IBM IMS Tools Base for z/OS
1.7

IMS Hardware Data Compression Extended User's Guide and Reference

SC27-9858-01

Note:

Before using this information and the product it supports, read the information in "Notices" on page 77.

Second Edition (January 2024)

This edition applies to Version 1.7 of IBM IMS Tools Base for z/OS (program number 5655-V93) and to all subsequent releases and modifications until otherwise indicated in new editions.

This edition replaces SC27-9858-00.

© Copyright International Business Machines Corporation 2000, 2024.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

About this information v

Chapter 1. IMS HD Compression Extended overview 1
- What's new in IMS HD Compression Extended 1
- How IMS HD Compression Extended works 2
- IMS HD Compression Extended features and benefits 2
- Data compression and expansion 4
- IMS HD Compression Extended components 5
  - Dictionary types 5
  - IMS HD Compression Extended - functions and utilities 7
  - IMS HD Compression Extended programs 9
  - Report types produced 10
- IMS HD Compression Extended scenarios 10
- Service updates and support information 11
- Product documentation and updates 11
- Accessibility features 12

Chapter 2. Configuring IMS HD Compression Extended 13

Chapter 3. Getting started 15
- IMS HD Compression Extended process roadmap 15
- Starting the IMS HD Compression Extended ISPF interface 17
- The IMS HD Compression Extended main menu 17

Chapter 4. Extracting data 21
- Image Copy Extract utilities 21
  - Image Copy Extract panel 21
  - Image Copy Extract JCL requirements 22
- HD Unload Extract utility 22
  - Extracting specific segments by using HD Unload Extract 23
  - HD Unload Extract panel 24
  - HD Unload Extract JCL requirements 24

Chapter 5. Analyzing segment data 27
- Analyzing segment compression by using Detail Analyze 27
- Detail Analyze panel 29
- Detail Analyze JCL requirements 30
- Detail Segment Compression Analysis report (HCO006) 31
- Segment Distribution by Compressed Length report (HCO007) 33

Chapter 6. Analyzing database data 35
- Analyzing segment or database compression by using Analyze 35
- Analyze panel 36
- Analyze JCL requirements 38
- Compression Projection report (HCO001) 39

Chapter 7. Building and relinking a dictionary 41
- Building a custom dictionary by using Build 41
- Creating a compression exit routine 42

Modifying a DBD 43
Relinking a compression exit routine 44
Build panel 45
Compression Exit Routine panel 47
Edit DBD panel 47
COMPRTN keyword 48
Build JCL requirements 49
Compression exit routine and DBD JCL requirements 51
Compression Projection report (HCO002) 51

Chapter 8. Examining an existing dictionary 55
Examining a dictionary by using Examine 55
Examine panel 56
Examine JCL requirements 57
Compression Projection report (HCO003) 58
Compare Old/New Compression Projection report (HCO004) 60

Chapter 9. Compressing your data 63
Compressing your data by using Compress 63
Compress panel 64
Compress JCL requirements 65

Chapter 10. Troubleshooting 67
Runtime error messages (HCO) 67
Abend codes 70
Abend U3315 70
Abend U3325 71
Return codes 74
Gathering diagnostic information 75

Notices 77

Index 81

About this information

IBM® IMS Tools Base for z/OS® IMS Hardware Data Compression Extended (also referred to as IMS HD Compression Extended) is an IMS Tools product that provides a set of utilities designed to help implement and manage compressed IMS databases.

These topics provide instructions for installing, configuring, and using IMS Hardware Data Compression Extended.

To use these instructions, you must have already installed IMS Hardware Data Compression Extended by completing the instructions in the *Program Directory for IBM Tools Base for z/OS*, GI10-8819, which is included with the product media and is also available on the IMS Tools Product Documentation page.

These topics are designed to help database administrators, system programmers, application programmers, and system operators perform the following tasks:

- Understand the capabilities of the functions that are associated with IMS Hardware Data Compression Extended
- Install and operate IMS Hardware Data Compression Extended
- Customize your IMS Hardware Data Compression Extended environment
- Diagnose and recover from IMS Hardware Data Compression Extended problems
- Use IMS Hardware Data Compression Extended with other IMS products

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

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

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

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

The IMS Tools Product Documentation web page includes:

- Links to IBM Documentation for the user guides ("HTML")
- PDF versions of the user guides ("PDF")
- Program Directories for IMS Tools products
- Technical notes from IBM Software Support, referred to as "Tech notes"
- White papers that describe product business scenarios and solutions

© Copyright IBM Corp. 2000, 2024

Chapter 1. IMS HD Compression Extended overview

IBM IMS Tools Base for z/OS IMS Hardware Data Compression Extended (also referred to as IMS HD Compression Extended) provides functions for compressing IMS data. IMS HD Compression Extended uses the z/OS hardware data compression (HDC) that is available on IBM processors.

IMS provides a basic utility that supports z/OS HDC. IMS HD Compression Extended provides enhanced functions that supports z/OS HDC that are not available in the basic IMS utility. IMS HD Compression Extended can be used to compress HDAM, HIDAM, PHDAM, PHIDAM, HISAM, and Fast Path databases.

The advantages of compression include:

- Reduced DASD storage requirements for databases
- Reduced database I/O
- CPU usage is not significantly increased

Compressing data conserves DASD and tape resources. It also reduces high RBA usage, which is especially beneficial to IMS databases that are approaching the 4 GB limit for VSAM or 8 GB limit for OSAM.

Compression can improve database response time and performance because fewer physical I/Os are needed to access data.

Topics:

- “What's new in IMS HD Compression Extended” on page 1
- “How IMS HD Compression Extended works ” on page 2
- “IMS HD Compression Extended features and benefits” on page 2
- “Data compression and expansion ” on page 4
- “IMS HD Compression Extended components” on page 5
- “IMS HD Compression Extended scenarios” on page 10
- “Service updates and support information” on page 11
- “Product documentation and updates” on page 11
- “Accessibility features” on page 12

What's new in IMS HD Compression Extended

This topic summarizes the technical changes for this edition.

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

Revision markers follow these general conventions:

- Only technical changes are marked; style and grammatical changes are not marked.
- If part of an element, such as a paragraph, syntax diagram, list item, task step, or figure is changed, the entire element is marked with revision markers, even though only part of the element might have changed.
- If a topic is changed by more than 50%, the entire topic is marked with revision markers (so it might seem to be a new topic, even though it is not).

Revision markers do not necessarily indicate all the changes made to the information because deleted text and graphics cannot be marked with revision markers.

© Copyright IBM Corp. 2000, 2024

SC27-9858-01 - January 2024

Description	Related APARs
Updated descriptions of the CYLS parameter. The following topics have been updated: • “Detail Analyze panel ” on page 29 • “Detail Analyze JCL requirements ” on page 30	N/A

SC27-9858-00 - July 2022

Description	Related APARs
Refreshed for IMS Tools Base 1.7.	N/A

How IMS HD Compression Extended works

IMS HD Compression Extended provides a set of tools that are designed to help implement and manage compressed IMS databases.

IMS HD Compression Extended provides:

- An easy-to-use approach for creating and maintaining compressed IMS databases
- Features to evaluate compression impact, build, verify, and archive dictionaries, implement compression, and monitor and maintain production compression routines
- Ability to use generic, enterprise, or custom compression dictionaries
- An easy-to-use ISPF interface

Segment data is expanded and compressed by the compression exit routine whenever a segment is accessed.

The compression routine contains two dictionaries, one for compression and one for expansion (collectively referred to as the dictionary).

An advantage of using IMS HD Compression Extended is that it does not depend on IMS being up, does not require that you schedule around IMS, and does not interfere with IMS.

A dictionary can be built with representative data or a subset of the actual data. The data does not have to be current; the backed up data on an IMS image copy or HD unload data set can work as well as segment data that is extracted by one of your own applications.

IMS HD Compression Extended features and benefits

You can save DASD and I/O resources while minimizing CPU usage by taking advantage of IBM Hardware Compression services.

Compression can improve database response time and performance, because fewer physical I/Os are needed to access data.

Because IMS HD Compression Extended uses hardware-assisted compression, accessing compressed data from an application does not significantly increase CPU usage. CPU usage is greatest when compression takes place, for example during reload (writing data). CPU usage is negligible when unloading compressed segments (reading data). Any added CPU usage is based entirely on the amount of compression and expansion required when an application processes the database. A minimal increase in CPU usage when reading is typical, but is offset by the reduced physical I/O.

IMS HD Compression Extended:

- Analyzes what compression savings are possible before creating a dictionary

- Determines what compression savings the generic dictionary supplied with the product could provide
- Determines what compression savings an existing custom dictionary could provide if used on a database other than the one from which it was created
- Evaluates compression savings by segment and provides a distribution of compressed segment lengths (to help you prepare a DBD for reloading a database and minimizing VSAM control interval (CI) splits)
- Creates a dictionary, compression exit routine, and reload DBD in preparation for implementing data compression
- Provides ISPF panels for all functions

IMS HD Compression Extended provides a platform to evaluate, build, verify, archive, implement, and manage the performance of compressed databases. Compressing databases reduces storage and I/O requirements. Hardware compression helps minimize the CPU usage associated with compression. IMS HD Compression Extended is designed around the activities needed to develop compression dictionaries, implement compression, and monitor compression performance. It provides an easy-to-use front end to interactively operate these activities. IMS HD Compression Extended:

- Provides an Analyze function that simulates a compression dictionary and reports on its effectiveness. Multiple input data sources are supported for sampling and comparison. Results can be evaluated easily to select the most effective dictionary for production use.
- Provides a Build function to build, verify, and archive a compression dictionary. Linking the dictionary to the base exit routine creates the exit needed for implementation and updates the DBDGEN source.
- Provides a Compress function to implement the compression service. JCL is built to reload the database with the new or revised compression exits.
- Provides an Examine function to evaluate various combinations of dictionaries and databases, and monitors installed compression performance as databases change over time. It also can evaluate the effectiveness of the generic dictionary or an enterprise dictionary when used on a database other than the one from which the source data was obtained.
- Supports compression of:
  - IMS High Availability Databases (HALDB) – PHDAM and PHIDAM
  - Full-function databases - HDAM, HIDAM, HISAM
  - Fast Path databases - DEDB
- Uses the z/OS hardware data compression (HDC) available on IBM processors and defaults to software implementation if hardware services are not available.
- Functions are ISPF menu-driven.

IMS HD Compression Extended and other IMS Tools

Many IMS tools provide database management features that are not available in IMS itself or provide enhancements to capabilities built into IMS.

IMS HD Compression Extended is only one of several IMS tools that provide enhancements to the process of database administration and change management.

Other IMS Tools products that can assist with database administration and change management include:

- IMS Database Solution Pack
- IMS Database Utility Solution
- IMS Fast Path Solution Pack
- IMS Recovery Solution Pack
- IMS Transaction Manager Solution Pack
- IMS Sequential Randomizer Generator

IMS HD Compression Extended is a vital part of managing your IMS database system.

Data compression and expansion

A general understanding of how data compression and expansion works helps you use IMS HD Compression Extended.

Hardware data compression uses the Ziv-Lempel compression technique, which uses a fixed number of bits to replace a variable number of bytes. This technique requires use of a dictionary. The dictionary does the translation between uncompressed and compressed data. Dictionaries compress and expand data as follows:

- Compression takes an input string of data and, using the compression dictionary, produces an output string of compression symbols.

  The symbols represent a fixed number of bits that replace the variable number of bytes that were input.
- Expansion takes an input string of compression symbols and, using the expansion dictionary, produces an output string of data.

A separate dictionary exists for compression symbols and expansion characters.

The dictionary-based compression exit routine created by IMS HD Compression Extended can compress the data of individual IMS segments or entire databases. The compression exit routine can compress segments and databases other than those from which the dictionary was originally created. This allows flexibility in the use of compression exit routines.

The amount of space saved by compression depends on how well the dictionary and the data that is being compressed correlate. Due to the finite size of the dictionary (64 KB), it is generally not possible for a single dictionary to hold all the entries necessary to produce maximum compression of all IMS databases. You can use, multiple dictionaries to compress and expand the different types of data in your IMS databases.

An IMS database must be unloaded and reloaded as part of the compression process. The data is expanded when unloaded and compressed by IMS at reload using the compression exit routine.

Compression and expansion are done by using the compression exit routine. The compression exit routine, which is specified in the DBD, is created by linking together a dictionary driver and a dictionary object data set. You can use the Build function to create the compression exit routine, including a dictionary driver.

IMS gives the compression exit routine control whenever a segment that requires compression or expansion is accessed. As shown in the following figure, the phrase `this is data` is compressed from 12 bytes down to 5 bytes.

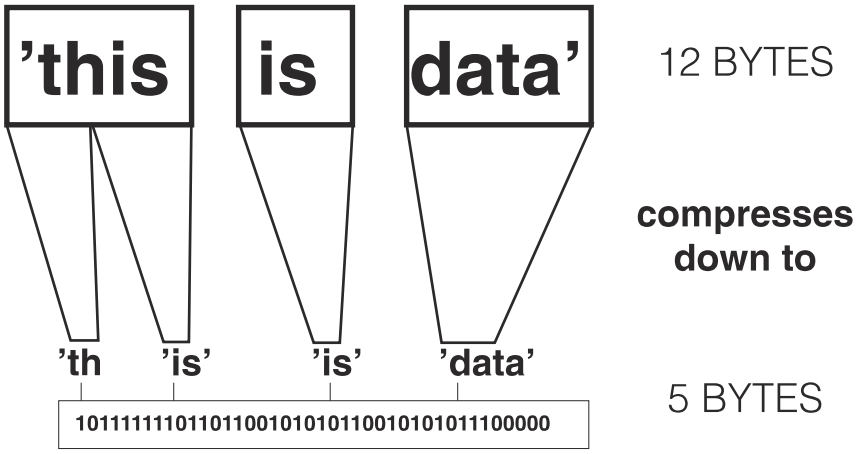

Figure 1. Compression example

When you build a custom dictionary, as each input string of data is read, a Ziv-Lempel tree begins to develop arranging parts of the data as dictionary entries. The total size of the dictionary is 64 KB because each entry is 16 bytes (8 for compression and 8 for expansion).

If sufficient data is supplied to build a dictionary, 4096 entries are included in the dictionary (2048 for compression and 2048 for expansion).

IMS HD Compression Extended components

IMS HD Compression Extended consists of several components.

The main components of IMS HD Compression Extended include:

Dictionary
The dictionary tells IMS how to compress and decompress data as it moves into and out of its location in storage. A good dictionary is an essential part of getting optimal compression on your systems.

An IMS HD Compression Extended dictionary exists as both an:

z/OS data set
The z/OS data set is used by the Examine function to report on the effectiveness of a dictionary.

Object data set
The object data set is used by IMS as part of its compression exit routine.

Functions and utilities
The functions and utilities that come with IMS HD Compression Extended are used to analyze your data and segments, build dictionaries, and compress your data.

Programs
The programs that come with IMS HD Compression Extended support the functions and utilities that are used to analyze your data and segments, build dictionaries, and compress your data.

Reports
The reports that are generated from IMS HD Compression Extended help you analyze what level of storage savings your current dictionary could achieve for you. They also project potential compression savings, and can be used to audit extract information.

Topics:

- “Dictionary types ” on page 5
- “ IMS HD Compression Extended - functions and utilities” on page 7
- “IMS HD Compression Extended programs ” on page 9
- “Report types produced ” on page 10

Dictionary types

Depending on your data and the amount of compression you need, you can select from four different dictionary types.

As shown in the following figure, IMS HD Compression Extended supports four types of dictionaries to help you find the best possible balance between DASD savings and dictionary management.

- Generic dictionary (provided with IMS HD Compression Extended)
- Enterprise dictionary
- Customized database dictionary
- Customized segment dictionary

The last three dictionaries are custom dictionaries that you create by using the Build function. The three dictionaries are not really different types of dictionaries, but rather different dictionary strategies. They are referred to as custom dictionaries.

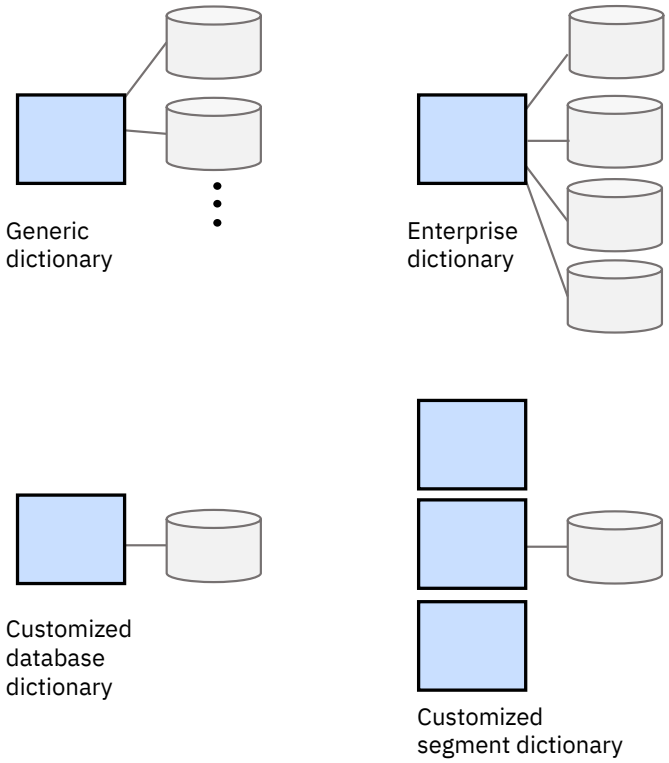

Figure 2. Different dictionary types

Generic dictionary

A generic dictionary is provided with IMS HD Compression Extended. The generic dictionary is developed from English-text data that contain frequently used character combinations. Typically the generic dictionary can help you achieve 40 - 60 percent compression.

The generic dictionary provides a single dictionary for compressing all of your data. The generic dictionary resides on the *hcohlq*.HCOGDIC1.HDCDDICT data set that you installed.

Use the Examine function to see how much space you could save by using the generic dictionary.

If using the generic dictionary produces less than a 40 percent savings in space or if the database does not contain primarily English text, consider using one of the other dictionary strategies.

Enterprise dictionary

An enterprise dictionary is created by you from your IMS data by using the Build function. Enterprise dictionaries typically define one dictionary per data type.

Selecting sample data for input to the Build function is a part of creating an enterprise dictionary. The sample data is the basis for the entries in the dictionary. The idea of an enterprise dictionary strategy is to create a customized dictionary for data in multiple databases, so you need to identify a representative set of data from the databases across your enterprise. For example, English-text data can be thought of as one group, alphanumeric data (such as names and addresses) as another, and numeric data a third. For each group of data, you can create an enterprise dictionary.

