG default field title
Printer panel 172
operator group 312, 315
OPERCMDS class 212
OPERLOG panel 255
including inactive log stream data 95
limiting amount searched 26
OPTIONS statement 23
parameters 23
ORIGIN default field title
CK panel 143
ORIGINAL default field title
Enclave panel 139
ORIGNODE default field title
Input Queue panel 147
Status panel 188
OSCONFIG default field title
SYS panel 187
OUT default field title
JC panel 150
OUT value 52
OUTBIN default field title
Job Data Set panel 153
Output Descriptors panel 165
Output Data Set panel
authorization to use, specifying 55, 329
Output Descriptors panel 164
variable field list 164
variable field list of
alternate, defining 65
primary, defining 65
output groups
SAF authority to 194, 201, 245
output of jobs
controlling
by issuing MVS/JES commands for 47
by issuing MVS/JES2 commands for 53
displaying
specifying authorization for 55, 59, 329
Output Queue panel 166
authorization to display 46, 235
authorization to display, specifying 235
jobs displayed on, controlling
by destination name 58, 59
by job name 61, 73
by job name prefix 66
by owner 65
variable field list of 166
alternate, defining 85
primary, defining 85
OUTPUT-HOLD-TEXT default field title
Held Output Queue panel 144
Output Queue panel 167
OVA default field title
Spool Volumes panel 183
OverFNam default field title
Spool Volumes panel 183
OverInto default field title
Spool Volumes panel 183
OVERLAYB default field title
Job Data Set panel 153
Output Descriptors panel 165
OVERLAYF default field title
Job Data Set panel 153
Output Descriptors panel 165
OverOccur default field title
Spool Volumes panel 183
overtypeable fields
setting up generic and discrete profiles 256
tables 258
overtyping fields
authorization for 47, 50
authorization for, SAF 194, 202, 255
reference for, by command level 73

OVERWARN default field title
RM panel 179
OWNER command
authorization 312, 314, 578
Owner default field title
NC panel 160
OWNER default field title
CKH panel 141
Display Active Users panel 136
Held Output Queue panel 144
Initiator panel 146
Input Queue panel 147
Job 0 panel 154
Job Data Set panel 151
LI panel 155
Output Queue panel 166
Printer panel 170
Process panel 174
Punch panel 175
Reader panel 177
Spool Offload panel 180
Status panel 188
OWNER keyword 312
OWNERAS default field title
Enclave panel 139
OWNERASX default field title
Enclave panel 139
OWNERID default field title
Address Space Memory panel 134
OWNERJOB default field title
Enclave panel 139
OWNERS default field title
Enqueue panel 141
OWNERSYS default field title
Enclave panel 139

P

P abbreviation in column data 86
Page panel 168
PAG parameter 66
variable field list of
alternate, defining 66
primary, defining 66
PAGE-CNT default field title
Job 0 panel 155
Job Data Set panel 151
Printer panel 170
Punch panel 175
Page-Lim-Hi default field title
LI panel 156
Page-Lim-Lo default field title
LI panel 156
Page-Limit default field title 161
PAGE-LIMIT default field title
LI panel 155
Printer panel 171
Spool Offload panel 180
Page-Limit-Hi default field title 161
Page-Limit-Lo default field title 161
PAGE-PRT default field title
Printer panel 170
Punch panel 175
PAGEDEF default field title
Job Data Set panel 152
Output Descriptors panel 164
PAGELIMH default field title
Printer panel 172
Spool Offload panel 182
PAGELIML default field title
Printer panel 172
Spool Offload panel 182
PAGERATE default field title
SYS panel 186
PAGING default field title
Display Active Users Panel 136
panel data set
DD statement in TSO logon procedure 27
specifying name and volume serial of 27
Panel scope
controlling 96, 98, 99
panels
description of
Address Space Memory 133
Authorized Program Facility 135
CK 142
CKH 141
Display Active Users 135
Dynamic Exits 138
Enclaves panel 139
Enqueue 140
Held Output panel 144
Initiator panel 146
Input Queue panel 147
LI panel 155
Link List 157
Link Pack Area 158
multi-access spool (MAS) panel 158
NC panel 160
NO panel 162
NS panel 161
Output Descriptors panel 164
Output Queue panel 166
PAG 168
PARM 169
Printer panel 169
Process panel 174
PUN panel 174
RDR panel 177
Spool Offload panel 180
Spool Volumes panel 182
SRCH 184
Status panel 188
SYS 186
System Symbols 185
Panels
Address Space Memory panel 133
Authorized Program Facility panel 135
Dynamic Exits panel 138
Enqueue panel 140
Link List panel 157
Link Pack Area panel 158
PAG panel 168
PARMLIB panel 169
SRCH panel 184
SYS panel 186
System Symbols panel 185
PARM default field title
CK panel 143
PARM panel scope
controlling 98
PARMLEN default field title
CK panel 143
PARMLIB panel 169
PARMFLD2 parameter 66
PARMFLDS parameter 66
variable field list of
alternate, defining 66
primary, defining 66
PARTNAM default field title
NODE panel 163
PARTNAM, default field title
Job Class panel 151
PartName default field title
MAS and JESPLEX panels 159
Spool Volumes panel 183
Password default field title
NS panel 162
PASSWORD default field title
Lines panel 156
password protection
bypassing 23
PATH default field title
NODE panel 163
pattern matching characters, for PREFIX 27
PAU default field title
Printer panel 171
Punch panel 176
PB abbreviation in column data 86
PBufAux default field title
MAS and JESPLEX panels 159
PBufCSA default field title
MAS and JESPLEX panels 159
PBufFixed default field title
MAS and JESPLEX panels 159
PDefault default field title
Printer panel 173
PEN default field title
NODE panel 162
PER default field title
Enclave panel 139
permitting other users to view your jobs 247
PGN default field title
Enclave panel 139
JC panel 150
PGNM default field title
JC panel 150
PHASE default field title
Input Queue panel 148
Status panel 189
PHASENAME default field title
Input Queue panel 148
Status panel 189
PI default field title
Reader panel 177
PID default field title
Process panel 174
PL default field title
JC panel 150
PLult field title
Reader panel 177
PMGR default field title
NODE panel 163

