

IMS
Version 14

Release Planning
(March 1, 2018 edition)



IMS
Version 14

Release Planning
(March 1, 2018 edition)



Note

Before you use this information and the product it supports, read the information in “Notices” on page 267.

March 1, 2018 edition.

This edition applies to IMS 14 (program number 5635-A05), IMS Database Value Unit Edition, V14.01.00 (program number 5655-DSE), IMS Transaction Manager Value Unit Edition, V14.01.00 (program number 5655-TM3), and to all subsequent releases and modifications until otherwise indicated in new editions.

© **Copyright IBM Corporation 1995, 2017.**

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

Contents

About this information	vii
How to use the Release Planning information	vii
Prerequisite knowledge	viii
How new and changed information is identified	viii
How to read syntax diagrams	viii
Accessibility features for IMS 14	x
How to send your comments	xi

Part 1. General planning information for IMS 14 1

Chapter 1. Hardware requirements	3
Processor requirements	3
System console requirements	3
Tape unit requirements	3
Coupling facility requirements	3
DASD requirements	4
Large sequential data set support hardware requirements	5
Multiple Systems Coupling hardware requirements	5
Parallel RECON access hardware requirements	6
Terminals and equipment supported by IMS 14	6
Remote Site Recovery hardware requirements	7
zIIP utilization hardware requirements	8
Chapter 2. Software requirements	9
Operating software requirements	9
IMS 14 base software requirements	9
Coexistence APARs	10
Database Resource Adapter (DRA) software requirements	10
Data sharing software requirements	10
HALDB Index/ILDS Rebuild utility free space function software requirements	10
IMS callout function software requirements	10
IMS catalog function software requirements	11
IMS management of ACBs function software requirements	11
ISC TCP/IP software requirements	11
IMS Connect software requirements	12
Java application program support in IMS 14	12
Software requirements for Java application programs that use the IMS Universal drivers or JDR resource adapter	12
Software requirements for Java applications that access IMS transactions	14
Open Database solution software requirements	15
Parallel RECON access software requirements	15
SQL support software requirements	15
User exit enhancements software requirements	15
IMS Enterprise Suite software requirements	16
CICS subsystems supported	16
Db2 for z/OS subsystems supported	16
IBM MQ subsystems supported	17

Intersystem Communication (ISC) subsystems supported	17
Programming languages used to write IMS 14	17
Programming languages supported	17
Application programs supported	18

Chapter 3. Packaging for IMS 14 19

Chapter 4. Installation considerations 21

Chapter 5. Migration to IMS 14 23

General migration recommendations	23
Migration steps	26
Migrating to IMS 14 from IMS Version 12 or earlier	28
Discontinuance of support	28
Specific migration considerations	31
Migrating to IMS 14: DB	31
Migrating to IMS 14: TM	37
Migrating to IMS 14: System.	44
Fallback considerations	68
DBRC fallback considerations	69
Dynamic definition for MODBLKS resources fallback considerations.	70
Dynamic definition of MSC resources fallback considerations	71
IMS catalog fallback	71
Fallback from IMS-managed ACBs to ACBLIBs	73
Falling back from IMSRSC repository support for dynamically defined MSC resources	75

Chapter 6. Coexistence with IMS 14 77

Overview of coexistence APARs	77
Determining which coexistence service needs to be installed	78
General coexistence considerations	78
Specific coexistence considerations.	79
Common Queue Server coexistence considerations	79
DBRC coexistence considerations	80
DEDDB Alter enhancement coexistence considerations	81
DRA coexistence considerations	82
Database utility coexistence considerations	82
Dynamic definition for MODBLKS resources coexistence considerations	82
Exit routine coexistence considerations	83
Fast Database Recovery (FDBR) coexistence considerations	83
HALDB alter enhancement coexistence considerations	83
HALDB support for 8-GB OSAM data sets enhancement coexistence considerations	83
IMS 64-bit storage manager coexistence considerations	84
IMS catalog coexistence considerations	84

	Coexistence between IMS systems that manage ACBs differently	84
	IMS repository function coexistence considerations	85
	IMS Connect coexistence considerations	87
	MSC coexistence considerations	88
	Open Database coexistence considerations	88
	OTMA coexistence considerations	89
	PST coexistence considerations	89
	Remote Site Recovery (RSR) coexistence considerations	89
	Security enhancements coexistence considerations	92
	Shared message queue coexistence considerations	93
	Synchronous program switch coexistence considerations	93
	Syntax Checker coexistence considerations	93
	Restrictions for IMS 14 during coexistence	93

Chapter 7. Message and code changes in IMS 14. 95

	New messages and codes for IMS 14	95
	Changed messages and codes for IMS 14	101
	Deleted messages and codes for IMS 14	104

Chapter 8. Log record changes in IMS 14 105

Part 2. IMS Version 14 enhancements 107

Chapter 9. IMS enhancement PTFs 109

	The IMS continuous enhancement process.	109
	October 2017 enhancement PTFs	110
	July 2017 enhancement PTFs	110
	June 2017 enhancement PTFs	110
	May 2017 enhancement PTFs	111
	March 2017 enhancement PTFs	111
	February 2017 enhancement PTFs.	111
	January 2017 enhancement PTFs	112
	IMS 14 base-level enhancements	112

Chapter 10. IMS Database Manager enhancements 115

	64-bit buffer enhancement for Fast Path high-speed processing	115
	Automatic management of Fast Path DEDB SDEP buffers enhancement	116
	Database Image Copy 2 utility (DFSUDMT0) enhancement for zEnterprise Data Compression	117
	DBRC migration and coexistence	117
	DBRC REPAIR.RECON command enhancement	118
	DEDB Alter function enhancements	120
	Dynamic database definition enhancements	123
	Data Definition Language (DDL) for IMS	124
	IMS management of ACBs enhancement	127
	PSB activation control	136
	IMS catalog activity record	136
	Fast Path control block changes	137
	FDBR in-doubt thread support for ESAF	138

	GSAM metadata enhancement.	139
	HALDB support for 8-GB OSAM data sets enhancement	140
	ODBM accounting enhancement	142
	Concurrent active thread enhancement	142
	OSAM 24-bit storage constraint relief enhancement	142
	SQL support enhancements.	143

Chapter 11. IMS Transaction Manager enhancements 145

	APPC flood control enhancement.	145
	IMS Connect support for cascading global transactions to IMS TM on different z/OS images	146
	ISC VTAM enhancement for ERP messages	149
	Java environment dynamic scripting enhancement	150
	JVM 64-bit support in Java dependent regions	152
	JVM usage statistics	154
	Mobile workload reporting enhancement	154
	MSC enhancements	156
	MSC 24-bit storage reduction	156
	MSC enablement enhancements	159
	MSC type-2 command enhancements	161
	Dynamic definition of MSC resources	162
	IMSRSC repository support for dynamically defined MSC resources	166
	OTMA enhancements.	169
	Conversation ID enhancement for QUERY	
	OTMATIC command	169
	Dynamic storage for OTMA descriptors	169
	OTMA dynamic SAPs enhancement	172
	OTMA security enhancements.	173
	OTMA tpipes parallelism enhancement	175
	Shared queues buffer pool enhancement	179
	Shared queues overflow feedback enhancement	179
	Synchronous callout enhancements	180
	ICAL call enhancement to support control data for synchronous callout requests	180
	Trace enhancement for synchronous callout processing	181

Chapter 12. IMS system enhancements 183

	Command enhancements overview	183
	Dynamic refresh enhancement for PWFI and WFI regions	189
	External Subsystem Attach Facility enhancements	191
	ESAF Associate Thread exit routine enhancement	191
	ESAF subsystem type enhancement	191
	ESAF connection pooling	193
	IMS 64-bit storage manager enhancement	193
	IMS Connect enhancements	194
	IMS Connect command enhancements	194
	Recording IMS Connect RACF statistics for IMS DB connections.	196
	Recording IMS Connect RACF statistics for IMS TM connections	197
	Timeout option for idle connections	198
	IMS Connect WTOR command interface option enhancement	199

IMSRSC repository enhancements	199
Automatic export to the IMSRSC repository at IMS checkpoints	200
Create RDDS from Log Records utility (DFSURCL0) enhancement	202
DELETE DEFN command enhancement for IMS IDs	203
QUERY command enhancement for unexported resources	203
IMS type-2 trace enhancement	204
Reduced total cost of ownership enhancements	205
User exit enhancements	206
IMS Monitor (IMSMON) user exit enhancement	206
Refreshable type-2 Automated Operator exit routine	207
ACEE creation and management enhancement	209

Part 3. IBM IMS Tools support for IMS 14 211

Chapter 13. IBM IMS Tools Solution Packs for z/OS 213

IBM IMS Database Solution Pack for z/OS, V2.1	213
IBM IMS Fast Path Solution Pack for z/OS, V1.3	213
IBM IMS Performance Solution Pack for z/OS, V1.3	214
IBM IMS Recovery Solution Pack for z/OS, V2.1	214
IBM Tools Base for z/OS, V1.5	214

| Chapter 14. IBM Management Console for IMS and Db2 for z/OS, V1.1 217

Chapter 15. IMS Application Management tools 219

IBM IMS Batch Terminal Simulator for z/OS, V4.1	219
IBM IMS Program Restart Facility for z/OS, V2.2	219

Chapter 16. IMS Backup and Recovery Management tools 221

IBM IMS DEDB Fast Recovery for z/OS, V2.2	221
IBM IMS High Performance Image Copy for z/OS, V4.2	221
IBM IMS Index Builder for z/OS, V3.1	221
IBM IMS Recovery Expert for z/OS, V2.2	222
IBM IMS Recovery Solution Pack for z/OS, V2.1	222

Chapter 17. IMS Database Administration tools 223

IBM IMS Sequential Randomizer Generator for OS/390, V1.1	223
IBM IMS Database Solution Pack for z/OS, V2.1	223

Chapter 18. IMS Data Replication tools 225

IMS Cloning Tool for z/OS, V1.2	225
InfoSphere Data Replication for IMS for z/OS, V11.3	225

Chapter 19. InfoSphere IMS tools 227

IBM InfoSphere Classic Change Data Capture for z/OS, V11.3	227
IBM InfoSphere Classic Federation Server for z/OS, V11.3	227
IBM InfoSphere Optim Test Data Management Solution for z/OS, 11.3	227
InfoSphere Data Replication for IMS for z/OS, V11.3	228

Chapter 20. IMS Performance Management tools 229

IBM IMS Buffer Pool Analyzer for z/OS, V1.4	229
IBM IMS Connect Extensions for z/OS, V2.4	229
IBM IMS Network Compression Facility for z/OS, V1.2	229
IBM IMS Performance Analyzer for z/OS, V4.4	230
IMS Problem Investigator for z/OS, V2.4	230
IBM Transaction Analysis Workbench, V1.3	231
IBM IMS Performance Solution Pack for z/OS, V1.3	231

Chapter 21. IMS Regulatory Compliance tools. 233

IBM InfoSphere Guardium Data Encryption for Db2 and IMS Databases, V1.2	233
IBM Security Guardium S-TAP for IMS.	233

Chapter 22. IMS System management tools 235

IBM IMS Command Control Facility for z/OS, V2.2	235
IBM IMS Extended Terminal Option Support for z/OS, V3.2	235
IBM IMS High Performance System Generation (SYSGEN) Tools for z/OS, V2.3	235
IBM IMS Workload Router for z/OS, V2.7.	236

Chapter 23. Tivoli IMS tools 237

Tivoli Decision Support for z/OS V1.8 or later	237
Tivoli Monitoring, V6.3.0	237
Tivoli AF/OPERATOR on z/OS V3.4.1	237
Tivoli AF/Integrated Resource Manager, V5.0.0	237
Tivoli OMEGACENTER Gateway on z/OS V3.4.1	237
Tivoli OMEGAMON XE for Db2 Performance Expert on z/OS, V5.2 or later	238
Tivoli OMEGAMON XE for Db2 Performance Monitor on z/OS, V5.2 or later	238
Tivoli OMEGAMON XE for IMS on z/OS, V5.3	238
Tivoli System Automation for z/OS, V3.5	238

Chapter 24. IMS TM management tools 239

IBM IMS Configuration Manager for z/OS, V2.2	239
IBM IMS Queue Control Facility for z/OS, V3.2	239
IBM IMS Sysplex Manager for z/OS, V1.3.	240

Chapter 25. IMS Utilities Management tools	241
IMS Cloning Tool for z/OS, V1.2	241
IBM IMS Database Control Suite for z/OS, V3.2	241
IBM IMS Database Reorganization Expert for z/OS, V4.1	242
IBM IMS High Performance Image Copy for z/OS, V4.2	242
IBM IMS High Performance Load for z/OS, V2.1	242
IBM IMS High Performance Unload for z/OS, V1.2	243
IBM IMS High Performance Prefix Resolution for z/OS, V3.1	243
IBM IMS High Performance Pointer Checker for z/OS, V3.1	243
IBM IMS Index Builder for z/OS, V3.1	244
IBM IMS Library Integrity Utilities for z/OS, V2.2	244
IBM IMS Database Solution Pack for z/OS, V2.1	245
IBM IMS Fast Path Solution Pack for z/OS, V1.3	245

Chapter 26. Miscellaneous IBM tools that support IMS 247

Part 4. Featured integration solutions for IMS 14 251

Chapter 27. IBM IMS Enterprise Suite for z/OS. 253

Chapter 28. Mobile and REST API solution with IBM z/OS Connect Enterprise Edition 257

Chapter 29. IBM Cognos Business Intelligence solution 259

Chapter 30. Data discovery with IBM Watson Explorer 261

Chapter 31. Integrated access with IBM DataPower Gateway 263

Part 5. Appendixes 265

Notices	267
Trademarks	269
Terms and conditions for product documentation	269
IBM Online Privacy Statement	270

Bibliography. 271

Index X-1

About this information

These topics provide general information to help you evaluate and plan for IMS™ 14. The topics describe the new features and enhancements for IMS 14, the hardware and software requirements for these new features and enhancements, considerations for migration and coexistence for IMS 14, and an overview of the IMS Tools that are enhanced to work with IMS 14.

This information is available in IBM® Knowledge Center.

For information about IMS Version 13, including its enhancements and considerations for migration and coexistence, see *IMS Version 13 Release Planning*, (GC19-3658).

When you install IMS, you will also need information from the following information units:

- *IMS Version 14 Installation*
- *IMS Version 14 System Definition*
- *IMS 14: Program Directory for Information Management System Transaction and Database Servers*

How to use the Release Planning information

Use the IMS 14 Release Planning information to learn about the new enhancements in IMS 14 and to plan your migration.

The Release Planning information is organized into the following parts:

- Part 1, “General planning information for IMS 14,” on page 1, which contains general information about the new release, including the following information:
 - A summary list of the enhancements included in IMS 14
 - An overview of the hardware and software requirements
 - Migration considerations for IMS 14
 - Coexistence considerations between the supported versions of IMS
 - A listing of the changed, new, and deleted messages and abends for IMS 14
 - A listing of the new and changed log records for IMS 14

- Part 2, “IMS Version 14 enhancements,” on page 107, which describes each new enhancement in IMS 14.

In the IBM Knowledge Center, the description of each enhancement includes links to all of the topics that are new or changed for the enhancement. On the web, see *IMS Version 14 enhancements*.

- Part 3, “IBM IMS Tools support for IMS 14,” on page 211, which describes IBM IMS Tools and products that support IMS 14.
- Part 4, “Featured integration solutions for IMS 14,” on page 251, which contains overviews of some of the new ways that you can further integrate IMS 14 and the rest of your IT architecture.

Note: If you are migrating to IMS 14 from IMS Version 12 or earlier, you must refer to the *IMS Release Planning Guide* for each IMS release that you are skipping for complete information about the enhancements, the software and hardware requirements, and the migration and coexistence considerations that are specific to that release.

Prerequisite knowledge

Before using this information, you should have knowledge of either the IMS Database Manager (DB) or the IMS Transaction Manager (TM). You should also understand basic z/OS® and IMS concepts, your installation's IMS system, and have a general knowledge of the tasks involved in project planning.

You can learn more about z/OS by visiting the “z/OS basic skills” topics in IBM Knowledge Center.

You can gain an understanding of basic IMS concepts by reading *An Introduction to IMS*, an IBM Press publication.

IBM offers a wide variety of classroom and self-study courses to help you learn IMS. For a complete list of courses available, go to the IBM Skills Gateway and search for IMS.

How new and changed information is identified

New and changed information in most IMS library PDF publications is denoted by a character (revision marker) in the left margin. The first edition (-00) of *Release Planning*, as well as the *Program Directory* and *Licensed Program Specifications*, do not include revision markers.

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.

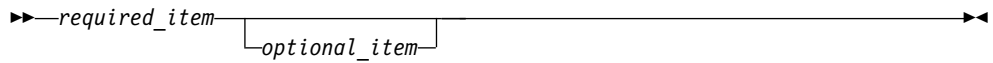
How to read syntax diagrams

The following rules apply to the syntax diagrams that are used in this information:

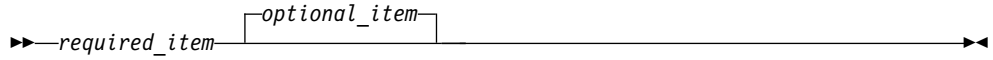
- Read the syntax diagrams from left to right, from top to bottom, following the path of the line. The following conventions are used:
 - The >>--- symbol indicates the beginning of a syntax diagram.
 - The ---> symbol indicates that the syntax diagram is continued on the next line.
 - The >--- symbol indicates that a syntax diagram is continued from the previous line.
 - The --->< symbol indicates the end of a syntax diagram.
- Required items appear on the horizontal line (the main path).

▶▶—*required_item*————▶▶

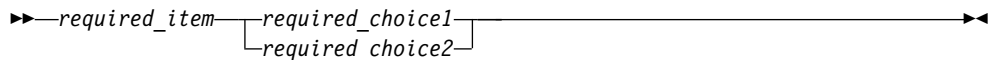
- Optional items appear below the main path.



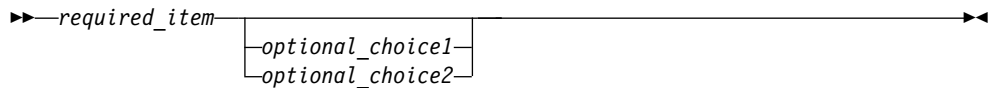
If an optional item appears above the main path, that item has no effect on the execution of the syntax element and is used only for readability.



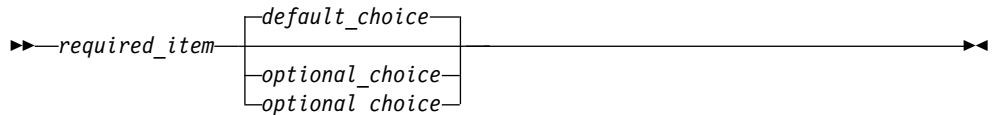
- If you can choose from two or more items, they appear vertically, in a stack. If you *must* choose one of the items, one item of the stack appears on the main path.



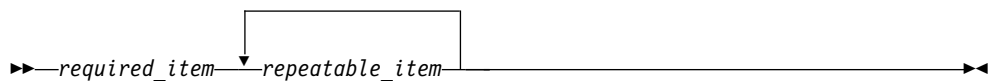
If choosing one of the items is optional, the entire stack appears below the main path.



If one of the items is the default, it appears above the main path, and the remaining choices are shown below.



- An arrow returning to the left, above the main line, indicates an item that can be repeated.

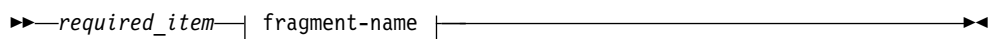


If the repeat arrow contains a comma, you must separate repeated items with a comma.



A repeat arrow above a stack indicates that you can repeat the items in the stack.

- Sometimes a diagram must be split into fragments. The syntax fragment is shown separately from the main syntax diagram, but the contents of the fragment should be read as if they are on the main path of the diagram.



fragment-name:

- In IMS, a b symbol indicates one blank position.
- Keywords, and their minimum abbreviations if applicable, appear in uppercase. They must be spelled exactly as shown. Variables appear in all lowercase italic letters (for example, *column-name*). They represent user-supplied names or values.
- Separate keywords and parameters by at least one space if no intervening punctuation is shown in the diagram.
- Enter punctuation marks, parentheses, arithmetic operators, and other symbols, exactly as shown in the diagram.
- Footnotes are shown by a number in parentheses, for example (1).

Accessibility features for IMS 14

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

Accessibility features

The following list includes the major accessibility features in z/OS products, including IMS 14. These features support:

- Keyboard-only operation.
- Interfaces that are commonly used by screen readers and screen magnifiers.
- Customization of display attributes such as color, contrast, and font size.

Keyboard navigation

You can access IMS 14 ISPF panel functions by using a keyboard or keyboard shortcut keys.

For information about navigating the IMS 14 ISPF panels using TSO/E or ISPF, refer to the *z/OS TSO/E Primer*, the *z/OS TSO/E User's Guide*, and the *z/OS ISPF User's Guide Volume 1*. These guides describe how to navigate each interface, including the use of keyboard shortcuts or function keys (PF keys). Each guide includes the default settings for the PF keys and explains how to modify their functions.

Related accessibility information

Online documentation for IMS 14 is available in IBM Knowledge Center.

IBM and accessibility

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

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 information, you can take one of the following actions:

- Click the **Contact Us** tab at the bottom of any IBM Knowledge Center topic.
- Send an email to imspubs@us.ibm.com. Be sure to include the book title and the publication number.

To help us respond quickly and accurately, please 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.

Part 1. General planning information for IMS 14

In addition to the new functions and enhancements that are available, IMS 14 has new hardware and software requirements, as well as new considerations for migration and coexistence between IMS systems.

These topics describe how the new IMS 14 functions might affect your installation, list new, changed, and deleted messages and codes, and describe the IBM IMS Tools that support IMS 14.

Chapter 1. Hardware requirements

IMS 14 has base hardware requirements. Some individual functions have additional hardware requirements.

Processor requirements

IMS 14 runs only in z/Architecture[®] mode on an IBM System z9[®] processor or later.

The following table lists the processors supported by IMS 14.

Table 1. Supported IBM processors for IMS 14

Machine name	Machine type
IBM System z9 Enterprise Class (formerly z9 [®] 109)	2094
IBM System z9 Business Class (z9 BC)	2096
IBM System z10 [®] Enterprise Class (z10 EC)	2097
IBM System z10 Business Class (z10 [™] BC)	2098
IBM zEnterprise [®] 196 (z196)	2817
IBM zEnterprise 114 (z114)	2818
IBM zEnterprise EC12 (zEC12)	2827
IBM zEnterprise BC12 (zBC12)	2828
IBM z13 [®]	2964
IBM z13s	2965
IBM z14	3906

For more information about IBM System z[®] processors, see System z Hardware at www.ibm.com/systems/z/hardware/.

System console requirements

The console requirements of z/OS Version 2 Release 1 or later apply.

Tape unit requirements

IMS supports IBM 3590 and later tape units (or equivalent products) for installation and maintenance. IMS supports tape block sizes greater than 32760 bytes for the output of the Database Image Copy utility (DFSUDMP0) and the Online Database Image Copy utility (DFSUICP0).

Coupling facility requirements

IMS 14 has the following coupling facility requirements.

A coupling facility level of 14 or later is required for the following IMS 14 functions:

- Operations Manager (OM) Audit Trail, if a coupling facility log stream is used

- Parallel RECON access support
- Repository Server Audit Log, if a coupling facility log stream is used
- Resource Manager (RM), if a resource structure is used
- Shared-EMH support
- Shared queues
- Sysplex data sharing (including data caching and VSO data sharing) with Internal Resource Lock Manager (IRLM) V2.3

For more information about coupling facility levels, see <http://www.ibm.com/systems/z/advantages/pso/cftable.html>.

DASD requirements

IMS 14 hardware requirements include several requirements for DASD.

During the binding of the IMS control blocks load modules (specifically during the bind of the IMS VTAM[®] control blocks load monitoring module), both the binder work data set SYSUT1 and IMS data sets IMS.SDFSRESL and IMS.SDFSJLIB must reside on a device that supports a record size of 18 KB or greater. For all other system libraries and working storage space, any device that is supported by the operating system is allowed.

For IMS database storage, any device that is supported by the operating system is allowed within the capabilities and restrictions of Basic Sequential Access Method (BSAM), Queued Sequential Access Method (QSAM), Overflow Sequential Access Method (OSAM), and Virtual Storage Access Method (VSAM).

You must preallocate and format the write-ahead data set (WADS) on a DASD device that supports Extended Count-Key-Data (ECKD[™]) architecture.

The fast replication function of the Database Image Copy 2 utility (DFSUDMT0) requires DASD controllers that support one of the following features:

- The concurrent-copy feature of DFSMS
- The FlashCopy[®] feature of the IBM Enterprise Storage Server[®] (ESS)
- The SnapShot feature of the IBM RAMAC Virtual Array (RVA) storage system

FlashCopy and SnapShot might require microcode from IBM to activate their functionality. Also, the source and target data sets (databases and image copies) must reside on the same ESS or RVA hardware.

The DASD storage requirements for the following items are described in the *Program Directory for Information Management System Transaction and Database Servers V14.01.00*:

- SMP/E system entries
- SMP/E data sets
- Target libraries
- Distribution libraries
- Install process
- Optional machine-readable material

The following types of data sets can be allocated in the extended addressing space (EAS) of an extended address volume (EAV):

- GSAM database data sets
- BPE external trace data sets
- OSAM database data sets
- VSAM database data sets
- Online log data sets (OLDSs), including large OLDS (for example, greater than 64 KB tracks)
- Write ahead data sets (WADSs)
- Restart data sets (RDSs)
- Message queue blocks data sets
- Long and short message data sets
- Terminal devices with UNITYPE = SPOOL or DISK
- RESLIB data sets (IMS.SDFSRESL)
- MODBLKS data sets for online change (IMS.MODBLKSA and IMS.MODBLKSB)
- Application control block library (ACBLIB) data sets
- DBRC RECON data sets (non-PRA)
- Database Image Copy utility (DFSUDMP0) data sets
- Database Image Copy 2 utility (DFSUDMT0) data sets
- Database Change Accumulation utility (DFSUCUM0) data sets
- Local online change data sets (IMS.MODSTAT)
- Global online change data sets (IMS.OLCSTAT)
- Partitioned data set extended (PDSE) data sets (IMS.SDFSJLIB, PGMLIB, SMPLTS, and External Subsystem Attach Facility (ESAF) load libraries)
- Time-controlled operations (TCO) data sets
- System log data sets (SLDSs)
- Recovery log data sets (RLDSs)
- HALDB Indirect List data sets (ILDSs)
- IMS Repository data sets
- MFS map library data sets produced by the MFS Language and Service utilities (IMS.FORMAT)
- IMS Trace facility external trace data sets
- IMS Monitor output data sets

Large sequential data set support hardware requirements

To take advantage of this support, hardware that has more than 65,535 tracks must be used.

Multiple Systems Coupling hardware requirements

When the physical link is channel-to-channel (CTC) and is dedicated to IMS, Multiple Systems Coupling (MSC) requires the System/370 CTC adapter or a logical channel on the IBM 3088, ESCON, or Fiber Channel connection (FICON®). MSC FICON CTC support requires that at least one IMS system be installed on an IBM zSeries machine with the FICON channel and FICON CTC microcode. The other side (IMS) can be any processor with a FICON channel.

Parallel RECON access hardware requirements

The parallel RECON access function requires a Parallel Sysplex[®] environment and DFSMS Transactional VSAM Services (DFSMSStvs). Therefore, parallel RECON access requires Coupling Facility (CF) hardware in the System z sysplex.

Terminals and equipment supported by IMS 14

IMS 14 supports SLU, LU, NTO, 3270, and Finance (3600) terminals, as well as other equipment such as printers and DASD devices.

The following tables list the terminals and other equipment supported by IMS 14.

In the table, the following abbreviations are used:

DSC	Data Stream Compatibility
ISC	Intersystem Communication
LU	Logical Unit
NTO	Network Terminal Option
PC	Personal Computer
PP	Program Product
SLU	Secondary Logical Unit
TTY	Teletypewriter equipment
VTAM	Virtual Telecommunications Access Method

Table 2. Terminals that are supported by IMS 14

Compatible product	SNA	Notes[®]
SLU 1 (for example, 3230, 3232, 3262, 3287, 3767, 3268, 3770, 3770P, 3790 (type 2 batch and bulk print), 4700, 5280, 5550, S/32, S/34, S/38, 8100)	VTAM	1, 2
SLU 2 (for example, 3179, 3180, 3276, 3278, 3279, 3290, 3790 (3270 DSC feature), 3600 Admin PP, 4700, 5280, 5520, 5550, 8100, 8775, S/34, Display writer)	VTAM	1, 2
SLU P (for example, 3600, 3630, 3650, 3680, 3770PC, 3790, 4700, 4730, 4736, 5520, 8100, S/34, Series/1)	VTAM	1, 2, 3, 4
LU 6.1 (ISC)	VTAM	1, 2
LU 6.2 (APPC)	VTAM	2
NTO (for example, 33/35, TTY, 2740, 2741, 3101, 3232, 3767, S/23)	VTAM	1, 2

Notes:

1. The IMS Message Format Service (MFS) is available for this device. MFS editing can be bypassed on a message-by-message basis.
2. IMS Fast Path supports the use of compatible terminals.
3. Although IMS provides sample code for this terminal, additional user coding is required.

4. IMS provides no device-resident code for this device. Additional user coding is required to attach it to IMS.

Table 3. Terminals that are supported by IMS 14, but withdrawn from IBM Marketing

IMS-supported product	Compatible product	Switched mode	Polled mode	Local mode	SNA	Notes
3270		VTAM	VTAM	VTAM	VTAM	1, 2
Finance (3600)	4700				VTAM	1, 2, 3

Notes:

1. The IMS Message Format Service (MFS) is available for this device. MFS editing can be bypassed on a message-by-message basis.
2. IMS Fast Path supports the use of compatible terminals.
3. Although IMS provides sample code for this terminal, additional user coding is required.

For the following table, refer to operating system descriptions for specific device types.

Table 4. Other equipment that is supported by IMS 14

IMS-supported product	Access
System console	MVS™ write-to-operator/write-to-operator-with-reply (WTO/WTOR)
Spool device	Basic Sequential Access Method (BSAM)
Card reader	BSAM
Printer	BSAM
Magnetic tape	BSAM
DASD devices	BSAM

Restriction: IMS 14 does not support BTAM devices (2740-1, 2740-2, 2741, 2780, System/3, and System/7).