Before you build an enterprise dictionary, choose one database from a group, then use an appropriate program to create a variable-blocked QSAM data set for that group. Repeat this process for each group. You create a dictionary for a group by using one of these QSAM data sets as input to the Build function. Only one dictionary can be created at a time, so run the Build function for each QSAM data set to create a dictionary for each group.

If you want fewer enterprise dictionaries, you can create a single one. However, more effort is needed to create a single enterprise dictionary than multiple ones because a QSAM data set has to be built by using sample data from each database so that the data set holds a cross-section of your database data. The amount of data that is typically needed to build a dictionary is 300 - 400 KB.

Customized database dictionary

A customized database dictionary can provide better compression than the generic or enterprise dictionary, but might require you to build more dictionaries.

A customized database dictionary is intended to compress only one database. A customized dictionary is created from the data in the database it is to compress. The dictionary entries are, therefore, oriented to data that is unique to the database.

Even though the object of this strategy is to have a separate dictionary for each database, you can still use a customized database dictionary to compress more than one database. If data in one database is like data in another or another group of databases, the customized database dictionary can serve as an enterprise dictionary. The ability to share dictionaries eliminates the need to have a dictionary for each database, yet provides almost customized compression.

Customized segment dictionary

A customized segment dictionary compresses one segment, or a group of segments, from a database. A dictionary is built for each segment or group of segments.

The cumulative effect of compressing each segment by using its own dictionary can achieve maximum compression savings. Implementation of a customized segment dictionary strategy generally requires that more dictionaries be built. Because each database can contain up to 255 segments, you can potentially need 255 dictionaries for a single database.

The segments that benefit most from a customized segment dictionary are those segments that use the most space or are read the most often, especially by sequential batch jobs. Compressing these segments can save I/O as well.

To evaluate which segments are good candidates for compression, use the Detail Analyze function. The Detail Analyze function shows the potential savings for each segment in a database without requiring you to actually build a permanent dictionary. You can then determine which segments need their own dictionaries built.

IMS HD Compression Extended - functions and utilities

IMS HD Compression Extended includes several functions that help you evaluate, build, verify, archive, implement, and manage the performance of compressed databases.

Effective database compression is achieved through knowledge of the data, the databases, and the effect of your dictionary type. IMS HD Compression Extended provides the functions and reports for analyzing various compression scenarios, extracting the sample data that feeds this process, building the dictionaries, and building the compression user exit routine.

The functions help you answer these questions:

- What dictionary strategy should I employ?
- Will compression using the IBM-supplied generic dictionary be satisfactory?
- Should I build an enterprise-wide dictionary based on my own database data?
- Should I compress each database by using its own customized database dictionary?
- Will it be worth the effort to build a customized segment dictionary for each segment of each database?

The following table lists and provides a brief description for each of the IMS HD Compression Extended functions and utilities.

Table 1. IMS HD Compression Extended functions and utilities

Function	Description
Extract	To obtain the data that is used in the other IMS HD Compression Extended functions, use the Extract function. The Extract function consists of multiple utilities: **Image Copy Extract utilities** The Image Copy Extract utilities create a data set of segment records. This data set might be a selection of one or more segments or it might be all segments from a database. This image copy extract data set can be operated on by all the IMS HD Compression Extended functions: • in its entirety • one segment at a time **HD Unload Extract utility** This utility is used only by the Detail Analyze function. Any data set to be operated on by the Detail Analyze function must be in HD unload format. If you do not have an HD unload data set, this utility can convert an image copy extract data set into HD unload format so that it can be operated on by the Detail Analyze function. The HD Unload Extract utility splits the HD unload data set into multiple data sets. Each data set contains one segment. These data sets are processed all in one pass of the Detail Analyze function. Alternatively, each data set could be operated on one at a time by the Analyze, Build, and Examine functions.
Examine	The Examine function reports the estimated percent of savings for sample data by using an existing dictionary. The Examine function can help you: • Determine whether the generic dictionary provides adequate compression savings. The Examine function compresses segments or databases using the generic dictionary. • Determine the ongoing effectiveness of a custom dictionary, or determine if a custom dictionary that was built for one segment or database can be effectively used to compress another segment or database. The old estimates from when the dictionary was built and the new estimates from compressing a different set of data are reported.
Analyze	The Analyze function builds a temporary dictionary from your sample data and then rereads your data to compress it. The Analyze function produces a single report projecting your compression savings for the sample data based on a custom dictionary. The Analyze function is limited to treating one entire data set or one selected segment at a time.

Table 1. IMS HD Compression Extended functions and utilities (continued)

Function	Description
Detail Analyze	The Detail Analyze function builds a temporary dictionary from your sample data, rereads your data to compress it, and then produces two reports. The Detail Analyze function can analyze multiple segments in a single pass. It provides statistics for both generic and custom dictionaries. Reports from the Detail Analyze function show: • Compression savings for each segment • Distributing lengths of the data portion of the segments after compression Use the Detail Analyze function to determine maximum segment length before building the DBD and when minimizing VSAM control interval (CI) splits.
Build	The Build function creates a single dictionary for whichever segment or database is selected and reports the estimated savings on the compressed part of the segment data. It also creates the load module that becomes the compression exit routine for that DBD (and any other DBDs your strategy calls for). After you determine the segments or databases for which you want to use customized compression, you use the Build function. From IMS's perspective, the compression exit routine is an IMS Segment Edit/ Compression exit routine.
Compress	The Compress function creates reload JCL and guides you through the reload process. After you create a dictionary and an IMS compression exit routine by using the Build function, you are ready to compress your data. Compression is done when you reload your database.

z/OS and object data sets can also be used as input to the Analyze, Build, and Examine functions and can be processed as a whole.

IMS HD Compression Extended programs

IMS HD Compression Extended consists of several programs and two dictionary driver modules.

The following table lists the program names and brief descriptions of the functions, utilities, and drivers included in IMS HD Compression Extended.

Table 2. IMS HD Compression Extended programs

Program name	Function or utility	Description
HCOZLDH0	HD Unload Extract utility	Creates a separate data set for each segment that you select for extraction from an IMS HD unload data set or IMS image copy data set.
HCOZLDE0	Detail Analyze function	Produces two reports. • “Detail Segment Compression Analysis report (HCO006) ” on page 31 • “Segment Distribution by Compressed Length report (HCO007) ” on page 33
HCOZLDU0	Analyze, Build, and Examine functions	Creates a dictionary z/OS format data set and reports.

Table 2. IMS HD Compression Extended programs (continued)

Program name	Function or utility	Description
HCOZLDX0 and HCOZLDXV	dictionary drivers	HCOZLDX0 and HCOZLDXV must be linked to the dictionary object data set to create the IMS compression exit routine. • Use HCOZLDX0 for full-function databases. • Use HCOZLDXV for variable-length Fast Path DEDB segments when padding is needed to force segments to the minimum segment length after compression is done.

HCOZLDH0, HCOZLDE0, and HCOZLDU0 run as standard z/OS batch jobs. The dictionary driver runs when an IMS application requests access to a segment that references the compression exit routine.

Report types produced

IMS HD Compression Extended produces three types of reports.

The following table lists and describes the three types of reports produced by IMS HD Compression Extended. The Segment Compression Analysis and Compression Projection reports can help you determine the best methods and settings for compression of your data. The auditing reports can be used to track where data has come from.

Table 3. IMS HD Compression Extended report types

Report type	Description
Segment Compression Analysis	These reports show, on a per segment basis, kilobytes used before and after compression. Compression savings are shown in kilobytes and as a percent of the original space before compression. A distribution of segment lengths after compression is also provided.
Compression Projection	These reports show minimum, maximum, and average segment size before and after compression and the average amount of space saved per segment. The Build, Analyze, and Examine functions each produce a version of this report.
Auditing	Created by the Extract functions.

IMS HD Compression Extended scenarios

You can use IMS HD Compression Extended to address many of your day-to-day business problems.

The following scenario illustrates how you can use IMS HD Compression Extended to address typical business problems.

Saving storage space

Your business has grown rapidly, and your data volume has become difficult to manage. To maximize your company's data storage strategies and to avoid having to purchase more computers, you decide to use IMS HD Compression Extended.

By compressing data, you can free up over 50% of your current systems storage capacity.

Reaching the IMS storage limits

If your IMS databases are approaching the 4 GB VSAM or 8 GB limit for OSAM and you do not want to convert to HALDB, use IMS HD Compression Extended to achieve enough data storage savings to continue using your existing systems. By compressing your data instead of purchasing more hardware, you save money and keep maintenance requirements lower. Fewer databases to monitor and maintain decreases the burden on database administrators and IT staff.

Service updates and support information

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

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

IBM Support: IMS Tools Base for z/OS

Product documentation and updates

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

Information on the web

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

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

The IMS Tools Product Documentation web page includes:

- Links to IBM Documentation for the user guides ("HTML")
- PDF versions of the user guides ("PDF")
- Program Directories for IMS Tools products
- Technical notes from IBM Software Support, referred to as "Tech notes"
- White papers that describe product business scenarios and solutions

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

http://www.redbooks.ibm.com

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

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

Receiving documentation updates automatically

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

To register with the My Notifications service:

1. Go to https://www.ibm.com/support/mynotifications
2. Enter your IBM ID and password, or create one by clicking **register now**.
3. When the My Notifications page is displayed, click **Subscribe** to select those products that you want to receive information updates about. The IMS Tools option is located under **Software** > **Information Management**.

4. Click **Continue** to specify the types of updates that you want to receive.
5. Click **Submit** to save your profile.

How to send your comments

Your feedback is important in helping us provide the most accurate and highest quality information. If you have any comments about this or any other IMS Tools information, see How to provide feedback in IBM Documentation.

When you provide feedback, include as much information as you can about the content you are commenting on, where we can find it, and what your suggestions for improvement might be.

Accessibility features

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

The major accessibility features in this product enable users to perform the following activities:

- Use assistive technologies such as screen readers and screen magnifier software. Consult the assistive technology documentation for specific information when using it to access z/OS interfaces.
- Customize display attributes such as color, contrast, and font size.
- Operate specific or equivalent features by using only the keyboard. Refer to the following publications for information about accessing ISPF interfaces:
  - *z/OS ISPF User's Guide, Volume 1*
  - *z/OS TSO/E Primer*
  - *z/OS TSO/E User's Guide*

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

Chapter 2. Configuring IMS HD Compression Extended

Information about configuring IMS HD Compression Extended and other IMS Tools Base components is provided in the *IMS Tools Base for z/OS Configuration Guide*.

For information about migrating an existing installation of IMS HD Compression Extended after upgrading to a new release of IMS Tools Base for z/OS, see “Relinking a compression exit routine ” on page 44.

© Copyright IBM Corp. 2000, 2024

Chapter 3. Getting started

To start using IMS HD Compression Extended, you need to learn how to start the product, how to use its ISPF interface, and how to complete the processes that are associated with the five phases of compressing data by using IMS HD Compression Extended.

Topics:

- "IMS HD Compression Extended process roadmap" on page 15
- "Starting the IMS HD Compression Extended ISPF interface" on page 17
- "The IMS HD Compression Extended main menu " on page 17

IMS HD Compression Extended process roadmap

You need to progress through five main phases when using IMS HD Compression Extended. The information in this topic provides a quick reference to those phases, the processes you need to run, and links to the topics that explain how to use the function or utility to complete the task.

The following diagram illustrates the online reorganization process flow for an IMS HD Compression Extended environment.

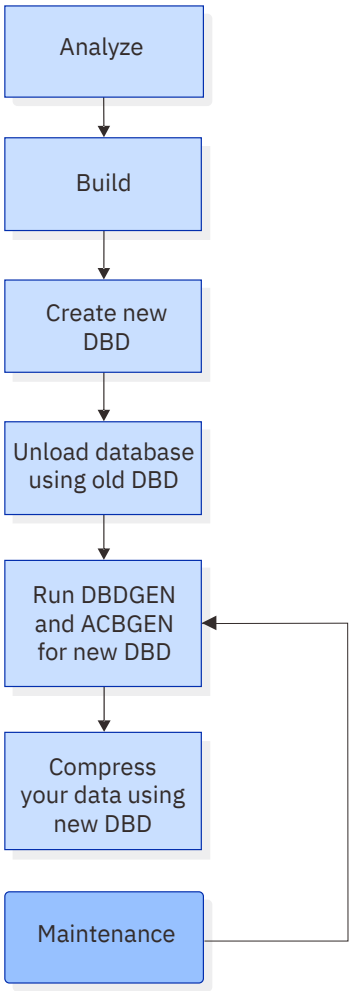

Figure 3. IMS HD Compression Extended process flow

© Copyright IBM Corp. 2000, 2024

Table 4. IMS HD Compression Extended phases

Phase	Instructions	Link to topic
Decide on a dictionary type	Decide on the type of dictionary that you want to use. If you are not sure, you can use the generic dictionary to test the effectiveness of data compression.	“Dictionary types ” on page 5
1. Analyze your data	Analyze your data to determine which segments to compress. If most of your data uses plain English text and 50% compression is acceptable, use the generic dictionary.	Use the Examine function to access either: • “Image Copy Extract utilities ” on page 21 • “HD Unload Extract utility ” on page 22
	If you want a high percentage of compression or have non-English text in the database, use a custom dictionary.	Either: • “Analyzing segment compression by using Detail Analyze ” on page 27 • “Analyzing segment or database compression by using Analyze ” on page 35
2. Build your dictionary	If you are using a custom dictionary strategy, then create your custom dictionary using the Build function.	“Building a custom dictionary by using Build ” on page 41
	Create the compression exit routine.	“Creating a compression exit routine ” on page 42
	Create new DBD. This compression exit routine is invoked just like any other IMS Segment Edit/Compression exit routine. For example, review the invocation of DFSCMPX0, which is described in *IMS Exit Routines*. The exit routine name is specified in the DBD SEGM statement COMPRTN parameter.	“Modifying a DBD ” on page 43
3. Unload the database	Unload the database using an appropriate IMS tool and the old DBD. An IMS database must be unloaded and reloaded as part of the compression process. The data is expanded when it is unloaded and compressed by IMS at reload using the compression exit routine.	Use IMS tools provided by IBM or other IMS tools provider.
4. Run DBDGEN and ACBGEN for new DBD	Run DBDGEN and ACBGEN for new DBD.	Use IMS tools provided by IBM or other IMS tools provider.

Table 4. IMS HD Compression Extended phases (continued)		
Phase	**Instructions**	**Link to topic**
5. Compress your data	Run the Compress function to compress and reload the database. Reload the database by using the new DBD. Data is compressed during the reload process. Later, segment data is expanded and compressed by the compression exit routine whenever a segment is accessed.	“Compressing your data by using Compress ” on page 63
Maintain active database compression	As a maintenance step, after applying a product update, relink the compression exit routine.	“Relinking a compression exit routine ” on page 44

Starting the IMS HD Compression Extended ISPF interface

You start the IMS HD Compression Extended ISPF interface by issuing a TSO command. The command that you enter depends on whether the HCOC00 REXX procedure is installed in your executable CLIST library.

About this task

Use the appropriate command to start IMS HD Compression Extended.

Procedure

1. Log on to TSO and specify a minimum REGION of (6144).
2. Select ISPF option 6, and issue the following command on the TSO command line.
   - If the REXX procedure is installed in your executable CLIST library, enter:

     ```
     HCOC00 'hlq'
     ```

   - If REXX procedure is not installed in your executable CLIST library, enter:

     ```
     EX 'hlq.SHCOCEXE(HCOC00)' 'hlq'
     ```

   Where *hlq* is the data set high-level qualifier to install IMS HD Compression Extended.

   When IMS HD Compression Extended starts, the main menu opens after the copyright panel.

The IMS HD Compression Extended main menu

You can access all basic functions of the product from the IMS HD Compression Extended main menu panel except Extract.

The following figure shows the IMS HD Compression Extended main menu panel. The Extract function is a subfunction and is accessed from the Analyze, Build, Detail Analyze, and Examine functions.

```
HCOP00 ---------------- IMS HD Compression Extended -------------------

OPTION  ===>

   COMPRESSION FUNCTIONS

   1  Detail Analyze - Analyze segment data

   2  Analyze        - Analyze database data

   3  Build          - Build a compression exit

   4  Examine        - Examine an existing dictionary

   5  Compress       - Compress database by executing Reload

Press END to exit.
```

Figure 4. IMS HD Compression Extended main menu panel

The options on this panel are:

Detail Analyze
Select this function to generate compression statistics at the segment level for each segment that you specify. This process involves:

- Extracting the segments to analyze
- Generating JCL for the Detail Analyze function
- Editing the completed JCL and then submitting a job

Analyze
Select this function to generate compression projections at the segment database level. This process involves:

- Extracting the segments to analyze
- Generating JCL for the Analyze function
- Editing the completed JCL and then submitting the job

Build
Select this function to build a dictionary, an IMS compression exit routine, and the appropriate DBD. This process involves:

- Extracting the segments to build a dictionary
- Generating JCL for the Build function
- Generating dictionary link-edit JCL
- Editing the DBD source
- Generating DBDGEN JCL
- Editing the completed JCL and then submitting the job

Examine
Select this function to examine an existing or generic dictionary's effectiveness with current data or data from a different database. This process involves:

- Extracting the segments to examine
- Generating JCL for the Examine function
- Editing the completed JCL and then submitting the job

Compress
Select this function to reload a database using the new DBD (data is compressed during the reload). This process involves:

- Generating database reload JCL

- Editing the completed JCL and then submitting the job

Chapter 4. Extracting data

To generate compression statistics or build a dictionary, you must extract segment data into a variable-blocked QSAM data set. The Extract function is accessed from the Detail Analyze, Analyze, Build, and Examine functions, and it is the first step in the compression process.

IMS HD Compression Extended provides utilities to help you extract data:

Image Copy Extract utilities
: Extract specified segments from an IMS image copy data set.

HD Unload Extract utility
: Extracts specified segments from an IMS HD unload data set or, in some cases, from the Image Copy Extract utilities output.

  To separate the segments that you have extracted into individual data sets, use the HD Unload Extract utility.

The output from these utilities is one or more QSAM data sets that contain extracted segments that can be used for creating a dictionary or analyzing compression.