point-and-shoot
custom property 99
POLSTAT default field title
CK panel 142
Port default field title
NC panel 160
NS panel 162
PORT default field title
Job Data Set panel 153
LI panel 156
PortName default field title 160
PORTNAME default field title
LI panel 156
NS panel 162
PORTNO default field title
Output Descriptors panel 165
POS default field title
Address Space Memory panel 134
Display Active Users panel 136
Input Queue panel 147
Status panel 188
post-SAF exit point 335
PPID default field title
Process panel 174
PR command
authorization to issue 235
PR panel
device name format 99
pre-SAF exit point 335
prefix
of job names, using to control jobs
with ISFPARMS 66
with SDSF's internal parameters 57, 58
PREFIX command
adding generic character 94, 331
authorization to issue 235
PREFIX parameter 66
PReq default field title
Reader panel 177
PREVCKPT default field title
Multi-Access Spool panel 158
primary field lists for SDSF panels 85, 87
primary option menu
case of 101
PrimTG default field title
MAS and JESPLEX panels 159
PRINT command
default SYSOUT class for 28
lines per page 26
Printer panel 169
authorization to display 46, 235
controlling, using job destination name 59
variable field list of 169
alternate, defining 85
primary, defining 85
printers
authorization to control 48, 73, 194, 296
controlling
by issuing MVS/JES commands 47
by issuing MVS/JES2 commands 53
printing
controlling lines per page 26
custom property 100
PRIO, default field title
Job Class panel 151
PRITRK, default field title
Job Class panel 151
PRMODE default field title
Held Output Queue panel 144
Job 0 panel 154
Job Data Set panel 152
Output Queue panel 167
Printer panel 170
Punch panel 176
problems
management of 555
Proc-Lines default field title
NC panel 160
PROC-LINES default field title
LI panel 155
SO panel 180
Process panel 174
authorization to display 46, 235
variable field list of
alternate, defining 67
primary, defining 67
PROCNAME default field title
CK panel 143
PROCS default field title
Address Space Memory panel 134
PROCSTEP default field title
Display Active Users pane 135, 136
Initiator panel 146
Job 0 panel 155
Job Data Set panel 151
profile descriptions 325
PROGRAMMER-NAME default field title
Held Output Queue panel 145
Input Queue panel 148
Output Queue panel 167
Status panel 188
PROMOTED default field title
Display Active Users panel 137
Enclaves panel 140
properties for use with the PROPERTY statement 92
PROPERTY statement 91
parameters 92
PROPLIST statement 91
parameters 92
PROT default field title
Spool Offload panel 182
PRT-PAGE default field title
Output Queue panel 166
PRT-REC default field title
Output Queue panel 166
PRTDEF default field title
NODE panel 164
PRTDEST default field title
Input Queue panel 147
Reader panel 177
Status panel 188
PRTFLD2 parameter 67
PRTFLDS parameter 67
PRTOPTNS default field title
Output Descriptors panel 165
PRTQUEUE default field title
Output Descriptors panel 165
PRTTSO default field title
NODE panel 164
PRTXWTR default field title
NODE panel 164
PRTY default field title
Held Output Queue panel 144
Input Queue panel 147
Output Queue panel 166
Status panel 188
PRVT default field title
NODE panel 163
PS command
authorization to issue 235
PS panel scope
controlling 99
PSEL default field title
Printer panel 171
PSFLD2 parameter 67
PSFLDS parameter 67
PTracks default field title
Spool Volumes panel 183
PTrackU default field title
Spool Volumes panel 183
PTYPE default field title
NODE panel 163
PUN command
authorization to issue 235
PUN panel
device name format 99
variable field list 174
variable field list of 174
Punch panel
authorization to display 46, 235
punches
authorization to control 298
protecting with SAF 297
PUNDEF default field title
NODE panel 164
PUNDEST default field title
Reader panel 177
PUNFLD parameter 67
PUNFLD2 parameter 67
PURGE ($C) command
authorization with ISFPARMS 48, 73
authorization with SAF 194
purging jobs and output
authorization for, specifying 48
PWCNTL default field title
NODE panel 164

Q

QHLD default field title
JC panel 149
query character, specifying 24
QUERY MODULE command 555
QUEUE default field title
SR panel 190
Status panel 188
QUEUE Default field title
Job 0 panel 154
Job Data Set panel 153
queue manager
communication between 346
on COMM statement 32

queue manager *(continued)*
remote 346
queues
defining 116, 346
defining remote 346
protecting 306
QUIESCE default field title
Display Active Users panel 137
Enclave panel 139