Remote Site Recovery hardware requirements

For basic Remote Site Recovery (RSR) functions, you need a Sysplex Timer, a high bandwidth control unit, and at least one tape unit.

RSR requires:

- A Sysplex Timer (if either data sharing or workload is spread across multiple CPCs)
- A high bandwidth control unit (such as a 3172)
- At least one tape unit at the tracking site

Coordinated Disaster Recovery support for IMS and DB2® requires that the DB2 logs reside on devices that support Extended Remote Copy (XRC).

zIIP utilization hardware requirements

| One or more IBM System z Integrated Information Processors (zIIPs) must be
| online on the machine at the time an IMS Connect, IMS ODBM, or IMS CQS
| address space is started in order to have any threads zIIP eligible and executed on
| a zIIP for that execution instance.

If no zIIPs are online when the address space is started, no work will be moved to a zIIP.

Chapter 2. Software requirements

IMS 14 has base software requirements. Some individual functions have additional software requirements.

Operating software requirements

IMS 14 and its various functions have specific operating software requirements.

Before you install IMS 14, check with your IBM Support Center or check either Information/Access or Service Link for additional preventive service planning (PSP) information that you need to be aware of. The PSP upgrade name for IMS 14 is IMS1400.

The z/OS service levels that are required for installation and execution are described in the *Program Directory for Information Management System Transaction and Database Servers V14.01.00*.

IMS 14 base software requirements

The base IMS 14 system runs on z/OS Version 2 Release 1 or later. Certain features and functions have additional software requirements.

IMS 14 requires the following minimum version, release, or modification levels (as long as those versions remain available and supported by IBM):

- z/OS Version 2 Release 1 (5650-ZOS) or later
 - The use of zEnterprise Data Compression services by the IMS Database Image Copy utility (DFSUDMT0) is dependent on the installation of DFSMSdss APAR OA42238 (PTF UA74782) in z/OS 2.1.
- IRLM Version 2.3 or later (5635-A04), if data sharing is used. IRLM Version 2.3 is delivered with IMS 14.

When using multiple IMS systems:

- On the same z/OS system, you need only one IRLM.
- Of different release levels on the same z/OS system, you can have one IRLM or you can use two or more IRLM address spaces. If two or more IMS systems share data and are running on the same z/OS system, they should use the same IRLM.
- On different z/OS systems for inter-processor block-level data sharing, you must have one IRLM on each z/OS system.

IMS 14 also operates in a virtual machine (VM) under control of z/OS. This environment is intended for use in a program development, testing, and non-XRF production environment.

Restrictions: The VM environment has the following restrictions:

- The Log Analysis utilities might yield inaccurate time-stamp results.
- If you operate the IMS 14 Transaction Manager under VM for production purposes and have specific throughput or terminal response-time requirements, plan to benchmark under VM to ensure that the proposed configuration meets your performance needs.

System-Managed CF Structure Duplexing is recommended, though not required, for the Resource Manager resource structure.

Coordinated Disaster Recovery support for IMS and DB2 requires the IMS 14 Remote Site Recovery (RSR) Recovery Level Tracking (RLT) feature.

Coexistence APARs

Certain functions of IMS 14 can coexist with IMS Version 12 and IMS Version 13 with the appropriate coexistence APARs applied.

See “Overview of coexistence APARs” on page 77 for an overview of the APARs that are needed for coexistence purposes.

Database Resource Adapter (DRA) software requirements

The version of the IMS DRA modules that are used by a DRA client must be the same version as the IMS with which the DRA client is communicating.

Recommendations:

- Concatenate the IMS.SDFSRESL library to the DRA client step library so the correct version of the DRA Startup/Router routine (DFSPRRC0) is loaded into the DRA client load library.
- Ensure that the DRA Startup Table (DFSPZPxx) points to the correct version of IMS.SDFSRESL.

Data sharing software requirements

For block-level data sharing, IRLM Version 2.3 or later is required. The IRLM is an independent component that is shipped with IMS 14. The IRLM must be defined as a z/OS subsystem. Block-level data sharing of databases is supported between all in-service levels of IMS.

HALDB Index/ILDS Rebuild utility free space function software requirements

The HALDB Index/ILDS Rebuild utility (DFSPREC0) requires four 2 GB data spaces to store and sort the rebuilt indirect list entries (ILEs) before reloading them into the ILDS.

IMS callout function software requirements

To support the IMS callout function, OTMA must be enabled in IMS and IMS Connect configured for callout support.

Also, one of the following components that is external to IMS is required:

- The IMS Enterprise Suite SOAP Gateway
- The IMS TM Resource Adapter
- An IBM WebSphere® DataPower® appliance
- A user-written IMS Connect client (TCP/IP application)

The synchronous program switch functionality, which was added in IMS Version 13, is a variation of the callout function. However, even though IMS processes synchronous program switch requests as OTMA transactions, the synchronous

program switch functionality does not require you to enable OTMA, IMS Connect, or any of the external software components in the preceding list.

Related concepts:

[🔗](#) IMS Enterprise Suite SOAP Gateway overview

[🔗](#) IMS TM Resource Adapter overview

IMS catalog function software requirements

The IMS catalog function has software requirements.

In IMS 14, if the IMS management of ACBs is enabled by ACBMGMT=CATALOG in the <CATALOG> section of the DFSDFxxx PROCLIB member, z/OS 2.1 APAR OA45400 for DFSMS must be applied via PTF UA73855.

Distributed environments that use the IMS catalog function can use the IMS Enterprise Suite Explorer for Development. The IMS catalog function also supports the use of the IMS Universal drivers, IMS Data Provider for Microsoft .NET, traditional IMS database query techniques, and batch processing in the z/OS environment.

IMS management of ACBs function software requirements

The IMS management of ACBs function has software requirements.

The Operations Manager (OM) and the Structured Call Interface (SCI) components of the IMS Common Service Layer (CSL) are required to add new or changed ACBs the IMS directory by using the IMPORT DEFN SOURCE(CATALOG) command.

If the IMS catalog is shared, the Resource Manager component of CSL is required.

In IMS 14, if you are running IMS on z/OS 2.1, the IMS management of ACBs function requires z/OS 2.1 DFSMS APAR OA45400 applied via PTF UA73855.

ISC TCP/IP software requirements

Software requirements for the IMS 14 ISC TCP/IP function include requirements for CICS®, ETO, IMS Connect, and a minimal CSL.

The software requirements for the IMS 14 ISC TCP/IP function are:

- IBM CICS Transaction Server for z/OS, Version 5.1 (or later) must be used.
- IMS Connect is required to provide TCP/IP socket connection support for IMS.
- The Common Service Layer (CSL) with at least the Structured Call Interface (SCI) and the Operations Manager (OM) is required. SCI is required for communications between IMS and IMS Connect and OM is required for type-2 command support.
- A single point of control (SPOC) program, such as the IMS TSO SPOC, must be used to issue type-2 commands to the OM API or REXX SPOC API.
- For each IMS subsystem that uses dynamically defined terminals with ISC TCP/IP, the IMS Extended Terminal Option (ETO) is required.

IMS Connect software requirements

IMS Connect has software requirements in addition to those of the base IMS product.

The software requirements for IMS Connect include:

- z/OS Communications Server V2R1.0 or later (TCP/IP).
- To implement security, z/OS Security Server RACF® or an equivalent product.
- To use the local option for client communications, there are additional software requirements. See IMS Transaction Manager Resource Adapter overview (Application Programming).
- To use Secure Sockets Layer (SSL), z/OS System SSL, a sub-component of z/OS Cryptographic System Services, is required. For information about z/OS encryption support available with the z/OS Cryptographic System Services SSL module, see *z/OS Cryptographic Services System SSL Programming (SC14-7495)*.

IMS 14 is the last version of IMS to support the IMS Connect SSL function.

Customers using this function should migrate to using IBM z/OS Communications Server Application Transparent Transport Layer Security (AT-TLS) to set up Secure Socket Layer (SSL) on TCP/IP connections to IMS Connect.

- To support the IMS Universal drivers or a user-written DRDA source server, an IMS Common Service Layer is required, including the Open Database Manager, the Operations Manager, and the Structured Call Interface.
- IMS Connect must have z/OS UNIX System Services superuser privileges, to ensure that IMS Connect can open ports.

IMS Connect XML Adapter support

The IMS Connect XML Adapter support in IMS 14, used with the IMS Enterprise Suite SOAP Gateway, requires IBM Developer for System z . Certain functions of the IMS Enterprise Suite SOAP Gateway might have additional software requirements.

Java application program support in IMS 14

Java™ applications that run in or access IMS 14 must meet specific software requirements.

Software requirements for Java application programs that use the IMS Universal drivers or JDR resource adapter

The IMS Universal drivers that Java application programs can use to access IMS data have software requirements.

IMS 14 requires the following software:

- z/OS UNIX System Services available at run time.
- Hierarchic File System (HFS) or zFS. For information on preparing HFS, see *z/OS UNIX System Services File System Interface Reference*.

In IMS 14, the IMS Universal drivers provide the IMS Java drivers and database resource adapters.

Note: The IMS Java dependent region (JDR) resource adapter reuses some of the interfaces and classes in the IMS Universal drivers. As a result, it is packaged as

part of the IMS Universal drivers starting IMS 14 APAR PI62871 (PTF UI40491) and has the same software requirements as the IMS Universal drivers.

The IMS Universal drivers have the following runtime software requirements:

- IBM SDK, Java Technology Edition, Version 7.0.1 or later (31-bit or 64-bit)
- One or more of the following conditional requirements:
 - For CICS applications, IBM CICS Transaction Server for z/OS Version 5.1 (5655-Y04) or later, as determined by the JDK version
 - For Db2[®] stored procedures:
 - Db2 11 for z/OS (5615-DB2) or later
 - Db2 10 for z/OS (5605-DB2)
 - For WebSphere applications, WebSphere Application Server for z/OS (5655-W65) or WebSphere Application Server for distributed platforms (5724-J08), Version 8.5 or later, as determined by the supported JDK level.
- RACF or an equivalent product
- The software requirements for the JDR resource adapter are the same as for the IMS Universal drivers.

Java application programs that use the IMS Universal drivers also require a way to generate the IMS database metadata, such as using the IMS Enterprise Suite Explorer for Development.

Note: Use binary-mode FTP to transfer the IMS Universal drivers JAR or RAR files to another system.

JAR and RAR files for type-4 connectivity

The following table describes the JAR and RAR files that provide type-4 connectivity for the IMS Universal drivers:

Table 5. Type-4 connectivity JAR and RAR files for Java applications that use the IMS Universal drivers

Driver	JAR or RAR file
IBM IMS Universal DL/I driver	<i>pathprefix</i> /usr/lpp/ims/ims14/imsjava/imsudb.jar
IBM IMS Universal JDBC driver	<i>pathprefix</i> /usr/lpp/ims/ims14/imsjava/imsudb.jar
IBM IMS Universal Database resource adapter	For use within WebSphere Application Server (both z/OS and distributed platforms): <i>pathprefix</i> /usr/lpp/ims/ims14/imsjava/rar/imsudbLocal.rar for JCA/JDBC local transaction processing only <i>pathprefix</i> /usr/lpp/ims/ims14/imsjava/rar/imsudbXA.rar for JCA/JDBC two-phase (XA) commit processing or local transaction processing <i>pathprefix</i> /usr/lpp/ims/ims14/imsjava/rar/imsudbLocal.rar for CCI local transaction support <i>pathprefix</i> /usr/lpp/ims/ims14/imsjava/rar/imsudbXA.rar for CCI two-phase commit (XA) transaction support

JAR and RAR files for type-2 connectivity

The following table describes the JAR and RAR files that provide type-2 connectivity for the IMS Universal drivers:

Table 6. Type-2 connectivity JAR and RAR files for Java applications that use the IMS Universal drivers

Driver	JAR file
IMS Universal DL/I driver	<i>pathprefix</i> /usr/lpp/ims/ims14/imsjava/imsudb.jar
IMS Universal JDBC driver	<i>pathprefix</i> /usr/lpp/ims/ims14/imsjava/imsudb.jar
IMS Universal Database resource adapter	For use within WebSphere Application Server (both z/OS and distributed platforms): <i>pathprefix</i> /usr/lpp/ims/ims14/imsjava/rar/imsudbLocal.rar for CCI programming interface to perform SQL or DL/I data operations <i>pathprefix</i> /usr/lpp/ims/ims14/imsjava/rar/imsudbJLocal.rar for JDBC programming interface to perform SQL data operations

When DriverType=2:

- The transaction scope is local (a unit of work is scoped to a particular connection). Multiple connections can have independent units of work associated with each connection.
- Application programs can issue local commit and rollback calls through either the JDBC Connection interface or the CCI LocalTransaction interface.
- ContainerManaged beans are supported, but require the following properties to be set in the EJB Deployment Descriptor:
 - In the Bean tab, specify the following properties under the LocalTransaction heading:
 - Boundary = BeanMethod
 - Resolver = ContainerAtBoundary
 - Unresolved action = Rollback
 - In the Assembly tab, set the transaction scope to NotSupported.

When DriverType=2_CTX:

- Specifies a global scope transaction model in which a unit of work can span multiple bean methods. RRS-managed transaction applications use this driver type. The container coordinates commit and rollback processing through RRS.
- Application programs can use the UserTransaction interface for explicit commit and rollback calls.

Software requirements for Java applications that access IMS transactions

Java applications that access IMS transactions must meet specific software requirements.

- Java programs that run in Java message processing (JMP) and Java batch processing (JBP) regions require IBM SDK for z/OS, Java Technology Edition, Version 7.0 or later.
- For programs that access transactions using the IMS TM Resource Adapter, see supported versions and software configurations.

The JAR file `imsudb.jar` is needed to support JMP and JBP regions.

Note: IMS 14 APAR PI62871 (PTF UI40491) is required. Because IMS Java dependent region (JDR) resource adapter reuses some of the interfaces and classes in the IMS Universal drivers, the classes and interfaces are packaged together as one .jar file, `imsudb.jar`. Previously, the `imsutm.jar` for the JDR resource adapter was packaged separately.

Open Database solution software requirements

To use the Open Database solution, IMS must be configured as an IMSplex and IMS Connect is required.

The Open Database solution requires IMS Connect, as well as the following Common Service Layer (CSL) components:

- Operations Manager (OM)
- Structured Call Interface (SCI)
- Open Database Manager (ODBM)

Important: Open Database Manager (ODBM) can only connect to the IMS systems that are of the same version as ODBM itself. In a mixed-version IMSplex, to limit ODBM connection to the IMS systems of the same version, list the eligible IMS systems as data stores in the `CSLDCxxx` member of the IMS PROCLIB data set.

Parallel RECON access software requirements

To use the parallel RECON access function of Database Recovery Control (DBRC), you must configure IMS as an IMSplex and install DFSMS Transactional VSAM Services (DFSMSStvs), a separately orderable feature of z/OS.

SQL support software requirements

For IMS to process SQL calls in the native host environment, COBOL Version 5 with IMS coprocessor support is required. With COBOL Version 5, all load modules must reside within a partitioned data set extended (PDSE). The IMS catalog must be enabled for this SQL support.


User exit enhancements software requirements

Exits to be queried or refreshed using type-2 commands must first be defined in the `<USER_EXITS>` section of the `DFSDFxxx` member of the IMS PROCLIB data set.


Some user exits are passed a standard user exit parameter list (SXPL), mapped by macro `DFSSXPL`. The SXPL contains a version number that can be used to identify what fields are present in the parameter list. If your user exit accesses a field that was added at a specific version of the parameter list beyond the base level for an IMS release, you should test the SXPL version number to ensure that the parameter list you were passed is at the correct version or higher before using the field.

In IMS 14, some older user exits are always passed a version 1 SXPL. All other user exits that are passed an SXPL receive a version 6 or later SXPL.

Related concepts:

 Refreshable exit routine types (Exit Routines)

Related reference:

 IMS standard user exit parameter list (Exit Routines)

IMS Enterprise Suite software requirements

IMS Enterprise Suite provides APIs, tools, and a web service solution for facilitating application development and extending access to IMS transactions and data.

IMS 14 can be used with the following versions of IMS Enterprise Suite, although some components or functions might have specific IMS requirements.

- IMS Enterprise Suite V3.2
- IMS Enterprise Suite V3.1

Related information:

IMS Enterprise Suite V3.2 overview

CICS subsystems supported

IBM CICS Transaction Server for z/OS can connect to IMS if minimum version requirements are met. Certain IMS 14 functions might include additional version requirements for CICS.

CICS Transaction Server for z/OS Version 4.1 (5655-S97) or later can connect to either the IMS 14 Database Manager (DB) or, using the appropriate TM interface, the IMS 14 Transaction Manager.

For the ISC TCP/IP function, IBM CICS Transaction Server for z/OS Version 5.1 or later, the Extended Terminal Option (ETO) feature, and IMS Connect are required.

The IMS Universal drivers require CICS Transaction Server for z/OS Version 4.1 (5655-S97) or later.

IBM CICS Transaction Server for z/OS Version 4.2 is required to support the CICS open transaction environment (threadsafe). CICS Version 4.2 takes advantage of the DRA open thread TCB option that was added by IMS Version 12 APAR/PTF PM31420/UK70991. By specifying the DRA open thread TCB option on an INIT request, a CCTL can avoid the overhead of TCB switching and improve parallel processing. When the DRA open thread TCB option is active, the DRA does not attach dedicated DRA thread task control blocks (TCBs). IMS DL/I calls can run on a CICS thread, eliminating the need for IMS to create additional TCBs for each thread and the need to switch between tasks during call processing.

For more information about the DRA open thread TCB option, see:

- INIT request (System Programming APIs)
- Database resource adapter startup table for CCTL regions (System Definition)

Db2 for z/OS subsystems supported

IMS 14 Transaction Manager can connect with Db2 for z/OS.

The IMS 14 Transaction Manager can be connected to any of the following Db2 products:

- Db2 11 for z/OS (5615-DB2) or later
- Db2 10 for z/OS (5605-DB2)

IMS/Db2 Coordinated Disaster Recovery Support requires the IMS 14 Remote Site Recovery (RSR) feature, and requires the databases to be registered with Recovery Level Tracking (RLT).

IBM MQ subsystems supported

IMS 14 supports IBM MQ.

IBM WebSphere MQ for z/OS V7.1 enhances the support for MQ message expiry and adds support for the OTMA protocol messages for resource monitoring.

Intersystem Communication (ISC) subsystems supported

Using Intersystem Communication (ISC), the IMS 14 Transaction Manager can be connected to IMS 14 and earlier systems, to IBM CICS Transaction Server for z/OS, and to user-written software.

The IMS 14 Transaction Manager can be connected to the following products by using ISC:

- IMS 14 (5635-A05)
- IMS Version 13 (5635-A04)
- IMS Version 12 (5635-A03)
- IBM CICS Transaction Server for z/OS Version 4.1 (5655-S97) or later

For the ISC TCP/IP function, IBM CICS Transaction Server for z/OS Version 5.1 or later and IMS Connect are required.

- User-written software

Programming languages used to write IMS 14

IMS 14 is written in High Level Assembler Release 6, PL/X, C, C++, and JDK Version 7.

Programming languages supported

You can write IMS applications in the supported versions of many programming languages.

You can write IMS applications in the currently supported versions of the following languages:

- Ada
- COBOL for OS/390® & VM
- Enterprise COBOL for z/OS

If you use Enterprise COBOL for z/OS Version 5.1, the data set that holds the output load modules of the compiler must be a PDSE.

For the latest version of COBOL for z/OS that is supported for a particular version of z/OS, see COBOL Migration Concerns going to New Release of the Compiler or z/OS.

- Enterprise PL/I for z/OS
- IBM High Level Assembler for z/OS & z/VM[®] & z/VSE[®]
- Java, using the IBM 31-bit SDK for z/OS, Java Technology Edition, V7, available from IBM Support Fix Central.
- PL/I for z/OS and OS/390
- TSO/E REXX
- VS Pascal
- z/OS C/C++

Application programs supported

IMS 14 supports application programs that are supported by IMS Version 12 and IMS Version 13.

All application programs that are supported under IMS Version 12 and IMS Version 13 are still supported under IMS 14. In general, you should not have to recompile, reassemble, or rebind an IMS application program to run under IMS 14.

Application programs that use the use the AIB INQY ENVIRON call might need to be changed. For more information about this situation, see “INQY call for system services migration considerations” on page 60.

Chapter 3. Packaging for IMS 14

The IMS product is packaged under several function modification identifiers (FMIDs). This packaging choice is in response to IMS internal requirements and is subject to change in the future.

The existence of an FMID does not imply that installation of the FMID is optional. Refer to the following table to determine which FMIDs are required, optional, or not applicable. Within the table the following notations are used:

R FMID installation is required.

O FMID installation is optional.

N The FMID is not applicable to this environment.

All FMIDs are installed outside of the Installation Verification Program (IVP). See the *Program Directory for Information Management System Transaction and Database Servers V14.01.00* for installation instructions.

Table 7. FMID requirements in IMS 14

FMID	Description	DB batch	DBCTL	DB/DC	DB/DC with XRF (1)	DCCTL
HIR2230	Internal Resource Lock Manager V2R3	O	O	O	O	N
HMK1400 (2)	System Services component IVP component Database Recovery Control Logging component IMS Connect	R	R	R	R	R
JMK1401	Database Manager function	R	R	R	R	N
JMK1402	Transaction Manager function	N	N	R	R	R
JMK1403	Extended Terminal Option feature	N	N	O	O	O
JMK1404	RSR Recovery-Level Tracking feature	O	O	O	O	O
JMK1405	RSR Database-Level Tracking feature	O	O	O	O	N
JMK1406	IMS Java On Demand features: IMS Universal drivers for access to IMS DB IMS TM Resource Adapter for access to IMS TM	O	O	O	O	O
JMK141Z	IMS Value Unit Editions: IMS Database Value Unit Edition V14.01.00 Program Number 5655-DSE IMS Transaction Manager Value Unit Edition V14.01.00 Program Number 5655-TM3	O	O	O	O	O

Note:

1. Although DCCTL with XRF is a supported combination, it is not included as an IVP option.
2. FMID installation is required even if the primary function provided by this FMID is not used.

Chapter 4. Installation considerations

Before you migrate and prepare your own systems, complete both the SMP/E install and the entire IVP process.

The following sections list the programming considerations for installing IMS and activating its functions.

IMS 14 is a complex product to install and prepare for execution. The SMP/E install and the IVP processes are not completed as a single process.

Recommendation: Complete both the SMP/E install and the entire IVP process, documented in *IMS Version 14 Installation*, before proceeding with the migration and preparation of your own systems.

The following table lists the sample jobs that are shipped with IMS 14.

Table 8. Sample installation jobs that are shipped with IMS 14

Job name	Job type	Description
DFSALA	SMP/E	Sample job to allocate and initialize a new SMP/E CSI data set (optional)
DFSALB	SMP/E	Sample job to allocate SMP/E data sets (recommended, but optional)
DFSALOC1	ALLOCATE	Sample job to allocate target and distribution libraries
DFSALOC2	ALLOCATE	Sample job to allocate target and distribution libraries for the On Demand features
DFSJSMKD	MKDIR	Sample job to invoke the supplied DFSJMKDR EXEC to allocate paths for the On Demand Features
DFSDDEF1	DDDEF	Sample job to define SMP/E DDDEFs for IMS
DFSDDEF2	DDDEF	Sample job to define SMP/E DDDEFs for the IMS Java On Demand Features
DFSAPPLY	APPLY	Sample APPLY job
DFSACCEP	ACCEPT	Sample ACCEPT job

The IVP process:

- Can be used for setting up and demonstrating an IMS system, as well as the testing of selected functions.
- Provides materials that you can use as a guide for working with your own IMS systems.

The IVP process includes:

- Data set allocation
- Post-installation activities on target libraries
- System definition activities
- SVC considerations
- Authorization considerations
- IMS system preparation activities

- IMS application preparation activities
- IMS system and application execution activities

There are no system programming or special programming considerations for IMS 14.

Chapter 5. Migration to IMS 14

The objective of migration support is to enable an orderly migration from your current release of IMS to IMS 14.

These topics describe general migration considerations for IMS 14 and migration considerations for specific new features and functions of IMS 14.

General migration recommendations

General migration tasks for current IMS installations include researching Preventive Service Planning (PSP), determining hardware and software requirements, backing up your system, and other tasks.

General migration recommendations for current IMS installations are:

- Contact IBM Software Support for current installation, migration, and problem resolution information, and ask for PSP for IMS.
- Read the *Program Directory for Information Management System Transaction and Database Servers V14.01.00* for the most current hardware requirements, software requirements, prerequisites, and installation information. For general installation information, see *IMS Version 14 Installation*.
- Review the service that has been applied to your current system. Determine if any critical service was released since your copy of the new IMS product was created. If critical service was released, install any that is appropriate for your installation of the new IMS release.
- Review the functions and enhancements for IMS 14.

Recommendation: Enable new functions and enhancements in production only after you have the new release up and running in production with your current IMS configuration. After the new IMS release supports your current configuration in production, then take advantage of new functions.

- Review changes to:
 - SMP/E, distribution, and system data sets
 - System definition macros
 - Log records
 - RECON records
 - Exit routines
 - Cataloged procedures
 - Control statement members in the IMS.PROCLIB data set
 - Utilities
 - Operator commands
 - Operating procedures
 - Messages and abend codes
- Determine the availability of updates to IBM IMS Tools, aids, and related products.
- Prepare a migration plan.
- Prepare a fallback plan. See “Fallback considerations” on page 68 for a sample list of items to consider when preparing a fallback plan.

- Apply the appropriate coexistence APARs/PTFs to your existing system. For a list of the coexistence APARs/PTFs, see “Overview of coexistence APARs” on page 77.
- Perform database recovery for any database data sets for which Extended Error Queue Elements (EEQEs) have been recorded in the DBRC Recovery Control (RECON) data set.
- If running the Common Service Layer (CSL) in an IMSplex made up of different versions of IMS, use the latest version of IMS for the CSL address spaces, such as SCI, OM, and so forth.
- Certain target and distribution library data sets must be either a PDS or a PDSE. The ADFSJLIB, SDFSJLIB, and ADFSLOAD data sets must be PDSE data sets. SDFSRESL data set must be a PDS. All other target (SDFSxxxx) and distribution (ADFSxxxx) libraries can be either PDS or PDSE.

Note: Other execution data sets, such as FORMAT, ACBLIB, and MODBLKS data sets, must be a PDS.

- Back up your system, including:
 - Databases and areas
 - RECON data sets
 - SMP/E data sets, distribution, and target libraries

Recommendation: Examine Hardware Data Compression (HDC) dictionaries when you migrate to a new release of IMS to determine if they incorporate IMS versions that are now out of service. Although rebinding dictionaries is not required when migrating to a new version of IMS, a gradual refresh of these dictionaries to a current release is a good practice.

- Validate your system definition source. You might want to merge the IVP source with your source.

The IMS IVP is used after the installation of a new IMS system. The IVP is used to verify the installation of IMS and can be used sporadically afterward. The IVP Variable Export utility mitigates the migration of IVP variables values between releases.
- Install the system prerequisites and your new IMS system (including the pre-generation service).

The complete set of IMS 14 modules that are needed for execution are built by a combination of SMP/E processing and running an *ALL* type of system definition process.

If needed, you can use the SMP/E GENERATE command to generate the JCL for jobs that build the modules that are not built during the system definition process.
- Install required service that was not included in the pre-generation service.
- Install any needed updates to IBM IMS tools, aids, and related products.
- Upgrade the RECON data set.
- Build application control blocks (ACBGEN).

Attention: Never use an ACB library built in one version of IMS with an IMS system of another version. You could cause an abend in the IMS control region or you could destroy some or all of your databases. Using an ACB library from one version of IMS with an IMS system of another version is a serious error. Carefully review your fallback plan and JCL. Also, protect your ACB library with RACF (or an equivalent security product), to make the ACB library impossible to use with an incompatible version of IMS.

- Validate cataloged procedures for users.
- Validate user-created members of the IMS.PROCLIB data set.
- Validate, reassemble, and rebind exit routines and user modifications, especially IMS Connect exit routines and code that uses IMS control blocks, such as database randomizers. Check your exit routines before reassembling. Sequence numbers changed in certain modules in IMS 14.
- Validate, reassemble, and rebind user programs that process log records. Some log record formats have changed.
- Validate and update operating procedures (for example, recovery, backup, and restart).
- If necessary, set the appropriate values for the AOIP, CMDP, DYNP, EMHB, FPWP, HIOP, LUMC, and LUMP parameters in the DFSPBxxx member of the IMS.PROCLIB data set to specify an upper limit on the amount of storage a pool can acquire. Specifying a limit is not recommended and should be done only after careful consideration. You can also use the IMS Syntax Checker to validate the values for the DFSPBxxx parameters. Consider the various execution parameters described in this information that can affect performance and migration.
- Ensure that appropriate dynamic allocation members are available to the new environment.
- Ensure that any custom routines and exits are available to the new environment (for example, database randomizers, secondary index parsing routines, and others).
- When using MSC to connect IMS systems with different releases, consider all message types (such as ISC, APPC, and OTMA) and the prefix sizes that accompany them.

Recommendation: When message queue data sets are used, make the MSGQ LRECL and block sizes identical across all IMS MSC systems. A problem can occur when an IMS system is migrated to a new release that uses messages with larger prefix sizes and new prefix segment types. When messages that contain these new and larger prefixes are sent to an earlier release of IMS, the new, and larger prefixes might not fit the message queues of the earlier release of IMS. This size mismatch can cause problems when the message is formatted and delivered to its destination, especially with MFS.

- Consider other products that can be affected by migration.
Any product that is dependent on the format and contents of the IMS log or the RECON data set is potentially affected. Examples of affected products or utilities are:
 - IMS Statistical Analysis utility
 - IMS Fast Path Log Analysis utility
 - IMS Log Transaction Analysis utility
 - IMS MSC Log Merge utility
 - CICS
 - IBM Tools
 - Non-IBM products, including user modifications
- After you start and test the IMS 14 system, monitor storage usage in both private/eprivate and CSA/ECSA for differences with previous releases. Make adjustments as necessary.

Migration steps

General migration tasks for current IMS installations include researching Preventive Service Planning (PSP), determining hardware and software requirements, backing up your system, and other tasks.

Unless otherwise indicated in a particular step, the following steps are typically performed by an IMS system programmer.