Topics:

- "Image Copy Extract utilities " on page 21
  - "Image Copy Extract panel " on page 21
  - "Image Copy Extract JCL requirements " on page 22
- "HD Unload Extract utility " on page 22
  - "Extracting specific segments by using HD Unload Extract " on page 23
  - "HD Unload Extract panel " on page 24
  - "HD Unload Extract JCL requirements " on page 24

Image Copy Extract utilities

The Image Copy Extract utilities extract segment data. To access the Image Copy Extract utilities, you can use several IMS HD Compression Extended functions.

You can access the Image Copy Extract utilities in multiple ways:

- From the ISPF panel-driven Detail Analyze, Analyze, and Examine functions
- Directly from JCL that is run under the IMS batch region controller

Image Copy Extract panel

The Image Copy Extract panel opens when you specify IMAGE COPY EXTRACT=Y on the Detail Analyze, Analyze, or Examine panels. The ISPF panel-driven functions can create JCL to process the extract data set as an input data set to the Detail Analyze, Analyze, and Examine functions.

The following figure shows a sample of the Image Copy Extract panel.

© Copyright IBM Corp. 2000, 2024

```
HCOP12  IBM IMS HD Compression -------- Function = DETAIL    ------------------


COMMAND ===>

ADDITIONAL DATA REQUIRED FOR IMAGE COPY EXTRACT

  DBD NAME                ==>  SMPORDP1
  DATABASE DD NAME        ==>  SMPORD1P
  IMS RESLIB              ==>  IMS.RESLIB
  DBDLIB                  ==>  IMS.DBDLIB

  EXTRACTED OUTPUT FILE PARAMETERS: VOLSER    ==>  USER01 UNIT     ==>  SYSDA
                                    PRIM CYLS ==>  4      SEC CYLS ==>  4
                                      BLKSIZE ==>  6400   (Press PF1 for help)

Press END to exit.
```

Figure 5. Image Copy Extract panel

The Image Copy Extract panel contains the following fields:

DBD NAME
Specify the name of the database DBD. This information is added to the PROCCTL DD control statement.

DATABASE DD NAME
Specify the database DD name, as specified in the DBD. If an overflow data set exists, you must add the overflow data set name to the JCL and the DD name to the PROCCTL control statement parameters. The generated JCL is returned in ISPF edit mode for final changes after this panel is processed.

IMS RESLIB
Specify the name of the IMS RESLIB in which this utility resides.

DBDLIB
Specify the data set name of the library that contains the DBD that is used by this utility.

Image Copy Extract JCL requirements

The JCL for the Image Copy Extract utilities can be created from your ISPF panel input. You can also create your own JCL.

For examples that can help you create your own JCL for the Image Copy Extract utilities, refer to the members in the *hlq*.SHCOSAMP library file. (Substitute the *hlq* variable with the installation data set high-level qualifier.)

Instructions for customizing the JCL are included in each member. The following table lists the sample JCL member for each function that can be used to access the Image Copy Extract utilities.

Table 5. JCL members for accessing the Image Copy Extract utilities

Function	Sample JCL member
Detail Analyze	HCOJOB1A
Analyze	HCOJOB2A
Build	HCOJOB3A
Examine	HCOJOB4A

HD Unload Extract utility

The HD Unload Extract utility prepares segment data for a customized segment dictionary.

The HD Unload Extract utility produces a separate QSAM data set for each segment you extract. You can extract segments from either of the following sources:

- An IMS HD unload data set
- The output data set from the Image Copy Extract, which is useful if you want to create a customized segment dictionary but do not have an IMS HD unload data set

You enter the names of the segments to be extracted as control statements in a SYSIN DD data set. You can specify up to 255 different segment names for each database unload.

Extracting specific segments by using HD Unload Extract

The HD Unload Extract utility extracts specified segments from an IMS HD unload data set (or, in some cases, from the Image Copy Extract utilities output). You run the HD Unload Extract utility by using the HCOZLDH0 program. HCOZLDH0 runs as a standard z/OS batch job.

Before you begin

DEDB unload data sets are not supported by this utility.

You can use either of the following methods to run the HD Unload Extract utility:

- From the ISPF panel-driven functions Detail Analyze and Examine
- Directly from JCL as a standard z/OS batch job

About this task

The following procedure describes how to run the utility from the ISPF panels.

Procedure

1. Allocate a PDS for the SYSIN data set to hold the extract control statements for each segment.
2. Determine the names of the segments that you want to extract.
3. From the ISPF panel of the Examine function, enter 2 in the **INPUT SOURCE** field to select the HD Unload Extract utility. If you are using the Detail Analyze function, the HD Unload Extract utility is selected automatically.
4. Optional: Evaluate potential compression savings by using the following functions:

   Detail Analyze function
   Build a separate data set for each segment that you want to evaluate.

   The Detail Analyze function shows you the compression savings each segment received from its corresponding customized segment dictionary.

   Examine function
   Build a separate data set for a segment to evaluate the compression savings that another existing dictionary can offer. The existing dictionary is compared to the customized segment dictionary for the segment.
5. Optional: If you want compression statistics for a complete database, process all the volumes of an IMS HD unload. If you do not need statistics for a complete database, use only one volume (or subset of one volume) to build a dictionary or to analyze the degree of compression that each segment receives.
6. Generate the JCL by exiting the ISPF panel.

   IMS HD Compression Extended can create all JCL that is required to run dictionary functions and the HD Unload Extract utility. You can use the ISPF dialogs for the Detail Analyze function and the Examine function to create JCL to process the SGOUT*nnn* DD extract data sets.
7. Edit the JCL.

   Create or change the SYSIN DD entries for the segments to be extracted. If you are using the ISPF panels, the SYSIN DD is opened in edit mode. Otherwise, you must add the necessary SYSIN DD entries to the JCL.
8. Interpret the output report to verify that the process completed successfully.

Results

When you run the HD Unload Extract utility from ISPF panels, each segment that is selected for extraction from the HD unload data set has its own output SGOUT*nnn* DD statement. The *nnn* is the sequence number of the segment name parameter in the SYSIN DD control statements.

The ISPF-built JCL sets up temporary data sets for these segments. You might want to create permanent data sets to save reprocessing time when you build the dictionary.

The report from the utility is sent to the HCO005 data set. The report ID is also HCO005.

HD Unload Extract panel

The HD Unload Extract panel opens when you use the Detail Analyze function. It also opens after you specify INPUT SOURCE=2 on the Examine panel.

The following figure shows a sample of the HD Unload Extract panel.

```
----------- IMS HD Compression Extended (Function = DETAIL    ) --------


COMMAND ===>


    DATASET NAAME OF PDS FILE FOR SYSIN CONTROL CARDS
   ==>

    MEMBER NAME            ==>









Press END to exit.
```

Figure 6. HD Unload Extract panel

The HD Unload Extract panel contains the following fields:

DATASET NAME OF PDS FILE FOR SYSIN CONTROL CARDS
Specify the name of the PDS that was allocated for a SYSIN data set.

MEMBER NAME
Specify the name of the member in the SYSIN data set that contains the control statements for the segments to be extracted. The member can be new or existing. After the member name is entered, the member is opened in ISPF edit so that you can create or change the control statements.

For an explanation of the control statements and format for the SYSIN data set, see “HD Unload Extract JCL requirements ” on page 24.

HD Unload Extract JCL requirements

The JCL for the HD Unload Extract utility is created for you from ISPF panel input. You can also create your own JCL.

The following example shows the JCL for the HD Unload Extract utility. You can use this sample to help you develop your own JCL for the HD Unload Extract utility.

```
//*********************************************************************
//* HD UNLOAD EXTRACT
//*********************************************************************
//EXTRACT  EXEC PGM=HCOZLDH0,PARM='dbdname',REGION=20M
//STEPLIB  DD DSN=hcohlq.SHCOLMOD,DISP=SHR,PARM='ADBDNAME'
//HCO005   DD SYSOUT=*,DCB=RECFM=FBA
//HDCDHIN  DD DSN=hlq.dbname.UNLOAD,DISP=OLD
//*-------------------------------------------------------------------
//* THIS EXAMPLE SHOWS segname1 AND segname2 BEING EXTRACTED FROM THE HD
```

```
//* UNLOAD FILE & WRITTEN OUT TO DDS SGOUT001 AND SGOUT002 RESPECTIVELY.
//*-------------------------------------------------------------------
//SGOUT001 DD DSN=&&SEG001,UNIT=SYSDA,DISP=(NEW,PASS),
//            DCB=hlq.dbname.UNLOAD.DSN,
//            SPACE=(CYL,(xxx),RLSE)
//SGOUT002 DD DSN=&&SEG002,UNIT=SYSDA,DISP=(NEW,PASS),
//            DCB=hlq.dbname.UNLOAD.DSN,
//            SPACE=(CYL,(yyy),RLSE)
//*-------------------------------------------------------------------
//* The following additional DD statements are also required. ISPF file
//* tailoring (or manual edits) might be used to build appropriate SYSIN
//* control stmts, SGOUTnnn DD cards and space allocation JCL parms.
//* The temporary data sets are passed and concatenated as input to the
//* HCOZLDE0 reporting function.
//*-------------------------------------------------------------------
//SYSIN    DD *
segname1
segname2
/*
```

The HD Unload Extract utility JCL references the following data sets:

EXEC

This statement must adhere to the following format:

```
//         EXEC PGM=HCOZLDH0,PARM='dbdname',REGION=20M
```

For the PARM parameter, specify the name of the DBD for the database that is being extracted. This name is used for reporting purposes only. No IMS interface is invoked.

STEPLIB

The input PDS that contains the HCOZLDH0 load module.

HCO005

The output data set that contains the HD Unload Extract utility report.

HDCDHIN

The HD unload (or the equivalent) data sets that are input to the HD Unload Extract utility. Concatenate all data sets if you want segment statistics to be totaled for the database.

SGOUT*nnn*

The temporary output data set for the extracted data of a single segment. Each segment that is specified in the SYSIN DD statement has its own output SGOUT*nnn* DD statement. Each output data set and segment name are matched using *nnn*, which is the sequence number where the segment name is specified in the SYSIN DD statement.

The number of cylinders to be allocated (*xxx* and *yyy* in the example JCL) must also be specified in each SGOUT*nnn* DD statement.

You must change the DISP= parameter if you do not want the data set to be temporary.

SYSIN

The control data set that is used to specify control statements for each segment to be extracted. You must add a unique parameter that names each segment to be extracted.

```
//SYSIN    DD  *
SEGNAME1
SEGNAME2
```

Use columns 1 - 8 to specify the segment name from the DBD. This parameter is required.

Chapter 5. Analyzing segment data

The Detail Analyze function is a reporting tool that lets you evaluate hardware data compression without creating a compression exit routine or reloading the database.

The Detail Analyze function can analyze multiple segments in a single pass. The Detail Analyze function skips all HD unload segment prefix data when it analyzes a segment. It includes a KEYEND SYSIN option, which lets you skip the key field or other fields at the beginning of a segment. You can also skip the segment length (LL) field for variable-length segments. These options let you do a selective compression.

Acceptable input to the Detail Analyze function is an IMS image copy or HD unload (or the equivalent) data set. Regardless of the source of input, you must run the HD Unload Extract utility before you run the Detail Analyze function. The HD Unload Extract utility creates separate data sets for the segments that you have selected.

The Detail Analyze function builds a temporary dictionary from your sample data, rereads your data to compress it, and then produces the following two reports:

- Segment Compression Analysis Report
- Segment Distribution by Compressed Length Report

These reports provide information about generic and customized segment dictionary space savings for each segment that you select for analysis. These reports can help you prepare the reload DBD and specify optimal VSAM CI splits.

You run the Detail Analyze function by using the HCOZLDE0 program. The entire analysis runs as a single-step, standard z/OS batch job.

Topics:

- “Analyzing segment compression by using Detail Analyze ” on page 27
- “Detail Analyze panel ” on page 29
- “Detail Analyze JCL requirements ” on page 30
- “Detail Segment Compression Analysis report (HCO006) ” on page 31
- “Segment Distribution by Compressed Length report (HCO007) ” on page 33

Analyzing segment compression by using Detail Analyze

You can use the Detail Analyze function to analyze the segment compression that you can achieve by using the generic and customized dictionaries.

Before you begin

To use the Detail Analyze function, you must first run the HD Unload Extract utility create a separate data set for each segment to be analyzed.

Restrictions:

- DEDB unload data sets are not supported by this function.
- A maximum of 255 segments can be analyzed per database.
- A segment smaller than 4 bytes is not analyzed for compression. Any such data is reported without change in both the before and after statistics.

About this task

You can use either of the following methods to run the Detail Analyze function:

© Copyright IBM Corp. 2000, 2024

- From ISPF as a panel-driven function
- Directly from JCL as a standard z/OS batch job

When you analyze many segments, the ISPF panel-driven function can save you time by building the required HD unload extract JCL, creating control statements that can be shared between the Detail Analyze function and the HD Unload Extract utility, and building the JCL for Detail Analyze.

The following procedure describes how to run the Detail Analyze function from the ISPF interface.

Procedure

1. Select the Detail Analyze function on the IMS HD Compression Extended main menu panel.
2. Make sure that the HD Unload Extract utility (HCOZLDH0) has been run.

   You can enter N for **IMAGE COPY EXTRACT**, or you can run the HD Unload Extract JCL.
3. Provide an acceptable source of input data.

   Acceptable input to the Detail Analyze function is either an IMS image copy or an HD unload (or the equivalent) data set.

   If you do not have an input source already available, you can access both the Image Copy Extract and HD Unload Extract utilities.
4. Optional: If you are using an image copy data set:
   a) Enter Y (yes) in the **IMAGE COPY EXTRACT** field on the Detail Analyze panel.

      Alternately, you can make the JCL for the Image Copy Extract utilities the first step in the batch job that you create, so that the JCL precedes the HD Unload Extract utility step.
   b) Specify the names of the segments that you want to extract.

   You can specify these values on the Image Copy Extract utilities panel. The Image Copy Extract utilities panel opens after you finish entering all the required fields on the Detail Analyze panel.
5. Optional: If you are using an HD unload data set, complete the following steps:
   a) Allocate a PDS for the SYSIN data set if one does not exist.

      This value is validated by the panel, so it must exist before you can generate the JCL.
   b) Create or change the SYSIN DD entries for the segments to be extracted.

   You can specify these values on the HD Unload Extract utility panel. The HD Unload Extract utility panel opens after you finish entering all the required fields on the Detail Analyze panel.
6. Specify a data set name or accept the default data set name.

   If you want the Detail Analyze function to generate compression statistics for an entire database, specify a control statement in the SYSIN data set for every segment in the database.

   Any segments that are not included in the SYSIN data set are not analyzed. When you run the Detail Analyze function from an ISPF panel, the function shares control statements with the HD Unload Extract utility.
7. Generate the JCL by exiting the ISPF panel.

   When you run the Detail Analyze function from an ISPF panel, the data sets that are produced by the HD Unload Extract utility are concatenated into the HDCDEIN DD statement of the generated JCL. The SYSIN PDS member that you select is opened in ISPF edit for creation, review, or update.
8. Optional: Edit the JCL if required.
9. Submit the generated JCL.
10. Interpret the output report to verify that the process completed successfully.

Results

The reports that are created by the Detail Analyze function are sent to the HCO006 and HCO007 data sets. The report IDs are the same name as the data sets.

The Detail Analyze function produces the following two reports:

Segment Compression Analysis Report
Use this report to evaluate the segments that you selected to see how effectively each was compressed and how much space was saved as a result. Space savings are shown for each segment based on a generic and customized segment dictionary. This report indicates which segments save the most space when they are compressed.

You can substitute an enterprise or customized database dictionary in place of the generic dictionary. This change can be made when the JCL that was created by using the ISPF Detail Analyze panel is edited. The amount of compression that is produced by one of these other dictionaries typically is better than the compression that is produced by the generic dictionary, but not as good as the compression that is produced by customized segment dictionaries.

Segment Distribution by Compressed Length Report
For each segment that you select, a distribution of lengths for the segment data after compression is provided. Use this distribution to help you specify the minimum segment size in the new DBD that will be used to reload the database. Consider a size from the upper range of lengths if the segment is updated frequently and is subject to segment splits.

Detail Analyze panel

The Detail Analyze panel opens when you select the Detail Analyze function on the IMS HD Compression Extended main menu panel.

The following figure shows a sample of the Detail Analyze panel.

```
---------- IMS HD Compression Extended --- Function = DETAIL   --------

COMMAND ===>
    IMAGE COPY EXTRACT     ==>                Y OR N
    INPUT DSNAME           ==>
    USER DSN HLQ           ==>
    DATABASE IDENTIFIER    ==>




Press END to exit.
```

Figure 7. Detail Analyze panel

The Detail Analyze panel contains the following fields:

IMAGE COPY EXTRACT
Type Y if your input data set (specified in INPUT DSNAME) is an IMS image copy data set. Otherwise, specify N.

INPUT DSNAME
Specify the fully qualified name of the input data set. This data set must be an IMS image copy or HD unload (or the equivalent) data set. If you specify Y in the **IMAGE COPY EXTRACT** field, this data set must be an IMS image copy data set.

USER DSN HLQ
Specify the high-level qualifier for the output data sets.

DATABASE IDENTIFIER
Specify a name to identify the database that contains the segments that are being analyzed. For example, you can use the name of the DBD.

Based on the values that you provide, IMS HD Compression Extended returns a panel for extracting data, creating the SYSIN control data set, and creating JCL.

If the SYSIN statements are created by using the HDCE ISPF panels, specify the segment name in columns 1-8, the KEYEND parameter in columns 10-14, the SEGTYPE parameter in column 16, and the CYLS parameter in columns 18-21. The value of the CYLS parameter is used as the allocation size of the temporary data set when HDCE extracts the segment and the value is ignored by the Detail Analyze process. For more information about the segment name, the KEYEND parameter, and the SEGTYPE parameter, see "Detail Analyze JCL requirements " on page 30.

The following code shows an example of the SYSIN statements using the HDCE ISPF panels.

```
SEGNAME1 00000 F 0100
SEGNAME2 00000 V 0100
```

Detail Analyze JCL requirements

The JCL for the Detail Analyze function can be created from your ISPF panel input. You can also create JCL for the Detail Analyze function manually.

If SYSIN DD DUMMY is used, all input segments are analyzed and are assumed to be fixed length with a KEYEND value of 00000.

If the SYSIN statements are built by using the HDCE ISPF panels, the value of the CYLS parameter remains in columns 18-21. The value is ignored by the Detail Analyze process.