R

RACF
brief summary of commands 205
conditional access 209
converting ISFPARMS to 323
creating a resource group profile 207
delegation of authority 206
establishing security 207
security administration 205
using RACLIST 208
using RACLIST REFRESH 208
RACLIST 208, 249, 250
RALEVEL default field title
Reader panel 178
RD-DATE default field title
Held Output Queue panel 145
Input Queue panel 148
Output Queue panel 167
Status panel 189
RD-TIME default field title
Held Output Queue panel 145
Input Queue panel 148
Output Queue panel 167
Status panel 189
RDR panel
device name format 99
variable field list 177
variable field list of 177
Reader panel
authorization to display 46, 235
readers
protecting with SAF 298
READY value 52
REAL default field title
Address Space Memory panel 134
Display Active Users Panel 136
REALAFC default field title
SYS panel 186
REALAFCB default field title
SYS panel 186
REASON default field title
CK panel 143
reassembling SDSF 351
REC-CNT default field title
Job 0 panel 155
Job Data Set panel 151
Printer panel 170
Punch panel 175
Reader panel 177
REC-PROC default field title
Reader panel 177
REC-PRT default field title
Printer panel 170
Punch panel 175
RECEIVE default field title
NODE panel 162
RECFM default field title
Authorized Program Facility panel 135
Job 0 panel 155
Job Data Set panel 154
Link List panel 157
Link Pack Area panel 158
PARMLIB panel 169
SRCH panel 185
RECPERTRK default field title
Spool Volumes panel 183
REFDATE default field title
Authorized Program Facility panel 135
Link List panel 157
Link Pack Area panel 158
PARMLIB panel 169
SRCH panel 185
REFRESH 208
Region default field title
Reader panel 178
REGION default field title
JC panel 150
RelConn default field title
NC panel 160
RELEASE ($A) command
authorization to issue 48, 73, 194
RELEASE ($O) command
authorization to issue 48, 50, 73, 194
releasing
jobs
authorization for, specifying 48
output of jobs
authorization for, specifying 48
remote queue
defining 346
REPEAT ($N) command
authorization to issue 48, 73, 194
Repeat-Find PF key
problems with 556
REPLY (R) command
authorization to issue 73
reply messages
protecting with SAF 304
REQSTATE default field title
RES panel 178
REQTIME default field title
Enqueue panel 141
REQTYPE default field title
Enqueue panel 141
requests
protecting with SAF 304
Reserved default field title
Spool Volumes panel 184
RESET (E) command
authorization to issue, specifying 48, 73, 194
RESFLD2 parameter 68
RESFLDS parameter 68
RESGROUP default field title
Display Active Users panel 136
Enclave panel 139
resource group class
GSDSF 206
RACLIST 207
Resource Monitor panel
authorization to display 46, 235
Resource Monitor panel *(continued)*
RMFLD2 parameter 68
RMFLDS parameter 68
variable field list 179
variable field list of primary, defining 68
Resource panel
variable field list 178
variable field list of alternate, defining 85
primary, defining 85
resources 178, 179
Resources panel
authorization to display 46
Rest default field title
NC panel 160
REST default field title
Lines panel 156
NODE panel 162
Rest-Int default field title
LI panel 157
NS panel 161
RESTART ($E) command
authorization to issue, specifying 48, 73, 194
Restart default field title
NS panel 161
RESTART default field title
LI panel 157
ResType default field title
Initiator panel 146
RESULT default field title
CK panel 142
CKH panel 141
RETAINF default field title
Output Descriptors panel 165
RETAINS default field title
Output Descriptors panel 165
RETRYL default field title
Output Descriptors panel 165
RETRYT default field title
Output Descriptors panel 165
return codes
ISFBROWSE 397
REXX
action characters 418
browsing 394
commands 414
diagnosing 449
examples 429
ISFACT 388
ISFBROWSE 394
ISFCALLS 379
ISFEXEC 380
ISFGET 401
ISFLOG 403
ISFRESET 410, 420
ISFSLASH 408
printing 399
security 448
special variables 410, 420
system 448
using with SDSF 377
REXXIN default field title
CK panel 143
REXXOUT default field title
CK panel 143

RFIND NOT ACTIVE message, ISPF 556
RM command
  authorization to issue 235
RMF (Resource Measurement Facility)
  bypassing 331
  Display Active Users panel 135
  exit not installed 558
  migration considerations 344
  services, protecting 344
RMLEVEL default field title
  Reader panel 178
RMT default field title
  Held Output Queue panel 144
  Input Queue panel 147
  Job Data Set panel 152
  Output Queue panel 166
  Printer panel 171
RMTSHR default field title
  LI panel 156
RNAMEL default field title
  Enqueue panel 141
RNUM default field title
  Held Output Queue panel 145
  Input Queue panel 148
  Output Queue panel 167
  Status panel 188
ROOM default field title
  Job Data Set panel 152
  Output Descriptors panel 165
ROUTE ($R) command
  authorization to issue, specifying 48, 73, 194
ROUTECD default field title
  SR panel 190
routing codes
  and WTOR messages displayed on the SYSLOG 44, 46
ROWACTIVE special variable 429
RPRIO default field title
  Reader panel 178
RPTCLASS default field title
  Display Active Users panel 137
  Enclave panel 139
RSID default field title
  Multi-Access Spool panel 159
RST default field title
  JC panel 150
RSYS command
  authorization to issue 235
RTPD default field title
  Spool Offload panel 182