The general steps for migrating a current IMS installation to a new version of IMS are:

1. Plan and prepare for the new version of IMS
 - a. Create a migration plan.
 - b. Verify software and hardware requisites that are listed in either the announcement letter or the Program Directory.
 - c. Identify the Migration/Coexistence maintenance that you need to apply by using either the SMP REPORT MISSINGFIX (FIXCAT) command or by checking the PSP Bucket.
 - d. Review the migration and coexistence considerations and the new functions in the IMS Release Planning guide. If you are skipping one or more releases, review the Release Planning Guide for each release you are skipping.
 - e. Determine requirements for your IMS Tools or vendor products.
 - For information about IBM IMS Tools:
 - IMS Information Management Tools and IMS Version 14 Compatibility
 - IMS Information Management Tools and IMS Version 13 Compatibility
 - Contact IBM Software Support for all other IBM Tools related to IMS.
 - Contact vendor support for non-IBM tools and products that are related to IMS.
 - f. Order the product. The product can be ordered from Shopz website.
2. Install the product.
 - a. Review the PSP Buckets by searching for the upgrade name IMS1400 at Preventive Service Planning buckets for mainframe operating environments website and reviewing the upgrade information. If you are skipping a release, also review the upgrade information for the release you are skipping.
 - Pay particular attention to the *Installation Information* and *General Information* sections.
 - b. Perform SMP/E processing (RECEIVE, APPLY, ACCEPT).
 - c. Run the IMS Installation Verification Program (IVP).
 - d. Customize to your environment. Coordinate with your security administrators and database administrators, as appropriate.
 - e. Re-assemble any user code that accesses IMS control blocks or log records.
 - f. Re-work USERMODS as necessary, including the DFSIDEF0 module, if used.
 - g. Ensure that appropriate dynamic allocation members are available to the new environment. This step might be performed by or with a database administrator.
 - h. Ensure that your user exits are available to the new environment.

- i. Optional: Optionally, compare the members in the current SDFSRESL library to the members in the new SDFSRESL library to make sure that nothing that IMS needs is missing in the new SDFSRESL library.
3. Prepare for implementation
- a. Implement migration and coexistence maintenance that was identified in step 1c on page 26.
 - b. Prepare the z/OS interfaces by, at a minimum, installing the IMS type-2 and type-4 SVCs. Depending on your IMS configuration, you might need to take other steps to prepare the z/OS interface. For more information, see z/OS interface considerations (System Administration).
 - c. Verify that the RECONs are ready for upgrade by issuing the DBRC command CHANGE.RECON UPGRADE CHECKUP. This step might be performed by or with a database administrator.
 - d. Upgrade the RECONs by issuing the DBRC command CHANGE.RECON UPGRADE. This step might be performed by or with a database administrator.
 - e. Prepare libraries for the new version by:
 - 1) Perform the system definition (SYSGEN) process with "ALL" specified on the IMSCTRL macro.
 - 2) Create a set of ACB libraries for the IMS version by using the new version of the ACB Maintenance utility. This step is typically performed by a database administrator.
Attention: ACBs must always be generated by using the utilities from the IMS version in which they are used.
 - 3) If necessary, change procedures and jobs, ensuring that the DRA and JCL in DRA client address spaces, such as CICS, Db2 for z/OS stored procedures (SPAS), and so forth, point to the correct SDFSRESL data set.
 - f. Prepare and test a fallback plan. See "Fallback considerations" on page 68. This step is typically performed with database administrators.
 - g. Compare the current amounts of CSA, ECSA, PRIVATE, and EPRIVATE storage that is used to that of the new IMS version.
 - 1) Compare the CSA storage requirements of the new and current IMS Versions and adjust the IMS region sizes as necessary.
 - 2) Adjust IMS pool sizes for the new version as necessary.
 - 3) Take SVC memory dumps of your current IMS address spaces before migration and keep them for comparison purposes.
 - h. Review existing automation and update as appropriate. See Chapter 7, "Message and code changes in IMS 14," on page 95. This step might be performed by an automation or operations group.
 - i. Determine HIPER and PE exposure.
 - 1) Obtain current enhanced hold data from the Enhanced HOLDDATA for z/OS website.
 - 2) SMP/E RECEIVE current enhanced hold data
 - 3) Generate and analyze HIPER and PE exposure report by issuing the SMP/E command REPORT ERRSYSMODS pointing to your new TARGET zone. If needed, contact IBM Software Support for help.
 - j. Back up product libraries of old system, for example SDFSRESL, MODBLKS.
 - k. Identify any databases that have extended error queue elements (EEQEs) by issuing the IMS command /DIS DB EEQE and recover the databases as appropriate. This step is typically performed by database administrators.

4. Implement the system
 - a. Confirm that no databases have EEQEs by issuing the IMS command /DIS DB EEQE. If any do, resolve them before you proceed. This step is typically performed by database administrators.
 - b. Shut down old system
 - c. Ensure that shutdown completes successfully
 - d. Ensure that the archive of the logs runs successfully (OLDS)
 - e. Incorporate the new product libraries into your environment
 - f. Implement any changes to automation This step might be performed by an automation or operations group.
 - g. Cold start the new system by issuing the IMS command /NRE CHKPT 0 FORMAT ALL
 - h. Test your application programs This step is typically performed by the respective lines of business.

After implementation is complete, consider how changes to IMS commands in the new version might impact your installation and operations. This step might be performed by an automation or operations group, or by system programmers.

Important: After implementation of the new version is complete, database administrators and others must use the new versions of the Database Recovery utility (DFSURDB0) and Change Accumulation utility (DFSUCUM0). All other utilities that process logs must be run by using the version of the IMS system that created the log.

Migrating to IMS 14 from IMS Version 12 or earlier

If you are migrating to IMS 14 from IMS Version 12 or earlier, your migration plan must take into account the requirements and enhancements that are introduced into IMS in the versions that you are skipping.

For example, if you are migrating to IMS 14 from IMS Version 12, in addition to the requirements and enhancements of IMS 14, your migration plan must also account for the requirements and enhancements that are introduced into IMS in IMS Version 13.

Review the Release Planning information specific to both the IMS version that you are migrating to and the IMS version that you are skipping.

Discontinuance of support

Support is discontinued for various utilities, macros, resource adapters, and functions.

Extended Recovery Facility (XRF) stabilized

Support for the IMS Extended Recovery Facility (XRF) function is stabilized. Enhancements made in IMS no longer include support for XRF.

LGEN system definition

As of July 1, 2017, support for LGEN system definition is discontinued in all versions of IMS. If the LGEN parameter is specified on the SYSTEM= parameter in the IMSCTRL macro, remove the LGEN parameter specification to use the standard IMS system definition.

After IMS 14 APAR PI83715 is installed, if the LGEN parameter is specified, IMS issues message G123 and the stage-1 assembly fails with return code 4.

IMS XML DB support

Support for IMS XML DB is discontinued in IMS 14. IMS Version 12 is the last release to support XML DB.

Remote Site Recovery (RSR)

IMS 14 is the last version of IMS to support Remote Site Recovery (RSR). Installations that use RSR can instead use a multisystem IMSplex that includes geographically dispersed cloned IMS systems for seamless failover protection and disaster recovery.

SECURITY stage-1 system definition macro

Support for the SECURITY macro is discontinued in IMS Version 13. Except for the TRANEXIT and SIGNEXIT keywords, use initialization parameters, such as RCLASS and SECCNT, to specify values that were formerly specified in the SECURITY macro.

In IMS Version 13, you no longer need to specify the TRANEXIT and SIGNEXIT keywords to use the Transaction Authorization exit routine (DFSCTRN0) and the Signon/off Security exit routine (DFSCSNG0). Instead, IMS automatically detects and loads these exit routines and the Security Reverification exit routine (DFSCTSE0) when they are linked in one of the STEPLIB or LINKLIST libraries.

For information about using initialization parameters for security, see Controlling security during system startup for DB/DC and DCCTL (System Administration).

Updates to IMS PDFs in the IBM Publication Center

To enable more frequent updates to the IMS PDFs in support of the continuous delivery and agile methodologies, new or updated PDFs of the IMS documentation will no longer be delivered through the IBM Publications Center.

Instead, all new or updated IMS PDFs will be available for download from PDF files of IMS documentation.

IMS PDFs that are already available in the IBM Publication Center will remain in the IBM Publication Center.

IMS Classic Java APIs

IMS 14 systems and later do not support the IMS Classic Java APIs. If you used these APIs, you must migrate to the IMS Universal drivers.

In addition, support for Db2 stored procedures to IMS through Java is no longer provided. You can use the JDBC drivers to direct Java access to IMS instead of through a Db2 stored procedure. Or, use Db2 stored procedures in COBOL applications to access IMS through ODBA.

IMS Connect support for SSL

IMS 14 is the last version of IMS to support the IMS Connect SSL function. Installations that use this function should migrate to using IBM z/OS Communications Server Application Transparent Transport Layer Security (AT-TLS) to set up Secure Socket Layer (SSL) on TCP/IP connections to IMS Connect.

IMS Connect support for LOCAL option connections from IMS TM Resource Adapter

IMS 14 is the last version of IMS to support the IMS Connect LOCAL option for connections between IMS Connect and IMS TM Resource Adapter. Reconfigure any connections that currently use the LOCAL option to use either TCP/IP or the WebSphere Optimized Local Adapter.

IMS Enterprise Suite Version 2.1

IMS Enterprise Suite Version 2.1 is the last release of IMS Enterprise Suite that includes the DLIModel utility plug-in. Customers using the IMS Enterprise Suite DLIModel utility plug-in should migrate to using the IMS Enterprise Suite Explorer for Development instead. Although IMS Explorer for Development includes enhancements to most IMS Enterprise Suite DLIModel utility plug-in functions, it does not provide support for IMS database web services or IMS XML DB. You can obtain this support through the following options:

- For IMS database web services, you can generate web-enabled IMS database queries by shell sharing IMS Explorer for Development with IBM Data Studio.
- The XML data support in IMS databases is discontinued in IMS 14. For earlier versions of IMS, the support is discontinued when IMS Version 12 is out of service.

XRF support for full-function databases

XRF support for the data sharing of full-function databases is discontinued in IMS 14.

MFS SOA support

IMS 14 is the last release to support IMS Message Format Service support for service-oriented architecture (MFS SOA).

Current[®] users should migrate to IBM Rational[®] Host On Demand.

MFS Web Enablement

IMS 14 systems and later do not support IMS MFS Web Enablement.

Support for MFS Web Enablement in IMS Version 13 extends only to current users of the function.

Current users should migrate to IBM Rational Host Access Transformation Services (HATS), which provides capabilities similar to those provided by MFS Web Enablement.

Specific migration considerations

Specific migration considerations apply when you are migrating from IMS Version 12 or IMS Version 13 to IMS 14.

The information about the IMS features and enhancements that were introduced in IMS Version 13 are not documented in the IMS 14 Release Planning information. If you are skipping one or more IMS releases when you migrate to IMS 14, be sure to review the Release Planning information that is specific to each release of IMS that you are skipping.

For example, if you are migrating to IMS 14 from IMS Version 11, also review the migration considerations in the following information:

IMS Version 13: Release Planning, GC19-3658

IMS Version 12: Release Planning, GC19-3019

PDF files of the Release Planning Guides for all IMS releases can be downloaded from the IBM Publications Center.

The Release Planning information for the IMS releases that are currently supported by IBM is available in the IBM Knowledge Center.

If a new function does not have any migration considerations, it is not discussed in the following topics.

Migrating to IMS 14: DB

Specific migration considerations apply when you are migrating from the IMS Version 12 or IMS Version 13 Database Manager to the IMS 14 Database Manager.

These topics describe the IMS considerations for migrating to the IMS 14 Database Manager.

Database recovery utilities migration considerations

The IMS 14 Database Recovery (DFSURDB0) and Database Change Accumulation (DFSUCUM0) utilities accept as input log, image copy, and change accumulation data sets created by IMS Version 12 or later.

Database versioning migration considerations

Database versioning introduces some specific considerations for migration.

Attention: If you are migrating an IMS system that uses database versioning to IMS 14, you must preserve in the catalog the DBD segment instances that define the prior version of any DBD version that is still in use. If the segment instances for the prior version of a DBD are not preserved, any application programs that use that prior version of the DBD can no longer access the database.

For example, if your IMS system uses version 0 and version 1 of DBD DBD1A in IMS Version 13, both versions of the DBD1A DBD are stored as separate segment instances in the same DBD1A record in the IMS catalog. Version 1 of DBD1A reflects the actual physical structure of the database. Version 0 of DBD1A is a "virtual" structure that exists only in the IMS catalog.

In this case, when you migrate to IMS 14:

- You regenerate version 1 of DBD1A by using the IMS 14 generation utilities. When you populate the IMS catalog, version 1 of DBD1A gets inserted as a new set of segments in the same record as all of the segment instances of DBD1A from IMS Version 13. The IMS 14 instance of version 1 now becomes the active version of the DBD and the IMS Version 13 instance of version 1 is no longer used.
- You do not regenerate version 0 of DBD1A. Instead, you preserve the IMS Version 13 instance of the DBD1A version 0 in the IMS catalog. Application programs that use version 0 in IMS 14 continue to use the segment instances that were generated in IMS Version 13.

In IMS Version 13 and later, all database definitions (DBDs) that do not specify the DBVER parameter show a version number of 0 as the value of the DBVER parameter, even when database versioning is not enabled.

DEDB Alter enhancement migration considerations

The DEDB Alter utility has requirements for migration to IMS 14.

To run the DEDB Alter utility, all IMS data-sharing systems must be at the IMS Version 13 or later level. The MINVERS value in the RECON data set must be 13.1 or later. The default value for MINVERS in IMS 14 is 12.1

If you are invoking the DEDB Alter utility for an IMS 14 function (such as ADDAREA), the DEDB Alter utility detects the version of the IMS system and terminates if the IMS version does not support the particular change.

HALDB alter enhancements migration considerations

In IMS Version 13, the HALDB alter enhancement introduced some specific considerations for migration.

HALDB alter requires the MINVERS value in the RECON data set to be set to 13.1 or later. The default value for MINVERS in IMS 14 is 12.1.

HALDB support for 8-GB OSAM data sets enhancement migration considerations

The HALDB support for 8-GB OSAM data sets enhancement introduces some specific considerations for migration to IMS 14.

If you are migrating from IMS Version 12 to IMS 14 or later, all IMS systems must be at IMS 14 or later and the MINVERS value in the RECON data set must be 14.1 before you can use 8-GB OSAM data sets in a HALDB database.

If you are migrating from IMS Version 13 to IMS 14, an IMS Version 13 system can access a HALDB database that uses 8-GB OSAM data sets if APAR PI23918 is installed on the IMS Version 13 and the MINVERS=13.1 and CDSLID=2 are specified. Without the APAR applied, all IMS systems must be at IMS 14 and the MINVERS=14.1 must be specified in the RECON data set.

The default value for MINVERS in IMS 14 is 12.1.

IMS catalog migration considerations

The IMS catalog is a HALDB PHIDAM database that can require special consideration when migrating to a new release, depending on changes to the IMS catalog itself or on changes made to other IMS functions that use the IMS catalog.

Normally, migrating an HALDB database requires that you unload the original database, change its DBD to the new version, reload the database with the new DBD, and then run the new version of the ACB Maintenance utility on the new DBDs and PSBs of the HALDB database.

However, when migrating the IMS catalog, you only need to run the new version of the ACB maintenance utility on the new DBDs and PSBs of the IMS catalog because the catalog DBD structures are identical across versions.

Storage use by the IMS catalog

When you enable the IMS management of ACBs, the new system data sets that are required to manage the ACBs can use slightly more storage than the ACB libraries. However, IMS manages these system data sets, so you do not need to allocate the storage or the data sets.

The IMS management of ACBs and the IMS catalog

In IMS 14, the IMS management of application control blocks (ACBs) introduces migration considerations for users of the IMS catalog.

When IMS manages ACBs, which are the runtime control blocks for active databases and program views, the IMS catalog replaces ACB libraries as the repository of the active ACBs. Consequently, backup and recovery procedures for the IMS catalog are critical. Create image copies of the IMS catalog and use them with the logs to recover the IMS catalog. In previous releases of IMS, you might have been able to rebuild the IMS catalog from your ACB libraries. However, when IMS manages ACBs, you can use SQL DDL to add or modify database and program views. Changes made by using DDL would not be contained in an ACB library.

The default access level of the IMS catalog changes from ACCESS=READ to ACCESS=UPDATE when the IMS management of ACBs is enabled.

Migration steps for an existing IMS catalog in systems that use ACB libraries:

Like other IMS databases, the IMS catalog is defined by a DBD and accessed through PSBs. The migration procedure for the IMS catalog differs depending on which version of IMS you migrate from, and if you are migrating multiple IMS systems that share the catalog. When migrating to IMS 14, you must re-generate the ACBs from the DBDs and PSBs that IMS 14 supports. Additionally, when the ACBs for the other databases and program views in the IMS system are re-generated for the new IMS release, the IMS catalog must be updated with those ACBs to keep the IMS catalog in sync with the active ACBs in use by the IMS system.

Ensure that enough storage is allocated to the IMS catalog to accommodate the insertion of a complete IMS 14 set of the existing DBD and PSB resources.

Updating an IMS catalog for a new release can take a long time. Before you populate the IMS catalog, consider tuning your buffer pools based on the number and types of segments in the IMS catalog so that the utility you use for population doesn't do many buffer steals.

The DBDs and PSBs for the catalog changed in IMS 14. If you are migrating to IMS 14 from IMS Version 13, you need to replace the existing IMS catalog DBDs and PSBs with the IMS 14 DBDs and PSBs.

The following steps update an existing IMS catalog for a new version of IMS.

Attention: Do not delete and rebuild the IMS catalog if the IMS catalog contains metadata that cannot be restored from the new version of the ACB library. Metadata that cannot be restored from the ACB library includes metadata that is required for database versioning, remarks, or metadata stored by non-IMS products.

1. If necessary, increase the amount of storage that is allocated to the IMS catalog.
2. Shut down IMS 14 if it is running.
3. Install the IMS 14 DBDs and PSBs for the IMS catalog from the IMS.SDFSRESL data set to the IMS.DBDLIB and IMS.PSBLIB data sets. You do not need to unload and reload the IMS catalog database, even though the IMS catalog DBD might have changed in the new release. The catalog DBDs are called DFSCD000 and DFSCX000. The PSBs are called DFSCPL00, DFSCP000, DFSCP001, DFSCP002, and DFSCP003. These are reserved names and cannot be changed or used for other resources.

If the IMS catalog is shared by multiple systems and you are migrating from IMS Version 13, you must create a temporary DBDLIB that you will install the IMS 14 DBDs into. This DBDLIB is used to hold the IMS 14 catalog DBDs until migration is complete.

4. Use the IMS 14 ACB Generation and Populate utility (DFS3UACB) to generate the ACBs for the IMS catalog and all your application PSBs and database DBDs and add them to the IMS catalog. The DFS3UACB utility generates the ACBs and updates the IMS catalog in a single job step, ensuring that the IMS catalog is in sync with the latest ACBs.

If the IMS catalog is shared by multiple systems, you must run the IMS 14 DFS3UACB utility as data sharing DL/I batch job with IRLM. If the utility cannot be ran with IRLM, you must take the IMS Version 13 catalog offline with a /DBR BD command while performing this step.

As an alternative to using the ACB Generation and Catalog Populate utility (DFS3UACB), you can run both the IMS 14 ACB Maintenance utility and the IMS Catalog Populate utility. The ACB Maintenance utility builds the ACBs. The IMS Catalog Populate utility populates the IMS catalog with the new versions of PSBs and DBDs.

5. Activate the ACB library. If the IMS catalog is shared by multiple systems and you are migrating from IMS Version 13, update the common DBDLIB with the catalog DBDs held in the DBDLIB you created in step 3, and delete the temporary DBDLIB.
6. Restart the IMS 14 system.

If you have any application programs that read the IMS catalog, review the structure of the IMS catalog to ensure that the application programs can handle any changes appropriately.

After testing is complete and you are certain that you do not need to fall back to the previous version of IMS, you can remove the DBD and PSB resource instances for the past IMS version from the IMS catalog by using the IMS Catalog Record Purge utility (DFS3PU10).

Related reference:

➔ IMS Catalog Record Purge utility (DFS3PU10) (System Utilities)

➔ ACB Generation and Catalog Populate utility (DFS3UACB) (System Utilities)

IMS management of ACBs migration considerations

When you enable the IMS management of runtime application control blocks (ACBs) for the first time, there are a number of considerations to take into account.

Initial load of the IMS Catalog by using the IMS Catalog Populate utility (DFS3PU00)

You must load the IMS catalog before enabling IMS management of ACBs. Use the IMS Catalog Populate utility (DFS3PU00) to populate the IMS catalog if you are using it for the first time with `MANAGEDACBS=SETUP` specified in the `SYSINP` DD statement in the utility JCL. For more information, see *Loading the IMS catalog with the DFS3PU00 utility (System Definition)*.

Migrating from a shared IMS catalog with multiple ACBLIBs

When the IMS management of ACBs is enabled, an IMS catalog functions like an ACB library for all of the IMS systems that use that IMS catalog. So, if you are migrating a multi-system environment in which the IMS systems share an IMS catalog but each system has its own ACB library, enabling the IMS management of ACBs effectively converts your IMS systems from using multiple, independent ACB libraries to sharing a single set of ACBs.

An IMS catalog supports only a single active instance of an ACB for each version of IMS that uses the IMS catalog. In the IMS catalog, each instance of an ACB is identified by the time stamp that is taken when the ACB was created. When you set up an IMS catalog to manage ACBs for multiple IMS systems that each used their own ACB library, if different instances of the same ACB appear in different ACB libraries, only one instance of the ACB is loaded into the IMS directory data set, a system-managed data set that is an extension of the IMS catalog. Usually, the instance that is loaded into the IMS directory is the instance with the most recent time stamp.

Before you enable the IMS management of ACBs, if multiple IMS systems use separate ACB libraries that contain common ACBs, make sure that all of the common ACBs in the separate libraries were built from the same DBD and PSB libraries.

After the IMS management of ACBs is enabled, any changes to an active ACB in the IMS catalog potentially impacts all IMS systems that share the IMS catalog.

Although it is not recommended to do so, you can enable a dedicated IMS catalog for each IMS system that needs separately maintained ACBs. However, the benefits of sharing an IMS catalog would be lost.

DBRC and IMS managed ACBs

DBRC requires access to your database definitions to process certain DBRC commands. In IMS systems that use ACB, DBD, and PSB libraries, DBRC retrieves the database definitions from the `DBDLIB` data set.

In an IMS system that manages ACBs, DBRC can retrieve the database definitions from the IMS catalog, if the name of the IMS catalog is entered in the RECON header or is specified when a DBRC command is entered.

You enter the IMS catalog name into the RECON header by using either DBRC command INIT.RECON or CHANGE.RECON.

The following DBRC commands are enhanced to either specify the name of an IMS catalog to use or to display the current default IMS catalog:

- CHANGE.DBDS
- CHANGE.PART
- CHANGE.RECON
- INIT.DB
- INIT.DBDS
- INIT.PART
- INIT.RECON
- LIST.RECON
- NOTIFY.REORG

IMS Catalog Definition exit routine (DFS3CDX0)

Existing IMS Catalog Definition exit routines (DFS3CDX0) must be updated when the IMS management of ACBs is enabled so that the specification for ACBMGMT in the exit routine matches the specification for ACBMGMT in the CATALOG section of the DFSDFxxx PROCLIB member.

IMS 14 updates the sample Catalog Definition exit routine with a flag that indicates whether the IMS management of ACBs is enabled.

Online processing impact

When the IMS management of ACBs is enabled and DDL statements are used to define or modify databases or program views, the online IMS system processes the DDL statements, updates the IMS catalog, and builds the runtime control blocks. So, the costs that are associated with this processing are incurred by the online IMS system.


This processing cost is avoided during the initial migration to IMS-managed ACBs because the batch DBD and PSB generation utilities are used to create the IMS 14 DBD and PSB control blocks. Also the IMS Catalog Populate utility can be run in batch mode when it updates the IMS catalog and sets up the IMS system to manage ACBs.

After the IMS management of ACBs is enabled, if your installation uses DDL heavily to define or modify databases and program views, you might see an increase in DL/I processing over the processing in IMS systems that are not enabled for the IMS management of ACBs.

Related tasks:

 [Enabling IMS management of ACBs \(System Definition\)](#)

Related reference:

 [IMS Catalog Definition exit routine \(DFS3CDX0\) \(Exit Routines\)](#)

Migrating to IMS 14: TM

IMS considerations for migrating from IMS Version 12 or IMS Version 13 Transaction Manager to IMS 14 Transaction Manager include how the enhancements to IMS affect migration, as well as how major IMS functions are affected by migration.

These topics describe the IMS considerations for migrating from IMS Version 12 or IMS Version 13 Transaction Manager to IMS 14 Transaction Manager.

APPC/IMS flood control migration considerations

In IMS 14, the new APPC/IMS flood control function is active by default and the IMS system will take action if the default flood thresholds are reached unless you change the default thresholds or disable the function.

The default thresholds are 5,000 and 1,000,000. If the number of active APPC conversations exceeds the default threshold of 5000, subsequent APPC requests are queued in 64-bit storage until IMS can accept them for processing. If the number of APPC requests that are queued in 64-bit storage exceeds 1,000,000, IMS stops all APPC input from z/OS. When the number of conversations gets close to the flood thresholds, IMS issues warning messages.

You can modify or disable APPC/IMS flood control by specifying the APPCMAXC= parameter in the DFSDCxxx PROCLIB member. To disable the function, specify APPCMAXC=0.

The /DISPLAY ACT command output is modified to include a MAXC= output field that shows the maximum number of APPC conversations that can be active before APPC requests are queued in 64-bit storage.

JVM 64-bit support migration considerations

Specify JVM=64 in the EXEC job control statement to enable the JVM 64-bit support. The JVM= parameter is valid only for Java batch processing (JBP) and Java message processing (JMP) regions.

By default, Java dependent regions load and run with a 31-bit Java virtual machine (JVM). You can modify the JVM addressing mode to 64-bit by specifying the JVM=64 parameter in the DFSJBP and DFSJMP procedure EXEC statement. The new JVM= parameter is valid only for the JBP and JMP regions.

When you specify the JVM= parameter on the EXEC PGM=DFSRRRC00 card for the JMP region, you must specify all positional parameters, PRLD=, SSM=, PARDLI=, MINTHRD= and MAXTHRD=, before the JVM= parameter. MAXTHRD= and MINTHRD= are two positional parameters that are hidden when the JVM= parameter is not provided.

By specifying JVM=64, Language Environment, the infrastructure that supports a 64-bit JVM, is also changed to the 64-bit addressing mode.

You can change to the default 31-bit addressing mode by specifying JVM=31.

To determine which addressing mode a JVM in a Java dependent region is using, you can check the JVM= parameter, the environment file, or the status messages in the job output.

Example

The following sample output shows status messages of a job that uses 31-bit JVM. In the 31-bit addressing mode, the DFSJVM00 module is used, and the LIBPATH points to a 31-bit JVM directory.

```
DFSJVM00: ENVIRON member name is DFSJVMEV
DFSJVM00: LIBPATH=/usr/lpp/java170/J7.0/bin/
DFSJVM00: JVMOPMAS member name is DFSJVMMS
```

In the following sample output of a job that uses 64-bit JVM, the DFSJVM64 module is used, and the LIBPATH points to a 64-bit JVM directory.

```
DFSJVM64: ENVIRON member name is DFSJVMEV
DFSJVM64: LIBPATH=/usr/lpp/java170/J7.0_64/bin/
DFSJVM64: JVMOPMAS member name is DFSJVMMS
```

Java environment scripting migration considerations

When the //STDENV DD statement is used, only the parameters in the shell scripts that are referenced by the //STDENV DD statement are used. Move the JVM configuration parameters from the DFSJVMMS and DFSJVMEV members of the IMS.PROCLIB data set to the shell scripts.

Before you use the //STDENV DD statement, ensure that you move the parameter configurations that were specified in the DFSJVMEV and DFSJVMMS members of the IMS.PROCLIB data set to the shell scripts that are referenced by the DD statement.

If the //STDENV DD statement is present, both the DFSJVMEV and DFSJVMMS PROCLIB members are ignored.

In the shell script, you must use the export shell command to export the environment variables.

APAR PI82480 introduces two variables under the //STDENV DD statement, **JZOS_ENABLE_OUTPUT_TRANSCODING** and **JZOS_OUTPUT_ENCODING**, that can be used to specify the encoding options for stdout and stderr. APAR PI82480 also increases the length limit of classpath under the //STDENV DD statement from 16K to 150K.

You can set the variable **JZOS_OUTPUT_ENCODING** under the //STDENV DD statement to specify the code page that is used to convert the raw bytes for stdout and stderr. If no code page is provided in **JZOS_OUTPUT_ENCODING**, the default code page in the current locale is used for stdout and stderr. You can use the variable **JZOS_ENABLE_OUTPUT_TRANSCODING** to enable or disable the code page that is specified in **JZOS_OUTPUT_ENCODING**. The encoding option is enabled by default. If **JZOS_ENABLE_OUTPUT_TRANSCODING** is set to false, then the code page specified in **JZOS_OUTPUT_ENCODING** is ignored and raw bytes are written to stdout and stderr.

Example 1

The following JCL is an example of how to use //STDENV DD statement.

```
//STDENV DD *
# This is a shell script which configures
# environment variables for the Java JVM.
# Variables must be exported to be visible to the launcher.
. /etc/profile
export JAVA_HOME=/usr/lpp/java/J7.0
```

```

export PATH=/bin:"${JAVA_HOME}"/bin
LIBPATH=/lib:/usr/lib:"${JAVA_HOME}"/bin
export LIBPATH="$LIBPATH":

# Customize your CLASSPATH here.
CLASSPATH="$CLASSPATH":myLibPath/imsudbimsxxxx.jar
for i in "${JAVA_HOME}/*.jar; do
    CLASSPATH="$CLASSPATH":"$i"
done
# Classpath length can be up to 150K
export CLASSPATH="$CLASSPATH":

# Use this variable to specify the encoding for DD STDOUT and DD STDERR.
export JZOS_OUTPUT_ENCODING=Cp1047
# Use this variable to enable or disable the encoding specified
# in JZOS_OUTPUT_ENCODING.
# The default is true. When the variable is set to false, the encoding
# specified in JZOS_OUTPUT_ENCODING is ignored.
export JZOS_ENABLE_OUTPUT_TRANSCODING=false
...