The following example shows the JCL for the Detail Analyze function. You can use this sample to help you develop your own JCL for the Detail Analyze function.

```
//*********************************************************************
//* DETAIL ANALYZE FUNCTION
//*********************************************************************
//STATS    EXEC PGM=HCOZLDE0,PARM='dbdname',REGION=8M
//STEPLIB  DD DSN=hcohlq.SHCOLMOD,DISP=SHR
//HCO006   DD SYSOUT=*,DCB=RECFM=FBA
//HCO007   DD SYSOUT=*,DCB=RECFM=FBA
//HDCDDICT DD DSN=hlq.HCOGDIC1.HDCDDICT,DISP=SHR
//HDCDWORK DD DSN=&&WORK01,UNIT=SYSDA,DISP=(NEW,PASS),
//            DCB=(LRECL=32756,BLKSIZE=32760,RECFM=VB),
//            SPACE=(CYL,(10,5),RLSE)
//HDCDEIN  DD DSN=&&SEG001,DISP=OLD
//         DD DSN=&&SEG002,DISP=OLD
//SYSIN    DD *
segname1 00000 f
segname2 00009 v
/*
```

The Detail Analyze JCL references the following data sets:

EXEC

This statement must adhere to the following format:

```
//         EXEC PGM=HCOZLDE0,PARM='dbdname',REGION=8M
```

For the PARM parameter, specify the name of the DBD for the segments that is being analyzed. This name is used for reporting purposes only. No IMS interface is invoked.

STEPLIB

The input PDS that contains load module HCOZLDE0.

HCO006

The output data set that contains the compression report for a generic or customized segment dictionary. If you substitute a different dictionary for the generic dictionary, the generic dictionary that is specified in the output is actually the substituted dictionary.

Each input segment is on a unique line.

HCO007
The output data set that contains the distribution report of compressed segment lengths.

Each input segment has a unique page.

HDCDDICT
The fully qualified name of the generic dictionary (*hlq*.HCOGDIC1.HDCDDICT) or the dictionary you substitute. The dictionary must have the z/OS data set format of 16 records of 4 KB each. Do not use the dictionary object data set.

HDCDWK01
The temporary output data set that is used for accumulating statistics.

HDCDEIN
The input DD of concatenated segment data sets from the HD Unload Extract utility.

SYSIN
The control data set that is used to specify control statements for each segment to be extracted. For each segment, you must specify unique parameters that identify the following attributes:

- The name of the segment to be extracted
- The end position of the segment key
- The segment type (fixed or variable)

columns 1 - 8
The segment name from the DBD. This parameter is required.

columns 10-14
The KEYEND parameter, which specifies the byte number in the segment where the key data ends. Five zeros (00000) means that key compression is wanted.

column 16
The SEGTYPE parameter, which specifies the segment type. You can specify either F (fixed) or V (variable). The 2-byte LL field for variable-length segments is skipped if V is specified.

The following code shows an example of how you specify SYSIN data set input.

```
//SYSIN    DD  *
SEGNAME1 00000 F
SEGNAME2 00000 V
```

Detail Segment Compression Analysis report (HCO006)

The Detail Segment Compression Analysis report (HCO006) shows the compression analysis report that is produced by the Detail Analyze function. IMS prefix length and access method usage are excluded from these statistics.

The following figure show a typical Detail Segment Compression Analysis report:

```
PAGE:    1                                IMS HD COMPRESSION EXTENDED (5655-V93)                          System Date: 2021.170
REPORT:  HCO006 V2.R2                        DETAIL SEGMENT COMPRESSION ANALYSIS                         System Time: 15.32.32

                                      ___________________________________________________
                                     |    DICTIONARY FUNCTION:  Detail Analyze           |
                                     |                 DBNAME:  DX41T001                 |
                                     |___________________________________________________|

----------------------|---------|-------------------------------------------------------------------------------------------------------
                      |         |                         DATA LENGTH AFTER KEYEND                                                    |
                      |         |-------------------------------------------------------------------------------------------------------
SEGMENT      KEY  SEG | DICT    |     BEFORE COMPRESS           AFTER COMPRESS                SAVINGS                 |        INPUT        BUILD
NAME         END  TYPE| TYPE    |  KBYTES USED  AVG LGTH      KBYTES USED  AVG LGTH          KBYTES   PCT |      RECORDS      RECORDS
----------------------|---------|-------------------------------------------------------------------------------------------------------
                      |         |                                                                                                       |
SEGMENTA       2   V  | Custom  |        3,291       247              162        12           3,129   95% |       13,625       11,602
                      | Generic|                                     524        39           2,767   84% |       13,625
                      |         |                                                                                                       |
SEGMENTC       0   V  | Custom  |       10,282       761              149        11          10,133   98% |       13,832       13,104
                      | Generic|                                     571        42           9,711   94% |       13,832
----------------------|---------|-------------------------------------------------------------------------------------------------------
                      |         |                                                                                                       |
TOTALS (KB)           | Custom  |       13,573                        311                     13,262   97% |
                      | Generic|                                   1,095                     12,478   91% |


                                                                END OF REPORT
```

Figure 8. Detail Segment Compression Analysis report

The Detail Segment Compression Analysis report contains the following information:

SEGMENT NAME
The name of the segment that was selected for analysis.

KEYEND
Byte number in the segment where the key data ends. Compression starts following the specified value. If you specify 00000, key data is included in the analysis.

SEGTYPE
The type of segment, which can be either variable (V) or fixed (F).

DICTTYPE
The type of dictionary that was used to compress the data for analysis.

DATA LENGTH AFTER KEYEND
Provides the following information:

KBYTES USED
Kilobytes of data that were read by the Detail Analyze function before and after the segment was compressed.

AVG LGTH
The average length of the segment data that was read by the Detail Analyze function before and after compression.

SAVINGS KBYTES
The number of kilobytes saved, which is determined as follows:

```
BEF COMPRESS KB USED - AFT COMPRESS KB USED
```

SAVINGS PCT
The percent of storage saved, which is determined as follows:

```
100 - ((AFT COMP KB USED / BEF COMP KB USED)*100)
```

INPUT RECORDS
The number of records that the Detail Analyze function analyzed for each segment.

BUILD RECORDS
The number of input records that the Detail Analyze function used to build the dictionary.

Segment Distribution by Compressed Length report (HCO007)

The Segment Distribution by Compressed Length report (HCO007) shows the compression analysis reports that are produced for each segment by the Detail Analyze function. A distribution report is created for each segment that is specified to the Detail Analyze function.

The report provides a count of the number of compressed segment lengths that fall into each segment range. This information can help you determine the maximum and minimum segment size in the new DBD. Consider a size from the upper range of lengths if the segment is updated frequently and is, therefore, subject to segment splits. For instructions on how to modify or create a DBD, see “Modifying a DBD ” on page 43.

The following figure shows the Segment Distribution by Compressed Length report that is produced by the Detail Analyze function. IMS prefix length and access-method usage are excluded from these statistics.

```
PAGE:    1                                   IMS HD COMPRESSION EXTENDED (5655-V93)                          System Date: 2021.170
REPORT:  HCO007 V2.R2                  SEGMENT SIZE DISTRIBUTION BY COMPRESSED DATA LENGTHS                  System Time: 15.32.32

                                     _____________________________________________________
                                    |    DICTIONARY FUNCTION:  Detail Analyze             |
                                    |                 DBNAME:  DX41T001                   |
                                    |           SEGMENT NAME:  SEGMENTC                   |
                                    |_____________________________________________________|

_____________________________________________________________________________________________________________________________
     LENGTH OF COMPRESSED          |   NUMBER AND PERCENT OF INPUT RECORDS WITHIN EACH LENGTH RANGE   |
     SEGMENT DATA AFTER KEYEND     |__________________________________________________________________|   INPUT RECORDS:        13,832
         (BYTE RANGES)             |         CUSTOM COMPRESSION              GENERIC COMPPRESSION     |
_____________________________________|__________________________________________________________________|_______________________________
                                   |                                                                  |
             1 - 15                |                0        0%                     0        0%      |
            16 - 23                |           11,077       80%                     0        0%      |
            24 - 31                |              199        1%                 6,153       44%      |
            32 - 39                |              602        4%                   504        3%      |
            40 - 47                |                1        0%                 1,106        7%      |
            48 - 55                |              400        2%                 1,672       12%      |
            56 - 63                |              249        1%                 2,556       18%      |
            64 - 71                |               84        0%                   188        1%      |
            72 - 79                |              810        5%                   282        2%      |
            80 - 87                |              294        2%                    73        0%      |
            88 - 95                |               38        0%                     4        0%      |
            96 - 103               |               78        0%                     0        0%      |
           104 - 127               |                0        0%                     0        0%      |
           128 - 135               |                0        0%                   298        2%      |
           136 - 143               |                0        0%                     0        0%      |
           144 - 151               |                0        0%                   199        1%      |
           152 - 207               |                0        0%                     0        0%      |
           208 - 215               |                0        0%                   399        2%      |
           216 - 271               |                0        0%                     0        0%      |
           272 - 279               |                0        0%                   199        1%      |
           280 - 327               |                0        0%                     0        0%      |
           328 - 335               |                0        0%                   199        1%      |
           336 - 32,767            |                0        0%                     0        0%      |
_____________________________________|__________________________________________________________________|_______________________________

                                                          END OF REPORT
```

Figure 9. Segment Size Distribution by Compressed Lengths report

The Segment Size Distribution by Compressed Lengths report contains the following information:

LENGTH OF COMPRESSED SEGMENT DATA AFTER KEYEND
The range of lengths across which compressed segment data is distributed.

CUSTOM COMPRESSION
The number of segments with lengths that were compressed to a size within that range. A percentage of the total number of segments is given for each range. The statistics are for a customized segment dictionary strategy.

GENERIC COMPRESSION
The number of segments with lengths that were compressed to a size within that range. A percentage of the total number of segments is given for each range. The statistics are for a generic dictionary strategy. If you substitute a different dictionary for the generic dictionary, the generic dictionary that is specified in the output is actually the substituted dictionary.

INPUT RECORDS
The number of records that the Detail Analyze function analyzed for the segment.

Chapter 6. Analyzing database data

The Analyze function is a reporting tool that lets you evaluate hardware data compression without creating a compression exit routine or reloading the database.

The Analyze function analyzes one entire data set or one selected segment at a time. The Analyze function builds a temporary dictionary from your sample data, rereads your data to compress it, and then produces a Compression Projection report.

The Compression Projection report details projected segment data compression. You can use this report to evaluate the database or segment space that would be saved if you created a dictionary to compress the data.

The Analyze function can evaluate enterprise or customized database dictionary solutions.

You run the Analyze function by using the HCOZLDU0 program. The entire analysis runs as a single-step, standard z/OS batch job.

Topics:

- "Analyzing segment or database compression by using Analyze " on page 35
- "Analyze panel " on page 36
- "Analyze JCL requirements " on page 38
- "Compression Projection report (HCO001) " on page 39

Analyzing segment or database compression by using Analyze

Use the Analyze function to produce a report to evaluate data space savings that would result if you created a dictionary to compress the input data. The ISPF panel functions help you create and submit JCL.

Before you begin

The Analyze function has a START BYTE option, which lets you adjust where compression begins in the input records. You can skip key data, unload-record prefix data, or specific fields in a segment. These choices let you do a selective compression.

START BYTE is different from the KEYEND SYSIN option in the Detail Analyze function. The Detail Analyze function input is always in HD unload format, whereas the Analyze function input can come from any variable-blocked QSAM input source. As a result, by using the Analyze function you can specify where compression starts in the input records.

The Analyze function supports DEDB unload data sets.

If your data is not an image copy data set, you can run the Analyze function without having to run the Extract function.

About this task

You can use either of the following methods to run the Analyze function:

- From ISPF as a panel-driven function
- Directly from JCL as a standard z/OS batch job

The following procedure describes how to run the utility from the ISPF panels.

Procedure

1. Specify the dictionary name.

© Copyright IBM Corp. 2000, 2024

2. Provide an acceptable source of input data.

   Acceptable input includes any of the following data set types:

   - DEDB unload data sets
   - Image copy extract data sets
   - HD unload extract data sets (or equivalent)

   If you do not have an input source already available, the function provides access to both the Image Copy Extract and HD Unload Extract utilities.
3. Run the Analyze function from ISPF as a panel-driven function

Option	Description
If you are using an image copy data set	From the Analyze function ISPF panel, enter Y (yes) in the **IMAGE COPY EXTRACT** field.
If you are using an HD unload data set	From the Analyze function ISPF panel, enter N (no) in the **IMAGE COPY EXTRACT** field.

4. Complete the other required fields on the ISPF panel.
5. Specify a data set name in the **INPUT HDCDIT DSNAME** field on the ISPF panel to generate statistics.

   This data set can be the same one that was specified in the **INPUT DSNAME** field.

   If a name is not specified, the INPUT DSNAME data set name is used.
6. Optional: If necessary, specify job card data on the Job Card Data panel.
7. Generate the JCL by exiting the ISPF panel.
8. Optional: Edit the JCL if required.
9. Submit the generated JCL.
10. Interpret the output report to verify that the process completed successfully.

Results

The report that is created by the Analyze function is sent to the SYSPRINT data set. The report ID is HCO001.

Output from the Analyze function is a report that shows projected segment data compression. The Analyze function produces a Compression Projection report. Use this report to evaluate (based on the input to the Analyze function) database or segment data space savings if a dictionary was created to compress the input data.

Analyze panel

The Analyze panel opens when you select the Analyze function on the IMS HD Compression Extended main menu panel.

The fields on this panel are used to open the analyze source for editing and to generate the analyze JCL. The following figure shows a sample of the Analyze panel.

```
----------- IMS HD Compression Extended (Function = ANALYZE   ) --------

COMMAND ===>

    DICTIONARY NAME        ==>

    IMAGE COPY EXTRACT     ==>               Y OR N

    INPUT DSNAME           ==>

    INPUT HDCDIT DSNAME    ==>

    USER DSN HLQ           ==>

    START BYTE             ==>

Press END to exit.
```

Figure 10. Analyze panel

The Analyze panel contains the following fields:

DICTIONARY NAME
Specify the dictionary name. You can specify up to eight characters for the name. This name is used for reporting purposes only; no permanent dictionary is created.

IMAGE COPY EXTRACT
Type Y if your input data set (specified in INPUT DSNAME) is an IMS image copy data set. Otherwise, specify N.

INPUT DSNAME
Specify the fully qualified name of the input data set. This data set must be an IMS image copy or HD unload (or the equivalent) data set. If you specify Y in the **IMAGE COPY EXTRACT** field, this data set must be an IMS image copy data set.

INPUT HDCDIT DSNAME
Specify the name of the data set that was used to generate compression statistics. If this data set is different from the one that was specified in INPUT DSNAME, statistics are generated for this (INPUT HDCDIT DSNAME) data set's space savings for the dictionary that was built from the INPUT DSNAME data set. For example, you might want to create a dictionary from one volume of an HD unload data set but use a different volume of the same data set to generate the statistics.

Specification of this field is optional. It defaults to the data set name that is specified in INPUT DSNAME.

USER DSN HLQ
Specify the high-level qualifier for the output data set.

START BYTE
Specify the relative byte number in the input source record at which compression analysis begins. Its point of reference is the beginning of the record, not the end of an unload utility record prefix (as is the case with the KEYEND parameter of the Detail Analyze function). The START BYTE option lets you adjust where compression analysis begins in the input records. The following figures show examples of the START BYTE option used for the HDCDIN and HDCDIT input data sets.

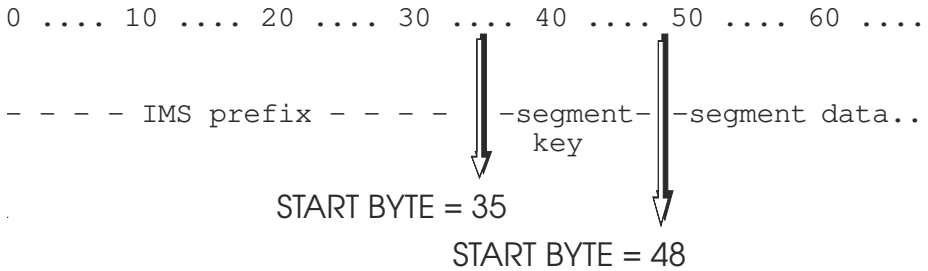

Figure 11. Example of the START BYTE option for an IMS Unload

You can specify the offset where the compression analysis begins for an IMS Unload or an Image Copy Extract record. As shown in Figure 11 on page 37, for example, if START BYTE = 35, the data that starts at position 36 is used for dictionary building and statistics generation. The IMS prefix data is ignored. If START BYTE = 48, the data that starts at position 49 is used for dictionary building and statistics generations. Both the IMS prefix data and the segment key are ignored.

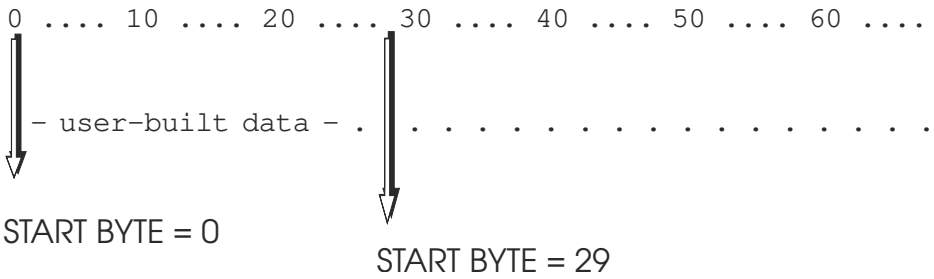

Figure 12. Example of the START BYTE option for a user-built record

You can specify the offset where the compression analysis for a user-built record begins. As shown in Figure 12 on page 38, for example, if you specify START BYTE = 0, the data that starts at position 1 is used for dictionary building and statistics generation. All the data in the record is used. If START BYTE = 29, the data that starts at position 30 is used for dictionary building and statistics generations. Unwanted user prefix data is ignored.

Analyze JCL requirements

You can create the JCL for the Analyze function by using ISPF panel input. You can also create your own JCL.

The following example shows the JCL for the Analyze function. You can use this sample to help you develop your own JCL for the Analyze function.

```
//*********************************************************************
//* ANALYZE FUNCTION
//*********************************************************************
//STATS    EXEC PGM=HCOZLDU0,PARM='dictnameAnnnn'
//STEPLIB  DD DISP=SHR,
//         DSN=hcohlq.SHCOLMOD
//SYSPRINT DD SYSOUT=*,DCB=RECFM=FBA
//SYSUDUMP DD SYSOUT=*
//HDCDIN   DD DISP=SHR,
//         DSN=hlq.input.data
//HDCDIT   DD DISP=SHR,
//         DSN=hlq.input.data
//HDCDOUT  DD DUMMY
```

The Analyze JCL references the following data sets:

EXEC
This statement must have the following format:

```
//         EXEC PGM=HCOZLDU0,PARM='dictnameAnnnnn'
```

where:

dictname
Specifies the 1- to 8-character name of the dictionary from long.