S

SAF
  grouping class 207
  migration considerations 311
SAF indeterminate exit point 336
SAF logging
  custom property 100
SAF security
  access authority 194
  action characters 211, 228
  APF 235
  auditing requests 192
  Authorized program facility data sets 235
SAF security *(continued)*
  checks on the CK and CKH panels 238
  classes 193
  converting ISFPARMS 311
  DEST command 239
  destination names 239
  destination operator authority 240
  Dynamic exit information 242
  DYNX 242
  enclaves 242
  ENQ 243
  Enqueue information 243
  establishing 205
    broad access considerations 207
    conditional access 209
    RACLIST command 208
    REFRESH command 208
  generated MVS and JES commands 212, 213, 256
  GROUP 191
  groups 312
    end users 318
    operators 315
    summary of profiles 321
    system programmers 313
  initiators 243
  ISFGRP 191
  JES2 resources on the RM panel 244
  JESPLEX members 251
  JESSPOOL class 240, 245
  job classes 245
  jobs 245
  lines 250
  Link list data sets 250
  LNK 250
  MAS members 251
  migration from ISFPARMS 561
  migration from ISFPARMS parameters 192, 561
  MVS and JES commands on command line 252
  network connections 253
  network servers 254
  nodes 254
  offloaders 296, 301
  output groups 245, 246
  overtypeable fields 255, 293
  PAG 294
  Page data sets 294
  PARM 295
  PARM data sets 295
  permitting other users to view your data 247
  printers 295
  protecting SDSF tasks 193
  punches 297
  readers 298
  resource names 193, 577
  scheduling environments 299
  SDSF commands 235
  SDSF resources 193, 211
  server 300
  supplementing 193, 329
  SYM 303
  SYS 303
  SYSIN/SYSOUT data sets 245, 246
SAF security *(continued)*
  SYSLOG 302
  System information 303
  system requests 304
  System Symbol information 303
  Table of SDSF Resource Names 577
  WLM enclaves 242
  WLM resources 298
SAFF default field title
  Input Queue panel 147
  Reader panel 177
  Spool Volumes panel 183
  Status panel 188
sample
  ISFPARMS module 18
  SAF security for groups 312
  SMP/E UCLIN jobs for automatic reassembly 351
sample ISFPARMS to restrict batch 376
SBURST default field title
  Printer panel 170
  Spool Offload panel 180
scaling, numeric values 86
SCHARS parameter 27
SCHDATE default field title
  CK panel 143
Scheduling Environment panel
  variable field list of
    alternate, defining 85
    primary, defining 85
scheduling environments 179
Scheduling environments panel
  authorization to display 46
Scheduling-Env default field title
  JC panel 150
SCHENV default field title
  Input Queue panel 147
  Status panel 188
SCHINT default field title
  CK panel 143
SCLASS default field title
  Printer panel 169
  Punch panel 175
  Spool Offload panel 180
SCLASS1-8 default field title
  Spool Offload panel 182
SCN default field title
  JC panel 150
SCOPE default field title
  Enclave panel 139
  Enqueue panel 141
SCPU default field title
  Display Active Users panel 136
screen size
  in batch 369
  specifying maximum supported 27
SCRSIZE parameter 27
SDEPTH, default field title
  Job Class panel 151
SDEST1 default field title
  Printer panel 170
  Punch panel 175
  Spool Offload panel 180
SDEST2 default field title
  Printer panel 170
SDEST2-4 default field titles
  Punch panel 175

SDEST3 default field title
Printer panel 170
SDEST4 default field title
Printer panel 170
SDISP default field title
Spool Offload panel 180
SDSF termination exit point 335
SDSF, multiple instances of 342
SE panel 179
search characters 27
Search panel 184
SecLabel default field title
Display Active Users panel 137
Initiator panel 146
Process panel 174
Reader panel 177
SECLABEL default field title
Held Output Queue panel 144
Input Queue panel 147
Job 0 panel 154
Job Data Set panel 152
Output Queue panel 166
Printer panel 170
Punch panel 176
Status panel 188
secondary JES2 subsystems 339
SecTG default field title
MAS and JESPLEX panels 159
SECTRK, default field title
Job Class panel 151
Secure default field title
NC panel 160
NS panel 162
SECURE default field title
LI panel 156
NODE panel 163
security conversion assist 323
SEFLD2 parameter 69
SEFLDS parameter 69
SEGID default field title
Job 0 panel 154
Job Data Set panel 152
Sel default field title
Spool Volumes panel 184
SELECT default field title
Printer panel 173
Punch panel 175
SELECT Default field title
Job 0 panel 155
Job Data Set panel 153
SELECT-RANGE default field title
Spool Offload panel 180
SelectModeName default field title
MAS and JESPLEX panels 159
sending comments to IBM xxi
SENDP default field title
NODE panel 163
SENTRS default field title
NODE panel 162
SEP default field title
Printer panel 170
Punch panel 175
SEPCHAR default field title
Printer panel 171
SEPDS default field title
Printer panel 170
Punch panel 175
SEQ default field title
Authorized Program Facility panel 135
Dynamic Exits panel 138
Link List panel 157
Link Pack Area panel 158
PARMLIB panel 169
SRCH panel 184
SEQMAX default field title
Dynamic Exits panel 138
server
accessing 109
changing options 122
communications 116
problems with 558
security 306
default, specifying 34
DISPLAY command 126, 555
displaying information about 125
group
defining 30
displaying information about 126
multiple levels of JES2 in 339
sample statements 30
starting communications 128
groups, adding to or modifying 115
groups, examples 111
initial state, specifying 32
input to 107
introduction 107
logging 109
multiple servers on a system 107
operator commands 118
options 118
processing statements 108
protecting with SAF 300
REFRESH command 127
refreshing ISFPARMS 127
registering with ARM 118
security 110
specifying 27
START,COMM command 128
starting 118
STOP,COMM command 129
stopping 130
stopping communications 129
system it runs on 31
XCF 34
SERVER default field title
Display Active Users panel 137
Process panel 174
SERVER parameter 27
SAF authorization for 235
SERVER statement 30, 31
SERVERGROUP statement 30, 31
SESSION default field title
LI panel 155
SETNAME default field title
Link Listpanel 157
SETUP default field title
Printer panel 171
Punch panel 176
SEVCODE default field title
CK panel 142
SEVERITY default field title
CK panel 142
SFCB default field title
Printer panel 170
Spool Offload panel 180
SFLH default field title
Printer panel 170
Spool Offload panel 180
SFORM2 default field title
Punch panel 175
Spool Offload panel 180
SFORM2-8 default field title
Printer panel 169
SFORM3 default field title
Punch panel 175
Spool Offload panel 180
SFORM4 default field title
Punch panel 175
Spool Offload panel 180
SFORM5 default field title
Punch panel 175
Spool Offload panel 180
SFORM6 default field title
Punch panel 175
Spool Offload panel 180
SFORM7 default field title
Punch panel 175
Spool Offload panel 180
SFORM8 default field title
Punch panel 175
Spool Offload panel 180
SFORMS default field title
Printer panel 169
Punch panel 175
Spool Offload panel 180
shared spool 158
sharing HASPINDX data set 349
SHOLD default field title
Spool Offload panel 180
shortcut keys 589
SHRMO default field title
Address Space Memory panel 134
SHRMOHWM default field title
Address Space Memory panel 134
SHRMONUM default field title
Address Space Memory panel 134
SID default field title
MAS panel 158
Multi-Access Spool panel 158
SIO default field title
Display Active Users Panel 136
SYS panel 186
SJOBNAME default field title
Printer panel 170
Punch panel 175
Spool Offload panel 180
Slash command
modifying the character used 94
SLCPU default field title
Display Active Users panel 137
SLEVEL default field title
MAS and JESPLEX panels 159
SLOTS default field title
PAG panel 168
SMC default field title
Enqueue panel 141
SMF default field title
SYS panel 186