```

Example 2

The following JCL provides an example of how to configure Java environment variables and options by using the //STDENV DD statement. You can separate the JVM configurations variables in multiple files and concatenate them in a single //STDENV DD statement. The member names here are provided as examples.

```

//JMP00001 JOB MSGLEVEL=1,MSGCLASS=E,CLASS=K,
//  LINES=999999,TIME=1440,REGION=0M,
//  MEMLIMIT=NOLIMIT
//*
//JMP00001 PROC CL1=001,CL2=000,CL3=000,
//  CL4=000,OPT=W,OVLA=0,
//  SPIE=0,TLIM=00,VALCK=0,
//  PCB=032,SOD=,STIMER=,
//  NBA=5,OBA=5,IMSID=IMS1,
//  AGN=,SSM=,PREINIT=,
//  ALTID=,PWFI=,APARM=,
//  LOCKMAX=,ENVIRON=,JVMOPMAS=,
//  PARDLI=,PRLD=,JVM=31
//REGION EXEC PGM=DFSRR00,
//          PARM=(JMP,&CL1&CL2&CL3&CL4,
//              &OPT&OVLA&SPIE&VALCK&TLIM&PCB,&STIMER,&SOD,&NBA,
//              &OBA,&IMSID,&AGN,&PREINIT,&ALTID,&PWFI,'&APARM',
//              &LOCKMAX,&ENVIRON,,&JVMOPMAS,&PRLD,&SSM,&PARDLI,,
//              &JVM)
//STEPLIB DD DSN=IMSVS.IMDO.SDFSJLIB,DISP=SHR
//          DD DSN=IMSVS.IMDO.SDFSRESL,DISP=SHR
//          DD DSN=CEE.SCEERUN,DISP=SHR
//          DD DSN=SYS1.CSSLIB,DISP=SHR
//STDENV DD DSN=h1q1.IMSCONF(IMS1ENV)
//          DD DSN=h1q2.IMSCONF(IMS1OPT)
//          DD DSN=h1q2.IMSCONF(IMS1MAIN)
//          DD DSN=h1q2.IMSCONF(IMS1DEBUG)
//PRINTDD DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//STDOUT DD SYSOUT=*
//STDERR DD SYSOUT=*
//*
//          PEND
//JMP00001 EXEC JMP00001

```

In the example, the *IMS1ENV* file is a shell script that configures the environment variables for the JVM. The *IMS1OPT* file specifies the IBM JVM runtime options,

which are typically prefixed with -X, and Java system properties, which are prefixed with -D. The *IMS1MAIN* file supplies arguments to the Java main method.

```
# IMS1ENV sample file
. /etc/profile
export JAVA_HOME=/usr/lpp/java/J7.0
export PATH=/bin:${JAVA_HOME}/bin
LIBPATH=/lib:/usr/lib:${JAVA_HOME}/bin
export LIBPATH="$LIBPATH":
CLASSPATH="$CLASSPATH":myLibPath/imsudbimsxxxx.jar
for i in "${JAVA_HOME}/*.jar; do
    CLASSPATH="$CLASSPATH":$i
done
export CLASSPATH="$CLASSPATH":

# IMS1DEBUG sample file
export DEBUG=Y

# IMS1OPT sample file
export JZOS_OUTPUT_ENCODING=Cp1047
export JZOS_ENABLE_OUTPUT_TRANSCODING=false

# IMS1MAIN sample file
# Use this variable to supply arguments to the Java main method
JZOS_MAIN_ARGS="gofast"
```

MSC migration considerations

The enhancements to Multiple Systems Coupling (MSC) in IMS 14 introduce a number of migration considerations.

Migration considerations for MSC exit routines:

Exit routines that use information from the MSC logical and physical blocks, such as the TM and MSC Message Routing and Control User exit routine (DFSMSCE0) and the Message Control/Error exit routine (DFSCMUX0), must be changed to either use callable services or use the FIND/SCAN function of the DFSCBTS macro to find the logical link blocks (LLBs) and the physical link control blocks (LCBs).

The exit routines can no longer use the MSCDLLB and MSCDLCB fields in the MSCD block.

User exit routines that reference MSC control blocks must run in 31-bit mode.

User exits in the IMS nucleus that reference MSC DDM or command modules with a VCON will need to be changed. Examples include the TM and MSC Message Routing and Control User Exit routine (DFSMSCE0) and the Message Control/Error exit routine (DFSCMUX0). MSC DDM modules are removed from the nucleus and now reside in 31-bit storage. Their addresses can be accessed from pointers in other control blocks.

The IMS nucleus no longer contains any MSC modules. The MSC modules that were in the nucleus in earlier releases of IMS are now in 31-bit storage. Their 31-bit addresses are stored in the MSC blocks. As a result of this change, MSC control blocks can no longer be sequentially scanned or their addresses calculated by multiplying the block length by the link number. The new pools (MSCP and MSCL) must be scanned using the DFSCBTS macro or FIND/SCAN control block callable services.

User exit routines that reference MSC modules in the nucleus must be changed to reference MSC modules in 31-bit storage, whose addresses are now saved in the MSCD.

The LLB is the main control block for MSC logical links and the LCB is the main control blocks for MSC physical links. The pointers within the MSC blocks that are used to tie the blocks together have not changed in IMS 14.

Note: Save old user exit routines, in case you need to fall back to a previous release of IMS.

Migration considerations for dynamic definition of MSC resources:

There are several considerations for migrating to IMS 14 support for the dynamic definition of MSC resources.

To migrate to using dynamic definition of MSC resources, evaluate existing transaction, LTERM, and MSNAME names for conflicts with dynamic MSC. Dynamic MSC reserves additional name prefixes, including MSNS and MSNI. If you already have existing transaction, LTERM, or MSNAMEs defined that start with the prefix MSNS or MSNI, the CREATE MSNAME command might not complete successfully. The CREATE MSNAME command defines an internal system MSNAME with the naming convention MSNIxxxx, where xxxx is the link number. If MSNIxxxx already exists as an LTERM, transaction, or MSNAME, the CREATE MSNAME command will fail.

The QUERY LTERM, QUERY MSLINK, QUERY MSNAME, and QUERY MSPLINK commands are changed to include L before all local headers in command output, as preconditioning to distinguish from the global output fields for the IMSRSC repository. If you have automation that looks for the headers, it might need to change to look for the new local headers that have L at the beginning.

Automation might need to be changed if it issues the following commands:

- /DISPLAY ASMT MSPLINK, QUERY MSLINK, or QUERY MSPLINK commands, because the resources are now displayed in alphabetic order instead of in system definition order.
- QUERY LTERM, QUERY MSLINK, QUERY MSNAME, or QUERY MSPLINK commands that parse the output headers, because all local headers now have an L prefix, to identify them as local values.
- QUERY MSLINK specified with SHOW(ALL), because SHOW(ALL) now displays the column for definitional attribute BACKUP.
- QUERY MSPLINK specified with SHOW(ALL), because SHOW(ALL) now displays columns for definitional attributes ASR, BACKUP, BUFSIZE, MAXSESS, and MODETBL.

IMS 14 manages link numbers differently after a cold start when logical links are deleted and re-created. Update your operating procedures to use link names instead of link numbers. The numbers for links remain the same across warm and emergency restarts, even if there are number gaps caused by deleted links. If links are created and deleted dynamically, resulting in gaps in link numbers, when an IMS system is cold started, the gaps are removed and the link numbers of the logical links that followed the gap are adjusted.

After migration, perform the following procedure to clean up system generation definitions:

1. Remove the MSC definitions from the stage 1 system definition, which include the following items:
 - MSPLINK, MSLINK, MSNAME, and NAME macros

- MSVID and SYSTEM=(MSVERIFY) parameters in the IMSCTRL macro
- SYSID parameter in the APPLCTN and TRANSACT macros

2. Run system generation.

Related concepts:

“Dynamic definition of MSC resources fallback considerations” on page 71

Migration considerations for IMSRSC repository support for dynamically defined MSC resources:

There are several considerations for migrating to IMS 14 IMSRSC repository support for dynamically defined MSC resources.

- If you use the IMSRSC repository to store dynamically defined MSC resources, ensure that automation and operational procedures that issue commands for MSC resources use type-2 commands, which specify link names, instead of type-1 commands, which specify link numbers. For example, instead of using the /RSTART LINK 10 command to start a link, use the UPDATE MSLINK NAME(*logical linkname*) START(COMM) command. During stage-1 system generation, the IMS system assigns numbers to logical links in the order in which the links are generated. However, the numbers for links are not stored in the IMSRSC repository. If logical links are referenced by using link numbers and are automatically imported from the IMSRSC repository, the numbers of the links are likely to change at the next IMS cold start.
- If you are using channel-to-channel (CTC) links, consider removing the DD definitions for the CTC links from the IMS JCL before you import the MSC resources from the IMSRSC repository. This allows the CTC addresses that are defined to be used for the CTC links that are imported from the IMSRSC repository.
- If you use the IMSRSC repository to store dynamically defined MSC resources, the DFSCLL3x and DFSCLR0x members of the IMS.SDFSRESL data set are no longer required. Because the DFSCLC0x member might contain non-MSC resources, the DFSCLC0x might still be required. After you are satisfied with the setup of your DRD environment for MSC resources, the DRD environment is running successfully, and your MSC resources are exported to the IMSRSC repository, you can remove the DFSCLL3x and DFSCLR0x members. For the DFSCLC0x member, update the member to remove MSC logical link path definitions. If automatic import is enabled, MSC resource definitions are imported during IMS cold start from the repository that contains the most current data. However, you can continue to use the DFSCLL3x, DFSCLR0x, and DFSCLC0x members as the source for your MSC resource definitions, instead of a repository. If you continue to use the DFSCLL3x, DFSCLR0x, and DFSCLC0x members, keep your system definition macros synchronized with the changes you make dynamically using DRD commands.
- If you migrate to using the IMSRSC repository for dynamically defined MSC resources but continue to use the DFSCLL3x, DFSCLR0x, and DFSCLC0x members of the IMS.SDFSRESL data set, keep resource definitions that are in the members synchronized with the resource definitions that are in the repository. This synchronization enables you to maintain viable DFSCLL3x, DFSCLR0x, and DFSCLC0x members if you must disable DRD and fall back to using the system generation process for MSC resources. To keep your DFSCLL3x, DFSCLR0x, and DFSCLC0x members synchronized with your online definitions, update your static macro definitions with the changes that you make dynamically using type-2 commands. When changes are made dynamically, perform a MSC system definition to add, change, or delete resources from the DFSCLL3x, DFSCLR0x, and DFSCLC0x members.

•
Recommendation: It is recommended that you enable automatic export to the IMSRSC repository only after both of the following tasks are completed:

- All RM systems are migrated to IMS 14 or later.
- For IMS 14 systems, the enabling APAR for IMSRSC repository for dynamically defined MSC resources, APAR PI50129, is applied.
- For IMS 15 systems, the enabling APAR for IMSRSC repository for dynamically defined MSC resources, APAR PI74957, is applied.

OTMA migration considerations

IMS 14 includes migration considerations for OTMA.

Migration considerations introduced in IMS 14

Dynamic storage for OTMA descriptors

Before IMS 14, only a value of 255 or 510 was displayed in messages DFS2384W and DFS3678E, which are the warning and error messages that are issued for approaching or exceeding the maximum allowable number of OTMA descriptors. In IMS 14, the value that is displayed in the error messages can be 255, 510, or another value that is specified on the DDESCMAX or MDESCMAX keywords in the DFSOTMA descriptor.

Migration considerations introduced in IMS Version 13

OTMA global flood control enhancement

The OTMA global flood control enhancement changed existing IMS error messages. The existing DFS4388W and DFS0793I messages have been enhanced to be consistent with the new DFS3428W and DFS3429E messages. Changes in your automation program might be needed if it detects the DFS4388W and DFS0793I messages.

If you use the DFSOTMA client descriptor to specify a global flood limit for all clients, the maximum number of OTMA client descriptors that you can specify in the DFSYDTx member of the PROCLIB data set is reduced from 255 to 254.

In previous releases, the command `/STA TMEMBER ALL INPUT #####` resulted in warning messages. It now turns on global flood control and rejects new input.

Changing the value with `/START TMEMBER ALL INPUT 200` command previously did not trigger the flood status update in the `/DIS OTMA`, and new input continued to be accepted. In IMS Version 13, the flood status takes effect immediately. If there are more messages waiting to be processed than the value:

- Message DFS3429E is displayed, which warns that THE TOTAL OTMA INPUT MESSAGES(TIB) HAVE REACHED THE GLOBAL LIMIT ZZZZ.
- `/DIS OTMA` shows a status of SERVER+FLOOD.
- New input is rejected.

The length of TIB and INPUT columns of IMS `/DISPLAY OTMA` command were increased by one so that these fields can display up to five digits of OTMA message numbers. The T27 DSECT and T28 DSECT of IMS `DISPLAY MACRO` were also updated for these length changes.

In IMS Version 13, the default for global flood controls continues to be a warning message, but the default threshold is now at 10,000 and the relief

at 50%. However, if the global flood limits have been activated by either specification of the DFSOTMA client descriptor or the /STA TMEMBER ALL INPUT command, reaching the 100% mark results in rejection of new transaction messages until the flood has been relieved at 50%. Additionally, warning messages will be issued at the 80% level.

OTMA Callable Interface enhancements

OTMA C/I no longer receives IMS DFS messages when OTMA NAKs (negative acknowledgement) input for asynchronous messages. They receive a return and reason code indicating the specific error.

Synchronous program switch migration considerations

In IMS Version 13, the synchronous program switch enhancements introduces migration considerations for the configuration of the MINVERS value in a shared queues environment.

In IMS 14, the default value of MINVERS in the RECON data set is 12.1. However, to use the synchronous program switch function in a shared queues environment, all of the participating systems must have a MINVERS value of at least 13.1.

XRF and Communication Controller for Linux on System z (CCL) migration considerations

IBM is withdrawing from support the Communication Controller for Linux on System z (CCL). The announced date for withdrawal is March 31, 2016.

If your installation uses CCL for IMS XRF tracking of VTAM terminals, consider replacing XRF entirely by setting up an IMSplex with two or more IMS systems and using VTAM Generic Resources (VGR). This option requires a Parallel Sysplex environment.

Migrating to IMS 14: System

IMS considerations for migrating from IMS Version 12 or IMS Version 13 systems to IMS 14 systems include how the enhancements to IMS affect migration, as well as how major IMS functions such as DBRC and dynamic resource definition (DRD) are affected by migration.

These topics describe the IMS considerations for migrating from IMS Version 12 or IMS Version 13 systems to IMS 14 systems.

CQS migration considerations

Migrate CQS and any CQS clients on the z/OS image at the same time. If doing so is not possible, CQS must be migrated before any of the CQS clients are migrated.

See “Common Queue Server coexistence considerations” on page 79 for information about CQS coexistence rules.

DBRC migration considerations

Migrating DBRC to IMS 14 includes several tasks, including upgrading the RECON data set.

These topics describe the considerations and tasks for migrating DBRC to IMS 14.

Changes to the RECON data set:

Certain records in the RECON data set are new or changed from the records in IMS Version 12 and IMS Version 13.

Changes introduced in IMS 14

In IMS 14, the following RECON records have changed fields:

- DSPDBHRC: New flag DBOSAM8G within the DMBFlags field is added to indicate that a HALDB database supports 8-GB OSAM database data sets.
- DSPPTNRC: New flag PTNOSAM8G within the PTNFlags field is added to indicate that within each HALDB partition the maximum capacity of each OSAM PHDAM and PHIDAM data set is 8 GB.
- DSPRCNRC: New field RCNCATLG is used to indicate the IMS catalog if one is being used.

Changes introduced in IMS Version 13

In IMS Version 13:

- The DSPPTRNC record may be increased in size. The partition record increases in size if, in preparation for HALDB alter, new OSAM block sizes or VSAM CI sizes are defined. 80 bytes are added to the end of the record based on a word boundary.
- The following RECON records have new or changed fields:
 - DSPDBHRC: For a HALDB database, DBALTER# and DBALTCM# fields are added to count the number of partitions to alter and the number of partitions that have completed the HALDB alter process.
For a partition database, new DBALTER and DBALTCMP flags are added to indicate that the alter process has started and completed. For a Fast Path Area, the DBALTER flag indicates that the area is being altered.
For a DEDB database, the new DBALTER# field is used to indicate that DEDB Alter is in progress.

REPAIR.RECON migration considerations:

The DBRC Command Authorization exit routine (DSPDCAX0) or RACF command authorization definitions might need to be updated to allow users to use the REPAIR.RECON command.

If you use security procedures to verify the authority of a user to issue DBRC commands, update the DBRC Command Authorization exit routine (DSPDCAX0) or RACF command authorization definitions to authorize the use of the REPAIR.RECON command. As with other DBRC commands, consider restricting access to only those users who must issue the command.

Related concepts:

 Security for DBRC commands and API requests (System Administration)

Upgrading the RECON data set:

A RECON upgrade batch command is provided to enable you to convert the recovery control (RECON) data set from an IMS Version 12 or IMS Version 13 format to an IMS 14 format. You are not required to change the MINVERS value to '14.1' when you migrate to IMS 14. Change this value only after you verify that you do not need to coexist with an earlier version of IMS, will not need to fall back, and when you need to use new functions that require the MINVERS value to be set to '14.1'.

Attention: Do not issue the IMS 14 CHANGE.RECON UPGRADE command to upgrade the RECON data sets until all IMS Version 12, IMS Version 13, and IMS 14 systems that access the RECON data sets have the correct supporting products in place and have been tested for IMS 14 support.

To upgrade an IMS Version 12 or IMS Version 13 RECON data set:

1. Apply the IMS 14 coexistence Small Programming Enhancements (SPEs) to all IMS Version 12 and IMS Version 13 systems before you upgrade the RECON data set. The coexistence SPEs (APARs/PTFs) are:
 - PI10131/UI22257 for IMS Version 12
 - PI10132/UI22258 for IMS Version 13

Attention: Jobs that access the RECON data set and do not create subsystem records, such as the Database Change Accumulation utility (DFSUCUM0) and the Database Recovery Control utility (DSPURX00), are not protected from having the RECON data set upgraded while they are running on a version of IMS that does not have the appropriate migration/coexistence SPE applied. When these types of jobs access the RECON data set after the upgrade, the results might be unpredictable. Ensure that no such jobs are running when you upgrade the RECON data set.

2. Ensure that you have two active RECON data sets (COPY1 and COPY2) and a spare data set when you upgrade the RECON data sets while other jobs are accessing them.
3. Before issuing the CHANGE.RECON UPGRADE command against the production RECON data sets, upgrade a copy of the production RECON data sets to verify that the upgrade will complete successfully.
4. Optionally, issue the CHANGE.RECON UPGRADE CHECKUP command. This command checks the status of the RECON data set and records to verify whether the RECON data set is in a state that allows an upgrade. No RECON data set records are changed as a result of issuing the CHANGE.RECON UPGRADE CHECKUP command.
5. Issue the CHANGE.RECON UPGRADE command by using either the IMS 14 DBRC Recovery Control utility (DSPURX00) or the IMS 14 DBRC Command API request. This command:
 - Upgrades the RECON data set without shutting down all IMS activity.
 - Uses the DBRC I/O recovery algorithms to recover from any failures during upgrade (so you do not need to back up the RECON data set before you upgrade).

After this command successfully completes, DBRC sets the value for MINVERS (the minimum version of IMS that can sign on to DBRC) to '12.1' if the value was less than '12.1'. You can display the MINVERS value in the output for a LIST.RECON command or a Query request using the DBRC API.

Recommendation: If you use DBRC command authorization, consider setting the RECON qualifier as part of your migration process. You can set the RECON qualifier when you upgrade by adding CMDAUTH parameters to the CHANGE.RECON UPGRADE command, or after the RECON has been upgraded by issuing a CHANGE.RECON CMDAUTH command. If CMDAUTH parameters are specified on the CHANGE.RECON UPGRADE command, the RECON is upgraded first and then other parameters (such as CMDAUTH) are processed.

For details about the CHANGE.RECON UPGRADE command, see *IMS Version 14 Commands, Volume 3: IMS Component and z/OS Commands*. For details about the Query request, see *IMS Version 14 System Programming APIs*.

6. Optionally, after the RECON data set for a system is upgraded, issue the REPAIR.RECON DMBNUM CHECKUP command against a copy of the production RECON data set. This command verifies that all of the data management block (DMB) numbers in the different records in the RECON data set are valid. If no problems are found, run the command once a year.

If problems are found when you verify the validity of the DMB numbers, issue the REPAIR.RECON DMBNUM UPDATE command against the production RECON data set when access to the RECON data set would be minimal.

Like the CHANGE.RECON UPGRADE, you can issue the REPAIR.RECON DMBNUM command by using either the IMS 14 DBRC Recovery Control utility (DSPURX00) or the IMS 14 DBRC Command API request.
7. When you are sure that a fallback to a previous IMS version is unnecessary and all systems that access the RECON data set are at an IMS 14 level, you can update the MINVERS value. Before you issue the CHANGE.RECON MINVERS('14.1') command, read "Minimum version value in the RECON data set" on page 69 to understand the ramifications involved in falling back to a previous version.

After you set the MINVERS level for an IMS system, system signon fails for earlier versions of IMS for online environments. All other jobs accessing the RECON data set fail DBRC initialization if the version of IMS used is lower than the MINVERS level.

Dynamic resource definition migration considerations

If you are migrating from IMS Version 12 or IMS Version 13, the process for enabling dynamic resource definition (DRD) in an IMS 14 system differs depending on whether your existing IMS system has DRD enabled. For MODBLKS resources, the process also depends on whether your resource definitions are stored in the IMSRSC repository or in the resource definition data set (RDDS).

Recommendations:

Before enabling dynamic resource definition or shared queues, evaluate any existing DFSINSX0 exit routines. The DFSINSX0 exit might need to be changed so that it checks whether LTERM creation is allowed before it accesses the USEQDATA parameter list that is related to LTERM processing. If LTERM creation is not allowed, the USEQDATA buffer address (INSXAUSQ) is zero.

After you enable DRD, ensure that the resource definitions are no longer loaded from the IMS.MODBLKS data set and the DFSCLL3x, DFSCLR0x, and DFSCLC0x members of the IMS.SDFSRESL data set during IMS cold starts. Otherwise, if you delete runtime resources with a DELETE command and then perform a cold start using the IMS.MODBLKS data set and the DFSCLL3x, DFSCLR0x, and DFSCLC0x members for the resource definitions, those runtime resources will reappear after the next IMS cold start. For MODBLKS and MSC resources, you can either perform another system definition that omits all those resources, or hereafter start IMS without the IMS.MODBLKS data sets and the DFSCLL3x, DFSCLR0x, and DFSCLC0x members defined. For MODBLKS resources, you can use the resource definitions from the RDDS instead. MODBLKS resources can be stored in the IMS.MODBLKS data set, RDDS or the IMSRSC repository, while MSC resources can be stored in the DFSCLL3x, DFSCLR0x, and DFSCLC0x members or the IMSRSC repository.

If resources need to be changed during migration, perform a MODBLKS online change operation or the system generation process for MSC resources on the IMS systems that have not migrated to DRD. Then, issue CREATE, UPDATE, and DELETE commands as necessary on the IMS systems that have migrated to DRD.

Recommendation: When migrating to DRD, for MODBLKS resources, use the IMSRSC repository instead of an RDDS, because the repository function is the strategic direction for IMS.

Related tasks:

 Enabling dynamic definition for IMS resource groups (System Definition)

Migrating a non-DRD-enabled IMS system to DRD and the IMSRSC repository:

To enable dynamic resource definition (DRD) with the IMSRSC repository, which is the recommended option for storing dynamically defined resources, you need to enable DRD and the Repository Server (RS) address space. For MODBLKS resources, you also need to import the resource definitions from the MODBLKS data set to the RDDS.

Recommendation: When you migrate to DRD for MODBLKS resources, use the IMSRSC repository instead of an RDDS because the repository function is the strategic direction for IMS.

To enable DRD with an IMSRSC repository:

1. Shut down IMS normally.
2. Define the following DRD parameters in the DYNAMIC_RESOURCES section of the DFSDFxxx member of the IMS.PROCLIB data set:
 - AUTOIMPORT=AUTO, which specifies that IMS automatically imports resource definitions during a cold start.

When AUTOIMPORT=AUTO, IMS searches data sources in the following order:

- a. IMSRSC repository
 - b. RDDS
 - c. MODBLKS data set and the DFSCLL3x member of the IMS.SDFSRESL data set
- During migration to IMS 14, if the IMSRSC repository is enabled and the DFSDFxxx member has the AUTOEXPORT=AUTO explicitly defined, automatic export of MODBLKS resources to the IMSRSC repository is enabled after IMS 14 is cold started. If dynamic definition of MSC resources is also enabled, MSC resources are also automatically exported to the repository after IMS 14 is cold started. Any resource definition changes (creates and updates) are automatically exported to the IMSRSC repository at the next checkpoint.

If you do not want to automatically export MODBLKS resource definitions to the IMSRSC repository, you must remove the AUTOEXPORT= parameter in the DYNAMIC_RESOURCES section of the DFSDFxxx member and either let it default to AUTO or use one of the following values:

- AUTOEXPORT= NO for no autoexport
- AUTOEXPORT = RDDS for autoexport to the RDDS

For MSC resources, automatic export must be enabled to store the MSC resources in the IMSRSC repository.

When you are ready to enable autoexport to the IMSRSC repository you must modify the AUTOEXPORT= parameter in the DYNAMIC RESOURCES section of the DFSDFxxx member to explicitly specify AUTOEXPORT=AUTO or AUTOEXPORT=REPO.

3. Enable DRD by specifying MODBLKS=DYN in either or both of the following locations:

- The DFSCGxxx member of the IMS.PROCLIB data set
- The DYNAMIC_RESOURCES section of the DFSDFxxx member of the IMS.PROCLIB data set

To enable DRD for MSC resources, you must also specify MSCRSCS=DYN in the MSC section of the DFSDFxxx member.

4. If you are enabling DRD for MSC resources, specify MSCREPO=Y in the MSC section of the DFSDFxxx PROCLIB member to enable the IMSRSC repository for MSC resources.
5. Specify attributes of the repository in the following members of the IMS PROCLIB data set:
 - FRPCFG member
 - BPE configuration parameters member
 - CSLRIxxx member
 - DFSDFxxx member
6. Create the repository catalog pairs of data sets.
7. Create the IMSRSC repository pairs of data sets.
8. Start the Repository Server (RS) address space.
9. Define the IMSRSC repository data sets to RS, which stores information about the IMSRSC repository in the RS catalog repository data sets.
10. Optional: Start the Common Queue Server (CQS) with a resource structure.
11. Enable the IMS Resource Manager (RM) to use the IMSRSC repository dynamically.
 - If the RM address space is running, issue the UPDATE RM command.
 - If RM is not running, restart RM so that it connects to the RS address space.
12. Before you start IMS, specify the EXEC parameters DFSDF=xxx and, if MODBLKS=DYN was specified in the DFSCGxxx member, CSLG=xxx. These parameters identify which DFSDFxxx and DFSCGxxx members in the IMS.PROCLIB data set to use.
13. Cold start IMS.

If the IMSRSC repository is empty, during cold start IMS imports the resource definitions from the IMS.MODBLKS data set and the DFSCLL3x, DFSCLR0x, and DFSCLC0x members of the IMS.SDFSRESL data set to create the runtime resource definitions. The online change process for the IMS.MODBLKS data set is now disabled.
14. After IMS is running, issue one of the following commands to export the runtime resource definitions to the IMSRSC repository:
 - For MODBLKS resources, the EXPORT DEFN TARGET(REPO) command
 - For all runtime resource definitions, including both MODBLKS and MSC resources, the /CHECKPOINT command

You can now use DRD to add (CREATE command), change (UPDATE command), or delete (DELETE command) the runtime resource definitions.

To ensure new and updated runtime resource definitions are recovered across an IMS cold start, issue the EXPORT DEFN TARGET(REPO) for MODBLKS resources or the /CHECKPOINT command for both MODBLKS and MSC resources to harden the definitions in the IMSRSC repository.

To ensure that deleted runtime resource definitions are not recovered across an IMS cold start, issue the DELETE DEFN command to delete the stored resource definitions from the IMSRSC repository.

Migrating a non-DRD-enabled IMS system to DRD and an RDDS:

To enable dynamic resource definition (DRD) with an resource definition data set (RDDS), you need to enable DRD, specify and allocate the RDDS, and import the resource definitions from the MODBLKS data set to the RDDS.

Recommendation: When migrating to DRD, use the IMSRSC repository instead of an RDDS, because the repository function is the strategic direction for IMS.

If resources need to be changed during migration, perform a MODBLKS online change operation on the IMS systems that have not migrated to DRD, and issue CREATE, UPDATE, and DELETE commands as necessary on the IMS systems that have migrated to DRD.

To enable DRD with an RDDS:

1. Shut down IMS normally.
2. Define the following DRD parameters in the <DYNAMIC_RESOURCES> section of the DFSDFxxx member of the IMS.PROCLIB data set:
 - RDDSДСN=(dsname_1, dsname_n)
These system definition data sets are for the resource definitions.
 - AUTOIMPORT=AUTO, which specifies that IMS automatically imports resource definitions. When AUTOIMPORT=AUTO, IMS checks the RDDS first. If the RDDS is empty, IMS loads the resource definitions from the IMS.MODBLKS data set. If the RDDS contains any resource definitions, IMS ignores any definitions in the IMS.MODBLKS data set.

Note: If AUTOIMPORT=MODBLKS is specified, instead of AUTOIMPORT=AUTO, your runtime definitions are imported from the IMS.MODBLKS data set during IMS cold start. Any changes you make to your online system will not be recovered across the cold start. You can either perform another system definition that includes the changes you made online, or hereafter start IMS with AUTOIMPORT=AUTO specified in the DFSDFxxx PROCLIB member.
 - AUTOEXPORT=AUTO, which specifies that IMS automatically exports resource definitions to the RDDS data sets, the IMSRSC repository, or both, during system checkpoints.
3. Enable DRD by specifying MODBLKS=DYN in either or both of the following locations:
 - The DFSCGxxx member of the IMS.PROCLIB data set
 - The <DYNAMIC_RESOURCES> section of the DFSDFxxx member of the IMS.PROCLIB data set
4. Allocate the system RDDS data sets that are specified on the RDDSДСN= parameter in the DFSDFxxx member.