A
The letter A, as the ninth character in the parameter list, indicates that you are running the Analyze function. This specification can be omitted. It defaults to A.

nnnnn
Specifies the value of the START BYTE option. This 1- to 5-character specification begins at character 10 in the parameter list. This specification can be omitted. It defaults to 0. Figure 11 on page 37 shows examples of use of this parameter.

STEPLIB
The input PDS that contains load module HCOZLDU0.

HDCDIN
The input data set that is used to build the temporary dictionary.

HDCDIT
The name of the data set that is used to generate compression statistics. If this data set is different from the one that is specified in HDCDIN, statistics are generated for this (HDCDIT) data set's space savings for the dictionary that was built from the HDCDIN data set. For example, you might want to create a dictionary from one volume of an HD unload data set but use a different volume of the same data set to generate the statistics.

Specifying a data set name in this field is optional. It defaults to the data set name that is specified in INPUT DSNAME.

HDCDOUT
Must be specified as DD DUMMY. A permanent dictionary is not created by the Analyze function.

Compression Projection report (HCO001)

The Compression Projection report (HCO001) shows the compression projection report that is produced by the Analyze function. You can use this report to evaluate the compression you could achieve using a custom dictionary, without having to create a compression exit routine or reloading the database.

Before you run the Compress function, lengths might not match the DBD-specified lengths. If a segment with an odd length value is defined in the DBD, a hexadecimal zero pad byte is added at the end of each segment by the IMS Unload utility. If such records are used as input to HDCE, the lengths differ from the DBD-specified lengths.

The following figure shows the compression projection report produced by the Analyze function for a custom dictionary. IMS prefix length and access method usage can be excluded from these statistics.

```
PAGE:    1                                   IMS HD COMPRESSION EXTENDED (5655-V93)                          System Date: 2021.178
REPORT:  HCO001 V2.R2                                  COMPRESSION PROJECTION                                 System Time: 15.19.00

                                        _____________________________________________________
                                       |      DICTIONARY FUNCTION:  Analyze                  |
                                       |          DICTIONARY NAME:  TESTDIC3                 |
                                       |_____________________________________________________|

--------------------|---------|-------------------------------------------------------------------------------------------|---------------------------
                    |         |                          DATE LENGTH FROM START BYTE                                      |
                    |         |___________________________________________________________________________________________|
           START    | DICT    |       BEFORE COMPRESS                AFTER COMPRESS            SAVINGS   |         INPUT         BUILD
            BYTE    | TYPE    |     MIN      AVG      MAX         MIN      AVG      MAX         AVG   PCT |       RECORDS       RECORDS
--------------------|---------|-------------------------------------------------------------------------------------------|---------------------------
                9   | Custom  |     36       45       46          12       15       23          30   67% |            20            20
--------------------------------------------------------------------------------------------------------------------------------------------

                                 The projected savings do not include IMS or access method usage.
                                             (PCT SAVINGS = AVG SAVINGS X 100 / AVG BEFORE)


                                                                END OF REPORT
```

Figure 13. Compression Projection report

The Compression Projection report that is generated by the Analyze function contains the following information:

START BYTE
The value that is specified in the START BYTE parameter.

DICT TYPE
The type of dictionary, which is always custom.

DATA LENGTH FROM START BYTE
Shows the following columns:

BEFORE COMPRESS MIN
Smallest input segment size before compression.

BEFORE COMPRESS AVG
Average input segment size before compression.

BEFORE COMPRESS MAX
Largest input segment size before compression.

AFTER COMPRESS MIN
Smallest input segment size after compression.

AFTER COMPRESS AVG
Average input segment size after compression.

AFTER COMPRESS MAX
Largest input segment size after compression.

SAVINGS AVG
Average savings of bytes per segment.

Bytes that are skipped as a result of the START BYTE specification are excluded.

SAVINGS PCT
The percent of storage that is saved as determined by the following formula:

```
(PCT SAVINGS = AVG SAVINGS X 100 / AVG BEFORE)
```

Bytes that are skipped as a result of the START BYTE specification are excluded.

INPUT RECORDS
The number of records that the Analyze function read from the HDCDIT data set to generate these statistics.

BUILD RECORDS
The number of records that the Analyze function read to build the temporary dictionary (from the HDCDIN data set).

Chapter 7. Building and relinking a dictionary

Use the Build function to help you build a dictionary.

After you have decided what segments or databases that you want to compress, you use the Build function to build a dictionary and an IMS compression exit routine. The dictionary that the Build function creates can be an enterprise, customized segment, or customized database dictionary.

The Build function performs the following tasks:

- Creates a dictionary and the compression exit routine that uses it when HDC is invoked
- Evaluates the space savings that are reported by the dictionary build process so that you can check that expected compression conforms to your earlier expectations (based on earlier analysis)
- Generates DBDGEN JCL after you edit the old DBD or create a DBD

The Build function has a START BYTE option, which lets you adjust where compression begins in the input records. You can also do selective compression by skipping key data, unload-record prefix data, or specific fields in a segment.

The Build function produces a Compression Projection report. You can compare the space savings that are projected in this report to the savings that are projected from an earlier analysis.

You run the Build function by using the HCOZLDU0 program. The Build function runs as a single-step, standard z/OS batch job.

Topics:

- “Building a custom dictionary by using Build ” on page 41
- “Creating a compression exit routine ” on page 42
- “Modifying a DBD ” on page 43
- “Relinking a compression exit routine ” on page 44
- “Build panel ” on page 45
- “Compression Exit Routine panel ” on page 47
- “Edit DBD panel ” on page 47
- “COMPRTN keyword” on page 48
- “Build JCL requirements ” on page 49
- “Compression exit routine and DBD JCL requirements ” on page 51
- “Compression Projection report (HCO002) ” on page 51

Building a custom dictionary by using Build

Use the Build function to build your custom dictionary. Creating a custom dictionary is optional, but typically custom dictionaries provide the most optimal data compression.

Before you begin

The Build function can use any of the acceptable input sources, including DEDB unload data sets.

Restrictions:

- Only one dictionary can be created with each execution of the Build function.
- When you use a HIDAM or HISAM database HD Unload file to build a dictionary, the beginning records are not a representative sample of the database. You must use an IMS HD Compression Extended extract utility or your own application in these cases.

You can use either of the following methods to run the Build function:

© Copyright IBM Corp. 2000, 2024

- From ISPF as a panel-driven function
- Directly from JCL that is run as a standard z/OS batch job

The ISPF panel functions help you create and submit JCL and modify the DBD.

About this task

The following procedure describes how to run the utility from the ISPF panels.

Procedure

1. Provide a variable-blocked QSAM data set that contains uncompressed database segments.

   You can create the QSAM data set by using an appropriate program or one of the following utilities:

   - Image Copy Extract utilities
   - HD Unload Extract utility
2. Run the Build function from ISPF as a panel-driven function. Either complete the required fields on the Build panels, or create the JCL yourself.
3. Specify a data set name in the **INPUT HDCDIT DSNAME** field on the ISPF panel to generate statistics.

   This data set must be the same one that was specified in the **INPUT DSNAME** field.

   If a name is not specified, the INPUT DSNAME data set name is used.
4. Optional: For selective compression, adjust where compression begins in the input records by using the **START BYTE** option.

   You can skip key data, unload-record prefix data, or specific fields in a segment.
5. Optional: Edit the DBD if required.

   If you specify Y for the **EDIT DBD SOURCE** field, the Job Card panel, the Compression Exit panel, and the DBD panel are displayed.
6. Generate the JCL by exiting the ISPF panel.
7. Optional: Edit the JCL if required.
8. Submit the generated JCL.
9. Interpret the output report to verify that the process completed successfully.

Results

The report that is created by the Build function is sent to the SYSPRINT data set. The report ID is HCO001.

The Build function produces a Compression Projection report. Compare the space savings that are projected in this report with the space savings that were projected from an earlier analysis.

Creating a compression exit routine

An essential part of using IMS HD Compression Extended is the creation of the exit routine that is used to compress your data.

About this task

You can use either of the following methods to create a compression exit routine:

- From ISPF as a panel-driven function
- By editing JCL

You create an IMS compression exit routine by linking together the following items:

- IMS HD Compression Extended dictionary
- HCOZLDX0 or HCOZLDXV (compression drivers)

Procedure

1. Run the Build function from ISPF as a panel-driven function, and specify Y for the **EDIT DBD SOURCE** field.
2. On the Compression Exit Routine panel, specify the following values:
   - Compression dictionary name
   - Compression driver name
   - Compression exit name

   The dictionary must be the first CSECT in the compression exit routine, and it must be on a page boundary.
3. Exit ISPF or continue on to the DBD panel.

Modifying a DBD

The database definition must point to the correct compression exit routine to use.

About this task

IMS runs compression or expansion by calling the compression exit routine that is specified in the COMPRTN keyword of the DBDGEN SEGM statement. You must create or modify the existing DBD to specify the compression exit routine.

Procedure

1. Add or change the COMPRTN keyword of the SEGM statement for each segment that contains compressed data. The COMPRTN keyword must specify either KEY or DATA compression, and must specify INIT. For more information, see "COMPRTN keyword" on page 48.

   The following example shows how to enter the COMPRTN keyword of the SEGM statement for fixed-length segments:

   ```
   SEGM  NAME=SEGMENTA,PARENT=0,BYTES=256,PTR=TB,
         COMPRTN=(DRVRSCNM,DATA,INIT)
   ```

   The following example shows how to enter the COMPRTN keyword of the SEGM statement for variable-length segments:

   ```
   SEGM  NAME=SEGMENTA,PARENT=0,BYTES=(256,16),PTR=TB,
         COMPRTN=(DRVRSCNM,DATA,INIT)
   ```

2. Modify the COMPRTN or the BYTES parameter to allow for 10 byte control information.

   If the format or contents of the data in a segment precludes compression, the HDCE exit routine adds control information that indicates that the segment cannot be compressed.

   The control information might lengthen the segment beyond the maximum length definition. For example, an application on a test IMS system uses a production compression exit. If the test application tries to insert an X'00' string segment and no X'00' data was present when the production dictionary was built, that input string cannot be compressed.

   Failure to increase the MAXBYTES field by 10 might result in an ABENDU0853. Follow these instructions to allow for 10 byte control information:

Segment type	Action
Fixed-length segment	Omit the PAD parameter, and omit the *max* parameter or specify 10 (default) for the *max* parameter of the COMPRTN keyword of the SEGM statement. If you want to use the PAD parameter to control segment padding, see step "3" on page 44.

Segment type	Action
Variable-length segment	Increase the *max bytes* parameter of the BYTES keyword of the SEGM statement by 10 from the maximum length before compression.
DEDB	If your database is a DEDB, make sure that the new MAXBYTES field is at least 120 bytes less than the VSAM control interval (CI) size. If not, you must redefine your VSAM CI size.
Segments that cannot be compressed due to data inconsistency	If a segment cannot be compressed due to data inconsistency, IMS manages the increased length internally with no affect to the application or DBD. For more information, see the topic "Database Manager exit routines" in *IMS Exit Routines*.

3. Optional: If you want to control segment padding, adjust the parameters.

   If the compressed segment is frequently replaced by applications, consider adjusting the padding length. When a segment is replaced and the new data length after compression becomes longer, it could cause data fragmentation, which affects database performance. By specifying appropriate padding, IMS adds padding space to the value for the segments that are heavily compressed and reduces fragmentation.

Segment type	Action

Fixed-length segment	Specify the padding length with the *max* parameter and the PAD parameter of the COMPRTN keyword of the SEGM statement. The *max* parameter, when specified with the PAD parameter, is treated as the padding length. In this case, the default length, which is 10 bytes, is added for control information.
	`SEGM NAME=SEGMENTA,PARENT=0,BYTES=256,PTR=TB,`
	`  COMPRTN=(DRVRSCNM,DATA,INIT,150,PAD)`
	Padding length and the PAD parameter are not supported for DEDBs.

Variable-length segment	Specify the padding length with the *min bytes* parameter of the BYTES keyword of the SEGM statement.
	`SEGM NAME=SEGMENTA,PARENT=0,BYTES=(266,150),PTR=TB,`
	`  COMPRTN=(DRVRSCNM,DATA,INIT)`
	To determine the padding value for each segment, review the Segment Distribution by Compressed Length report (HCO007). For details, see “Segment Distribution by Compressed Length report (HCO007) ” on page 33.

4. Run an ACBGEN for online databases.

   Running ACBGEN is required whenever you modify a DBD.

Relinking a compression exit routine

You must relink all IMS compression exit routines if you upgrade IMS Tools Base for z/OS to a new release or apply maintenance to the IMS compression driver program (HCOZLDX0 or HCOZLDXV).

About this task

You can use generic, enterprise, and customized compression dictionaries that you used with a prior release. Segments that were compressed with a prior release can be expanded with this release without unloading and reloading databases, as long as the same dictionary is used.

You can use sample job HCORLNKD in the sample library to link the updated HCOZLDX0 program into your existing IMS compression exit routines.

Procedure

1. Put a copy of the old exit in a backup library (to be used if the new link edit fails).
2. Create a copy of the old exit on the test system.
3. Create a copy of the database on the test system.
4. Link the new exit into test user.RESLIB. Use the old exit (see DDNAME DD1 in Example) and the new exit driver (HCOZLDX0 or HCOZLDXV, see DDNAME DD2 in Example) as **SYSLIN** input.
5. Do one of the following steps:
   - Unload and reload the test database.
   - Create a DFSDDLT0 job that does **GET NEXT**s and updates through the test database to check expansion and compression.
6. After testing is successful:
   a) Create production database backups by using the image copy or unload utilities.
   b) Install the updated exits into the production IMS exit library using IMS exit update procedures.

Example

The following sample JCL shows how sample job HCORLNKD can be used to link HCOZLDX0 or HCOZLDXV to the IMS compression exit routines.

```
//LINK2  EXEC PGM=IEWL,
//       PARM='SIZE=(880K,64K),RENT,NCAL,LET,XREF,LIST=ALL'
//SYSPRINT DD SYSOUT=*
//DD1      DD DSN=hlq.user.RESLIB,DISP=SHR   old compression exit
//DD2      DD DSN=hcohlq.SHCOLMOD,DISP=SHR   new HCOZLDX0
//SYSLMOD  DD DSN=hlq.user.RESLIB,DISP=SHR   new compression exit
//SYSUT1   DD UNIT=SYSDA,DISP=(,DELETE),SPACE=(CYL,(10,1),RLSE)
//SYSLIN  DD *
  INCLUDE DD2(HCOZLDX0)        /* new module HCOZLDX0               */
  INCLUDE DD1(compexit)        /* old compression exit routine's name*/
  ORDER HCOZHDCD,HCOZLDX0      /* dictionary must be first          */
  PAGE     HCOZHDCD
  ENTRY HCOZLDX0
  NAME compexit(R)             /* new compression exit routine      */
```

Build panel

The Build panel opens when you select the Build function on the IMS HD Compression Extended main menu panel.

The fields on this panel are used to generate the Build JCL. The following figure shows a sample of the Build panel.

```
----------- IMS HD Compression Extended ---- Function = BUILD   --------

COMMAND ===>
    DICTIONARY NAME        ==>
    IMAGE COPY EXTRACT     ==>               Y OR N
    INPUT DSNAME           ==>
    INPUT HDCDIT DSNAME    ==>
    USER DSN HLQ           ==>
    USER DSNS VOLSER       ==>            UNIT RECORD DEVICE ==>
    EDIT DBD SOURCE        ==>               Y OR N
    START BYTE             ==>

Press END to exit.
```

Figure 14. Build panel

The Build panel contains the following fields:

DICTIONARY NAME
Specify the dictionary name. You can specify up to eight characters for the dictionary name. This name is used as the name for these output data sets:

- HDCDDICT
- HDCDSTAT
- HDCDOUT

IMAGE COPY EXTRACT
Type Y if your input data set (specified in INPUT DSNAME) is an IMS image copy data set. Otherwise, specify N.

INPUT DSNAME
Specify the fully qualified name of the input data set. This data must already be processed by an extract utility or be in variable-blocked QSAM format for input source data.

INPUT HDCDIT DSNAME
Specify the name of the data set that is used to generate compression statistics. If this data set is different from the one that is specified in INPUT DSNAME, statistics are generated for this (INPUT HDCDIT DSNAME) data set's space savings for the dictionary that was built from the INPUT DSNAME data set. For example, you might want to create a dictionary from one volume of an HD unload data set but use a different volume of the same data set to generate the statistics.

Specification of this field is optional. It defaults to the data set name that is specified in INPUT DSNAME.

USER DSN HLQ
Specify the high-level qualifier for the output data set.

USER DSNS VOLSER
Specify the volume ID, if needed, to which the HDCDDICT, HDCDSTAT, and HDCDOUT data sets are output.

UNIT RECORD DEVICE
Specify the device that is assigned to the output data sets.

EDIT DBD SOURCE
Specify whether you want to edit the DBD source to change the compression-sensitive SEGM statements. If you select Y, JCL for DBDGEN is built. If you select N, no action is taken.

START BYTE
Specify the relative byte number in the input source record at which compression analysis begins. Its point of reference is the beginning of the record, not the end of an unload utility record prefix, as is the case with the KEYEND parameter of the Detail Analyze function. START BYTE lets you adjust where compression analysis begins in the input records. Figure 11 on page 37 and Figure 12 on page 38 show examples of this parameter.

Compression Exit Routine panel

The Compression Exit Routine ISPF panel opens to prompt you for information to build the dictionary link-edit JCL and the compression exit routine.

The Compression Exit Routine panel is displayed as part of running the Build function through the ISPF interface. The following figure shows a sample of the Compression Exit Routine panel.

```
------- IBM IMS HD Compression -------- Function = BUILD     ------------------


COMMAND ===>


  COMP NAME      ==>  HDCDICT1   COMPRESSION DICTIONARY NAME.

  DRIVER NAME    ==>  HCOZLDXV   COMPRESSION DRIVER NAME. (HCOZLDX0/HCOZLDXV)

  EXIT NAME      ==>  COMPRTN1   COMPRESSION EXIT NAME.

  USER RESLIB WHERE COMPRESSION EXIT MUST BE PLACED
    ==>  IMSTOOLS.HSDCE.PGMLIB


Press END to exit.
```

Figure 15. Compression Exit Routine panel

The exit routine panel contains the following fields:

COMP NAME
Specify the name of the dictionary that is to be linked to the dictionary driver. Typically, this name is the same as the DICTIONARY NAME in Figure 14 on page 46.