SMP/E (System Modification Program Extended)
UCLIN jobs 351
SMS default field title
Authorized Program Facility panel 135
Link List panel 157
Link Pack Area panel 158
PARMLIB panel 169
SRCH panel 184
SNODE1 default field title
Printer panel 172
SNODE2 default field title
Printer panel 172
SNODE3 default field title
Printer panel 172
SNODE4 default field title
Printer panel 172
SO command
authorization to issue 235
Socket default field title
NS panel 161
SOCKETN default field title
LI panel 156
SODISP default field title
LI panel 157
Spool Offload panel 181
SODISP2 default field title
LI panel 157
Spool Offload panel 181
SODISP3 default field title
LI panel 157
Spool Offload panel 181
SODISP4 default field title
LI panel 157
Spool Offload panel 181
SODsp default field title
NC panel 161
SOFLD2 parameter 69
SOFLDS parameter 69
SOWNER default field title
Printer panel 170
Punch panel 175
Spool Offload panel 180
SP command
authorization to issue 235
SP default field title
Display Active Users panel 136
SPAGING default field title
Display Active Users panel 136
special variable 425
SPEED default field title
Lines panel 156
SPIN default field title 189
Display Active Users panel 137
Input Queue panel 148
SPIN Default field title
Job 0 panel 155
Job Data Set panel 153
SPINNABLE default field title
Job Data Set panel 154
spool data set
data sets, number of supported by SDSF 25
password protecting 23
RACF protecting 23
specifying prefix for 25
Spool Offload panel
authorization to display 46, 235
variable field list 180
variable field list of 180
alternate, defining 85
primary, defining 85
spool offloaders
protecting with SAF 296, 301
SAF authorization 203
Spool Volumes panel
authorization to display 46, 235
SPFLD2 parameter 69
SPFLDS parameter 69
variable field list 182
variable field list of
alternate, defining 69
primary, defining 69
SPOOLPCT default field title
SYS panel 186
SPRMODE1 default field title
Printer panel 170
Punch panel 175
Spool Offload panel 180
SPRMODE2 default field title
Printer panel 170
SPRMODE2-4 default field titles
Punch panel 175
SPRMODE3 default field title
Printer panel 170
SPRMODE4 default field title
Printer panel 170
SQA default field title
Address Space Memory panel 134
SQAPCT default field title
Address Space Memory panel 134
SYS panel 186
SR command
authorization to issue 235
SR default field title
LI panel 156
SRANGE default field title
Printer panel 170
Punch panel 176
SRCH panel
SRCH parameter 70
variable field list of
alternate, defining 70
primary, defining 70
SRFLD2 parameter 70
SRFLDS parameter 70
SRNum default field title
NC panel 160
SRNUM default field title
NODE panel 163
SRout1 default field title
Printer panel 172
SRout2 default field title
Printer panel 172
SRout3 default field title
Printer panel 172
SRout4 default field title
Printer panel 172
SrvClass default field title
Initiator panel 146
SRVCLASS default field title
Enclave panel 139
Input Queue panel 147
SRVCLASS default field title *(continued)*
Status panel 188
SrvJobNm default field title
NS panel 162
SrvName default field title 161
SSAFF default field title
Spool Offload panel 180
SSCHEDULING—ENV default field title
Spool Offload panel 181
SSCP default field title
SYS panel 187
SSI (subsystem interface) request 76, 78
SSIGNON default field title
NODE panel 163
SSMODE default field title 188
Input Queue panel 148
SSRVCLASS default field title
Spool Offload panel 181
SSTYPE default field title
Enclave panel 139
ST command
adding generic character 94, 331
authorization to issue 235
ST default field title
LI panel 156
ST-DATE default field title
Held Output Queue panel 145
Input Queue panel 148
Output Queue panel 167
Status panel 189
ST-TIME default field title
Held Output Queue panel 145
Input Queue panel 148
Output Queue panel 167
Status panel 189
Stack default field title
NS panel 161
START ($S) command
authorization to issue 48, 73, 194
Start-Date-Time default field title
MAS and JESPLEX panels 159
Start-Type default field title
MAS and JESPLEX panels 159
STARTBY default field title
Input Queue panel 149
STATDATE default field title
SYS panel 187
STATE default field title
CK panel 142
Process panel 174
statements
using to define
fields overtypeable on SDSF panels 47
MVS and JES commands that can be issued 47
Status default field title
NC panel 160
STATUS default field title
Authorized Program Facility panel 135
CK panel 142
CKH panel 141
Display Active Users panel 136
Enclave panel 139
Enqueue panel 141
Held Output Queue panel 144