5. Specify the IMS EXEC parameters DFSDF=xxx and, if MODBLKS=DYN was specified in the DFSCGxxx member, CSLG=xxx. These parameters identify which DFSDFxxx and DFSCGxxx members to use in the IMS.PROCLIB data set.
6. Cold start IMS. If the RDDS data sets are empty, during cold start IMS imports the resource definitions in the IMS.MODBLKS data set to create the runtime resource definitions. The online change process for the IMS.MODBLKS data set is now disabled.
7. Start using DRD to add (CREATE command), change (UPDATE command), or delete (DELETE command) the runtime resource definitions.
8. At system checkpoint time, IMS automatically exports the definitions to the RDDS if any changes have been made. To export any changes immediately after they have been made, either issue a /CHE command, or issue an /EXPORT command.
9. For subsequent cold starts of this IMS, specify AUTOIMPORT=RDDS in the <DYNAMIC_RESOURCES> section of the DFSDFxxx member of the IMS.PROCLIB data set. When AUTOIMPORT=RDDS is specified, IMS automatically imports resource and descriptor definitions from the RDDS during cold start processing.

Migrating a DRD-enabled IMS system that uses an RDDS to use the IMSRSC repository:

To migrate an IMS system in which dynamic resource definition is enabled from an RDDS to the IMSRSC repository, you need to enable the Repository Server (RS) address space, and import the resource definitions from the RDDS to the IMSRSC repository.

To migrate a DRD-enabled IMS system from an RDDS to the IMSRSC repository:

1. Create a non-system RDDS that contains definitions for the current runtime and descriptor resources. Use any of the following methods to create this RDDS:
 - Run the Create RDDS from the Log Records utility (DFSURCL0)
 - Run the Create RDDS from the MODBLKS utility (DFSURCM0)
 - Run the DRD IMS SYSGEN stage 1 pre-parser utility (DFSURST0)
 - Run the Copy RDDS utility (DFSURCP0)
 - Use the EXPORT command, if IMS is running
2. Specify attributes of the repository in the following members of the IMS PROCLIB data set:
 - FRPCFG member
 - BPE configuration parameters member
 - CSLRIxxx member
 - DFSDFxxx member
3. Create the repository catalog pairs of data sets.
4. Create the IMSRSC repository pairs of data sets.
5. Start the Repository Server (RS) address space.
6. Define the IMSRSC repository data sets to RS, which stores information about the IMSRSC repository in the RS catalog repository data sets.
7. Optional: Start the Common Queue Server (CQS) with a resource structure.
8. Enable the IMS Resource Manager (RM) to use the IMSRSC repository dynamically:
 - If the RM address space is running, issue the UPDATE RM command.

- If RM is not running, restart RM so that it connects to the RS address space.
9. Populate the IMSRSC repository with resource and descriptor definitions by running the batch RDDS to Repository utility (CSLURP10).
 10. Cold start IMS with AUTOIMPORT=AUTO specified or use the UPDATE IMS command if IMS is running.

Requirement: If AUTOEXPORT=RDDS or AUTOEXPORT=AUTO is specified in the DFSDFxxx PROCLIB member and RDDSs are defined, IMS continues to export resource definitions to the RDDSs after each system checkpoint, even if IMS is migrated to use the IMSRSC repository. To disable the AUTOEXPORT to RDDS function so that the RDDSs are no longer used:

1. Modify the DFSDFxxx PROCLIB member to specify AUTOEXPORT=REPO, AUTOEXPORT=AUTO, or AUTOEXPORT=NO.
2. Modify the DFSDFxxx PROCLIB member to remove the RDDS definitions.

If you use the UPDATE IMS command to turn off automatic export while IMS is up, the automatic export to the repository is also disabled.

Migrating MSC macro definitions to the IMSRSC repository:

You can migrate your MSC macro definitions to the IMSRSC repository to store the definitions in a single, centralized location for all IMS systems in an IMSplex. Migrating MSC definitions to the IMSRSC repository also ensures that the definitions are saved across an IMS cold start.

To migrate MSC macro definitions to the IMSRSC repository, complete the following steps:

1. Enable automatic export and automatic import by defining both of the following parameters in the DYNAMIC_RESOURCES section of the DFSDFxxx member of the IMS.PROCLIB data set:
 - AUTOEXPORT=AUTO or AUTOEXPORT=REPO
 - AUTOIMPORT=AUTO
2. Enable dynamic resource definition for MSC resources by specifying MSCRSCS=DYN in the MSC section of the DFSDFxxx member. Also ensure that MODBLKS=DYN is specified in either or both of the following locations:
 - The DFSCGxxx member of the IMS.PROCLIB data set
 - The COMMON_SERVICE_LAYER section of the DFSDFxxx member of the IMS.PROCLIB data set
3. Enable the IMSRSC repository for MSC resources by specifying MSCREPO=Y in the MSC section of the DFSDFxxx member.
4. Specify attributes of the repository in the following members of the IMS PROCLIB data set:
 - FRPCFG member
 - BPE configuration parameters member
 - CSLRIxxx member
 - DFSDFxxx member
5. Create the repository catalog pairs of data sets.
6. Create the IMSRSC repository pairs of data sets.
7. Start the Repository Server (RS) address space.
8. Define the IMSRSC repository data sets to RS, which stores information about the IMSRSC repository in the RS catalog repository data sets.

9. If you are enabling more than one IMS Resource Manager (RM), start the Common Queue Server (CQS) with a resource structure.
10. Enable RM to use the IMSRSC repository dynamically.
 - If the RM address space is running, issue the UPDATE RM command.
 - If RM is not running, restart RM so that it connects to the RS address space.
11. Specify the MSC=Y execution parameter in your startup procedure to initialize the MSC function.
12. If one or more logical link paths are not defined in an MSNAME stage-1 system definition macro, specify the SYSID= parameter in the MSC section of the DFSDFxxx member to define one or more local system IDs (SYSIDs) for the IMS system.
13. If you are cold starting IMS with IMSRSC repository support for MSC resources enabled for the first time and you want MSC resources to be automatically exported to the IMSRSC repository at the cold start, ensure that the following data sets contain the current MSC resource definitions:
 - The DFSCLL3x member of the IMS.SDFSRESL data set, for both MSC physical and logical link definitions
 - The DFSCLC0x member of the IMS.SDFSRESL data set, for MSC logical link path definitions
 - The DFSCLR0x member of the IMS.SDFSRESL data set, for remote logical terminal definitions
14. Specify the EXEC parameters DFSDF=xxx and, if MODBLKS=DYN was specified in the DFSCGxxx member, CSLG=xxx. These parameters identify which DFSDFxxx and DFSCGxxx members in the IMS.PROCLIB data set to use.
15. Cold start IMS. If the IMSRSC repository does not contain MSC resources, the MSC resource definitions that were generated during the system generation process are imported into the IMS system to create the runtime resource definitions. The runtime resource definitions are then automatically exported to the IMSRSC repository at the cold start.

If one of the following conditions are true, ensure that at least one system identifier (SYSID) is specified in the MSC section of the DFSDFxxx PROCLIB member. Otherwise, IMS cold start abends with ABENDU0741 M0DID MS0ABEND. At least one of the SYSIDs should be the lowest SYSID you plan to define for that IMS system.

 - AUTOIMPORT=AUTO is specified in the DYNAMIC_RESOURCES section of the DFSDFxxx member with no MSC resources in the IMSRSC repository and no MSC resources are defined in stage-1 system definition macros.
 - AUTOIMPORT=REPO is specified in the DYNAMIC_RESOURCES section of the DFSDFxxx member with no MSC resources in the IMSRSC repository.
 - AUTOIMPORT=NO is specified in the DYNAMIC_RESOURCES section of the DFSDFxxx member.
16. Optional: Issue the QUERY command for the type of MSC resource that you exported to the IMSRSC repository with the SHOW(DEFN) keyword specified. The local definitions in the IMS system and the global definitions in the IMSRSC repository are displayed. All of the MSC resource attributes should be identical between IMS and the IMSRSC repository.

After you migrate MSC macro definitions to the IMSRSC repository, the MSC resource definitions that you create or update since the last automatic export are


exported to the IMSRSC repository at the next IMS checkpoint. The IMS checkpoint can be initiated either by issuing the /CHECKPOINT command or automatically by the IMS system.

If you are using channel-to-channel (CTC) links, consider removing the DD definitions for the CTC links from the IMS JCL before you import the MSC resources from the IMSRSC repository. This allows the CTC addresses that are defined to be used for the CTC links that are imported from the IMSRSC repository.

If you use the IMSRSC repository to store dynamically defined MSC resources, the DFSCLL3x and DFSCLR0x members of the IMS.SDFSRESL data set are no longer required. Because the DFSCLC0x member might contain non-MSC resources, the DFSCLC0x might still be required. After you are satisfied with the setup of your DRD environment for MSC resources, the DRD environment is running successfully, and your MSC resources are exported to the IMSRSC repository, you can remove the DFSCLL3x and DFSCLR0x members. For the DFSCLC0x member, update the member to remove MSC logical link path definitions. If automatic import is enabled, MSC resource definitions are imported during IMS cold start from the repository that contains the most current data. However, you can continue to use the DFSCLL3x, DFSCLR0x, and DFSCLC0x members as the source for your MSC resource definitions, instead of a repository. If you continue to use the DFSCLL3x, DFSCLR0x, and DFSCLC0x members, keep your system definition macros synchronized with the changes you make dynamically using DRD commands.

If you migrate to using the IMSRSC repository for dynamically defined MSC resources but continue to use the DFSCLL3x, DFSCLR0x, and DFSCLC0x members of the IMS.SDFSRESL data set, keep resource definitions that are in the members synchronized with the resource definitions that are in the repository. This synchronization enables you to maintain viable DFSCLL3x, DFSCLR0x, and DFSCLC0x members if you must disable DRD and fall back to using the system generation process for MSC resources. To keep your DFSCLL3x, DFSCLR0x, and DFSCLC0x members synchronized with your online definitions, update your static macro definitions with the changes that you make dynamically using type-2 commands. When changes are made dynamically, perform a MSC system definition to add, change, or delete resources from the DFSCLL3x, DFSCLR0x, and DFSCLC0x members.

Related concepts:


 [Maintaining your dynamic resource definition environment \(System Definition\)](#)


Related tasks:

 [Defining the IMSRSC repository \(System Definition\)](#)

Related reference:


 [UPDATE RM command \(Commands\)](#)

 [/CHECKPOINT command \(Commands\)](#)

 [QUERY LTERM command \(Commands\)](#)

 [QUERY MSLINK command \(Commands\)](#)

 [QUERY MSNAME command \(Commands\)](#)

 [QUERY MSPLINK command \(Commands\)](#)

| [↗](#) DYNAMIC_RESOURCES section of the DFSDFxxx member (System
| Definition)

| [↗](#) MSC section of the DFSDFxxx member (System Definition)

| [↗](#) FRPCFG member of the IMS PROCLIB data set (System Definition)

| [↗](#) BPE configuration parameter member of the IMS PROCLIB data set (System
| Definition)

| [↗](#) CSLRIxxx member of the IMS PROCLIB data set (System Definition)

| **Related information:**

| [↗](#) 0741 (Messages and Codes)

| **RDDS JCL migration consideration:**

If you are migrating to IMS 14 from an IMS Version 12 system that does not have APAR/PTF PM35197/UK76853 installed and you have JCL that was created by the IMS Manage Resources ISPF panels (option 2.7.1.2) to create an RDDS from log records, and intend to use that JCL again, you must regenerate that JCL after migrating to IMS 14.

To regenerate the JCL, execute option 2.7.1.2 using the same field settings that created the original job and save the resulting JCL.

If you have not saved the JCL from the create job or do not intend to reuse that JCL, no action is required.

| **Exit routine migration considerations**

There are migration considerations for some exit routines when you migrate to IMS 14.

Service and enhancements to IMS can change or add information in the output of certain commands. Any exit routine or automation program that parses the output from these commands might need to be modified. Service changes to the output of a command are documented in ++HOLD statements that describe the changes.

| **Migration considerations introduced in IMS 14**

| In IMS 14 you can dynamically refresh and query type-2 Automated Operator user exit routines (DFSABOE00 and other AOIE type exit routines).

| To enable dynamic refresh and query support for the existing type-2 Automated Operator user exit routine (DFSABOE00), specify TYPE=AOIE,EXITS=(DFSABOE00) on the EXITDEF parameter in the USER_EXITS section of the DFSDFxxx member of the PROCLIB data set. No changes to the exit routine are required.

| After support for dynamically refreshing and querying type-2 Automated Operator user exit routines is enabled, you can modify, add, or delete the exit routines by issuing the REFRESH USEREXIT TYPE(AOIE) command. You can also display information about the exit routines with the QUERY USEREXIT TYPE(AOIE) command.

| **Migration considerations introduced in IMS Version 13**

If you are migrating to IMS 14 from IMS Version 12, you must also consider the enhancements to exit routines that were introduced in IMS Version 13.

IMS Version 13 introduces enhancements to the following exit routines:

- Signon/off security exit routine (DFSCSGN0)
- Security Reverification exit routine (DFSCCTSE0)
- Transaction Authorization exit routine (DFSCSTRN0)

For information about migrating these security exit routines, see “Security enhancements migration considerations” on page 66.

In IMS Version 13, you can dynamically refresh and query more types of exit routines.

To enable dynamic refresh and query support for existing exit routines, specify the exit routines on the EXITDEF parameter in the USER_EXITS section of the DFSDFxxx member. No changes to the exit routine itself are required.

After support for dynamically refreshing and querying an exit routine type is enabled, you can modify, add, or delete the exit routines of that type by issuing the REFRESH USEREXIT TYPE(*type*) command. You can also display information about the exit routines with the QUERY USEREXIT TYPE(*type*) command.

IMS Version 13 enables the following user exit types for the QUERY USEREXIT and REFRESH USEREXIT commands:

User exit type	Exit routine name
BSEX	Build Security Environment User Exit (DFSBSEX0)
LOGEDIT	Log Edit User Exit (DFSFLGE0)
LOGWRT	Logger User Exit (DFSFLGX0)
NDMX	Non-Discardable Messages User Exit (DFSNDMX0)
OTMAIOED	OTMA Input/Output Edit User Exit (DFSYIOE0)
OTMAYPRX	OTMA Destination Resolution User Exit (DFSYPRX0)
OTMARTUX	OTMA Resume TPIPE Security User Exit (DFSRTUX)
RASE	Resource Access Security User Exit (DFSRAS00)

IMS 64-bit storage manager migration considerations

In IMS 14, the QUERY POOL command is enhanced to provide more accurate feedback for syntax checking for the TYPE and SHOW keywords.

In earlier versions of IMS, when building QUERY POOL output with completion code text, the output was built with the CCText output header right-justified instead of left-justified. In IMS 14, the CCText header is now left-justified as it is with other type-2 commands.

IMSplex migration considerations

Migrating an IMSplex from one version of IMS to another is a complex process because many factors are involved and many different configurations are possible.

The following considerations apply when planning to migrate an IMSplex:

- Multiple Resource Managers (RMs) and Operations Managers (OMs) can run simultaneously in an IMSplex.
- Only one Structured Call Interface (SCI) can run at any given time in an IMSplex on a single logical partition (LPAR).

- All Common Queue Server (CQS) clients connected to a CQS address space must be stopped before shutting down that CQS.
- CQS clients are limited as to which version of CQS they can connect to. For more information about these rules, see “Common Queue Server coexistence considerations” on page 79.
- Install all appropriate coexistence service onto IMSplexes before attempting to migrate.
- If you are running multiple IMS systems on one logical partition (LPAR), migrate one IMS at a time.
- If you are running multiple LPARs, migrate one LPAR at a time.
- If you activate the automatic RECON loss notification and parallel RECON access functions within the same IMSplex, you must use the CHANGE.RECON IMSPLEX command to ensure that all DBRCs in the IMSplex are using the same IMSplex name that is specified in the RECON data set. If you use the DBRC SCI Registration exit routine (DSPSCIX0) or the IMSPLEX EXEC parameter before issuing the CHANGE.RECON IMSPLEX command, message DSP1136A is issued and subsequent jobs fail due to an unavailable RECON data set.
- Open Database Manager (ODBM) can only connect to the IMS systems that are of the same version as ODBM itself. In a mixed-version IMSplex, to limit ODBM connection to the IMS systems of the same version, list the eligible IMS systems as data stores in the CSLDCxxx member of the IMS PROCLIB data set.

Example 1: Migrating IMS systems on a single LPAR

The following IMSplex migration example assumes:

- Continuous availability is a high priority.
- All coexistence APARs have been installed.
- All IMS systems in the IMSplex are running on one LPAR (LPAR1).
- One IMS Version 12 (V12) CQS that is using a resource structure.
- One RM and one OM running on the LPAR.
- Two V12 IMS systems (named IMSA and IMSB) that are running on LPAR1 and participating in data sharing and shared queues.

To migrate the IMS systems in this example IMSplex to IMS 14 (V14):

1. Prepare the IMSA libraries and data sets for IMS 14 (for example, perform a system definition, set up JCL, and so on).
2. Stop the V12 SCI.
3. Start the V14 SCI.
4. Start the V14 OM.
Because there is only one LPAR in this example, and one OM must be running at all times to provide OM services, the higher-level OM is started before shutting down the lower-level OM.
5. Stop the V12 OM.
6. Shut down IMSA.
7. Shut down IMSB.
8. Stop the V12 RM.
9. Stop the V12 CQS.
10. Start the V14 CQS.
11. Start the V14 RM.
12. Cold start IMSA.

13. Restart IMSB.
14. Test IMSA.
15. After IMSA (V14) has tested successfully, proceed to the next step. If IMSA does not pass this testing step, consider whether IMSA needs to fall back to IMS Version 12 and repeat this process after any problems are fixed.
16. Stop the V12 OM.
17. Take IMSB offline and migrate it to IMS 14.

Example 2: Migrating IMS systems on multiple LPARs

The operational environment for this example consists of two LPARs and a total of three IMS systems that are participating in shared queues and data sharing.

The following IMSplex migration example assumes:

- Continuous availability is a high priority.
- All coexistence APARs have been installed.
- An example IMSplex consisting of:

LPAR1

- One V12 CQS
- One Common Service Layer (CSL), consisting of an OM, RM, and SCI
- One V12 IMS system named IMSC

LPAR2

- One V12 CQS
- One Common Service Layer (CSL), consisting of an OM, RM, and SCI
- Two V12 systems (named IMSD and IMSE)

Recommendation: Migrate less complex LPARs before migrating more complex LPARs.

To migrate the IMS systems in this example IMSplex to IMS 14 (V14):

1. Migrate the subsystems on LPAR1 first by performing the following actions:
 - a. Prepare the IMSC libraries and data sets for IMS 14 (for example, perform a system definition, set up JCL, and so on).
 - b. Stop the V12 SCI.
 - c. Start the V14 SCI.
 - d. Stop the V12 OM (assuming the same OM name is used before and after the migration).
 - e. Start the V14 OM.
 - f. Shut down IMSC.
 - g. Stop the V12 RM.
 - h. Shut down the V12 CQS.
 - i. Start the V14 CQS.
 - j. Start the V14 RM.
 - k. Coldstart IMSC.
 - l. Test IMSC. If IMSC does not pass this testing step, consider whether IMSC needs to fall back to IMS Version 12 and repeat this process after any problems are fixed.

2. After validating that the migration of IMSC on LPAR1 is successful, start migrating the IMS subsystems on LPAR2 with the following steps:
 - a. Prepare the IMSD libraries and data sets for IMS 14 (for example, perform a system definition, set up JCL, and so on).
 - b. Stop the V12 SCI.
 - c. Start the V14 SCI.
 - d. Stop the V12 OM.
 - e. Start the V14 OM.
 - f. Shut down IMSD.
 - g. Shut down IMSE.
 - h. Stop the V12 RM.
 - i. Stop the V12 CQS.
 - j. Start the V14 CQS.
 - k. Start the V14 RM.
 - l. Cold start IMSD.
 - m. Restart IMSE.
 - n. Test IMSD. If IMSD does not pass this testing step, consider whether IMSD needs to fall back to IMS Version 12 and repeat this process after any problems are fixed.
 - o. After IMSD (V14) has tested successfully, proceed to the next step.
 - p. Stop the V12 OM.
 - q. Take IMSE offline and migrate it to IMS 14.

Example 3: Migrating IMS systems on multiple LPARs (with IMS Connect involved)

The operational environment for this example consists of two LPARs and a total of three IMS systems that are participating in shared queues and data sharing.

The following IMSplex migration example assumes:

- Continuous availability is a high priority.
- All coexistence APARs have been installed.
- An example IMSplex consisting of:

LPAR1

- One V12 CQS
- One Common Service Layer (CSL), consisting of an OM, RM, and SCI
- One V12 IMS system named IMSF
- One IMS Connect that is communicating with IMSF, IMSG, and IMSH

LPAR2

- One V12 CQS
- One Common Service Layer (CSL), consisting of an OM, RM, and SCI
- Two V12 IMS systems named IMSG and IMSH

Recommendation: Migrate less complex LPARs before migrating more complex LPARs.

To migrate the IMS systems in this example IMSplex to IMS 14 (V14):

1. Migrate the subsystems on LPAR1 first by performing the following actions:

- a. Prepare the IMSF libraries and data sets for IMS 14 (for example, perform a system definition, set up JCL, and so on).
 - b. Stop the V12 SCI.
 - c. Start the V14 SCI.
 - d. Stop the V12 OM (assuming the same OM name is used before and after the migration).
 - e. Start the V14 OM.
 - f. Shut down IMS Connect.
 - g. Shut down IMSF.
 - h. Stop the V12 RM.
 - i. Shut down the V12 CQS.
 - j. Start the V14 CQS.
 - k. Start the V14 RM.
 - l. Coldstart IMSF.
 - m. Start IMS Connect. If you start IMS Connect before IMSF, IMS Connect issues a datastore unavailable message.
 - n. Test IMSF. If IMSF does not pass this testing step, consider whether IMSF needs to fall back to IMS Version 12 and repeat this process after any problems are fixed.
2. After validating that the migration of IMSF on LPAR1 is successful, start migrating the IMS subsystems on LPAR2 with the following steps:
- a. Prepare the IMSG libraries and data sets for IMS 14 (for example, perform a system definition, set up JCL, and so on).
 - b. Stop the V12 SCI.
 - c. Start the V14 SCI.
 - d. Start the V14 OM.
 - e. Stop the V12 OM.
 - f. Shut down IMSG.
 - g. Shut down IMSH.
 - h. Stop the V12 RM.
 - i. Stop the V12 CQS.
 - j. Start the V14 CQS.
 - k. Start the V14 RM.
 - l. Cold start IMSG.
 - m. Restart IMSH.
 - n. Test IMSG. If IMSG does not pass this testing step, consider whether IMSG needs to fall back to IMS Version 12 and repeat this process after any problems are fixed.
 - o. After IMSG (V14) has tested successfully, proceed to the next step.
 - p. Stop the V12 OM.
 - q. Take IMSH offline and migrate it to IMS 14.

INQY call for system services migration considerations

Application programs that use the AIB INQY ENVIRON call might need to be changed.

IMS Version 12 APAR/PTF PM42909/UK78069 increased the length of the data returned by the INQY ENVIRON call from 100 bytes to 108 bytes. As a result, the

minimum size of the AIB I/O area required for the INQY ENVIRON call increased from 152 bytes to 160 bytes and the offsets of the optional recovery token and application parameter fields have changed. If you need to modify the size of your application AIB I/O area, it is highly recommended you set the size to 512 bytes.

Any application programs that referenced the recovery token or application parameter field by its offset prior to IMS Version 12 APAR/PTF PM42909/UK78069 must be modified for continued access to the field, as described in the following recommendations. These fields must not be referenced by offset.

Recommendations: To account for expansion in the length of the reply data, specify an I/O area length of 512 bytes.

To reference the field that contains the recovery token or the application parameter string, code your application programs to locate the field by using the address of the field that is returned in the data output of the INQY ENVIRON call. This is the only valid programming technique to reference the recovery token field and the application parameter string field. No other programming technique should be used to reference these fields.

The recovery token or the application parameter string are optional and therefore are not always returned. If they are not returned, the value in the address field is zero.

If you are migrating to IMS 14 from IMS Version 12 without APAR/PTF PM42909/UK78069 and your applications uses the AIB INQY ENVIRON call, ensure that the application area for the INQY ENVIRON data is a minimum of 108 bytes and corresponding I/O area is a minimum of 160 bytes.

Application programs that issue the INQY ENVIRON call that have an I/O area less than the minimum size receive a status code AG. The I/O area will be filled with partial data, as much as would fit in the I/O area provided. AIBOALEN contains the length of the data returned in the I/O area. AIBOAUUSE contains the output area length that is required to receive all the data for the INQY ENVIRON call.

IVP enhancements migration considerations

Use the IMS installation verification program (IVP) after installing a new IMS system, to verify the installation of that new system, and use it subsequently for other purposes (for example, as an educational tool). The IVP dialogs are replaced when a new release of IMS is installed.

The IVP Variable Export utility mitigates the migration of IVP variables values between releases.

IMS Connect migration considerations

Many migration considerations pertain to migrating IMS Connect from IMS Version 12 or IMS Version 13 to IMS 14.

Consider the following items when planning to migrate IMS Connect support to IMS 14:

- In IMS Version 13, IMS Connect requires a region size that is at least 170 MB larger than the regions size that was required in IMS Version 12. The increase supports the IMS Connect use of CPOOL instead of STORAGE OBTAIN in the XCF Message exit.

- Automated programs that issue IMS Connect type-2 commands might need to be modified to handle certain type-2 commands that are now processed synchronously in IMS 14. IMS Connect type-2 commands that are processed synchronously return the full results of the command in the command response, instead of returning part of the response to the system console. The commands that are processed synchronously also include new completion codes. The following commands are processed synchronously in IMS 14:
 - CREATE IMSCON TYPE(DATASTORE) command
 - CREATE IMSCON TYPE(PORT) command
 - UPDATE IMSCON TYPE(DATASTORE) START(COMM) command
 - UPDATE IMSCON TYPE(IMSPLEX) START(COMM) command
 - UPDATE IMSCON TYPE(MSC) START(COMM) command
 - UPDATE IMSCON TYPE(ODBM) START(COMM) command
 - UPDATE IMSCON TYPE(PORT) START(COMM) command
 - UPDATE IMSCON TYPE(RMTIMSCON) START(COMM) command
- In IMS Version 13, the IMS Connect recorder trace records are expanded to capture the entire messages that are sent and received on DRDA sockets, as well as all messages that are sent to and received from OM and ODBM via SCI. Consequently, the amount of trace information produced in IMS Version 13 might be significantly higher than in previous releases. Make sure that BPE external tracing is enabled to account for the increase in information.
- Automated application programs that interpret the output from the VIEWHWS and QUERY MEMBER commands must be changed to take advantage of:
 - New output fields added in IMS 14
 - New output fields added in IMS Version 13

IMS repository function migration considerations

There are several migration considerations related to the IMS repository enhancements.

Automatic export to the IMSRSC repository

If AUTOEXPORT=AUTO is specified in the DFSDFxxx member and IMS is enabled with the IMSRSC repository, autoexport to the IMSRSC repository is enabled at system checkpoints. To disable the autoexport to the IMSRSC repository function during migration to IMS 14, modify the DFSDFxxx member in one of the following ways:

- Remove the AUTOEXPORT= specification so that AUTOEXPORT will default to AUTO and not trigger the autoexport to the IMSRSC repository. If IMS also has system RDDS, then autoexport will be done to the RDDS.
- Modify to AUTOEXPORT=NO so that no autoexport is enabled.
- Modify to AUTOEXPORT=RDDS so that autoexport is only to the system RDDS and not the IMSRSC repository.

Remember: Autoexport to the IMSRSC repository is not enabled if AUTOEXPORT=AUTO is defaulted to. You must explicitly specify autoexport in the DFSDFxxx member.

If you have AUTOEXPORT=AUTO defined with IMSRSC repository enabled and no system RDDS defined, IMS Version 13 and earlier will disable autoexport (AUTOEXPORT=N) since no system RDDS data sets are defined. A DFS3374W message is issued to indicate that autoexport is disabled.

If you have AUTOEXPORT=AUTO explicitly specified and not defaulted to with IMSRSC repository enabled and no system RDDS defined, IMS 14 will, at the end of the next checkpoint, export the changed definitions automatically since the last export to the IMSRSC repository.

If you have AUTOEXPORT=AUTO defined with IMSRSC repository enabled and system RDDS defined, IMS Version 13 and earlier will export the changed definitions automatically to the system RDDS at the end of the next checkpoint.

If you have AUTOEXPORT=AUTO defined with IMSRSC repository defined and system RDDS defined, IMS 14 will, at the end of the next checkpoint, export the changed definitions automatically since the last export to the IMSRSC repository. IMS will also export the changed definitions automatically to the system RDDS. The automatic export to the system RDDS will occur even if the automatic export to the IMSRSC repository did not succeed.

Create RDDS from Log Records utility (DFSURCL0) migration considerations

Run the same version of the DFSURCL0 utility as the version of the IMS that produced the IMS logs being used as input.

For example, use the IMS 14 DFSURCL0 utility with IMS 14 log data sets. If you run with mixed versions, the results can be unpredictable.

Remote Site Recovery migration

To avoid a loss of disaster recovery protection during the migration period, plan for an orderly migration to IMS 14 of IMS systems that participate in Remote Site Recovery (RSR). Migrating RSR to IMS 14 does not depend on any non-IMS software other than that required for RSR to operate.

Planning and administration tasks

The system programmer, system administrator, and the database administrator must carefully plan the migration to IMS 14 of IMS systems that participate in RSR. Also, automation programs and procedures might need to be created or modified based on messages issued during an RSR migration.

To take advantage of added IMS function, RSR users must migrate from IMS Version 12 and IMS Version 13 to IMS 14. The migration path must be chosen carefully for the IMS subsystems that are participating in RSR. Migration of any IMS system that participates in RSR includes all the steps that are required to move to the new level.

To migrate from IMS Version 12 or IMS Version 13 to IMS 14 in an RSR environment:

1. Migrate the tracking IMS and Transport Manager Subsystem (TMS) to the IMS 14 level.
2. Migrate the active site TMS that is running isolated log sender (ILS) to the IMS 14 level. You must complete this step before or during migration of the first active IMS. Stop ILS on all the other TMSs so that the tracking IMS does not attempt to establish conversations with an ILS at an earlier level.
3. Migrate the rest of the active IMS subsystems and active site TMS systems to the IMS 14 level as necessary.