DRIVER NAME
Specify the name of the compression dictionary driver.

EXIT NAME
Specify the name of the IMS compression exit routine.

USER RESLIB WHERE COMPRESSION EXIT MUST BE PLACED
Specify the name of the library that will contain the IMS compression exit routine.

Edit DBD panel

The Edit DBD ISPF panel opens when you select EDIT DBD SOURCE = Y on the Build panel. The fields on this panel are used to open the DBD source for editing and to generate the DBDGEN JCL.

The Edit DBD ISPF panel is displayed as part of specifying the information needed for the Build function. The following figure shows a sample of the Edit DBD panel.

```
----------- IMS HD Compression Extended ---- Function = BUILD   --------

COMMAND ===>

  DBD MACRO LIBRARY DATASET NAME
    ==>
  DBD SOURCE LIBRARY
    ==>
  DBD SOURCE LIBRARY MEMBER NAME
    ==>
  NEW DBD LOAD LIBRARY
    ==>
  NEW DBD LOAD LIBRARY MEMBER NAME
    ==>





Press END to exit.
```

Figure 16. Edit DBD panel

The edit DBD panel contains the following fields:

DBD MACRO LIBRARY DATASET NAME
Specify the data set that contains the library where DBD macro files are located.

DBD SOURCE LIBRARY
Specify the library where database definition DBD files are located.

DBD SOURCE LIBRARY MEMBER NAME
Specify the member name of the library where DBD files are located.

NEW DBD LOAD LIBRARY
Specify the name that you want to give to the DBD load library.

NEW DBD LOAD LIBRARY MEMBER NAME
Specify the name that you want to give to the DBD load library.

COMPRTN keyword

The COMPRTN keyword of the SEGM statement for each segment that contains data must be modified to allow for data compression.

Purpose

The COMPRTN keyword statement specifies the compression exit routine.

```
►► COMPRTN= ── (EXITNAME ──┬─ ,DATA ─┬──┬─ ,INIT ─┬──┬─ ,max ──┬─ ,PAD ─┬──┬── ) ──►◄
                           └─ ,KEY ──┘  └─────────┘  └─────────┘        └──┘
```

Parameters for variable-length and fixed-length segments

The following parameters apply to both variable-length and fixed-length segments:

EXITNAME
Specifies the name of the user-supplied compression exit routine. This name must be a 1- to 8-character alphanumeric value and must not be the same as any other name in `IMS.RESLIB`.

DATA
Specifies that the indicated exit routine compresses or modifies data fields only. Sequence fields must not be modified, nor data fields that change the position of the sequence field in respect to the start of

the segment. DATA is the default value if a compression routine is named but no operand is selected. When you use the DATA option, only data after the key field is compressed. If your key field is not first in your records, using this option might not result in the best possible compression.

KEY
Specifies that the exit routine can compress or modify any fields within the named segment. This parameter is invalid for the root segment of a HISAM database. Use this option to achieve maximum data compression.

INIT
Indicates that initialization and termination processing control are required by the segment exit routine. When this parameter is specified, the edit or compression routine gains control after a database opens and after a database closes.

Parameters for fixed-length segments

The following parameters apply to fixed-length segments:

max
When specified without the PAD keyword, the *max* value specifies the maximum number of bytes by which fixed-length segments can increase during compression exits. You can specify a value between 1 - 32,767 bytes. This keyword is optional. If omitted, the default value 10 is used.

When specified with the PAD keyword, the *max* value specifies the padding length. For more information, see the description of the PAD keyword.

PAD
Use the PAD keyword to indicate that the numeric value that is supplied in the previous parameter of *max* is used for padding. The numeric range from 1 to 32767 still applies; however, it is used in a different way. The value specifies a size to which an inserted segment is padded when the compression of that segment results in a length less than the PAD value. This keyword is optional.

Examples

The following example shows how to enter the COMPRTN keyword of the SEGM statement for fixed-length segments:

```
SEGM  NAME=SEGMENTA,PARENT=0,BYTES=256,PTR=TB,
      COMPRTN=(DRVRSCNM,DATA,INIT,150,PAD)
```

The following example shows how to enter the COMPRTN keyword of the SEGM statement for variable-length segments:

```
SEGM  NAME=SEGMENTA,PARENT=0,BYTES=(266,150),PTR=TB,
      COMPRTN=(DRVRSCNM,DATA,INIT)
```

Build JCL requirements

The JCL for the Build function can be created from your ISPF panel input. You can also create your own JCL.

The following example shows a portion of the Build JCL. You can use this sample to help you develop your own JCL for the Build function.

```
//*****************************************************************
//* BUILD FUNCTION
//*****************************************************************
//STATS    EXEC PGM=HCOZLDU0,PARM='dictnameBnnnn'
//STEPLIB  DD DISP=SHR,
//         DSN=hcohlq.SHCOLMOD
//SYSPRINT DD SYSOUT=*,DCB=RECFM=FBA
//SYSUDUMP DD SYSOUT=*
//HDCDIN   DD DISP=SHR,
//         DSN=hlq.input.data
//HDCDIT   DD DISP=SHR,
```

```
//         DSN=hlq.input.data
//HDCDOUT  DD DISP=(,CATLG,DELETE),
//         DSN=hlq.dictname.OBJECT,
//         SPACE=(TRK,(8,4)),
//         UNIT=uuuu,VOL=SER=ssssss,
//         DCB=(LRECL=80,BLKSIZE=3200,RECFM=FB)
//HDCDDICT DD DISP=(,CATLG,DELETE),
//         DSN=hlq.dictname.HDCDDICT,
//         SPACE=(TRK,(8,4)),
//         UNIT=uuuu,VOL=SER=ssssss,
//         DCB=(LRECL=4096,BLKSIZE=4096,RECFM=FB)
//HDCDSTAT DD DISP=(,CATLG,DELETE),
//         DSN=hlq.dictname.HDCDSTAT,
//         SPACE=(TRK,(1)),
//         UNIT=uuuu,VOL=SER=ssssss,
//         DCB=(LRECL=80,BLKSIZE=80,RECFM=FB)
```

The Build JCL references the following data sets:

EXEC
This statement must have the following format:

```
//         EXEC PGM=HCOZLDU0,PARM='dictnameBnnnnn'
```

where:

dictname
Specifies the 1- to 8-character name of the dictionary.

B
The letter B, as the ninth character in the parameter list, indicates that you are running the Build function.

nnnnn
Specifies the value of the START BYTE option. This 5-character specification begins at character 10 in the parameter list. This specification is optional. It defaults to 0. Figure 11 on page 37 shows examples of use of this parameter.

STEPLIB
The input PDS that contains load module HCOZLDU0.

HDCDIN
The input data set that is used to build a dictionary.

HDCDIT
The name of the data set that is used to generate compression statistics. If this data set is different from the one that is specified in HDCDIN, statistics are generated for the HDCDIT data set's space savings for the dictionary that was built from the HDCDIN data set. For example, you might want to create a dictionary from one volume of an HD unload data set but use a different volume of the same data set to generate the statistics.

HDCDOUT
The output data set that contains the generated dictionary in object format. This dictionary is linked with the dictionary driver when the compression exit routine is created.

HDCDDICT
The output data set that contains the generated dictionary in the z/OS format of 16 records of 4 KB each. This data set can be used later as input to the Examine function.

HDCDSTAT
The output data set that contains the compression statistics. This data set can be used later as input to the Examine function.

Compression exit routine and DBD JCL requirements

If you specified EDIT DBD SOURCE = Y on the Build panel, the Compression Exit Routine and DBD panels are displayed.

When you have made your changes, JCL for creating the compression exit routine and a new DBD are created for you.

The following example shows a portion the JCL that is used to specify the DBD for the Build function. You can use this sample to help you develop your own JCL for the Build function.

```
//*******************************************************************
//* LINK DICTIONARY INTO LOAD LIBRARY
//*******************************************************************
//LINK1    EXEC PGM=IEWL,
//        PARM='SIZE=(180K,28K),RENT,REFR,NCAL,LET,XREF,LIST=ALL'
//SYSLIN   DD DISP=SHR,
//         UNIT=uuuu,VOL=SER=ssssss,
//         DSN=hlq.dictname.OBJECT
//SYSLMOD  DD DISP=SHR,
//         DSN=hlq.user.RESLIB(dictname)
//SYSUT1   DD UNIT=SYSDA,DISP=(,DELETE),SPACE=(CYL,(1,1),RLSE)
//SYSPRINT DD SYSOUT=*
//*
//*-----------------------------------------------------------------
//* LINK DICTIONARY WITH HCOZLDX0 GENERIC DICTIONARY DRIVER
//*-----------------------------------------------------------------
//LINK2   EXEC PGM=IEWL,COND=(4,LT,LINK1),
//       PARM='SIZE=(880K,64K),RENT,REFR,NCAL,LET,XREF,LIST=ALL'
//SYSPRINT DD SYSOUT=*
//DD1      DD DISP=SHR,
//         DSN=hlq.user.RESLIB
//DD2      DD DISP=SHR,
//         DSN=hcohlq.SHCOLMOD
//SYSLMOD  DD DISP=SHR,
//         DSN=hlq.user.RESLIB
//SYSUT1   DD UNIT=SYSDA,DISP=(,DELETE),SPACE=(CYL,(10,1),RLSE)
//SYSLIN  DD *
  CHANGE   dictname(HCOZHDCD)    /* dictionary name must be first */
  INCLUDE DD1(dictname)          /* name of dictionary object */
  INCLUDE DD2(HCOZLDX0)          /* name of dictionary driver */
  PAGE     HCOZHDCD
  ENTRY HCOZLDX0
  NAME compexit(R)               /* name of compression exit routine */
/*
//*-----------------------------------------------------------------
//* ASSEMBLE DBD WITH CHANGED COMPRESSION EXIT ROUTINE
//*-----------------------------------------------------------------
//STEP1    EXEC PGM=ASMA90,REGION=0M,
//           PARM='OBJECT,NODECK,NOUSING'
//SYSLIB   DD DISP=SHR,
//         DSN=hlq.ims.MACLIB
//SYSLIN   DD UNIT=SYSDA,DISP=(,PASS),SPACE=(80,(100,100)),
//            DCB=BLKSIZE=400
//SYSPRINT DD SYSOUT=*
//SYSUT1   DD UNIT=SYSDA,DISP=(,DELETE),SPACE=(1700,(200,50))
//SYSIN    DD DISP=SHR,
//         DSN=hlq.lib.dbdscr(dbdname)
//*-----------------------------------------------------------------
//* LINK DBD INTO DBDLIB
//*-----------------------------------------------------------------
//STEP2    EXEC PGM=IEWL,PARM='XREF,LIST',
//           COND=(0,LT,STEP1),REGION=0M
//SYSLIN   DD DSN=*.STEP1.SYSLIN,DISP=(OLD,PASS)
//SYSPRINT DD SYSOUT=*
//SYSLMOD  DD DISP=SHR,
//         DSN=hlq.lib.dbdlib(dbdname)
//SYSUT1   DD UNIT=SYSDA,DISP=(,DELETE),SPACE=(1024,(100,10))
```

Compression Projection report (HCO002)

The Compression Projection report (HCO002) shows the compression analysis reports that project your data savings by using the compression dictionary that is specified for the Build function.

IMS prefix length and access method usage can be excluded from these statistics.

Before you run the Compress function, lengths might not match the DBD-specified lengths. If a segment with an odd length value is defined in the DBD, a hexadecimal zero pad byte is added at the end of each segment by the IMS Unload utility. If such records are used as input to HDCE, the lengths differ from the DBD-specified lengths.

The following figure shows the compression projection report produced by the Build function. IMS prefix length and access method usage can be excluded from these statistics.

```
PAGE:    1                                  IMS HD COMPRESSION EXTENDED (5655-V93)                          System Date: 2021.178
REPORT:  hco002 V2.R2                               COMPRESSION PROJECTION                                   System Time: 15.18.40

                                        _____________________________________________________
                                       |      DICTIONARY FUNCTION:  Build                    |
                                       |          DICTIONARY NAME:  TESTDIC2                 |
                                       |_____________________________________________________|

---------------------|---------|-------------------------------------------------------------------------------------------------------------------|-----------------------------
                     |         |                    DATE LENGTH FROM START BYTE                                                                    |
                     |         |-------------------------------------------------------------------------------------------------------------------|
           START     | DICT    |       BEFORE COMPRESS                AFTER COMPRESS                 SAVINGS                                          |         INPUT          BUILD
            BYTE     | TYPE    |    MIN      AVG      MAX        MIN      AVG      MAX        AVG   PCT                                          |       RECORDS        RECORDS
---------------------|---------|-------------------------------------------------------------------------------------------------------------------|-----------------------------
                     |         |                                                                                                                   |
               9     | Custom  |     38       52       76          9       18       42         34   65%                                          |           169          4,906
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------

                                The projected savings do not include IMS or access method usage.
                                            (PCT SAVINGS = AVG SAVINGS X 100 / AVG BEFORE)


                                                                END OF REPORT
```

Figure 17. Compression Projection report

The Compression Projection report that is generated by the Build function contains the following information:

START BYTE
The value that is specified in the START BYTE parameter.

DICT TYPE
The type of dictionary, which is always custom.

DATA LENGTH FROM START BYTE
Shows the following columns:

BEFORE COMPRESS MIN
Smallest input segment size before compression.

BEFORE COMPRESS AVG
Average input segment size before compression.

BEFORE COMPRESS MAX
Largest input segment size before compression.

AFTER COMPRESS MIN
Smallest input segment size after compression.

AFTER COMPRESS AVG
Average input segment size after compression.

AFTER COMPRESS MAX
Largest input segment size after compression.

SAVINGS AVG
Average savings of bytes as determined by the following formula:

```
(BEFORE COMPRESS AVG - AFTER COMPRESS AVG)
```

Bytes that are skipped as a result of the START BYTE specification are excluded.

SAVINGS PCT
The percent of storage that is saved as determined by the following formula:

```
(PCT SAVINGS = AVG SAVINGS X 100 / AVG BEFORE)
```

Bytes that are skipped as a result of the START BYTE specification are excluded.

INPUT RECORDS
The number of records that the Build function read from the HDCDIT data set to generate these statistics.

BUILD RECORDS
The number of input records that the Build function read from the HDCDIN data set to build the dictionary.

Chapter 8. Examining an existing dictionary

The Examine function is a reporting tool that lets you compare hardware data compression for alternate dictionaries.

The Examine function reports the compression savings that an existing IMS HD Compression Extended dictionary can provide when it is used to compress data that is different from the data that was used to create the dictionary.

The Examine function has the following two subfunctions:

Examine Generic
This subfunction compresses segments or databases by using the generic dictionary. This subfunction produces the HCO003 report, which can help you evaluate whether the generic dictionary provides adequate compression savings.

Examine Compare
This subfunction compresses segments or databases by using an existing custom dictionary. This subfunction produces the HCO004 report, which can help you evaluate the effectiveness of compression in the following situations:

- Your enterprise or custom dictionary in compressing data that is different from the data that was originally used to create the dictionary.
- Your dictionary in compressing data after a major update of your database. The Examine function can also be used to periodically check that compression of your data is optimal.

To allow selective compression, the Examine function has a START BYTE option, which lets you adjust where compression begins in the input records. You can also choose data selectively by skipping key data, unload-record prefix data, or specific fields in a segment. Figure 11 on page 37 show examples of the START BYTE option.

You run the Examine function by using the HCOZLDU0 program. It runs as a single-step, standard z/OS batch job.

Topics:

- “Examining a dictionary by using Examine ” on page 55
- “Examine panel ” on page 56
- “Examine JCL requirements ” on page 57
- “Compression Projection report (HCO003) ” on page 58
- “Compare Old/New Compression Projection report (HCO004) ” on page 60

Examining a dictionary by using Examine

Use the Examine function to produce reports about potential compression savings when you use an existing dictionary.

Before you begin

The Examine function supports all the acceptable data input sources.

Requirement: The output data sets HDCDDICT and HDCDSTAT must exist before a custom dictionary can be examined.

You can use either of the following methods to run the Examine function:

- From ISPF as a panel-driven function
- Directly from JCL as a standard z/OS batch job

© Copyright IBM Corp. 2000, 2024

About this task

To determine the ongoing effectiveness of the dictionary, use the old dictionary to compress a current data set of the same segment or database.

To determine if the dictionary could be used on another segment or database, use the dictionary to compress that data.

The following procedure describes how to run the utility from the ISPF panels.

Procedure

1. Provide an acceptable source of input data, or use the Examine function to access the Image Copy utility or the HD Unload utility directly.

   Acceptable input data includes the following sources:

   - Image Copy File
   - HD Unload File
   - Other - enter INPUT DSNAME (QSAM data set)
2. Run the Examine function from ISPF as a panel-driven function.

   Either complete the required fields on the Examine panel.
3. Specify the data set name in the **INPUT DSNAME** field to generate statistics.
4. Optional: For selective compression, adjust where compression begins in the input records by using the **START BYTE** option.

   You can skip key data, unload-record prefix data, or specific fields in a segment.
5. Generate the JCL by exiting the ISPF panel.
6. Optional: Edit the JCL if required.
7. Submit the generated JCL.
8. Interpret the output report to verify that the process completed successfully.

Results

The report that is created by the Examine function is sent to the SYSPRINT data set. The report that is produced depends on the type of database that is examined. The Compression Projection (HCO003) report is produced when the generic dictionary is examined. The Compare Old/New Compression Projection (HCO004) report is produced when a custom dictionary is examined.

Examine panel

You can use the Examine panel to specify the parameters that are required to examine an existing compression dictionary.

The Examine ISPF panel opens when you select the Examine function on the IMS HD Compression Extended main menu panel.

```
----------- IMS HD Compression Extended (Function = EXAMINE   ) ------------

COMMAND ===>

    DICTIONARY NAME        ==>

    INPUT SOURCE           ==>             1  = Image Copy File
                                           2  = HD Unload File
                                           3  = Other - enter INPUT DSNAME

    INPUT DSNAME           ==>

    USER DSN HLQ           ==>

    START BYTE             ==>

   DATABASE IDENTIFIER     ==>



Press END to exit.
```

Figure 18. Examine panel

The Examine panel contains the following fields:

DICTIONARY NAME
Specify the name of the existing dictionary to be examined. Enter HCOGDIC1 if the generic dictionary is being examined. The name that you specify is used to find the existing dictionary and statistics data sets (HDCDDICT and HDCDSTAT) that are created by the Build function.