STATUS default field title *(continued)*
Initiator panel 146
Input Queue panel 147
JESPLEX panel 158
LI panel 155
NODE panel 162
NS panel 161
Output Queue panel 166
PAG panel 168
Printer panel 169
Process panel 174
PUN panel 174
RDR panel 177
RM panel 179
Spool Offload panel 180
Spool Volumes panel 183
SRCH panel 184
Status panel 188
STATUS field
JC panel 149
Status panel 188
authority to display 46, 235
jobs displayed on, controlling
by destination name 59
by job name 61, 73
by job name prefix 66
by owner 65
variable field list of 188
alternate, defining 85
primary, defining 85
STC value 52
STCID default field title
CK panel 143
STCNUM default field title
SYS panel 186
STEPN default field title
Address Space Memory panel 134
STEPNAME default field title
Display Active Users panel 135
Initiator panel 146
Job 0 panel 155
Job Data Set panel 151
STFLD2 parameter 70
STFLDS parameter 70
STNum default field title
NC panel 160
STNUM default field title
NODE panel 163
STOKEN default field title
Dynamic Exits panel 138
STOP ($P) command
authorization to issue 48, 73, 194
STOP command 130
authorization to issue 73
stopping the server 130
storage estimates for MQSeries page sets 345
storage estimates for WebSphere MQ page sets 345
STORCRIT default field title
Display Active Users panel 137
STREAM default field title
NODE panel 164
STT default field title
Spool Volumes panel 183
Stunted default field title
Spool Volumes panel 183
SubGroup default field title
Held Output Queue panel 145
Input Queue panel 148
Output Queue panel 167
Status panel 189
SUBNET default field title
NODE panel 163
SUBSYS default field title
Address Space Memory panel 134
Enclave panel 139
subsystem interface (SSI) request 76, 78
subsystem name, JES2, specifying 26
subsystem name, JES3, specifying 26, 341
SUBUSER default field title
Input Queue panel 148
Status panel 189
SUCS default field title
Printer panel 170
Spool Offload panel 181
summary of changes
as updated April 2016 xxiii
as updated December 2015 xxv
as updated March 2014 xxv
as updated March 2016 xxiv
as updated October 2016 xxiii
Summary of changes xxvi
SUS default field title
Punch panel 176
SUSPEND default field title
Printer panel 173
SVOL default field title
Printer panel 172
Punch panel 175
Spool Offload panel 180
SVOL2 default field title
Printer panel 172
SVOL2-4 default field title
Punch panel 176
SVOL3 default field title
Printer panel 172
SVOL4 default field title
Printer panel 172
SWA default field title
JC panel 150
Reader panel 178
SWAPR default field title
Address Space Memory panel 134
Display Active Users panel 136
SWRITER default field title
Printer panel 170
Punch panel 175
SWRITER. default field title
Spool Offload panel 181
symbols, using in ISFPARMS 22
SYNCTOL default field title
Multi-Access Spool panel 159
SYSID (system ID)
parameter in ISFPARMS 71
specifying in Multi-Access Spool environment 71
system ID of, specifying 71
SYSID command
authorization to issue 235
SysID default field title
NC panel 161
NODE panel 163
SYSID default field title
Held Output Queue panel 145
Initiator panel 146
LI panel 156
NS panel 162
Output Queue panel 166
Printer panel 171
Punch panel 176
Reader panel 177
Spool Offload panel 182
SYSID3 (system ID)
parameter in ISFPARMS 71
specifying in JES3 environment 71
SYSIN (system input)
displaying for jobs
authorization for, specifying 55
SYSIN/SYSOUT
SAF authorization 194, 245
SYSLEV default field title
CK panel 143
SYSLEVEL default field title
Address Space Memory panel 134
Authorized Program Facility panel 135
Dynamic Exits panel 138
Enclave panel 139
Link List panel 157
Link Pack Area panel 158
PAG panel 168
PARMLIB panel 169
Process panel 174
SYS panel 186
System Symbol panel 185
SYSLOG
protecting with SAF 302
SYSLOG index
block size of, calculating 25
buffers, specifying 27
name of 25
volume serial of 26
SYSLOG panel
system ID of SYSLOG displayed on 71
SYSMSU default field title
SYS panel 187
SYSNAME
command 61
SYSNAME command
authorization to issue 235
SysName default field title 161
NODE panel 163
SYSNAME default field title
Address Space Memory panel 134
Authorized Program Facility panel 135
CK panel 143
CKH panel 142
Dynamic Exits panel 138
Enclave panel 139
Enqueue panel 141
Initiator panel 146
Input Queue panel 148
JESPlex panel 159
LI panel 156
Link List panel 157
Link Pack Area panel 158
Multi-Access Spool panel 159