Pre-14 library retention

Retain all pre-IMS 14 libraries that are required for processing the pre-IMS 14 tracking log data sets until there is no possibility of needing the pre-IMS 14 log data sets. If a tracking log data set must be recovered for any reason, it must be processed by the Log Recovery utility (DFSULTR0) from the same version of IMS. Therefore, even after a remote takeover, you might need to retain pre-IMS 14 libraries at the remote site, even if all active IMS subsystems have been migrated to IMS 14.

Takeover considerations

Active IMS systems must restart at the remote site after a remote takeover to access to tracking log data sets (for normal and emergency restart). The new active IMS systems must run at the same IMS version level as the old active IMS systems that created the log records that are used for restart processing.

Handling error situations

If the tracking IMS terminates abnormally while at the pre-IMS 14 level, perform an emergency restart at the pre-IMS 14 level and successfully shut down before you migrate to IMS 14. Doing so allows the tracking IMS to create an RSR position data set that is used during cold start processing of the new IMS 14 RSR tracker. The RSR position data set helps the RSR tracker reestablish routing positions on the various PRILOG records.

If a deallocate or close error message for the RSR position data set is issued by the IMS Version 12 or IMS Version 13 tracking IMS during shutdown, resolve the problem that is causing the error before you migrate the tracking IMS system to IMS 14. If the problem causes an error during RSR position data set allocation such that the tracking IMS is unable to find or access the position data set, you can continue normal cold-start processing. However, normal cold-start processing will result in the tracking IMS processing only log data from new active IMS systems because the tracking IMS has no indication of where the previous tracking IMS stopped. You can restart the tracking IMS at the pre-IMS 14 level if a new RSR position data set needs to be created to alleviate this problem.

The following steps describe what can happen if an active IMS spans the migration of a tracking IMS and an RSR position data set is not available:

1. Log data is received at the tracking site from an active IMS Version 12 or IMS Version 13 system before migrating the tracking IMS to IMS 14 (the active PRILOG is not closed).
2. The tracking IMS migrates to an IMS 14 level but the active IMS does not shut down (the PRILOG did not close in the active site RECON data sets).
3. After a cold start, the IMS 14 tracker is unable to find the RSR position data set.
4. The user elects to continue cold start processing.
5. Log data is received from the active IMS with the same PRILOG start time as was processed by the previous release tracking IMS. This log data is not applied to any remote site database.

RSR migration path to IMS 14

The following table describes steps for migrating IMS and TMS from IMS Version 12 or IMS Version 13 to IMS 14, and the combinations of IMS systems and resources that are supported. In the table:

- The Steps column shows the order in which you can migrate the components of the RSR complex (where A = Active and R = Remote).
- The bold type in a table cell shows in which IMS version the current changes occur.

Table 9. Supported paths and configurations for migrating RSR to IMS 14

Steps		Active site state			Remote site state			Notes
A	R	IMS	TMS/ILS	RECON	IMS	TMS/ILS	RECON	
0	0	pre-V14	pre-V14	pre-V14	pre-V14	pre-V14	pre-V14	1
1	1	pre-V14	pre-V14	V14	pre-V14	pre-V14	V14	2
1	2	pre-V14	pre-V14	V14	V14	V14	V14	3
2	2	pre-V14	V14	V14	V14	V14	V14	4
3	2	V14	V14	V14	V14	V14	V14	5

Note:

1. (0,0) - This is the initial state. All subsystems at both the active and remote sites are running at the pre-IMS 14 level.
2. (1,1) - Both RECON data sets are upgraded by using the CHANGE.RECON UPGRADE command. It does not matter which site (active or remote) is run first. It only matters that both RECON data sets are upgraded before proceeding.
3. (1,2) - The tracking IMS and remote site TMS are migrated to IMS 14. These IMS systems can be migrated separately, but to ensure integrity, simultaneous migration is preferable.
4. (2,2) - The active site TMS is migrated to IMS 14. This can be accomplished as part of migrating the active IMS to IMS 14. All non-IMS 14 isolated log senders (ILSs) must be stopped and ILS started on the IMS 14 TMS so that missing log data from the IMS 14 IMS can be sent to the tracking IMS.
5. (3,2) - The active IMS subsystems are migrated to IMS 14. Each IMS subsystem can be migrated independently.

Alternatives to RSR migration and coexistence

The alternatives to RSR migration and coexistence are:

- Require IMS users of the RSR function to migrate all systems that are participating in RSR to IMS 14 simultaneously.
- Disable the RSR function while active site systems migrate to IMS 14.

Changed interfaces

The interface between the log router and various other components has been changed to support multiple release log record handling. The interfaces of the following components have changed:

- Active site IMS logger
- Active system tracker
- DBRC

- DL/I tracker
- Fast Path tracker
- Isolated log sender

Security enhancements migration considerations

The SECURITY macro is no longer used in IMS Version 13 and later IMS systems. Also, some security exit routines might need to be changed when migrating from IMS Version 12 or earlier.

SECURITY macro migration considerations

IMS Version 12 is the last version to support the SECURITY macro. You can use initialization parameters to specify most of the SECURITY macro keyword values.

If the SECURITY macro is specified in an IMS Version 13 or later IMS system, IMS issues message G115 and continues processing the stage 1 system definition input. For fallback purposes, you can leave the SECURITY macro in the IMS Version 13 stage 1 system definition input, in which case, you can ignore message G115.

To prepare you IMS system for the removal of the SECURITY macro, move your security specifications from the SECURITY macro to initialization parameters, use the following procedure:

1. Before you migrate to IMS 14, replace the SECLVL and TYPE security specifications in the SECURITY macro with the corresponding security specifications in the DFSPBxxx PROCLIB member.

The following table shows SECLVL and TYPE specifications in the SECURITY macro parameters and the parameter specifications in the DFSPBxxx PROCLIB member that replace them.

Table 10. EXEC parameters available in IMS Version 12 listed by the SECURITY macro parameters that they replace

SECURITY macro parameter	Corresponding DFSPBxxx parameter
SECLVL=NOSIGN	SGN=N
SECLVL=SIGNON	SGN=Y
SECLVL=FORCSIGN	SGN=F
SECLVL=NOTRAN	TRN=N
SECLVL=TRANAUTH	TRN=Y
SECLVL=FORCTRAN	TRN=F
TYPE=NORACTRM+NORACFCM	RCF=N
TYPE=RACFTERM	RCF=T
TYPE=RACFCOM	<ul style="list-style-type: none"> • RCF=C (specifies command authorization for ETO terminals) • RCF=S (specifies command authorization for ETO terminals and statically defined terminals)

Table 10. EXEC parameters available in IMS Version 12 listed by the SECURITY macro parameters that they replace (continued)

SECURITY macro parameter	Corresponding DFSPBxxx parameter
TYPE=RACFTERM+RACFCOM	<ul style="list-style-type: none"> • RCF=A (specifies command authorization for both ETO and statically defined terminals, sign on security, and transaction authorization) • RCF=Y (specifies command authorization for ETO terminals, sign on security, and transaction authorization)
TYPE=NORAS	ISIS=N
TYPE=RASRACF	ISIS=R
TYPE=RASEXIT	ISIS=C
TYPE=RAS	ISIS=A
TYPE=SIGNEXIT, TYPE=NOSIGNEX	IMS detects and loads the exit routines that are linked in one of the IMS STEPLIB libraries or in LINKLST. See "Security exit routines migration considerations."
TYPE=TRANEXIT, TYPE=NOTRANEX	IMS detects and loads the exit routines that are linked in one of the IMS STEPLIB libraries or in LINKLST. See "Security exit routines migration considerations."

2. Move SECCNT parameter to the COMM or MSGEN macro.
3. Move the RCLASS and SECCNT parameter specifications from the SECURITY macro to the appropriate PROCLIB member.
 - For DB/DC or DCCTL systems, you can move both the RCLASS and SECCNT parameter specifications to the DFSDCxxx PROCLIB member.
 - For all system types, you can move the RCLASS parameter specification to the DFSPBxxx PROCLIB member.
4. Comment out the SECURITY macro in the stage 1 system definition input. For fallback purposes, do not delete it.

If different versions of IMS use the same PROCLIB members, do not comment out the SECURITY macro. In this case, you can ignore the message and return code issued by the IMS 14 system definition and initialization processes. For more coexistence information, see "Security enhancements coexistence considerations" on page 92.
5. Start using IMS 14 with the security options specified in initialization parameters.
6. When there is no need to fall back to or coexist with a previous version of IMS, remove the SECURITY macro from the stage 1 system definition input.

For information about using initialization parameters for security, see "Controlling security during system startup for DB/DC and DCCTL" in the *IMS Version 14 System Administration*.

Security exit routines migration considerations

If you use either the Signon/off Security exit routine (DFSCSGN0) or the Transaction Authorization exit routine (DFSCSTRN0), you do not need to specify an initialization parameter to load these exit routines. IMS detects and loads the exit

routines that are linked in one of the IMS STEPLIB libraries or in LINKLST. A new message, DFS1937I, is issued when they are loaded.

As of IMS Version 13, the following security exit routines are no longer bound to the IMS nucleus:

- Signon/off security exit routine (DFSCSGN0)
- Security Reverification exit routine (DFSCTSE0)
- Transaction Authorization exit routine (DFSCTRN0)

Before IMS Version 13, if DFSCTSE0 was used, you included the DSECT in DFSCTRN0. As of IMS Version 13, all the security exit routines can be linked independently.

Recommendations:

- Split DFSCTSE0 and DFSCTRN0 into two separate load modules.
- Link the modules as reentrant (RENT) and AMODE/RMODE 31. If the load modules are not linked as reentrant, IMS loads them multiple times.
- If you change DFSCSGN0, DFSCTRN0 or DFSCTSE0, make sure to keep a copy of the old version for fallback purposes.

If the exit routines cannot be linked separately or cannot use a common work area, they must be linked in the following manner:

- If the CSECT of DFSCTSE0 is part of DFSCTRN0 source, DFSCTSE0 must be linked as an ALIAS of DFSCTRN0.
- If virtual address spaces are used to exchange data between DFSCSGN0, DFSCTRN0, and DFSCTSE0, then DFSCTSE0 and DFSCSGN0 must be linked as ALIASs of DFSCTRN0.

Syntax Checker enhancements migration considerations

The Syntax Checker assists with IMS release-to-release migrations by providing the ability to convert supported IMS.PROCLIB members from one release to the other.

When you use the Syntax Checker to check parameters for earlier releases of IMS, you must verify that the correct release number is displayed.

Fallback considerations

Major functions of IMS are generally not compatible with earlier versions.

Consider the following guidelines when preparing your migration fallback plan. This information is a guide to fallback inhibitors, and is not comprehensive:

- If your IMS system uses IMS-managed ACBs...
- The ACB Maintenance utility, ACBGEN, is required.
- For each IMS that you are migrating to a pre-14 level, complete the following steps:
 1. Ensure that the status of all databases and PSBs updated by IMS 14 are correct.
 2. Resolve DBRC issues. See “DBRC fallback considerations” on page 69.
 3. Shut down IMS 14.
 4. Install the version of IMS that you want.
 5. Cold start the IMS.
- You can use the IBM IMS Queue Control Facility for z/OS (QCF) to requeue IMS 14 messages to IMS Version 12 or IMS Version 13 message queues.

If you fall back from IMS 14 to IMS Version 13, ensure that the IMS Version 13 DBRC Coexistence APAR PI10132 (PTF UI22258) is applied to the IMS Version 13 system.

If you fall back from IMS 14 to IMS Version 12, ensure that the IMS Version 12 DBRC Coexistence APAR PI10131 (PTF UI22257) is applied to the IMS Version 12 system.

DBRC fallback considerations

Certain steps must be taken to revert the level of DBRC from IMS 14 to IMS Version 12 or IMS Version 13.

Minimum version value in the RECON data set

If the MINVERS value is set to '14.1' and you need to fall back from IMS 14 to an earlier version, you must lower the MINVERS value because jobs in an earlier release cannot access the RECON data if the current MINVERS value is set to '14.1'.


To change the MINVERS value to '12.1', complete the following steps:

1. Shut down all IMS 14 subsystems.
2. Ensure that all IMS 14 subsystem records have been removed from the RECON data set. Issue a LIST.SUBSYS command to see the subsystem records in the RECON data set. Delete all IMS 14 subsystem records in the RECON data set using the DELETE.SUBSYS command.
3. If any HALDB databases are defined to support 8-GB OSAM data sets, take the appropriate actions to change the maximum OSAM data set size from 8 GB to 4 GB. Otherwise, the CHANGE.RECON MINVERS('12.1') command will fail, and message DSP1256E will be issued.
4. Reset the MINVERS value by issuing a CHANGE.RECON MINVERS('12.1') command using IMS 14. If any databases or areas are in the process of being altered, messages DSP1249E, DSP1250E, and DSP1251E are issued, and the CHANGE.RECON command will fail. The structure changes need to either finish or be undone before the MINVERS value is lowered.

To change the MINVERS value to '13.1', complete the following steps:

1. Shut down all IMS 14 subsystems.
2. Ensure that all IMS 14 subsystem records have been removed from the RECON data set. Issue a LIST.SUBSYS command to see the subsystem records in the RECON data set. Delete all IMS 14 subsystem records in the RECON data set using the DELETE.SUBSYS command.
3. If the cross-DBRC service level ID (CDSLID) is not set to 2 and any HALDB databases are defined to support 8-GB OSAM data sets, the CHANGE.RECON MINVERS('13.1') command will fail, and message DSP1256E will be issued. To prevent the command from failing, take one of the following actions:
 - Complete the appropriate steps to change the maximum OSAM data set size for each HALDB from 8 GB to 4 GB.
 - If the appropriate maintenance for HALDB support of 8-GB OSAM data sets has been applied to all of the IMS Version 13 resident libraries (RESLIBs) that access the RECON data set, include the CDSLID(2) parameter in the CHANGE.RECON MINVERS('13.1') command in the next step.
4. Reset the MINVERS value by issuing a CHANGE.RECON MINVERS('13.1') command using IMS 14.

Related tasks:

 Changing the maximum OSAM data set size for a HALDB from 8 GB to 4 GB (Database Administration)

Dynamic definition for MODBLKS resources fallback considerations

You can fall back from an IMS system that has dynamic definition (DRD) enabled for MODBLKS resources to an earlier version of IMS.

To fall back from an IMS 14 system that uses the IMS repository function to an IMS Version 13 or IMS Version 12 system that uses a resource definition data set (RDDS):

1. Ensure that you have a non-system RDDS that contains a complete set of all the resource definitions before shutting down IMS 14.
2. Cold start the IMS Version 13 or IMS Version 12 system and import the resource definitions from the RDDS.

To fall back from an IMS system that has DRD enabled and uses an RDDS to an IMS system that does not use DRD:

1. Ensure that the IMS.MODBLKS data set has a complete set of all the resource definitions while you have DRD enabled. This set of resource definitions includes resource definitions that were originally in the IMS.MODBLKS data set at cold start time, and the resource definitions (or removal thereof) for the resources that were added, changed, or deleted dynamically. Having this set enables you to fall back to using online change for the IMS.MODBLKS data set with all the resources that were defined either by system definition originally or dynamically.

Optionally, you can use the Resource Definition Data Set (RDDS) Extraction utility (DFSURDD0) to extract the resource definitions to create Stage-1 macro statements from the stored resource definitions in an RDDS.

2. Shut down IMS normally.
3. Remove or change the MODBLKS keyword to enable online change for the IMS.MODBLKS data set:
 - Remove the MODBLKS keyword from the DFSCGxxx member or the DFSDFxxx member of the IMS.PROCLIB data set (or from both members). If both members are defined, any values specified in the DFSCGxxx member override the values specified in the DFSDFxxx member.
 - Change the value of the MODBLKS keyword from DYN to OLC in the DFSCGxxx member or in the <COMMON_SERVICE_LAYER> section of the DFSDFxxx member of the IMS.PROCLIB data set (or in both members).
4. Ensure that the IMS JCL includes the MODBLKS DD statement.
5. Cold start IMS. An IMS cold start creates runtime resource definitions from the stored resource definitions in the IMS.MODBLKS data set. The online change process for the IMS.MODBLKS data set is now enabled. Variations of the dynamic resource definition CREATE, DELETE, and UPDATE commands that change definitions are no longer permitted.
6. Reinstigate your in-house procedures that use the online change process for the IMS.MODBLKS data set and disable the procedures that use DRD commands.
7. Perform other pertinent tasks related to falling back to a previous version.

Dynamic definition of MSC resources fallback considerations

You can fall back from an IMS system that has dynamic definition enabled for MSC resources to an earlier version of IMS.

Complete the following procedure to fall back from an IMS system that has dynamic definition enabled for MSC resources to an IMS system that does not have dynamic definition enabled for MSC resources:

1. Keep your stage 1 system definition synchronized with dynamic definition of MSC resources.
2. Run system generation for stage 1 system definition that includes MSC resources, which are the following:
 - a. MSPLINK, MSLINK, MSNAME, and NAME macros
 - b. IMSCTRL macro with MSVID and SYSTEM=(MSVERIFY) parameters specified
 - c. APPLCTN and TRANSACT macros with the SYSID parameter specified
3. Restore old copies of MSC user exits.
4. Restore old copies of automation programs that issue QUERY LTERM, QUERY MSLINK, QUERY MSNAME, and QUERY MSPLINK commands and parse the output.
5. Remove new MSC parameters from the DFSDFxxx member of the IMS PROCLIB data set.
6. Cold start IMS at a previous IMS release level.
7. Complete other pertinent tasks related to falling back to a previous version of IMS.
8. Disable Common Service Layer (CSL) with the Structured Call Interface (SCI) and the Operations Manager (OM) if these are not needed for any other IMS function.
9. Confirm that dynamic definition is disabled for MSC resources by issuing the QUERY MEMBER TYPE(IMS) command, and ensuring that the command output does not include DYNMSC in the local attributes.

IMS catalog fallback

An IMS system in which the IMS catalog is enabled can fall back to an earlier version of IMS.

If you are falling back to a prior version of IMS in which the IMS catalog was enabled, unless the DBD and PSB instances from the prior version were purged from the IMS catalog, you do not need to change the IMS catalog.

If the DBD and PSB instances from the prior version were not purged, fallback to the IMS catalog of the prior version is achieved by making the ACB library (ACBLIB) of the prior version the active ACBLIB again. The active ACBLIB determines which DBD and PSB instances are active in the IMS catalog.

If the DBD and PSB instances from the prior version were purged from the IMS catalog, you must repopulate the IMS catalog with the DBD and PSB instances from the prior version after you make the ACBLIB from the prior version active.

After fallback is complete, you can leave the IMS 14 DBD and PSB instances in the IMS catalog or you can purge them to save storage.

The following figure shows the IMS catalog before fallback. The ACBLIB for the new version is active. The new DBDs are used in the DMB pool of the new version of IMS and the new DBD instances in the DBD record in the IMS catalog are used.

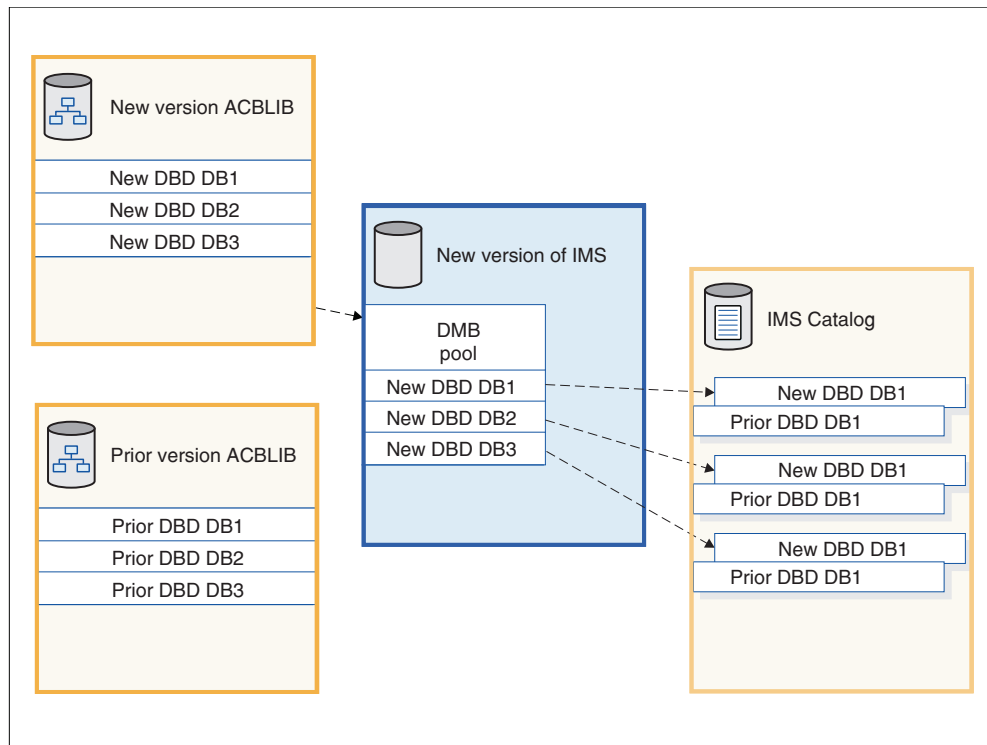


Figure 1. DBDs that IMS uses before fallback

The following figure shows the IMS catalog after fallback. The ACBLIB for the prior version is made active again. The prior version of IMS loads the prior DBDs into the DMB pool and uses the prior DBD instances in the DBD records in the IMS catalog.

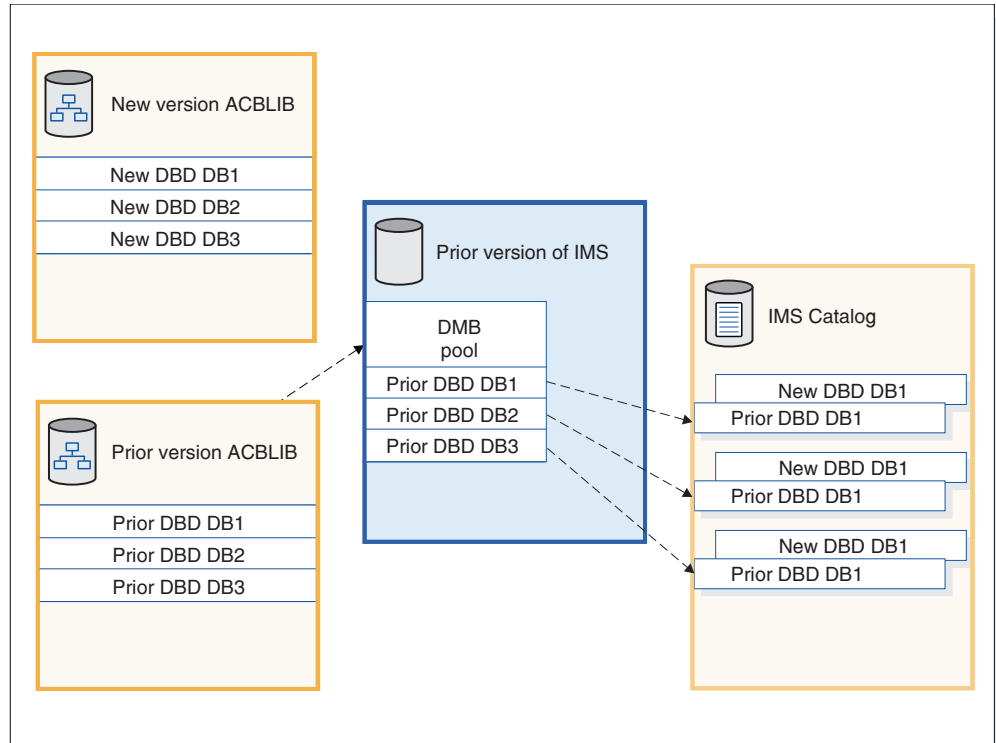


Figure 2. DBDs that IMS uses after fallback

Fallback from IMS-managed ACBs to ACBLIBs

The procedure for falling back from the IMS management of ACBs to the installation-managed ACB libraries (ACBLIBs) is different depending on whether your installation has activated any new or modified databases or program views since the IMS management of ACBs was enabled.

Falling back from IMS-managed ACBs before resource updates are activated

If your installation has not activated any new or modified databases or program views since enabling the IMS management of ACBs and your ACB, DBD, and PSB libraries are unchanged, the procedure for falling back from the IMS management of ACBs to the installation-managed ACB libraries (ACBLIBs) involves updating the DFSDFxxx member, changing various references to the IMS catalog back to references to the ACB, DBD, and PSB libraries, and restarting IMS.

The following procedure does not include steps for falling back from an IMS catalog. Disabling the IMS catalog is not required when falling back to installation-managed ACBLIBs.

The following procedure is generally applicable to both falling back within a release and falling back to a prior release of IMS.

1. In the <CATALOG> section of the DFSDFxxx PROCLIB member, specify ACBMGMT=ACBLIB.
2. If you use the IMS Catalog Definition exit routine (DFS3CDX0), modify it to indicate that the IMS management of ACBs is not enabled.
3. If necessary, update any JCL to reference the ACBLIB, DBDLIB, and PSBLIB.

4. If DBRC was configured to refer to the IMS catalog for database definitions, reconfigure DBRC to refer to the DBDLIB instead by issuing the CHANGE.RECON command with NOCATALG specified.
5. Restart IMS.

Before running the IMS Catalog Populate utility (DFS3PU00) again on a system that uses ACB libraries, remove the MANAGEDACBS control statement from the JCL.

If necessary, revert any tools or products that were updated to run in the IMS-managed ACBs environment so that they use the ACB, DBD, and PSB libraries again.

Falling back from IMS-managed ACBs to installation-managed ACBLIBs

If your installation has activated new or modified databases or program views after enabling the IMS management of ACBs, the procedure for falling back to installation-managed ACB libraries (ACBLIBs) includes recreating the DBD, PSB, and ACB libraries from the active resources in the IMS catalog.

The following procedure does not include steps for falling back from an IMS catalog. Disabling the IMS catalog is not required when falling back to installation-managed ACBLIBs.

The following procedure is generally applicable to both falling back within a release and falling back to a prior release of IMS.

1. Build the input statements for DBD Generation and PSB Generation utilities from the IMS catalog by using the DFS3LU00 utility.
2. Build the DBD and PSB libraries by running the DBD and PSB Generation utilities. If you are falling back to a prior release of IMS, be sure to use the utilities for that release of IMS.
3. Build the ACB library by providing the DBD and PSB libraries as input to the ACB Maintenance utility. If you are falling back to a prior release of IMS, be sure to use the utilities for that release of IMS.
4. In the <CATALOG> section of the DFSDFxxx PROCLIB member, specify ACBMGMT=ACBLIB.
5. If you use the IMS Catalog Definition exit routine (DFS3CDX0), modify it to indicate that the IMS management of ACBs is not enabled.
6. If necessary, update any JCL to reference the ACBLIB, DBDLIB, and PSBLIB.
7. If DBRC was configured to refer to the IMS catalog for database definitions, reconfigure DBRC to refer to the DBDLIB instead by issuing the CHANGE.RECON command with NOCATALG specified.
8. Restart the IMS system.

Remove the MANAGEDACBS control statement from the JCL for the IMS Catalog Populate utility (DFS3PU00) before running the utility again on a system that uses ACB libraries.

If necessary, revert any tools or products that were updated to run in the IMS-managed ACBs environment so that they use the ACB, DBD, and PSB libraries again.

Falling back from IMSRSC repository support for dynamically defined MSC resources

You can fall back from an IMS system that has IMSRSC repository support for dynamically defined MSC resources to an earlier version of IMS.

1. Keep your stage 1 system definition synchronized with dynamic definition of MSC resources.
2. Issue the DELETE DEFN command for all of the MSC resource types that have resource definitions in the IMSRSC repository for the IMS that is falling back. Issue the DELETE DEFN command with keywords FOR(IMSID(IMS1)) and NAME(*) for each MSC resource type. For example, if IMS1 is falling back, to delete all MSC resources, issue the following commands in the following order:

```
DELETE DEFN TARGET(REPO) NAME(*) TYPE(TRAN)FOR(IMSID(IMS1))
DELETE DEFN TARGET(REPO) NAME(*) TYPE(LTERM)FOR(IMSID(IMS1))
DELETE DEFN TARGET(REPO) NAME(*) TYPE(MSNAME)FOR(IMSID(IMS1))
DELETE DEFN TARGET(REPO) NAME(*) TYPE(MSLINK)FOR(IMSID(IMS1))
DELETE DEFN TARGET(REPO) NAME(*) TYPE(MSPLINK)FOR(IMSID(IMS1))
```

3. Run system generation for stage 1 system definition that includes MSC resources, which are the following:
 - MSPLINK, MSLINK, MSNAME, and NAME macros
 - IMSCTRL macro with MSVID and SYSTEM=(MSVERIFY) parameters specified
 - APPLCTN and TRANSACT macros with the SYSID parameter specified
4. Restore old copies of MSC user exits.
5. Restore old copies of automation programs that issue QUERY LTERM, QUERY MSLINK, QUERY MSNAME, and QUERY MSPLINK commands and parse the output.
6. Remove new MSC parameters from the DFSDFxxx member of the IMS PROCLIB data set.
7. Cold start IMS at a previous IMS release level.
8. Complete other pertinent tasks related to falling back to a previous version of IMS.
9. Confirm that dynamic definition is disabled for MSC resources by issuing the QUERY MEMBER TYPE(IMS) command, and ensuring that the command output does not include DYNMSC in the local attributes.

Chapter 6. Coexistence with IMS 14

Restrictions and compatibility considerations apply for coexistence of IMS 14 with earlier versions of IMS.

If an IMS 14 enhancement, or any other IMS function, has no coexistence or compatibility considerations, it is not described in these topics.

Overview of coexistence APARs

IMS Version 12 and IMS Version 13 must have certain APARs installed to coexist with IMS 14.

The following table describes the coexistence APARs and PTFs needed for various IMS functions. For detailed information about individual APARs and PTFs, go to the IMS home page at www.ibm.com/ims and click **Support**.

Table 11. APARs and PTFs needed for IMS 14 coexistence with IMS Version 12 and IMS Version 13

IMS 14 function	IMS Version 12 coexistence APAR/PTF	IMS Version 13 coexistence APAR/PTF	Additional information
DBRC RECON data sets	PI10131/UI22257	PI10132/UI22258	For more information, see "DBRC coexistence considerations" on page 80.
IMS repository enhancements - IMSRSC repository change list	PM80588/UK92589	None	For more information, see "IMS repository function coexistence considerations" on page 85.
DEDB Alter ADDAREA function	None	PI29543/UI24793	For more information, see "DEDB Alter enhancement coexistence considerations" on page 81.
IMS repository enhancements - Automatic export	PI27283/UI23503	PI27285/UI23504	For more information, see "IMS repository function coexistence considerations" on page 85.
IMS Connect support for OTMA tpipe parallelism	PM93880/UK98633	PM93878/UK98632	For more information, see "OTMA tpipe parallelism enhancement" on page 175.
Security enhancements	<ul style="list-style-type: none">PM48204/UK74051PM73558/UK82617	None	IMS Version 13 and IMS 14 do not support the SECURITY macro. You can use initialization parameters to specify all of the SECURITY macro keyword values. Initialization parameters for the RCLASS and SECCNT keywords are delivered through the APARs/PTFs listed.