IMS HD Compression Extended uses *hlq.dictname*.HDCDDICT and *hlq.dictname*.HDCDSTAT as the naming convention for these data sets. Specify the high-level qualifier in the **USER DSN HLQ** field. If the dictionary and statistics data sets do not conform to this convention, you must change the JCL.

INPUT SOURCE
Specify the input to the Examine function.

INPUT DSNAME
Specify the fully qualified name of the input data set. If you specified the INPUT SOURCE as 1, you must enter an image copy file data set name. If you specified the INPUT SOURCE as 2, you must enter an HD unload file data set name. If you specified the INPUT SOURCE as 3, you must enter a variable-blocked QSAM data set.

USER DSN HLQ
Specify the high-level qualifier for the input dictionary and statistics data sets.

START BYTE
Specify the relative byte number in the input source record at which compression analysis begins. Its point of reference is the beginning of the record, not the end of an unload utility record prefix as is the case with the KEYEND parameter of the Detail Analyze function. Specifying a value in the **START BYTE** field lets you adjust where compression analysis begins in the input records. Figure 11 on page 37 and Figure 12 on page 38 show examples of this parameter.

DATABASE IDENTIFIER
Specify a valid DBD name to identify the database whose segments are being examined.

Examine JCL requirements

The JCL for the Examine function can be created from your ISPF panel input. You can also create your own JCL.

The following example shows the JCL for the Examine function. You can use this sample to help you develop your own JCL for the Examine function.

```
//*********************************************************************
//* EXAMINE FUNCTION
//*********************************************************************
//STATS    EXEC PGM=HCOZLDU0,PARM='dictnameMnnnn'
//STEPLIB  DD DISP=SHR,
//         DSN=hcohlq.SHCOLMOD
//SYSPRINT DD SYSOUT=*,DCB=RECFM=FBA
//SYSUDUMP DD SYSOUT=*
//HDCDIN   DD DISP=SHR,
//         DSN=hlq.input.data
//HDCDDICT DD DISP=SHR,
//         DSN=hlq.dictname.HDCDDICT
//HDCDSTAT DD DISP=SHR,
//         DSN=hlq.dictname.HDCDSTAT
```

The Examine JCL references the following data sets:

EXEC
This statement must have the following format:

```
//         EXEC PGM=HCOZLDU0,PARM='dictnameMnnnnn'
```

where:

dictname
Specifies the 1- to 8-character name of the dictionary to be examined.

M
The letter M, as the ninth character in the parameter list, indicates that you are running the Examine function.

nnnnn
Specifies the value of the START BYTE option. This 5-character specification begins at character 10 in the parameter list. This specification can be omitted. It defaults to 0. Figure 11 on page 37 shows examples of this parameter.

STEPLIB
The input PDS that contains load module HCOZLDU0.

HDCDIN
The input data set.

HDCDDICT
The name of the generic dictionary (*hlq*.HCOGDIC1.HDCDDICT) or your custom dictionary (*hlq.dictname*.HDCDDICT).

Use this parameter to specify the dictionary that is to be examined.

The dictionary must have the z/OS data set format of 16 records of 4 KB each.

The Build function created this data set for the dictionary that you defined. IMS HD Compression Extended provided this data set for the generic dictionary.

Do not use the dictionary object data set form of either dictionary.

HDCDSTAT
The name of the custom dictionary statistics data set (*hlq.dictname*.HDCDSTAT).

This data set was created by the Build function when the dictionary was created.

Specify DD DUMMY when you examine the generic dictionary.

Compression Projection report (HCO003)

The Compression Projection report (HCO003) shows the compression projection report that is produced by the Examine function when the generic dictionary is being examined.

Before you run the Compress function, lengths might not match the DBD-specified lengths. If a segment with an odd length value is defined in the DBD, a hexadecimal zero pad byte is added at the end of each

segment by the IMS Unload utility. If such records are used as input to HDCE, the lengths differ from the DBD-specified lengths.

The following figure shows the compression projection report produced by the Examine function when the generic dictionary is being examined. IMS prefix length and access method usage can be excluded from these statistics.

```
PAGE:    1                                IMS HD COMPRESSION EXTENDED (5655-V93)                        System Date: 2021.178
REPORT:  HCO003 V2.R2                              COMPRESSION PROJECTION                               System Time: 15.19.55

                                      _____________________________________________________
                                     |      DICTIONARY FUNCTION:  Examine                  |
                                     |          DICTIONARY NAME:  HCOGDIC1 (IBM Generic)   |
                                     |_____________________________________________________|

--------------------|---------|-------------------------------------------------------------------------|------------------------------
                    |         |                        DATE LENGTH FROM START BYTE                      |
                    |         |-------------------------------------------------------------------------|
           START    | DICT    |       BEFORE COMPRESS                AFTER COMPRESS            SAVINGS    |         INPUT          BUILD
            BYTE    | TYPE    |     MIN      AVG      MAX        MIN      AVG      MAX        AVG   PCT |       RECORDS        RECORDS
--------------------|---------|-------------------------------------------------------------------------|------------------------------
                    |         |                                                                         |
                0   | Generic|     47       61       85         39       48       68         13 **21% |           169            n/a
--------------------------------------------------------------------------------------------------------------------------------------

                              The projected savings do not include IMS or access method usage.
                                        (PCT SAVINGS = AVG SAVINGS X 100 / AVG BEFORE)
                            **Better results will be achieved by building a custom dictionary.


                                                                  END OF REPORT
```

Figure 19. Compression Projection report

The Compression Projection report that is generated by the Examine function contains the following information:

START BYTE
The value specified in the START BYTE parameter.

DATA LENGTH FROM START BYTE
Shows the following columns:

BEFORE COMPRESS MIN
Smallest input segment size before compression.

BEFORE COMPRESS AVG
Average input segment size before compression.

BEFORE COMPRESS MAX
Largest input segment size before compression.

AFTER COMPRESS MIN
Smallest input segment size after compression.

AFTER COMPRESS AVG
Average input segment size after compression.

AFTER COMPRESS MAX
Largest input segment size after compression.

SAVINGS AVG
Average savings of bytes as determined by the following formula:

```
(BEFORE COMPRESS AVG - AFTER COMPRESS AVG)
```

Bytes that are skipped as a result of the START BYTE specification are excluded.

SAVINGS PCT
The percent of storage saved as determined by the following formula:

```
(PCT SAVINGS = AVG SAVINGS X 100 / AVG BEFORE)
```

Bytes that are skipped as a result of the START BYTE specification are excluded.

If the projected saving is less than 40%, consider creating a custom dictionary.

INPUT RECORDS
The number of records that the Examine function read to generate these statistics from the HDCDIT data set.

BUILD RECORDS
Not applicable.

Compare Old/New Compression Projection report (HCO004)

The Compare Old/New Compression Projection report (HCO004) shows the report that is produced by the Examine function when compression that uses a custom dictionary is compared to compression that uses the generic dictionary.

The following figure shows the compression projection report produced by the Examine function when a custom dictionary is being examined. IMS prefix length and access method usage can be excluded from these statistics.

```
PAGE:    1                                IMS HD COMPRESSION EXTENDED (5655-V93)                          System Date: 2021.178
REPORT:  HCO004 V2.R2                     COMPARE OLD/NEW COMPRESSION PROJECTION                          System Time: 15.19.01

                                      _____________________________________________________
                                     |      DICTIONARY FUNCTION:  Examine Compare          |
                                     |          DICTIONARY NAME:  TESTDIC2                 |
                                     |_____________________________________________________|

 ------------------------------|------------------------------------------------------------------------------------------------------------
                               |                     DATE LENGTH FROM START BYTE                                                            |
                               |____________________________________________________________________________________________________________|
 SOURCE     START              |       BEFORE COMPRESS                 AFTER COMPRESS                 SAVINGS                 |          INPUT
 OF STATS    BYTE              |    MIN      AVG      MAX        MIN      AVG      MAX        AVG   PCT                     |        RECORDS
 ------------------------------|------------------------------------------------------------------------------------------------------------
                               |                                                                                                            |
 OLD            9              |     38       52       76          9       18       42         34   65%                     |           169
 NEW            9              |     82      482   10,026          9       41      290        441   91%                     |        31,257
 ------------------------------------------------------------------------------------------------------------------------------------------

                                Stats in the OLD row pertain to input records used in the original BUILD of the dictionary.
                                Stats in the NEW row pertain to new input records compressed using the old dictionary.
                                The projected savings do not include IMS or access method usage.
                                            (PCT SAVINGS = AVG SAVINGS X 100 / AVG BEFORE)


                                                                  END OF REPORT
```

Figure 20. Compare Old/New Compression Projection report

The Compare Old/New Compression Projection report contains the following information:

SOURCE OF STATS
Shows the following columns:

OLD
Compression statistics that were produced when the Build function created the custom dictionary.

NEW
Compression statistics that were produced by the Examine function if the new input records were compressed by the existing custom dictionary.

These statistics let you compare the compression results of your dictionary against a different data set.

Before you run the Compress function, lengths might not match the DBD-specified lengths. If a segment with an odd length value is defined in the DBD, a hexadecimal zero pad byte is added at the end of each segment by the IMS Unload utility. If such records are used as input to HDCE, the lengths differ from the DBD-specified lengths.

START BYTE
The value that is specified in the START BYTE parameter.

DATA LENGTH FROM START BYTE
Shows the following columns:

BEFORE COMPRESS MIN
Smallest input segment size before compression.

BEFORE COMPRESS AVG
Average input segment size before compression.

BEFORE COMPRESS MAX
Largest input segment size before compression.

AFTER COMPRESS MIN
Smallest input segment size after compression.

AFTER COMPRESS AVG
Average input segment size after compression.

AFTER COMPRESS MAX
Largest input segment size after compression.

SAVINGS AVG
Average savings of bytes per segment as determined by the following formula:

```
(BEFORE COMPRESS AVG - AFTER COMPRESS AVG)
```

Bytes that are skipped as a result of the START BYTE specification are excluded.

SAVINGS PCT
The percent of storage saved as determined by the following formula:

```
(PCT SAVINGS = AVG SAVINGS X 100 / AVG BEFORE)
```

Bytes that are skipped as a result of the START BYTE specification are excluded.

INPUT RECORDS
The number of records that the Examine function read to generate these statistics.

Chapter 9. Compressing your data

After you create a dictionary and an IMS compression exit routine, the Compress function is the last of the IMS HD Compression Extended functions that you must run.

Compression is done when IMS reloads the database. The Compress function can create the JCL to reload your database if you are using one of the following reload utilities:

- DFSURG10 (for full-function databases)
- FABCUR3 (for DEDB databases)

If you prefer to use your existing reload JCL, you can skip the instructions for using the Compress function.

Although a dictionary and an IMS compression exit routine can be built at any time, data compression requires an unload and reload of the database. The most recent database unload must be used as input to the reload. The unload must be in uncompressed format, and the database must not have been available for update since the unload. Otherwise, database integrity is jeopardized.

The Compress function does not produce any reports other than those that are produced by the reload utilities.

The ISPF panel functions for Compress help you create and submit the reload JCL.

Topics:

- “Compressing your data by using Compress ” on page 63
- “Compress panel ” on page 64
- “Compress JCL requirements ” on page 65

Compressing your data by using Compress

Use the Compress function to reload a database by using the new DBD and to compress your data.

Before you begin

Restrictions:

- The restrictions of the reload utility you are using apply.
- The Compress function can be run only from ISPF panels.

About this task

The following procedure describes how to run the Compress function from the ISPF panels.

Procedure

1. Provide an acceptable unload data set.
   Provide a reload input data set (typically the unloaded database data set name).
2. Complete the required fields on the ISPF panel.
3. Generate the JCL by exiting the ISPF panel.
4. Optional: Edit the JCL if required.
5. Submit the generated JCL.
6. Check the reload job output to make sure that the job completed successfully.
7. Run an ACBGEN before making the database available to users.

© Copyright IBM Corp. 2000, 2024

Results

Output is the reloaded database data sets.

Compress panel

The Compress panel opens when you select the Compress function on the IMS HD Compression Extended main menu panel.

The Compress panel prompts you for the parameters that are needed to run compression. The following figure shows a sample of the Compress panel.

```
---- IMS HD Compression Extended --- Function = COMPRESS  --------


COMMAND ===>
  RELOAD DATABASE USING THE NEW DBD             * THESE FIELDS ARE OPTIONAL
  TAPE INPUT           ==>                             (Y OR N)
  RELOAD INPUT DSN     ==>
  IDCAMS CNTL DSNAME   ==>
  IDCAMS CNTL MEMBER   ==>
  IMS RESLIB DSN       ==>
  USER RESLIB          ==>
  NEW DBD LIBRARY DSN  ==>
  NEW DBD NAME         ==>
  DATABASE DSN         ==>
  DATABASE DD NAME     ==>
* DB INDEX CLUSTER     ==>
* DB INDEX DD NAME     ==>
* DURIWRK DSN          ==>
* DURDBDFN DSN         ==>
* IMS RECON1-3 PREFIX  ==>
Press END to exit.
```

Figure 21. Compress panel

The Compress panel contains the following fields:

TAPE INPUT
Type Y if your reload input data set is on tape. Otherwise, specify N.

RELOAD INPUT DSN
Specify the name of the reload input data set (typically the unloaded database data set name).

IDCAMS CNTL DSNAME
Specify the name of the IDCAMS control data set.

IDCAMS CNTL MEMBER
Specify an IDCAMS control data set member that contains the IDCAMS DEFINE parameters for the database that is being reloaded.

Tip: Adjust the CI and size parameters to maximize compression. For more information, see "Segment Distribution by Compressed Length report (HCO007) " on page 33.

IMS RESLIB DSN
Specify the name of the IMS RESLIB data set.

USER RESLIB
Specify the name of the data set on which the IMS compression exit routine resides.

NEW DBD LIBRARY DSN
Specify the name of the data set that contains the new DBD.

NEW DBD NAME
Specify the name of the new DBD that was built by using the compression exit routine.

DATABASE DSN
Specify the name of the input data set that contains the database that is to be reloaded. If there are multiple data set groups, you must add DD statements for them when the JCL is opened to be edited.

DATABASE DD NAME
Specify the name of the database to be reloaded (used as a DD name in the DFSURGL0 execution step). If you are reloading multiple data set groups, you must add DD statements for them when the JCL is opened to be edited.

DB INDEX CLUSTER
Specify the cluster name of the database to be reloaded (if required).

DB INDEX DD NAME
Specify the name of the index to be reloaded if required. You can use the name of the index that was used as a DD name in the DFSURGL0 execution step.

DURIWRK DSN
Specify the name of the DURIWRK data set that is used by the FABCUR3 utility (for DEDB reload only).

DURDBDFN DSN
Specify the name of the DURDBDFN data set that is used by the FABCUR3 utility (for DEDB reload only).

IMS RECON1-3 PREFIX
Specify the data set name prefix of the three IMS RECON data sets (if they are required).

Compress JCL requirements

The JCL to reload the database is opened in ISPF edit. It consists of the basic code that is needed to support the standard DFSURG10 or FABCUR3 reload utilities. You might have to complete the JCL to meet your requirements.

For additional information, see the other IMS HD Compression Extended JCL requirements.

Chapter 10. Troubleshooting

The topics in this section provide you with technical references to help you troubleshoot and diagnose IMS HD Compression Extended problems.

Topics:

- “Runtime error messages (HCO)” on page 67
- “Abend codes” on page 70
- “Return codes” on page 74
- “Gathering diagnostic information” on page 75

Runtime error messages (HCO)

Use the information in these messages to help you diagnose, troubleshoot, and solve IMS HD Compression Extended problems.

Message format

IMS HD Compression Extended messages adhere to the following format:

```
HCOnnnnx text
```

where:

HCO
Indicates that the message was issued by IMS HD Compression Extended

nnnn
Indicates the message identification number

x
Indicates the severity of the message:

A
Indicates that operator intervention is required before processing can continue.

E
Indicates that an error occurred, which might or might not require operator intervention.

I
Indicates that the message is informational only.

W
Indicates that the message is a warning to alert you to a possible error condition.

Each message also includes the following information:

Explanation:
Explains what the message text means, why it occurred, and what its variables represent.

System action:
Explains what the system will do in response to the event that triggered this message.

User response:
Describes whether a response is necessary, what the appropriate response is, and how the response will affect the system or program.

Problem determination
Helps you understand the kind of fault you have, and the appropriate course of action.

© Copyright IBM Corp. 2000, 2024

HCO3589E **OPEN FAILED FOR CONTROL FILE**

Explanation

After an OPEN macro was issued for a control statement data set, DCBOFLGS is not zero.

System action

A USER 3589 abend is issued.

User response

Correct the error, and rerun the job.

Problem determination

Check for any obvious runtime errors.

HCOZ0001I ***cccccccc* HDC DICTIONARY BUILT IN DD HDCDOUT FILE**

Explanation

HCOZLDU0 issues the message HCOZ0001I when the utility successfully completes with a return code of 0. *cccccccc* is the dictionary name that was passed as a parameter to the utility.

System action

The utility ends normally.

User response

None. This message is informational.

Problem determination

No error.

HCOZ0002E ***cccccccc* HDC DICTIONARY NOT BUILT, BECAUSE SOURCE DATA DOES NOT MATCH EXPANDED DATA. THE FIRST 10 MISMATCHED RECORDS WILL BE PRINTED BEFORE THE COMPRESSION STATISTICS**

Explanation

Message HCOZ0002E is issued when mismatched source and expanded data is detected. A dictionary object data set is not built. *cccccccc* is the dictionary name that was passed as a parameter to the utility.

System action

The utility ends normally.

User response

Contact your database administrator or IBM Software Support.

Problem determination

No error.

HCOZ0003W **HDC DICTIONARY NOT BUILT, BECAUSE MVS CSRCMPSC SERVICE IS NOT INSTALLED**

Explanation

Message HCOZ0003W is issued when the MVS™ compression and expansion service CSRCMPSC is not installed. An HDC dictionary cannot be built.

System action

The utility ends normally.

User response

Install MVS CSRCMPSC service, and rerun the utility.

Problem determination

No error.

HCOZ0004E **HDC DICTIONARY NOT BUILT, BECAUSE MVS CSRCMPSC SERVICE HAD A LOGIC ERROR**

Explanation

Message HCOZ0004E is issued when a logic error is detected while using the MVS CSRCMPSC compression and expansion service macro. An HDC dictionary cannot be built.

System action

The utility ends normally.

User response

Contact IBM Software Support.

Problem determination

No error.

HCOZ0005E ***ddname* INPUT FILE EMPTY OR INCOMPLETE. PROCESSING TERMINATED**

Explanation

Message HCOZ0005E is issued when an empty data set is detected. The ddname of the empty data set is named in the message.