SYSNAME default field title *(continued)*
NS panel 162
PAG panel 168
PARMLIB panel 169
Printer panel 171
Process panel 174
Punch panel 176
Reader panel 177
RM panel 179
Spool Offload panel 182
SR panel 189
System Symbol panel 185
SYSOUT (system output)
class, specifying default for 29
displaying for jobs
specifying authorization for 55, 59, 329
SYSOUT display authority exit point 334
SYSOUT parameter 28
sysplex
defining server group 30
ISFPARMS in 20
server group examples 111
sysplex DA 61, 558
disabling 331
migration considerations 344
SYSPLEX default field title
CKH panel 142
SYS panel 186
SYSSYM, default field title
Job Class panel 151
system commands, console used to issue 48, 342
SYSTEM default field title
Display Active Users panel 136
system field
RES panel 178
system ID, specifying 71
system input (SYSIN)
displaying for jobs
authorization for, specifying 55
system log
protecting with SAF 302
System Modification Program Extended (SMP/E)
ISFASMP 351
UCLIN jobs 351
system output (SYSOUT)
class, specifying default for 29
displaying for jobs
specifying authorization for 55, 59, 329
System panel 186
SYS parameter 71
variable field list of
alternate, defining 71
primary, defining 71
system programmer group 312, 313
system requests
protecting with SAF 304
system symbols
disabling 93
System symbols panel
SYMFLDS parameter 70
variable field list of
primary, defining 70
System Symbols panel 185
system symbols, using in ISFPARMS 22
SYSTEMS default field title
SE panel 179
SzAAP% default field title
Display Active Users panel 137
SzIIP% default field title
Display Active Users panel 137

T

T abbreviation in column data 86
table build exit point 330, 337
Target default field title
Spool Volumes panel 184
TB abbreviation in column data 86
TDEPTH default field title
JC panel 150
terminal names
using to define a group of users 36
terminal screen, largest supported 27
TGNUM default field title
Input Queue panel 147
Spool Volumes panel 183
Status panel 188
TGPCT default field title
Input Queue panel 147
Spool Volumes panel 183
Status panel 188
TGUE default field title
Spool Volumes panel 183
Time default field title
Reader panel 178
TIME default field title
SR panel 190
TIMEE default field title
Process panel 174
RM panel 179
TIMEOUT interval, default 28
TIMEOUT parameter 28
title 86
TITLE default field title
Job Data Set panel 152
Output Descriptors panel 164
tokens 390
TOT-ERRS default field title
LI panel 155
Tot-Lines default field title
NC panel 160
TOT-LINES default field title
Input Queue panel 148
LI panel 155
SO panel 180
Status panel 189
TOT-PAGE default field title
Held Output Queue panel 144
Output Queue panel 166
TOT-REC default field title
Held Output Queue panel 144
Output Queue panel 166
TOTERRS default field title
PAG panel 168
TP Default field title
Job Data Set panel 153
TP26 default field title
JC panel 150
TP6 default field title
JC panel 150
TPACCT default field title
Job Data Set panel 154
TPDATEE default field title
Job Data Set panel 154
TPDATER default field title
Job Data Set panel 153
TPJNAME default field title
Job Data Set panel 153
TPJOBID default field title
Job Data Set panel 153
TPTIMEE default field title
Job Data Set panel 153
TPTIMER default field title
Job Data Set panel 153
Tr default field title
NC panel 160
NS panel 161
TR default field title
Lines panel 156
NODE panel 163
Printer panel 171
Punch panel 176
TRACE parameter 28
trademarks 595
TRAN-ACT default field title
Display Active Users panel 137
TRAN-RES default field title
Display Active Users panel 137
TRANS default field title
Printer panel 172
TRANS value 52
translation tables 101
TRANSMIT default field title
NODE panel 162
TRANSP default field title
Lines panel 156
TRC default field title
Printer panel 173
TRCLASS parameter 29
TRDEF statement
syntax 105
TRKCELL default field title
Printer panel 172
TRKPERCYL default field title
Spool Volumes panel 183
TRKPERTG default field title
Spool Volumes panel 183
TRTAB statement 101
syntax 102
using for defining code pages 101
using UPCTAB parameter with 101
using VALTAB parameter with 101
TSO authority, using to define a group of users 36
TSO logon procedure
adding DD statements to 27
TSO user group 312
TSOAUTH parameter 40
TSU value 52
TSUNUM default field title
SYS panel 186
Type default field title
NC panel 160
Spool Volumes panel 183

TYPE default field title
Display Active Users panel 136
Enclave panel 139
Held Output Queue panel 145
Initiator panel 146
Input Queue panel 147
LI panel 155
Output Queue panel 167
PAG panel 168
Printer panel 172
Process panel 174
Punch panel 175
Reader panel 177
Spool Offload panel 180
Status panel 188
System Symbol panel 185
TYPE parameter 86
TYPE=DEST parameter 90

U

UCLIN 351
UCLIN jobs 351
UCS default field title
Held Output Queue panel 144
Job 0 panel 154
Job Data Set panel 152
Output Queue panel 167
Printer panel 170
UCSV default field title
Printer panel 171
UIC default field title
SYS panel 186
UJP default field title
JC panel 150
ULOG panel
authority to display, specifying 305
extended console 305
CICS 306
NetView 306
UnAlloc default field title
Initiator panel 146
UNALLOC parameter 29
unallocating dynamically allocated data sets 29
Unit default field title
NC panel 160
UNIT default field title
Enqueue panel 141
LI panel 155
Printer panel 171
Punch panel 175
Reader panel 177
Spool Offload panel 182
UPCTAB parameter 71, 104
lowercase to uppercase, converting from 71
requesting a code page, using for 71
translate tables, defining 71
UPDREASON default field title
CK panel 143
uppercase, specifying 101
UseCount default field title
Initiator panel 146
USENUM default field title
PAG panel 168
RM panel 179
USEPCT default field title
PAG panel 168
RM panel 179
user authorization 475
user exit routine, SDSF 329
user groups, defining 36
user ID
using to define a group of users 36
using to restrict
authority to issue commands for jobs 47
authority to view SYSOUT and SYSIN for jobs 55
jobs displayed on SDSF panels 65, 66
user interface
ISPF 589
TSO/E 589
USERDATA default field title
Enqueue panel 141
USERDATA Default field title
Job Data Set panel 153, 165
USERDATE default field title
CK panel 143
USERID value 50, 56, 65, 66
USERLIB default field title
Job Data Set panel 153
Output Descriptors panel 165
UserPages default field title
MAS and JESPLEX panels 159
using SDSF
batch processing 369
USO default field title
JC panel 150