Table 11. APARs and PTFs needed for IMS 14 coexistence with IMS Version 12 and IMS Version 13 (continued)

IMS 14 function	IMS Version 12 coexistence APAR/PTF	IMS Version 13 coexistence APAR/PTF	Additional information
MSC enhancements - IMSRSC repository support for dynamically defined MSC resources	PI49329/UI32564	PI49334/UI32565	For more information, see "MSC coexistence considerations" on page 88

Determining which coexistence service needs to be installed

You can identify the outstanding service that needs to be installed on your IMS Version 12 and IMS Version 13 systems to enable them to coexist with IMS 14 by using SMP/E and the IMS 14 FIXCAT category.

The FIXCAT category for IMS 14 is `IBM.Coexistence.IMS.V14` with the keyword `IMSV14COEX/K`.

The following is an example of the steps to determine what IMS service needs to be installed on IMS Version 12 or IMS Version 13 to coexist with IMS 14:

1. Download the current enhanced HOLDDATA
2. SMP/E RECEIVE the current enhanced HOLDDATA
3. Run the SMP/E REPORT MISSINGFIX command pointing to your IMS Version 12 or IMS Version 13 zone (for example):

```
SET BOUNDARY (GLOBAL) .
REPORT MISSINGFIX ZONES (targetzone)
FIXCAT(IBM.Coexistence.IMS.V14)
```

Related information:

Enhanced HOLDDATA for z/OS

IBM Fix Category Values and Descriptions

General coexistence considerations

Although IMS 14 can coexist with earlier versions of IMS, general coexistence considerations apply.

IMS 14 can coexist with earlier versions, so existing applications and data can be used without change. Migration and coexistence support is provided for IMS Version 12 and IMS Version 13.

The following general coexistence considerations apply:

- You must build new application control blocks (ACBs) for all existing program specification blocks (PSBs) and database definitions (DBDs).
- An all-system generation and a cold start are required for online systems (DBCTL, DB/DC, DCCTL). All data sets must be formatted when IMS is initialized the first time. To ensure that the data sets are formatted, specify the `FORMAT ALL` keywords on the cold start command (`/NRESTART CHECKPOINT 0` or `/ERESTART COLDSYS`).
- If you are installing multiple versions of IMS in the same processor, you can continue to use the prior versions of the IMS SVCs with the prior versions of IMS. However, the IMS 14 SVCs are downward compatible with IMS Version 12 and IMS Version 13. Only IMS 14 requires the IMS 14 SVCs.

- As of IMS Version 13, the old IMS static resource cleanup module, DFSMRCL0, is no longer shipped with IMS. DFSMRCL0 was used in IMS Version 8 and earlier. All currently supported versions of IMS use a dynamic resource cleanup module (DFSMRC20).

Recommendation: If you have not already done so, remove the DFSMRCL0 zap installed in IEAVTRML and the LPA direct pointer to the IMS SDFSRESL data set.

- For DB/DC and DCCTL online systems, the MFS format library is a required data set, regardless of whether MFS is used. DBCTL systems do not require an MFS format library.
- Utilities and logs
You might need to change programs that process the log because some log records that are created by database changes have been modified. For a list of the log records that are new, deleted, or changed for IMS 14, see Chapter 8, “Log record changes in IMS 14,” on page 105.
- **Extended checkpoint restriction:** You cannot use extended checkpoint to restart applications across different releases of IMS.

Specific coexistence considerations

Functional areas of or enhancements to IMS 14 have specific coexistence considerations.

The following topics describe specific coexistence considerations for IMS 14.

Common Queue Server coexistence considerations

Restrictions limit how Common Queue Servers (CQSs) from earlier versions of IMS can coexist.

General CQS coexistence rules

The following general coexistence rules apply to CQS in IMS 14:

- An IMS Version 12 or an IMS Version 13 can register with an IMS 14 CQS.
- An IMS 14 cannot register with a pre-IMS 14 CQS.
- An IMS Version 13 cannot register with a pre-IMS Version 13 CQS.
- An IMS Version 12 cannot register with a pre-IMS Version 12 CQS.
- User or vendor-written CQS clients that want to register with an IMS 14 CQS must be compiled with IMS Version 12 or later CQS macros.
- User or vendor-written CQS clients that are compiled with the IMS 14 CQS macros cannot register with a pre-IMS 14 CQS.
- User or vendor-written CQS clients that are compiled with the IMS Version 13 CQS macros cannot register with a pre-IMS Version 13 CQS.
- User or vendor-written CQS clients that are compiled with the IMS Version 12 CQS macros cannot register with a pre-IMS Version 12 CQS.
- An IMS Version 12 CQS, an IMS Version 13 CQS, and an IMS 14 CQS can connect to the same coupling facility structure.
- Any supported version of CQS can run on the same central processing complex (CPC).

For example:

- An IMS 14 CQS supports clients that are assembled with the IMS Version 12, IMS Version 13, or IMS 14 CQS macros.
- A client that is assembled with the IMS 14 CQS macros can register only with an IMS 14 CQS. The IMS 14 client cannot register with an IMS Version 12 or an IMS Version 13 CQS. Similarly, a client that is assembled with the IMS Version 13 CQS macros cannot register with an IMS Version 12 CQS.

DBRC coexistence considerations

An IMS 14 Database Recovery Control (DBRC) instance can coexist with an IMS Version 12 DBRC or an IMS Version 13 DBRC if you install the DBRC coexistence small programming enhancements (SPEs) to the IMS Version 12 or IMS Version 13 systems, and upgrade your RECON data set to the IMS 14 format by issuing a CHANGE.RECON UPGRADE command. Additional coexistence considerations apply to other DBRC functions.

SPEs are available for IMS Version 12 and IMS Version 13 that enable the coexistence of the earlier version DBRC with DBRC for IMS 14. With the appropriate SPE installed, IMS 14 and earlier version DBRCs can share the upgraded RECON data set. If the RECON data set has not been upgraded to IMS 14, the SPE has no effect. After the RECON data set has been upgraded, the SPE enables DBRC to convert records to the appropriate release format, depending on whether the record is being written to or the record is being read from the RECON data set. The SPE does not, however, enable the earlier level DBRC for the new functions delivered with DBRC in IMS 14.

The following coexistence SPEs must be installed to the IMS Version 12 or IMS Version 13 DBRC for coexistence purposes:

- IMS Version 12: PI10131/UI22257
- IMS Version 13: PI10132/UI22258

Restriction: After a RECON data set is upgraded to the IMS 14 level, it is not accessible to any pre-IMS 14 system that does not have the DBRC Coexistence SPE installed.

The MINVERS level must be set to the lowest level of IMS that uses or shares the RECON data sets.

Attention: Set a region size of 0M for all pre-IMS 14 Log Archive utility (DFSUARC0) jobs when an upgraded RECON data set is used. Unpredictable results can occur for any DBRC job that reads, writes, or changes any RECON record when inadequate storage is available for the job.

DBRC application coexistence considerations

There are two versions of each DBRC API (DSPAPI) macro: Version 1.0 (delivered with IMS Version 9) and Version 2.0 (delivered with IMS Version 10 and later). DBRC applications compiled with the Version 1.0 DSPAPI macros work without modification or reassembly with Version 2.0 of the DBRC API. However, these applications cannot use any of the newer functions (such as AUTH) or newer options (such as READONLY=YES) that are supported in the Version 2.0 macros.

For IMS Version 9, the default version level of the DSPAPI macros is 1.0. For IMS Version 10 and later, the default version level of the DSPAPI macros is 2.0. If a DBRC application that was originally assembled with the version 1.0 macros is

reassembled using an IMS Version 10 or later macro library, and does not specify VERSION=1.0, the application might not work as expected because of the default version change.

Recommendation: If the functionality of an IMS Version 9 DBRC application remains unchanged and is reassembled using an IMS Version 10 or later macro library, specify VERSION=1.0 on the DSPAPI macros.

The enhancements in Version 2.0 of the DBRC API are available only with IMS Version 10 and later. These enhancements can be used only by DBRC applications that are compiled with Version 2.0 or later of the DBRC API.

The output blocks for the IMS 14 DBRC API requests contain more information than the output blocks for earlier-version API requests. DBRC applications must use the latest maps to access the additional information.

In a coexistence environment, the RECON data set can be managed (read, updated, and so on) using the functions that each sharing DBRC supports. For example, a shared RECON data set could be:

- Updated by a DBRC instance using the batch DBRC commands that are processed by the Database Recovery Control (DBRC) utility (DSPURX00), and queried by using the Query requests through Version 1.0 of the DBRC API
- Updated or queried by an IMS Version 10 or later DBRC using either the batch DBRC commands or API requests through Version 2.0 of the DBRC API

IMS utilities coexistence considerations

The Batch Backout, Log Recovery, and Log Archive utilities must be run on the IMS release that created the logs. For example, an IMS Version 13 Batch Backout utility cannot back out logs that were created on IMS 14.

The Database Recovery and Change Accumulation utilities must be run on the IMS release that is the same as the highest level of the logs that are being used. For example, an IMS Version 13 Database Recover utility can use logs that were created on IMS Version 12 or IMS Version 13, but not on IMS 14.

DEDB Alter enhancement coexistence considerations

The DEDB Alter utility has requirements for coexistence in IMS 14.

To run the DEDB Alter utility, all IMS data-sharing systems must be at the IMS Version 13 or later level. The MINVERS value in the RECON data set must be 13.1 or later. The default value for MINVERS in IMS 14 is 12.1.

If you are using function of the DEDB Alter utility that was added in IMS 14 or later, such as ADDAREA, all IMS data-sharing systems must be at the IMS 14 or later level. The MINVERS value in the RECON data set must be 13.1 or later. If the MINVERS value in the RECON data set is not set to 13.1 or later, the DEDB Alter utility terminates without performing the requested function.

You must install IMS Version 13 APAR/PTF PI29543/UI24793 so that the DEDB Alter utility terminates with a non-zero return code when the ADDAREA function is attempted on a DEDB area that is shared by an IMS Version 13 system.

DRA coexistence considerations

The version of the IMS DRA modules that are used by a DRA client must be the same version as the IMS with which the DRA client is communicating. A DRA client that communicates with multiple versions of IMS must use the appropriate version of the IMS DRA modules for each version of IMS.

Recommendations:

- Concatenate the IMS.SDFSRESL library to the DRA client step library so the correct version of the DRA Startup/Router routine (DFSPRRC0) is loaded into the DRA client load library.
- Ensure that the DRA Startup Table (DFSPZPxx) points to the correct version of IMS.SDFSRESL.

Database utility coexistence considerations

Coexistence restrictions apply to the use of some database utilities.

JCL from earlier versions of IMS might need to be changed because IMS 14 DBRC uses time-stamp precision to the microsecond.

Recommendation: Use DBRC with all IMS 14 database change accumulation and database recovery jobs, especially during migration and coexistence.

Attention: Other utility programs (such as Log Merge and Log Analysis) work properly only when they process data that was created by an IMS subsystem or batch application program that is at the same release level as the utility program. Otherwise, the results are unpredictable and the output is unreliable.

Restrictions:

- IMS 14 utilities must be used whenever the input data for a DBDS contains log, image copy, or change accumulation records created by the IMS 14 system.
- Log records generated by a particular version of IMS must use that same version (or later) of the Database Recovery (DFSURDB0) and Database Change Accumulation (DFSUCUM0) utilities to process those logs. For example, the IMS Version 13 utilities can process either IMS Version 12 or IMS Version 13 log records, but cannot process IMS 14 log records.

Dynamic definition for MODBLKS resources coexistence considerations

If you use dynamic resource definition (DRD) for MODBLKS resources there are a few points to consider if different versions of IMS coexist.

When an instance of IMS 14 is using an IMSRSC repository and is coexisting with an earlier-version IMS that is using an RDDS, changes to online resources or descriptors made on the IMS 14 system are not exported to the earlier-version RDDS unless an EXPORT DEFN TARGET(RDDS) command is issued.

Recommendation: When changes are made to DRD runtime resources or descriptors in a multi-version IMSplex, store the definitions for those resources and descriptors in the IMSRSC repositories and RDDS as appropriate.

The Destination Creation exit routine (DFSINSX0) supports an IMSplex in which some systems are enabled for DRD and some are not. If the DFSINSX0 exit routine is set up to create runtime resource definitions (and the same exit is used across the IMSplex), the routing behavior differs, depending on whether the master is

enabled for DRD or not. If the master has DRD enabled, the created definition is routed to one or all IMS systems in the IMSplex. If the master does not have DRD enabled, you cannot route the created definition to other IMS systems in the IMSplex. The local exit creates the definition for a local DRD-enabled system.

Related concepts:

“IMS repository function coexistence considerations” on page 85

Exit routine coexistence considerations

Exit routines that run in IMS Version 12 or IMS Version 13 will work without modification in IMS 14. They will not, however, be able to use the latest functions without being modified.

Exit routines that run in multiple versions of IMS must be sensitive to the version of the Standard User Exit Parm List (SXPL). The version number of the SXPL can change if fields are added to the SXPL in a release of IMS.

For example, an exit routine that runs in a mixed-version IMSplex that needs to access the SXPLASWA field in the SXPL, which was introduced in IMS Version 11, must look for SXPLVER6 or later.

SXPLVER6 is the most recent version of the SXPL.

If security exit routines are to be shared across multiple versions, see “Security enhancements coexistence considerations” on page 92.

Related concepts:

“User exit enhancements” on page 206

“Exit routine migration considerations” on page 55

Fast Database Recovery (FDBR) coexistence considerations

An FDBR region must be at the same release level as the IMS system that it is tracking.

HALDB alter enhancement coexistence considerations

The HALDB alter enhancement introduces some specific considerations for coexistence with IMS Version 13.

HALDB alter requires the MINVERS value in the RECON data set to be set to 13.1. The default value for MINVERS in IMS Version 13 is 11.1.

HALDB support for 8-GB OSAM data sets enhancement coexistence considerations

The HALDB support for 8-GB OSAM data sets enhancement introduces some specific considerations for coexistence with IMS 14.

This enhancement supports coexistence with IMS Version 13 systems in data sharing environments only if SPE APAR PI23918 has been applied to all the IMS Version 13 resident libraries (RESLIBS) that access the RECON data set. The maintenance enables HALDB support for 8-GB OSAM data sets for IMS Version 13. The MINVERS value in the RECON data set must be 14.1 with the CDSLID value set to 2.

IMS 64-bit storage manager coexistence considerations

If the QUERY POOL command is routed to an IMS 14 system and an IMS 13 or IMS 12 system at the same time, the QUERY POOL command will have mixed results.

IMS catalog coexistence considerations

If you use the IMS catalog, certain considerations apply when IMS systems of different release levels coexist.

The IMS catalog is release independent. IMS systems at different release levels can use the same IMS catalog. However, because database definitions (DBDs), program views (PSBs), and application control blocks (ACBs) are release specific, the IMS catalog must contain a complete set of release-specific metadata for each release of IMS that uses the IMS catalog.

Consequently, when IMS systems at different release levels share the IMS catalog, the amount of storage that the IMS catalog requires increases by the amount of storage that is required to store the metadata from the ACB libraries of each release that uses the IMS catalog.

You can determine the amount of additional storage that the IMS catalog requires for a new release by providing the ACB library of the new release as input to the IMS Catalog Populate utility (DFS3PU00) and running the utility in analysis mode.

The DBD and PSB metadata for a given release of IMS must be added to the IMS catalog either by DDL or by using the generation and population utilities that are provided with that release.

Earlier versions of IMS cannot process any segments or fields that are added to the IMS catalog DBD by a new release of IMS.

Coexistence between IMS systems that manage ACBs differently

Before you enable the IMS management of ACBs for the first time in a data sharing environment, migrate all IMS Version 13 and earlier systems in the data sharing group to the new IMS release.


After all IMS systems that share an IMS catalog are migrated to the new release, IMS systems that use DBD, PSB, and ACB libraries can coexist in the data sharing group with IMS systems that use IMS-managed ACBs. However, the ACB, DBD, and PSB libraries must be kept in sync with the IMS catalog. Changes made to databases and program views in an IMS catalog that is enabled for ACB management are not automatically propagated to IMS systems that still use ACB, DBD, and PSB libraries.

To ensure that the ACB, DBD, and PSB libraries are consistent with the resources in an IMS catalog that is enabled for IMS ACB management, use one of the following methods:

- Use the IMS Catalog Library Builder utility (DFS3LU00) to create the ACB, DBD, and PSB libraries for the IMS systems that still require them from the active databases and program views in the IMS catalog. This method establishes the IMS catalog as the definitive source of active database and program views for all IMS systems. This is the recommended method.

- Use the generation utilities to create the ACB, DBD, and PSB libraries for all IMS systems. Distribute them among the IMS systems that still require them and use them to populate the IMS catalog.

Related reference:

 [IMS Catalog Library Builder utility \(DFS3LU00\) \(System Utilities\)](#)

IMS repository function coexistence considerations

The IMS repository function in IMS 14 can coexist with lower-level versions of IMS.

When you specify the EXPORTNEEDED control statement on the Create RDDS from Log Records utility (DFSURCL0), it is recommended that you run the same version of the DFSURCL0 utility as the version of the IMS that produced the IMS logs being used as input. For example, use the IMS 14 DFSURCL0 utility with IMS 14 log data sets. Otherwise, the results can be unpredictable.

Automatic export to the IMSRSC repository is supported only by an IMS 14 Resource Manager (RM) or later. Any lower-level RM systems that receive a request to update the repository for an automatic export will fail. IMS Version 12 APAR PI27283 and IMS Version 13 APAR PI27285 are open for the coexistence to not support RM to handle the AUTOEXPORT parameter. It is recommended that you enable automatic export to the IMSRSC repository only after all RM systems have been migrated to IMS 14.

An IMS change list is created only if the command master IMS is IMS Version 13 or later and the RM that processes change list requests is at V13 (1.6) level or higher.

Two possible scenarios for keeping stored resource definitions synchronized across multi-version IMSplex systems are shown as follows:

Scenario 1: Making attribute changes in a mixed environment of IMS systems that use RDDS and IMSRSC repository

In this scenario:

- IMSA is running IMS Version 12, using DRD with an RDDS.
- IMSB is running IMS Version 13, using DRD with an RDDS.
- IMSC is running IMS 14 and using DRD with an IMSRSC repository.
- All three IMS systems are in the same IMSplex and using shared queues.

The following steps illustrate changing an attribute of a transaction and storing its resource definition for scenario 1.

1. Check for work in progress by issuing a QRY TRAN SHOW(WORK) command. Either wait for the work to finish or address the work in progress.
2. When the transaction is not in use, an attribute of a transaction is changed on each IMS system by issuing an UPDATE or CREATE command.
3. IMSA and IMSB store the changed resource definition into their respective RDDSs by issuing the EXPORT DEFN TARGET(RDDS) command. IMSC stores the changed resource definition into its IMSRSC repository by issuing the EXPORT DEFN TARGET(REPO) command.

If AUTOEXPORT=AUTO is enabled, the changes are written to the system RDDS at IMSA and IMSB at the next checkpoint and to the IMSRSC repository for IMSC at the next checkpoint.

4. Work for the transaction is restarted.

Important: During migration to IMS 14, IMSRSC repository is enabled if the DFSDFxxx member has AUTOEXPORT=AUTO explicitly defined and autoexport export to the IMSRSC repository is enabled after IMS 14 is cold started. Any resource definition changes (creates and updates) will be automatically exported to the IMSRSC repository at the next checkpoint.

If you do not want to enable the autoexport to the IMSRSC repository, you must remove the AUTOEXPORT= parameter in the DYNAMIC RESOURCES section of the DFSDFxxx member and let it default to AUTO or you must modify the member to either of these values:

AUTOEXPORT= NO for no autoexport

AUTOEXPORT = RDDS for autoexport to the RDDS

To enable autoexport to the IMSRSC repository, you must modify the AUTOEXPORT= parameter in the DYNAMIC RESOURCES section of the DFSDFxxx member to specify AUTOEXPORT=AUTO or AUTOEXPORT=REPO. In Scenario 1, if IMSC has AUTOEXPORT=AUTO explicitly specified, autoexport to the IMSRSC repository is enabled. To disable autoexport, modify the DFSDFxxx member for IMSC to specify AUTOEXPORT= NO.

Scenario 2: Changing a transaction definition on one IMS and propagating the change to the other IMS systems

In this scenario:

- IMSA is running IMS Version 12, using DRD with an RDDS, and using an IMS 14 CSL.
- IMSB is running IMS Version 13, using DRD with an RDDS, and using an IMS 14 CSL.
- IMSC and IMSD are running IMS 14 and are using DRD with a single IMSRSC repository.
- All four IMS systems are participating in shared queues.

The following steps illustrate changing an attribute of a transaction and storing its resource definition for scenario 2.

1. Check for work in progress by issuing a QRY TRAN SHOW(WORK) command. Either wait for the work to finish or address the work in progress.
2. When the transaction is not in use, change an attribute of the transaction on IMSC by issuing an UPDATE command.
3. Store the changed transaction definition in the IMSRSC repository for IMSC and IMSD by issuing either of the following EXPORT command from IMSC.

```
EXPORT DEFN TARGET(REPO) SET(IMSID(IMSC,IMSD))
EXPORT DEFN TARGET(REPO) SET(IMSID(*))
```
4. Update the runtime definition of the transaction on IMSD by importing the stored resource definition from the IMSRSC repository by issuing the IMPORT DEFN SOURCE(REPO) command
5. Export the changed transaction definition from IMSC to a non-system RDDS by issuing the EXPORT DEFN TARGET(RDD) command.
6. Update the runtime definition of the transaction on IMSA and IMSB by importing the stored resource definition from the non-system RDDS by issuing the IMPORT DEFN SOURCE(RDD) command.

7. Export the changed transaction definition from IMSA and IMSB to their respective system RDDSs by issuing the EXPORT DEFN TAEGET(RDDS) command.
8. Work for the transaction is restarted.

Coexistence with IMS Version 13

The following IMS Version 13 APAR/PTF must be installed in an IMS Version 13 system to enable coexistence between an IMS 14 Resource Manager (RM) instance and an IMS Version 13 RM instance:

PI27285/UI23504

Prevents an IMS Version 13 RM address space from processing requests for automatic export to the IMSRSC repository from an IMS 14 system.

Coexistence with IMS Version 12

The following IMS Version 12 APAR/PTF must be installed in an IMS Version 12 system to enable coexistence between an IMS 14 Resource Manager (RM) instance and an IMS Version 12 RM instance:

PM80588/UK92589

Prevents an IMS Version 12 RM address space from processing QUERY, UPDATE, and DELETE commands that are issued against an IMS Version 13 or later IMSRSC repository change list.

PI27283/UI23503

Prevents an IMS Version 12 RM address space from processing requests for automatic export to the IMSRSC repository from an IMS 14 system.

IMS Connect coexistence considerations

The IMS 14 IMS Connect function supports IMS Version 13 and IMS Version 12, although new functions might not be available when connecting with the older versions.

IMS Connect instances from IMS Version 13 or IMS Version 12 might also be able to connect to an IMS 14 system, but coexistence APARs might be required.

- Support for cascading global transactions from the IMS TM resource adapter to an IMS TM system on a different LPAR is supported only after both IMS Connect and the IMS control region are migrated to IMS 14.

If IMS Connect receives a global transaction from IMS TM resource adapter when IMS Connect and IMS are not on the same LPAR and either IMS Connect, IMS, or both, are not at IMS 14, the transaction is rejected with a NAK message that includes a sense code of X'2F'.

- IMS Version 12 and IMS Version 13 instances of IMS Connect can connect to OTMA tpipes that support parallel processing of RESUME TPIPE requests after the appropriate coexistence APARs are applied. The APARs are required only for IMS Version 12 and IMS Version 13 instances of IMS Connect that connect to tpipes in which tpipe parallelism is enabled.

Previous versions of IMS Connect require these compatibility fixes:

- IMS Connect Version 13 APAR/PTF PM93878/UK98632
- IMS Connect Version 12 APAR/PTF PM93880/UK98633
- The IMS Connect function in IMS Version 13 and later reports its health to z/OS Workload Manager (WLM), but the IMS Connect functions in IMS Version 12 and earlier do not report their health. This difference can affect how the z/OS

Sysplex Distributor routes messages when different versions of IMS Connect coexist, because only the workload of the IMS Connect function in IMS Version 13 and later is accurately reported.

- IMS 14 introduces new and changed IMS type-2 commands for IMS Connect. If a type-2 command is broadcast to an IMSplex in which multiple versions of IMS Connect coexist, instances of IMS Connect from earlier versions reject commands that include keywords that they do not support and ignore commands that they do not recognize.

MSC coexistence considerations

You can connect an IMS 14 Transaction Manager subsystem to other supported versions of IMS by using Multiple Systems Coupling (MSC).

The following IMS versions can coexist with an IMS 14 system in an MSC network:

- IMS Version 13 (5635-A04)
- IMS Version 12 (5635-A03)

For an IMS that was generated with MSC resources to coexist with IMS 14, any MSC user exits that access MSC LLB control blocks (logical links) or MSC LCB control blocks (physical links) must be changed to access those blocks using callable services or DSCBTS services. This change is required regardless of whether dynamic MSC is enabled.

You might need to update an MSC resource definition in an IMSplex in which an IMS 14 system that uses an IMSRSC repository coexists with an IMS Version 13 or earlier IMS system. In this scenario, the MSC resource updates that are made on the IMS 14 system by using the UPDATE commands must be made on the IMS Version 13 or earlier IMS systems by coding stage-1 system definition macros.

If the IMSRSC repository support for dynamically defined MSC resources is enabled for IMS 14, apply the following coexistence APARs to the RM systems in the IMSplex:

- IMS 14: APAR/PTF PI49208/UI32429
- IMS Version 13: APAR/PTF PI49334/UI32565
- IMS Version 12: APAR/PTF PI49329/UI32564

Use one of the following methods to install the coexistence APARs that are listed in the preceding list:

- First, install IMS Version 12 APAR/PTF PI49329/UI32564, IMS Version 13 APAR/PTF PI49334/UI32565, and the IMS 14 preconditioning APAR/PTF PI49208/UI32429 on all RM systems in the IMSplex. Then, install the IMS 14 preconditioning APAR/PTF PI45186.
- Install all of the following APARs on all RM systems in the IMSplex at the same time:
 - IMS Version 12 APAR/PTF PI49329/UI32564
 - IMS Version 13 APAR/PTF PI49334/UI32565
 - IMS 14 preconditioning APAR/PTF PI49208/UI32429
 - IMS 14 preconditioning APAR/PTF PI45186

Open Database coexistence considerations

Coexistence restrictions apply to the use of Open Database.

The IMS Universal drivers that come with IMS 14 can be used by Java applications that access IMS Version 13 or IMS Version 12 databases as long as all the IMS systems are part of a mixed-version IMSplex that includes an IMS 14 system (along with its ODBM and IMS Connect address spaces).

Open Database Manager (ODBM) can only connect to the IMS systems that are of the same version as ODBM itself. In a mixed-version IMSplex, to limit ODBM connection to the IMS systems of the same version, list the eligible IMS systems as data stores in the CSLDCxxx member of the IMS PROCLIB data set.

OTMA coexistence considerations

When an IMS 14 OTMA coexists with earlier versions, certain considerations apply.

IMS Version 12 and IMS Version 13 instances of IMS Connect can connect to OTMA tpipes that support parallel processing of RESUME TPIPE requests after the appropriate coexistence APARs are applied. The APARs are required only for IMS Version 12 and IMS Version 13 instances of IMS Connect that connect to tpipes in which tpipe parallelism is enabled.

For connections from IMS Version 12 and IMS Version 13 instances of IMS Connect, support for the parallel processing of RESUME TPIPE requests can be enabled only from the OTMA client descriptors.

Previous versions of IMS Connect require these compatibility fixes:

- IMS Connect Version 13 APAR/PTF PM93878/UK98632
- IMS Connect Version 12 APAR/PTF PM93880/UK98633

PST coexistence considerations

The number of program specification tables (PSTs) that can be allocated during system initialization in IMS Version 13 can be larger than in previous versions. The larger number of PSTs in IMS Version 13 is not recognized by previous versions of IMS.

In a mixed-version IMSplex, commands that reference the MAXRGN parameter and specify a value greater than 999 pass validation at the Operations Manager (OM) layer because OM uses the highest IMS version grammar. However, if any these commands are routed to an IMS system that is a lower version than IMS Version 13, the command fails on that IMS system. The affected commands are:

- CHANGE TRAN
- CREATE TRAN
- CREATE TRANDESC
- UPDATE TRAN
- UPDATE TRANDESC

Remote Site Recovery (RSR) coexistence considerations

RSR coexistence supports configurations that meet certain version level requirements.

IMS 14 RSR coexistence supports configurations in which:

- The tracking site Transport Manager Subsystem (TMS) is at the IMS 14 level and the active site is at IMS Version 12 or IMS Version 13 level.

- The active site RECON data sets are at the IMS Version 12 or IMS Version 13 level (with the IMS 14 DBRC coexistence SPE applied).
- The active TMS that is running ILS is at the IMS Version 12 or IMS Version 13 level.
- One or more active IMS systems are at the IMS Version 12 or IMS Version 13 level.

The following table describes the supported configurations of a simple set of subsystems in an RSR Global Service Group (GSG). Some unsupported configurations are not included because a prerequisite configuration is also not supported. Some unsupported configurations are included even though a prerequisite configuration is supported so that the reason for non-support can be explained.