System action

The utility ends normally.

User response

Notify the database administrator.

Problem determination

No error.

HCOZ0006E **HCOGDIC1 DICTIONARY NAME RESERVED FOR "EXAMINE" USE ONLY**

Explanation

Message HCOZ0006E is issued if the generic dictionary reserved name of HCOGDIC1 is used with the Build or Analyze function. The HCOGDIC1 name can only be used for the Examine function.

System action

The utility ends normally.

User response

Notify the database administrator.

Problem determination

None.

HCOZ0007E ***ssssssss* HAD INVALID SYSIN CONTROL INPUT. SEE REPORT OUTPUT.**

Explanation

HCOZLDH0 issues this message when invalid, duplicate or no segment names are supplied in the SYSIN input data set.

HCOZLDE0 issues this message when invalid segment names, key end, fixed data, or variable data is read in the SYSIN input data set.

ssssssss is the SYSIN segment name.

System action

The utility ends before dictionary processing begins.

User response

Check the output report for details of the errors and correct the errors.

Problem determination

Check the output report for details of the errors.

HCOZ0008E **START BYTE SPECIFIED IN PARAMETER LONGER THAN QSAM LL FIELD. PROGRAM TERMINATED.**

Explanation

HCOZLDU0 issues this message if a record shorter than the value specified in the START BYTE parameter is detected in either the HDCDIT (statistics generation) or HDCDIN (dictionary creation) data set.

System action

The utility ends before dictionary creation or analysis is completed.

User response

If the HDCDIN or the HDCDIT data set is an HD unload data set, the short record is probably an unload header or trailer record. Use the HD Unload Extract utility, HCOZLDH0, to remove these records and any segment records not needed for the dictionary.

Otherwise, ensure that the START BYTE parameter correctly reflects the record lengths in the input data sets.

Problem determination

The HDCDIN or HDCDIT data set contains segment records not needed for the dictionary, or the START BYTE parameter is invalid.

Abend codes

The explanations and user responses provided in this abend code reference can help you diagnose, troubleshoot, and solve IMS HD Compression Extended problems.

For each abend code, the following information is provided where applicable:

Explanation:
Explains what the abend code means, why it occurred, and what its variable entry fields are (if any).

System action:
Explains what the system will do next.

User response:
Describes whether a response is necessary, what the appropriate response is, and how the response will effect the system or program.

Problem determination
Helps you understand the kind of fault you have, and the appropriate course of action.

Abend U3315

Abend issued for HCOZLDH0, HCOZLDE0, and HCOZLDU0.

Abend code U3315 is issued with the following reason codes:

Reason code 1

Explanation

The utility failed to get storage in the initialize dictionary module, or an internal error occurred in that module.

System action

The utility ends abnormally.

User response

Increase the region size and rerun the utility. If the problem persists, contact IBM Software Support.

Problem determination

A storage shortage occurred.

Reason code 2

Explanation

The utility failed because it passed invalid parameters to the build work dictionary module.

System action

The utility ends abnormally.

User response

Contact IBM Software Support.

Problem determination

Invalid parameters were passed to the build work dictionary module (an internal logic error occurred).

Reason code 3

Explanation

The utility failed because it passed invalid parameters to the build dictionary module.

System action

The utility ends abnormally.

User response

Contact IBM Software Support.

Problem determination

Invalid parameters were passed to the build dictionary module (an internal logic error occurred).

Reason code 4

Explanation

The utility failed because it passed invalid parameters to the validate dictionary module, or a bad dictionary was created.

System action

The utility ends abnormally.

User response

Contact IBM Software Support.

Problem determination

Invalid parameters were passed to the validate dictionary module (an internal logic error occurred), or a bad dictionary was created.

Reason code 5

Explanation

The utility failed because it detected an open or close error when accessing a data set.

System action

The utility ends abnormally.

User response

Check that the data set indicated by the ABEND has been created with the correct attributes and is available to the utility. If not, then make the necessary corrections and rerun the utility. If there are no apparent data set allocation or access problems, contact IBM Software Support.

Problem determination

An open or close error was detected while accessing a data set.

Reason code 6

Explanation

The utility failed because a logic error occurred in the statistics or data integrity module during data compression or expansion.

System action

The utility ends abnormally.

User response

Contact IBM Software Support.

Problem determination

A logic error occurred while attempting data compression or expansion.

Abend U3325

Abend issued for HCOZLDX0 and HCOZLDXV.

Abend code 3325 is used by the compression exit routine and issues one of the following reason codes. The reason code can be found in the last byte of Register 14.

Reason code 1

Explanation

Invalid function code was passed to the compression macro.

System action

The utility ends abnormally.

User response

Contact IBM Software Support.

Problem determination

A logic error occurred while attempting data compression or expansion.

Reason code 2

Explanation

Sequence field is not completely in the segment.

System action

The utility ends abnormally.

User response

Contact IBM Software Support.

Problem determination

A logic error occurred while attempting data compression or expansion.

Reason code 3

Explanation

Compression or expansion input length is negative.

System action

The utility ends abnormally.

User response

Contact IBM Software Support.

Problem determination

A logic error occurred while attempting to perform data compression or expansion.

Reason code 4

Explanation

INIT processing is not set for open.

System action

The utility ends abnormally.

User response

Check that the DBD is correct for the database access requirements and database type.

Problem determination

A logic error occurred while attempting to perform data compression or expansion.

Reason code 5

Explanation

No dictionary address.

System action

The utility ends abnormally.

User response

Contact IBM Software Support.

Problem determination

A logic error occurred while attempting to perform data compression or expansion.

Reason code 6

Explanation

Dictionary not on page boundary.

System action

The utility ends abnormally.

User response

Check that the link edit for the IMS segment edit or compression exit has been performed correctly.

Problem determination

A logic error occurred while attempting to perform data compression or expansion.

Reason code 7

Explanation

No eye catcher in dictionary.

System action

The utility ends abnormally.

User response

Check that the link edit for the IMS segment edit or compression exit has been performed correctly.

Problem determination

A logic error occurred while attempting to perform data compression or expansion.

Reason code 8

Explanation

No compression service available.

System action

The utility ends abnormally.

User response

Ensure that the application is running on a machine that supports hardware data compression.

Problem determination

An error occurred while attempting to access the hardware data compression microcode or the MVS software simulation code.

Reason code 9

Explanation

GETMAIN failure for full-function database call.

System action

The utility ends abnormally.

User response

Contact IBM Software Support.

Problem determination

A logic error occurred while attempting to perform data compression or expansion.

Reason code 10

Explanation

Minimum segment size for Fast Path database is invalid.

System action

The utility ends abnormally.

User response

Check that the DBD is correct for the database minimum and maximum segment lengths.

Problem determination

An invalid segment length was passed to the data compression process.

Reason code A

Explanation

GETMAIN failure for DEDB Fast Path database call.

System action

The utility ends abnormally.

User response

Contact IBM Software Support.

Problem determination

A logic error occurred while attempting to perform data compression or expansion.

Reason code B

Explanation

Compression input length < 2 bytes.

System action

The utility ends abnormally.

User response

Notify the database administrator.

Problem determination

A segment shorter than 2 bytes was detected during compression processing.

Reason code C

Explanation

Expansion input length < 2 bytes.

System action

The utility ends abnormally.

User response

Notify the database administrator.

Problem determination

A segment shorter than 2 bytes was detected during expansion processing.

Reason code D

Explanation

Hardware instruction issued a nonzero return code.

System action

The utility ends abnormally.

User response

Contact IBM Software Support.

Problem determination

A logic error occurred while attempting to perform data compression or expansion.

Reason code E

Explanation

Software simulation issued nonzero return code.

System action

The utility ends abnormally.

User response

Contact IBM Software Support.

Problem determination

A logic error occurred while attempting to perform data compression or expansion.

Reason code F

Explanation

INIT processing is not set for close.

System action

The utility ends abnormally.

User response

Check that the DBD is correct for the database access requirements and database type.

Problem determination

A logic error occurred while attempting to perform data compression or expansion.

Return codes

IMS HD Compression Extended messages are often accompanied by return codes.

From the HD Unload Extract utility

The following return codes are issued by the HD Unload Extract utility:

Return code	Meaning
0	The utility ended successfully.
8	The utility did not execute successfully. No data was found for the segments that are specified in the SYSIN control statements.
20	The utility ended unsuccessfully and issued the accompanying HCOZ0005E message because an input data set was empty. Processing could not continue.
24	The utility ended unsuccessfully and issued the accompanying HCOZ0006E message. The SYSIN segment name was invalid or missing.

From the Detail Analyze function

The following return codes are issued by the Detail Analyze function.

Return code	Meaning
0	The utility ended successfully.
12	The utility ended unsuccessfully and issued the accompanying HCOZ0003W message because MVS CSRCMPSC is not installed on the machine.
20	The utility ended unsuccessfully and issued the accompanying HCOZ0005E message because an input data set is empty.
24	The utility ended unsuccessfully and issued the accompanying HCOZ0006E message. The SYSIN segment name was invalid or missing.

From the Analyze, Build, and Examine functions

The following return codes are issued by the Analyze, Build, and Examine functions.

Return code	Meaning
0	The utility ended successfully and issued the accompanying HCOZ0001I message.

Return code	Meaning
8	The utility ended successfully and issued the accompanying HCOZ0002E message, but did not build a dictionary object data set because data integrity checks were detected between a source sequential variable-length input record and its equivalent re-expanded record.
12	The utility ended unsuccessfully and issued the accompanying HCOZ0003W message because z/OS CSRCMPSC is not installed on the machine.
20	The utility ended unsuccessfully and issued the accompanying HCOZ0005E message because an input data set was empty. Processing could not continue.

Gathering diagnostic information

Before you report a problem with IMS HD Compression Extended to IBM Software Support, gather the appropriate diagnostic information.

Procedure

Provide the following information for all IMS HD Compression Extended problems:

- A clear description of the problem and the steps that are required to re-create the problem
- All messages that were issued as a result of the problem
- Product release number and the number of the last program temporary fix (PTF) that was installed
- The version of IMS that you are using and the type and version of the operating system that you are using

Provide additional information based on the type of problem that you experienced:

For online abends, provide the following information

- A screen capture of the panel that you were using when the abend occurred
- The job log from the TSO session that encountered the abend
- The job log from the server
- A description of the task that you were doing before the abend occurred

For errors in batch processing, provide the following information

- The complete job log
- Print output
- Contents of any data sets that were used during the processing

Notices

This information was developed for products and services offered in the U.S.A.

This material may be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Director of Licensing
IBM Corporation
North Castle Drive

© Copyright IBM Corp. 2000, 2024

Armonk, NY 10504-1785
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

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 on the web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml.

Other company, product, and service names may be trademarks or service marks of others.

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions:

Applicability: These terms and conditions are in addition to any terms of use for the IBM website.

Personal use: You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use: You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights: Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED,

INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

Privacy policy considerations

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM's Privacy Policy at http://www.ibm.com/privacy and the section titled "Cookies, Web Beacons, and Other Technologies" in IBM's Online Privacy Statement at http://www.ibm.com/privacy/details. Also, see the "IBM Software Products and Software-as-a-Service Privacy Statement" at http://www.ibm.com/software/info/product-privacy.

Index

A

abend codes 70
ACBGEN 2, 43
accessibility
    overview 12
accounting information
    changing 10
Analyze function
    description 7, 35
    execution 9
    generating statistics 35
    HCOZLDU0 program 9
    input and output data sets 35, 38
    JCL 35, 38
    panel 36
    report 39
    using 35
Analyze panel 36
analyzer component
    overview 5

B

Build function
    creating a compression exit routine 42
    description 7, 41
    execution 9
    generating statistics 41
    HCOZLDU0 program 9
    input and output data sets 41, 49
    JCL
        generating 41
        requirements for Build function 49
        requirements for compression exit routine and DBD 51
    modifying or creating a DBD 43
    panels
        Build 45
        Compression Exit Routine 47
        Edit DBD 47
    report 51
    restrictions 41
    using 41
Build panel 45

C

codes 70
Compare Old/New Compression Projection report 60
components 5
Compress function
    description 7, 63
    execution 9
    HCOZLDX0 program 9, 44
    HCOZLDXV program 9, 44
    JCL 65
Compress function *(continued)*
    panel 64
    restrictions 63
    using 63
Compress panel 64
compression exceptions 43
compression exit routine
    creating 42
    description 2
    JCL 51
    relinking 44
Compression Exit Routine panel 47
Compression Projection report
    for Analyze 39
    for Build 51
    for Examine 58
COMPRTN keyword 43, 48
configuration 13
cookie policy 77
custom dictionary 5
customized database dictionary 5
customized segment dictionary 5

D

data sets
    HDCDDICT
        for Build 49
        for Detail Analyze 30
        for Examine 57
    HDCDIT 36, 49
    HDCDOUT 49
    PROCCTL 22
    QSAM 21
    SEGIN 22
    SGOUTnnn 23, 24
    SYSIN 24
data sharing support
    scenario 2
DBD (database description)
    description 2
    JCL 51
    modifying or creating 43
    modifying the MAXBYTES field 43
DEDB unload data set 35
Detail Analyze function
    description 7, 27
    execution 9
    generating statistics 27
    HCOZLDE0 program 9
    input and output data sets 27, 30
    JCL 27, 30
    panel 29
    reports
        Detail Segment Compression Analysis 31
        overview 27
        Segment Distribution by Compressed Length 33

Detail Analyze function *(continued)*
using 27
Detail Analyze panel 29
device types
changing 10
DFSURG10 utility 63
diagnostic information
gathering 75
dictionary
creating 41
customized database 5
customized segment 5
enterprise 5
generic 5
input data 41
object data set 5
strategies 5
z/OS data set 5
Ziv-Lempel technique 4
dictionary driver 2
documentation
accessing 11
sending feedback 11

E

Edit DBD panel 47
enterprise dictionary 5
error messages 67
Examine function
description 7, 55
execution 9
generating statistics 55
HCOZLDU0 program 9
input and output data sets 55, 57
JCL 55, 57
panel 56
reports 58, 60
restrictions 55
using 55
exceptions, compression 43
exit routines, compression 42
Extract function
description 7, 21
HD Unload Extract utility 22
Image Copy Extract utilities 21

F

FABCUR3 utility 63
formatter component
overview 5

G

generic dictionary 5

H

hardware data compression (HDC) 1
HCO001 report 39
HCO002 report 51
HCO003 report 58
HCO004 report 60
HCO005 report 23
HCO006 report 31
HCO007 report 33
HCORLNKD 44
HCOZLDE0 program 9
HCOZLDH0 program 9
HCOZLDU0 program 9
HCOZLDX0 44
HCOZLDX0 program 9
HCOZLDXV program 9
HD Unload Extract panel 24
HD Unload Extract utility
description 22
evaluating compression results 23
execution 9
HCOZLDH0 program 9
input and output data sets 23, 24
JCL 23, 24
report 23
restrictions 23
using 23
HDC (hardware data compression) 1
HDCDDICT data set 49, 57
HDCDIT data set 36, 49
HDCOUT data set 49

I

Image Copy Extract panel 21
Image Copy Extract utilities
description 21
execution 9
input and output data sets 22
JCL 22
panel 21
image copy support 2
IMS exit 44
IMS HD Compression Extended main menu panel 17
input data sets
Analyze function 38
Build function 49
Detail Analyze function 30
Examine function 57
HD Unload Extract utility 24
Image Copy Extract utilities 22
input to dictionary 41

J

JCL (job control language)
Analyze function 35, 38
Build function 41, 49
Compress function 65
compression exit routine 51
Detail Analyze function 27, 30
edit DBD 51
Examine function 55, 57
HD Unload Extract utility 23, 24
Image Copy Extract utilities 22
reports
Segment Distribution by Compressed Length 27
Segment Distribution by Compressed Length report 27

JCL (job control language) *(continued)*
to link HCOZLDX0 44

K

KEYEND SYSIN option 27

L

legal notices
cookie policy 77
notices 77
product documentation 77
programming interface information 77
trademarks 77

M

maintenance
applying 44
MAXBYTES field 43
maximum segment length 43
messages
overview 67

N

notices 77

O

object data set 5
output data sets
Analyze function 35, 38
Build function 41, 49
data sets
SYSIN 30
Detail Analyze function 27, 30
Examine function 55, 57
HD Unload Extract utility 23, 24
Image Copy Extract utilities 22
SYSIN data set 30

P

panels
Analyze 36
Build 45
Compress 64
Compression Exit Routine 47
Detail Analyze 29
Edit DBD 47
Examine 56
HD Unload Extract utility 24
Image Copy Extract 21
IMS HD Compression Extended main menu 17
parser component
overview 5
performance considerations 2
problems
diagnostic information about 75
PROCCTL data set 22
process roadmap 15
product documentation terms and conditions 77
programming interface information 77

Q

QSAM data set 21

R

reader comment form 11
reference
abend codes 70
relinking 44
reports
Analyze function 39
Build function 51
Compare Old/New Compression Projection (for Examine) 60
Compression Projection
for Analyze 39
for Build 51
for Examine 58
description 10
Detail Analyze function 27
Examine function 58
HCO001 39
HCO002 51
HCO005 23
HCO006 31
HCO007 33
Segment Compression Analysis 27, 31
Segment Distribution by Compressed Length 27, 33
restrictions
Build function 41
Compress function 63
Examine function 55
HD Unload Extract utility 23
return codes 74
REXX procedure 17

S

sample members
HCORLNKD 44
scenarios
changing accounting information 10
changing device types 10
screen readers and magnifiers 12
SEGIN data set 22
SEGM statement 43, 48
Segment Compression Analysis report 27, 31
Segment Distribution by Compressed Length report 27, 33
segments
maximum length 43
not compressible 43
selective compression 41, 55, 56
service information 11
SGOUTnnn data set 23, 24
space considerations 2
START BYTE option
Analyze function 35
Build function 41

START BYTE option *(continued)*
    Examine function 55
starting
    IMS HD Compression Extended 17
statistics
    Analyze function 35
    Build function 41
    Detail Analyze function 27
strategies for dictionary 5
summary of changes 1
support
    required information 75
support information 11
SYSIN data set 24

T

technotes 11
trademarks 77

U

upgrading
    IMS Tools Base 44
user-written unload program 4
utilities
    DFSURG10 63
    FABCUR3 63
    HD Unload Extract 22
    Image Copy Extract 21

Z

z/OS
    data set 5
    hardware data compression 1
Ziv-Lempel, explained 4

Product Number: 5655-V93

SC27-9858-01