V

VALIDATE default field title
Spool Offload panel 182
VALTAB parameter 72, 104
checking for valid characters, using for 72
requesting a code page, using for 72
VALUE 49
VALUE default field title
System Symbol panel 185
variable field list
for SDSF panels, defining 85
VERBOSE default field title
CK panel 143
VERIFYP default field title
NODE panel 163
VERSION default field title
Multi-Access Spool panel 159
viewing page-mode output
authorization
initialization statements 50
ISFPARMS 50, 54, 72
SAF 245, 247, 312
specifying unit name of temporary file 72
VIO default field title
PAG panel 168
VIO parameter 72
VMUSER default field title
SYS panel 186
VOLS default field title
Spool Offload panel 182
VolSer default field title
Spool Volumes panel 183
VOLSER default field title
Authorized Program Facility panel 135
Link List panel 157
Link Pack Area panel 158
PAG panel 168
PARMLIB panel 169
SRCH panel 184
VS/JES commands
authorization for, specifying
with ISFPARMS parameters 47
with statements parameters 47
VTr default field title
NC panel 160
NS panel 161
VTR default field title
LI panel 157

W

WAIT-CNT default field title
JC panel 149
WAITERS default field title
Enqueue panel 141
WAITEXC default field title
Enqueue panel 141
WAITSHR default field title
Enqueue panel 141
WARNPCT default field title
RM panel 179
WebSphere MQ
cluster 33
configuration 344
considerations 344
help with 344
libraries 344
objects
defining manually 33
delete option 33
replace option 33
queue prefix 33
queues 116, 346
queues, protecting 306
remote queues 346
sample DEFINE commands 116
security 300
storage estimates 345
WebSphere MQ, APAR for large messages 344
WHEN statement 20
who can use SDSF, specifying 46, 330
WITH default field title
Input Queue panel 149
WLM resources 178
WLM scheduling environments 179
Work-Selection default field title
NC panel 161
WORK-SELECTION default field title
LI panel 155
Printer panel 170
Punch panel 175
Spool Offload panel 181

WORKLOAD default field title
Display Active Users panel 136
Enclave panel 139
WPOS default field title
Input Queue panel 147
Status panel 188
write-to-operator (WTO) messages, issuing for commands 47
write-to-operator-with-reply (WTOR) messages
displayed on SYSLOG panel
specifying in ISFPARMS 44
error messages 508
WRITER default field title
Printer panel 170
WTOINT default field title
MAS and JESPLEX panels 159
WTOLIM default field title
MAS and JESPLEX panels 159
WTONUM default field title
CK panel 142
WTORNUM default field title
SYS panel 186
WTORs, controlling 371
WTOTYPE default field title
CK panel 142
WTR default field title
Held Output Queue panel 144
Job 0 panel 154
Job Data Set panel 152
Output Queue panel 65, 167
WTRID default field title
Punch panel 176

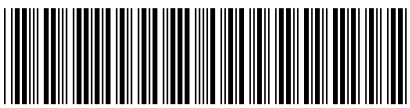
X

XBM default field title
JC panel 150
XBMPROC, default field title
Job Class panel 151
XCF application server
specifying 34
XCMD parameter 72
XDSP parameter 72
XDSPD parameter 55, 60, 73
XEQ-CNT default field title
JC panel 149
XEQDEST default field title
Reader panel 177
XLPROC parameter 40
XNAMEREQ default field title
NODE panel 164
XSTATUS parameter 73
XTNAME parameter 40
XUID parameter 40

Z

z/OS Version 2 Release 1 updates
browse 8
column changes 13
console name modification 12
CPU% normalization 13
long job class names 9
print 11
printing 11
scope of panels 13
SET DISPLAY 9
SET DUPDS 12
sort 8
symbols 9
z/64-bit addressing 8
z/OSMF 7
z/OSMF 352
adding the SDSF task 7, 353
properties file 7, 353
requirements 7, 352
security 354
settings 353
TSO logon proc 353
zAAP columns
custom property 97
zAAP use
custom property 97
zAAP-Time default field title
Display Active Users panel 137
ENC panel 139
zAAP-Use% default field title
Display Active Users panel 137
ZAAPNTIM default field title
Display Active Users panel 137
Enclaves panel 140
ZAAPNUM default field title
SYS panel 187
zACP-Time default field title
Display Active Users panel 137
ENC panel 139
zICP-Time default field title
ENC panel 140
zIIP-Time default field title
ENC panel 140
ZIIPCPTM default field title
Display Active Users panel 137
ZIIPNTIM default field title
Display Active Users panel 137
Enclaves panel 140
ZIIPNUM default field title
SYS panel 187
ZIIPTIME default field title
Display Active Users panel 137
ZIIPUSE default field title
Display Active Users panel 137

Product Number: 5650-ZOS

Printed in USA

SA23-2274-08