The simple set of configurations in the table assumes:

- A single active IMS
- A single TMS that is running the Isolated Log Transport
- A single set of active site RECON data sets
- A single tracking IMS
- A single remote site TMS
- A single set of remote site RECON data sets

The table also describes valid and invalid RSR coexistence configurations. The following notation is used to denote the state of the subsystems and RECON data sets in this table:

- pre-V14: IMS Version 12 or IMS Version 13
- pre-V14 + SPE: IMS Version 12 or IMS Version 13, with the appropriate IMS 14 DBRC coexistence SPE applied
- V14: IMS 14

Table 12. Valid RSR coexistence subsystem configurations and migration progression

Active site state			Remote site state			Notes
IMS	TMS/ILS	RECON	IMS	TMS	RECON	
pre-V14	pre-V14	pre-V14	pre-V14	pre-V14	pre-V14	Supported.
pre-V14 + SPE	pre-V14	pre-V14	pre-V14	pre-V14	pre-V14	Supported.
pre-V14 + SPE	V14	V14	pre-V14	pre-V14	pre-V14	Supported.
pre-V14	V14	pre-V14	pre-V14	pre-V14	pre-V14	Not supported. The local TMS/ILS cannot be migrated to IMS 14 if the RECON data sets have not been upgraded to IMS 14.
pre-V14 + SPE	pre-V14 + SPE	pre-V14	pre-V14	pre-V14	pre-V14	Supported.
V14	pre-V14 + SPE	pre-V14	pre-V14	pre-V14	pre-V14	Not supported. The RECON data sets must be upgraded to IMS 14 before running any IMS at the IMS 14 level. The active IMS cannot be at the IMS 14 level if the local site TMS that is running ILS is not at the same version level. The local site IMS cannot be at a later version level than the remote IMS.

Table 12. Valid RSR coexistence subsystem configurations and migration progression (continued)

Active site state			Remote site state			Notes
IMS	TMS/ILS	RECON	IMS	TMS	RECON	
pre-V14 + SPE	pre-V114 + SPE	V14	pre-V14	pre-V14	pre-V14	Supported.
pre-V14	V14	V14	pre-V14	pre-V14	pre-V14	Not supported. The local site TMS that is running ILS cannot be at the IMS 14 level if the tracking IMS is at the pre-14 level. Also, if the active RECON is at IMS 14, the IMS must have the SPE applied.
V14	V14	V14	pre-V14	pre-V14	pre-V14	Not supported. The local site TMS that is running ILS and the active IMS cannot be at the IMS 14 level if the tracking IMS is at the pre-14 level. The active IMS also cannot be at the IMS 14 level.
pre-V14	pre-V14	pre-V14	pre-V14 + SPE	pre-V14	pre-V14	Supported.
pre-V14 + SPE	pre-V14	pre-V14	pre-V14 + SPE	pre-V14	pre-V14	Supported.
pre-V14 + SPE	pre-V14 + SPE	pre-V14	pre-V14 + SPE	pre-V14	pre-V14	Supported.
V14	pre-V14 + SPE	V14	pre-V14 + SPE	pre-V14	pre-V14	Not supported. An active IMS cannot be at the IMS 14 level if the tracking IMS is not at the IMS 14 level. The active site TMS that is running ILS also cannot be at the pre-14 level or at the pre-14 + SPE level. 1 on page 92
pre-V14	pre-V14 + SPE	pre-V14	pre-V14 + SPE	pre-V14	pre-V14	Supported 1 on page 92.
pre-V14 + SPE	pre-V14 + SPE	pre-V14	pre-V14 + SPE	pre-V14	pre-V14	Supported 1 on page 92.
pre-V14 + SPE	pre-V14 + SPE	V14	pre-V14 + SPE	pre-V14	pre-V14	Supported 1 on page 92.
pre-V14 + SPE	pre-V14 + SPE	V14	V14	pre-V14	pre-V14	Not supported. The tracking IMS cannot be at the IMS 14 level if the remote site RECON data sets are not upgraded to IMS 14. 1 on page 92 2 on page 92
pre-V14 + SPE	pre-V14 + SPE	V14	pre-V14 + SPE	pre-V14 + SPE	V14	Supported.
V14	pre-V14 + SPE	V14	pre-V14 + SPE	pre-V14 + SPE	V14	Not supported. The active IMS cannot be at the IMS 14 level if neither the tracking IMS nor the active site TMS that is running ILS is not at the IMS 14 level.
pre-V14 + SPE	pre-V14 + SPE	V14	V14	V14	V14	Supported.
V14	pre-V14 + SPE	V14	V14	V14	V14	Not supported. The active IMS cannot be at the IMS 14 level if the active site TMS that is running ILS is not at the IMS 14 level.

Table 12. Valid RSR coexistence subsystem configurations and migration progression (continued)

Active site state			Remote site state			Notes
IMS	TMS/ILS	RECON	IMS	TMS	RECON	
pre-V14 + SPE	V14	V14	V14	V14	V14	Supported.
V14	V14	V14	V14	V14	V14	Supported.

Note:

1. The remote site TMS can also be at the IMS Version 12 or IMS Version 13 level.
2. Regardless of the configuration of the active site subsystems and the RECON data sets, this state is not supported.

Security enhancements coexistence considerations

If an IMS 14 subsystem shares the DFSPBxxx or DFSDCxxx PROCLIB members or security exit routines with an IMS Version 12 or earlier subsystem, coexistence considerations apply.

SECURITY macro coexistence considerations

Use the following procedure to enable multiple versions of IMS to share DFSPBxxx or DFSDCxxx PROCLIB members:

1. Install the following security small programming enhancement (SPE) APARs/PTFs:
 - IMS Version 11 - PM48203/UK74050
 - IMS Version 12 - PM48204/UK74051
2. Ensure that all of the security options are specified as initialization parameters and not parameters of the SECURITY macro.

For information about using initialization parameters for security, see "Controlling security during system startup for DB/DC and DCCTL" in the *System Administration* information.
3. Leave the SECURITY macro in the IMS 14 stage 1 system definition input because the earlier versions of IMS require the SECURITY macro. Ignore message G115.

Security exit routine coexistence considerations

In IMS Version 11 and IMS Version 12, the following exit routines must be linked into the IMS nucleus.

- Security Reverification exit routine (DFSCCTSE0)
- Signon/off security exit routine (DFSCSGN0)
- Transaction Authorization exit routine (DFSCSTRN0)

This linking arrangement enables the exit routines to use virtual address spaces to exchange data between them.

Another requirement in earlier versions of IMS was that the CSECT of DFSCSTRN0 had to be within DFSCCTSE0 if DFSCCTSE0 was used.

In IMS 14, these exit routines will run without modification. However, they must be linked differently.

- DFSCCTSE0 must be linked as an ALIAS of DFSCSTRN0.

Recommendation:

- Link DFSCCTSE0 and DFSCCTRN0 as reentrant and AMODE/RMODE 31. If the load modules are not linked as reentrant, IMS loads them twice.
- If v-type address constants are used by DFSCCTSE0, DFSCCTRN0, and DFSCCTSGN0 to exchange data between the exit routines, DFSCCTSE0 must be linked as an ALIAS of DFSCCTSGN0.

Recommendation:

- Link DFSCCTSE0 and DFSCCTRN0 as reentrant and AMODE/RMODE 31. If the load modules are not linked as reentrant, IMS loads them three times.

Shared message queue coexistence considerations

Instances of IMS Version 12, IMS Version 13, and IMS 14 can share an IMS message queue in an IMSplex.

Synchronous program switch coexistence considerations

Coexistence considerations for synchronous program switch include the configuration of the MINVERS value in a shared queues environment

In IMS 14, the default value of MINVERS in the RECON data set is 12.1. However, to use the synchronous program switch function in a shared queues environment, all of the participating systems must have a MINVERS value of 13.1 or later.

Syntax Checker coexistence considerations

The IMS 14 Syntax Checker supports IMS Version 12, IMS Version 13, and IMS 14.

Be sure that the version shown is correct when you use the Syntax Checker to check the parameters of earlier versions.

Restrictions for IMS 14 during coexistence

There are no general restrictions for IMS 14 in a coexistence environment; however, certain restrictions apply to new functions.

| The HALDB support for 8-GB OSAM data set enhancement supports coexistence
| with IMS Version 13 systems in data sharing environments only if SPE APAR
| PI23918 has been applied to all the IMS Version 13 resident libraries (RESLIBS) that
| access the RECON data set. The minimum version (MINVERS) in the RECON data
| set must be 14.1 or it must be 13.1 with the CDSLID value set to 2.

The HALDB alter and DEDB alter functions do not support coexistence with IMS Version 12 or earlier systems in data sharing environments. Both functions require a minimum version (MINVERS) of 13.1 in the RECON data set.

The synchronous program switch function does not support coexistence with IMS Version 12 or earlier systems in shared queues environments. The synchronous program switch function requires a minimum version (MINVERS) of 13.1 in the RECON data set.

Chapter 7. Message and code changes in IMS 14

IMS 14 includes new and changed messages and codes. Also, many messages were deleted from IMS 14.

Messages and codes that were added or changed in earlier versions of IMS are not listed.

For messages and codes that were added or changed through the service process, the APARs or PTFs contain appropriate ++HOLD cards to indicate the updates.

Recommendation: Always check the ++HOLD cards in the service stream for information about new, deleted, or changed messages and abend codes.

New messages and codes for IMS 14

IMS 14 includes new CSL, CQS, DFS, DSP, and HWS messages, and new abend, OTMA, DL/I, and SQL codes.

New CQS and CSL messages

The following messages are new:

CQS0223I
CQS0247W
CQS0248I
CSL2505W

New DFS messages

The following messages are new:

DFS0745E
DFS094I
DFS480E
DFS529E
DFS0533E
DFS0534I
DFS3487
DFS3497
DFS3505E
DFS3507E
DFS3513E
DFS3515E
DFS3525E
DFS3529
DFS3537E
DFS3539E
DFS3541E
DFS3543I

DFS3545
DFS3551E
DFS3575I
DFS3631E
DFS3642E
DFS3664I
DFS3679E
DFS3706I
DFS3722I
DFS3723E
DFS3734I
DFS3735I
DFS3755E
DFS3780I
DFS3793E
DFS4145I
DFS4146W
DFS4154W
DFS4155W
DFS4156W
DFS4157W
DFS4158I
DFS4160E
DFS4186E
DFS4187E
DFS4188E
DFS4194E
DFS4195E
DFS4196E
DFS4197E
DFS4198E
DFS4324E
DFS4325W
DFS4326E
DFS4328E
DFS4330I
DFS4331E
DFS4332E
DFS4333E
DFS4334W
DFS4335E
DFS4336W
DFS4337E
DFS4338W
DFS4339E

DFS4340E
DFS4341E
DFS4344E
DFS4349E
DFS4369I
DFS4370I
DFS4371W
DFS4372I
DFS4373I
DFS4374W
DFS4375I
DFS4376I
DFS4377I
DFS4379W
DFS4387I
DFS4390I
DFS4391E
DFS4519E
DFS4520I
DFS4521I
DFS4522I
DFS4523E
DFS4524E
DFS4525E
DFS4526E
DFS4527E
DFS4529E
DFS4530E
DFS4531I
DFS4532E
DFS4533I
DFS4534I
DFS4535I
DFS4536I
DFS4537I
DFS4538I
DFS4539I
DFS4540E
DFS4541E
DFS4542I
DFS4545I
DFS4546I
DFS4547W
DFS4548E
DFS4551E

|
|
|
|

DFS4552W
DFS4554E
DFS4560E
DFS4561E
DFS4562E
DFS4587E
DFS4591E
| DFS4593E
| DFS4594W
| DFS4597I
| DFS4598I
| DFS4599I
DFS4615E
| DFS4631E
| DFS4632E
DFS4682I
DFS4683I
DFS4684I
DFS4685I
DFS4686E
DFS4672E
DFS4673E
DFS4674E
DFS4675E
DFS4676E
| DFS4679E
DFS4682I
DFS4683I
DFS4684I
DFS4685I
DFS4686E
| DFS4719E
| DFS4735I
| DFS4736I
| DFS4741I
| DFS4744E
| DFS4747E
| DFS4768E
| DFS4760E
| DFS4761E
| DFS4742E
| DFS4770I
| DFS4778I
| DFS4784E
| DFS4785W

DFS4830I

New DSP messages

The following messages are new:

DSP1206E
DSP1254I
DPS1255W
DSP1256E
DSP1257E
DSP1258I
DSP1259I
DSP1263W
DSP1264E
DSP1265E
DSP1266E
DSP1267E
DSP1268E
DSP1269E
DSP1270E
DSP1271A
DSP1272A
DSP1273E
DSP1274E
DSP1275E
DSP1276A

New HWS messages

The following messages are new:

HWSD0205W
HWSO1330W

New abend codes

The following abend codes are new:

0037
0053
0220
0330
0529
0565
0569
3001

New DL/I status codes

The following status codes are new:

BY
DF

New AIB return and reason codes

The following status codes are new:

0100/0224
0100/120C

0100/1210
0100/1218
0104/1200
0104/1204
0104/1208
0104/1214
0108/0700
0108/0704
0108/0708
0108/0710

New component codes

The following component codes are new:

OTMA 0037
SQL -102
SQL -103
SQL -107
SQL -110
SQL -113
SQL -170
SQL -440
SQL -540
SQL -612
SQL -624
SQL -637
SQL -644
SQL -802
SQL -8019
SQL -8020
SQL -8021
SQL -8022
SQL -8023
SQL -8024
SQL -8025
SQL -9000
SQL -9001
SQL -9002
SQL -9003
SQL -9005
SQL -9006
SQL -9020
SQL -9021
SQL -9022
SQL -9023
SQL -9024
SQL -9030
SQL -9050
SQL -9051
SQL -9052
SQL -9053
SQL -9054
SQLIMSSTATE values and common error codes

Changed messages and codes for IMS 14

IMS 14 includes changed DFS, DSP, and HWS messages, and changed abend, component, and AIB codes.

Changed DFS messages

The following messages are changed:

| DFS554A
| DFS686W
| DFS0769I
| DFS0842I
| DFS0843I
| DFS1913I
| DFS1919I
| DFS1929I
| DFS2073I
| DFS2153
| DFS2153
| DFS2180I
| DFS2342E
| DFS2384W
| DFS2385E
| DFS2440W
| DFS2469W
| DFS2504I
| DFS2505I
| DFS2508I
| DFS2519I
| DFS2504I
| DFS2679A
| DFS2930I
| DFS3177E
| DFS3262E
| DFS3373I
| DFS3374W
| DFS3410I
| DFS3422X
| DFS3423E
| DFS3437E
| DFS3494E
| DFS3497A
| DFS3678E
| DFS3704E
| DFS3722I
| DFS3724I
| DFS3735I
| DFS4401E
| DFS4573E
| DFS4610E
| DFS4620E
| DFS4622E

Changed DSP messages

The following messages are changed:

DSP0012I
DSP0141I
DSP0222I
DSP0223I
DSP0224I
DSP0383A
DSP0398E
DSP0843I
DSP1076I
DSP1100I
DSP1126I
DSP1235W
DSP1236E
DSP1242E
DSP1243W
DSP1244E
DSP1245W
DSP1246E
DSP1253W

Changed HWS messages

The following messages are changed:

HWSI1665E
HWSS0742W

Changed abend codes

The following abend codes are changed:

0029
0071
0104/0450
0168
0220
0403
0513
0565
0630
0657
0741
0711
1060
3044
3045
3048
3049
3053
3055
3422

Changed component codes

The following component codes are changed:

DBCTL DRA return codes
OTMA 002F
SQL -122
SQL -206
BPE 0045E
BPE 3400

Changed AIB return and reason codes

The following status codes are changed:

0000/0000 (no change, but now used for IR call also)
0004/0004
0100/000C
0900

Deleted messages and codes for IMS 14

Several messages were deleted in IMS 14.

Table 13. Messages deleted in IMS 14

DFS3512I	DFS4612E
DFS3514I	DSP1205E
DFS3516I	G008
DFS3518I	G1211
DFS3520I	
DFS3522I	
DFS3524I	
DFS3526I	
DFS3528I	
DFS3530I	
DFS3532I	
DFS3534I	
DFS3536I	
DFS3538I	
DFS3540I	
DFS3546I	
DFS3548I	
DFS3550I	
DFS3552I	
DFS3554I	
DFS3556I	
DFS3558I	
DFS3559A	
DFS3560I	
DFS3562I	
DFS3564I	
DFS3566I	
DFS3567I	
DFS3568I	
DFS3569I	
DFS3570I	
DFS3572I	
DFS3574I	
DFS3577A	
DFS3593I	
DFS3599I	

Chapter 8. Log record changes in IMS 14

The following table lists the log records that are new or changed in IMS 14.

Table 14. New or changed log records for IMS 14

X'22'	X'36'	X'4518'
X'2203'	X'38'	X'4519'
X'2224'	X'4031'	X'56000009'
X'2225'	X'4032'	X'56FA'
X'2226'	X'4033'	X'59'
X'2230'	X'4036'	X'5961'
X'31'	X'4037'	X'63'
X'35'	X'4038'	X'66'
	X'4039'	X'6701'
	X'4098'	
	X'4513'	

In IMS Version 13 and later, certain existing statistics fields in the type X'07' and the type X'56FA' records now include I/O counts and times relating to Fast Path DEDB I/O that is issued by an IMS dependent region. Prior to IMS Version 13, Fast Path DEDB I/O statistics were not included in these fields.

Table 15. Fast Path DEDB statistics recorded in X'07' and X'56FA' log records

X'07' record field	X'56FA' record field	Description
DLRVSAMR	TPVSAMR	VSAM READ I/O count
DLRVSAMW	TPVSAMW	VSAM WRITE I/O count
DLRTOTIO	TPTOTIO	Total DLI I/O count
DLRTMEIO	TPTDBIO	Elapsed DB I/O time

Note: Only Fast Path I/O that is issued while under the dependent region is included in these statistics. Statistics about Fast Path write I/Os that are issued asynchronously by the IMS control region are not included.

For details about the DSECT names that are associated with these log records and descriptions of why the log record was issued, see the topic Log records (Diagnosis) in *IMS Diagnosis* information.

Part 2. IMS Version 14 enhancements

The enhancements to introduced in IMS 14 span the following areas of IMS: system definition, administration, troubleshooting, and reference.

The following topics provide planning information for the enhancements to IMS 14.

Chapter 9. IMS enhancement PTFs

Until a new version of IMS is released, IBM continues to enhance the most recently released version of IMS. The enhancements, which can include new IMS function or support for new technologies or products, are released as PTFs.

The following topics provide an overview of the IMS continuous delivery model and list the newest IMS enhancements with the date and number of the PTF in which they are delivered.

The releases are presented in descending order, where the most recent release is listed first.

The IMS continuous enhancement process

IMS uses a continuous delivery model to enhance the most recent in-service version of IMS. In the continuous delivery model, as soon as new IMS function or support for new technology is developed and tested, it is immediately released in a PTF.

Continuous delivery enhancements are applied only to the most recent in-service version of IMS. As soon as a new version of IMS is released, continuous delivery enhancements are applied only to the new version of IMS, and the previous version is designated as a long term support release and is no longer enhanced.

The continuous release of enhancements delivers the IMS functionality and IMS support for new technology that your business needs when your business needs it, without having to wait for a new version of IMS.

The continuous release of enhancements also delivers even higher quality code, because enhancements are delivered when they are ready, without the pressure to match the development and test schedule of a particular enhancement to the schedule of a major release that was set before development on the enhancement even began.

Enhancements that are delivered in function-level releases do not require you to reinstall IMS to adopt new functions or enable new support. Most new functionality or support for new technology that would require you to reinstall IMS or that would be particularly disruptive to your IMS installation is held for delivery on a release boundary.

New enhancements are delivered as PTFs through the normal service process for the most recent in-service version of IMS.

Some new functionality remains disabled by default after you install the PTF, so that you can enable it when you are ready. Typically, you enable disabled enhancements by using a function activation method, such as a command or new parameter. Other new functionality, due to either technical or strategic requirements or because it has no impact until you choose to use it, is enabled by installing the PTF.

October 2017 enhancement PTFs

PTFs that contain enhancements to IMS 14 were released in October 2017.

IMS system enhancements overview

PTFs for the following IMS 14 system were released in October 2017:

“IMS Connect WTOR command interface option enhancement” on page 199
PTFs UI50870, APAR PI85790. Other prerequisite PTFs might also be required.

July 2017 enhancement PTFs

PTFs that contain enhancements to IMS 14 were released in July 2017.

IMS system enhancements overview

PTFs for the following IMS 14 system were released in July 2017:

Timeout option for idle connections, APAR PI83616
PTF UI48752. Other prerequisite PTFs might also be required.

June 2017 enhancement PTFs

PTFs that contain enhancements to IMS 14 were released in June 2017.

IMS Database Manager enhancements

PTFs for the following IMS 14 Database Manager enhancements were released in June 2017:

STAGE,DELETE option for the IMS Catalog Populate utility (DFS3PU00)
PTF UI46904, APAR PI76125. Other prerequisite PTFs might be required.

After PI76125 is installed, the IMS Catalog Populate utility no longer scratches and recreates the staging data set by default when MANAGEDACBS=STAGE is specified. To scratch and recreate the staging data set when STAGE is specified, you must now specify MANAGEDACBS=(STAGE,DELETE).

DEDB Sequential Dependent Delete utility (DBFUMDL0) SETS A SPECIFIED DMACXVAL

PTF UI48236, APAR PI73397. Other prerequisite PTFs might be required.

IMS Transaction Manager enhancements overview

PTFs for the following IMS 14 Transaction Manager enhancements were released in June 2017:

Support for calling the DFSMSCE0 exit when issuing the IMS ICAL call for a synchronous program switch

PTF UI47892, APAR PI80003. Other prerequisite PTFs might also be required.

The ability to call the DFSMSCE0 exit routine enables support for the workload router.

IMS system enhancements overview

PTFs for the following IMS 14 system were released in June 2017:

Enhance report of IMS locking

PTFs UI47337 and UI47338, APAR PI70549. Other prerequisite PTFs might also be required.

Log record X'56FA' is updated to contain lock request statistics for transaction level statistics logging.

“Recording IMS Connect RACF statistics for IMS DB connections” on page 196

PTF UI47653, APAR PI80202. Other prerequisite PTFs might also be required.

May 2017 enhancement PTFs

PTFs that contain enhancements to IMS 14 were released in May 2017.

IMS Database Manager enhancements

PTFs for the following IMS 14 Database Manager enhancements were released in May 2017:

IMS DDL extensions for CREATE DATABASE, data sharing support, APAR PI69585

PTF UI45595. Other prerequisite PTFs might be required.

IMS Transaction Manager enhancements overview

PTFs for the following IMS 14 Transaction Manager enhancements were released in May 2017:

APPC flood control enhancement, APAR PI68466

PTFs UI47190 and UI47191. Other prerequisite PTFs might also be required.

Java environment scripting enhancement, APAR PI68127

PTF UI46000. Other prerequisite PTFs might also be required.

March 2017 enhancement PTFs

PTFs that contain enhancements to IMS 14 were released in March 2017.

IMS Transaction Manager enhancements overview

PTFs for the following IMS 14 Transaction Manager enhancements were released in March 2017:

MFS protected field validation enhancement, APAR PI51565

PTFs UI45620 and UI45621. Other prerequisite PTFs might also be required.

APAR PI51565 is a forward fit of an enhancement that was also released in IMS Version 12 and IMS Version 13.

February 2017 enhancement PTFs

PTFs that contain enhancements to IMS 14 were released in February 2017.

IMS Transaction Manager enhancements overview

PTFs for the following IMS 14 Transaction Manager enhancements were released in February 2017:

“Shared queues buffer pool enhancement” on page 179, APAR PI71929

PTFs UI44745 and UI44747, and any prerequisite PTFs that might be required to enable this function.

January 2017 enhancement PTFs

PTFs that contain enhancements to IMS 14 were released in January 2017.

IMS Database Manager enhancements

PTFs for the following IMS 14 Database Manager enhancements were released in January 2017:

Allow CNBA=0 to be specified for CCTL connectors, APAR PI60717

PTF UI44116. Other prerequisite PTFs might be required.

APAR PI60717 is a forward fit of an IMS Version 13 enhancement. For more information, see *IMS Version 13 Release Planning*, GC19-3658.

IMS Transaction Manager enhancements overview

PTFs for the following IMS 14 Transaction Manager enhancements were released in January 2017:

“IMSRSC repository support for dynamically defined MSC resources” on page 166, APAR PI50129

PTF UI44232 and prerequisite PTFs are required to enable this function.

IMS 14 base-level enhancements

The enhancements that are included in the initial *base-level* release of IMS 14 affect all areas of the product: IMS Database Manager, IMS Transaction Manager, and IMS system.

IMS Database Manager enhancements overview

IMS 14 Database Manager includes the following enhancements:

- “64-bit buffer enhancement for Fast Path high-speed processing” on page 115
- “Automatic management of Fast Path DEDB SDEP buffers enhancement” on page 116
- “Database Image Copy 2 utility (DFSUDMT0) enhancement for zEnterprise Data Compression” on page 117
- “DBRC migration and coexistence” on page 117
- “DBRC REPAIR.RECON command enhancement” on page 118
- “DEDB Alter function enhancements” on page 120
- “Dynamic database definition enhancements” on page 123
- “Fast Path control block changes” on page 137
- “FDBR in-doubt thread support for ESAF” on page 138
- “GSAM metadata enhancement” on page 139
- “HALDB support for 8-GB OSAM data sets enhancement” on page 140

- “ODBM accounting enhancement” on page 142
- “OSAM 24-bit storage constraint relief enhancement” on page 142
- “SQL support enhancements” on page 143

IMS Transaction Manager enhancements overview

The IMS 14 Transaction Manager includes enhancements to the following functions: callout, Intersystem Communication (ISC), Multiple Systems Coupling (MSC), and Open Transaction Manager Access (OTMA).

- “APPC flood control enhancement” on page 145
- “IMS Connect support for cascading global transactions to IMS TM on different z/OS images” on page 146
- “ISC VTAM enhancement for ERP messages” on page 149
- “JVM usage statistics” on page 154
- “MSC enhancements” on page 156
- “OTMA enhancements” on page 169
- “Shared queues overflow feedback enhancement” on page 179
- “Synchronous callout enhancements” on page 180

IMS system enhancements overview

IMS 14 includes enhancements to the overall IMS system, including both the IMS Database Manager and the IMS Transaction Manager.

- “Command enhancements overview” on page 183
- “Dynamic refresh enhancement for PWFI and WFI regions” on page 189
- “External Subsystem Attach Facility enhancements” on page 191
- “IMS 64-bit storage manager enhancement” on page 193
- “IMS Connect command enhancements” on page 194
- “IMSRSC repository enhancements” on page 199
- “Reduced total cost of ownership enhancements” on page 205
- “User exit enhancements” on page 206

Chapter 10. IMS Database Manager enhancements

The IMS 14 enhancements to IMS Database Manager (IMS DB) include enhancements that enable dynamic database definition, enhancements to the Open Database Manager (ODBM), and other enhancements that improve usability and performance for users of IMS DB.

The Chapter 12, “IMS system enhancements,” on page 183 in IMS 14 might also impact IMS DB.

64-bit buffer enhancement for Fast Path high-speed processing

In IMS 14, the 64-bit buffer enhancement for Fast Path high-speed processing introduces the option to place the utility private buffers for the high-speed sequential processing (HSSP) and the High-Speed DEDB Direct Reorganization (DBFUHDR0) utilities in 64-bit common storage. Placing the buffers in 64-bit storage reduces IMS usage of 31-bit storage extended common storage (ECSA).

Before IMS 14, the utility private buffers for the HSSP and High-Speed DEDB Direct Reorganization (DBFUHDR0) utilities were always allocated in 31-bit extended common storage (ECSA). When the buffers are in ECSA, processing units of work (UOWs) with a very large number of segments can be prone to fail because of ECSA shortages. When ECSA shortages exist, HSSP cannot be run against DEDB areas and the High-Speed DEDB Direct Reorganization utility cannot be used to reorganize portions of a DEDB area, preventing you from taking advantage of high-speed processing for Fast Path DEDBs.

New options now enable the utility private buffers to be allocated in 64-bit common storage to reduce the need for ECSA. Allocating the buffers in 64-bit storage not only allows the HSSP and High-Speed DEDB Direct Reorganization utilities to run against very large UOWs, but also increases the number of high-speed utilities and HSSP application programs that can be run in parallel.

Restrictions

When you make an image copy of a Fast Path DEDB with HSSP, the utility private buffers for HSSP cannot be in 64-bit common storage. The buffers must be in 31-bit extended common storage.

Changes to administering IMS

If you enable 64-bit buffers for the High-Speed DEDB Direct Reorganization utility, you might need to increase the value of the BUFNO command control statement to increase the number of allocated buffer sets.

For HSSP, the DFSCCTL DD statement supports a new SETU control statement with the keywords FPBP64U=Y or FPBP64U=N. Similarly, for the High-Speed DEDB Direct Reorganization utility, the SYSIN DD statement supports the new keywords FPBP64U Y or FPBP64U N. If none of these keywords are specified, the utility private buffers are allocated in 31-bit extended common storage (ECSA) by default.

When you are processing units of work that have a very large number of segments, consider using the new keywords to have the buffers allocated in 64-bit common storage to avoid ECSA shortages.

The Fast Path 64-bit buffer manager does not have to be enabled to have the utility private buffers allocated in 64-bit common storage.

Changes to troubleshooting for IMS

Message DFS0533E is issued if you try to make an HSSP image copy and the utility private buffers are in 64-bit common storage.

For a list of the messages and codes that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Impact to utilities

The utility private buffers for the HSSP and High-Speed DEDB Direct Reorganization utilities can be allocated in 64-bit common storage or in 31-bit extended common storage. The output for the utility jobs indicates when 64-bit storage was used for the buffers.

Automatic management of Fast Path DEDB SDEP buffers enhancement

In IMS 14, IMS can be configured to automatically clean up Fast Path DEDB sequential dependent segments (SDEPs) in online buffers by writing them to the physical database at intervals that you specify.

If there is an imbalance in workload among multiple IMSes, you might be unable to physically delete SDEPs from a DEDB Area. To avoid this problem, this enhancement enables automatic periodic clean up of the SDEPs.

To enable this function, you need to specify parameters in the FASTPATH section of the DFSDFxxx member of the IMS PROCLIB data set. You can specify either a time of day or a frequency for the SDEP buffers to be cleaned up.

Restrictions

This enhancement is useful only if you have DEDBs that use SDEPs and data sharing.

Changes to installing and defining IMS

To configure IMS to automatically clean up Fast Path DEDB sequential dependent segments (SDEPs), this line item introduces several new parameters in the FASTPATH section of the DFSDFxxx member of the IMS PROCLIB data set:

- SDEPAUTO
- SDEPEXP
- SDEPFREQ
- SDEPTIME

Changes to administering IMS

For a list of topics for administering IMS that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

Changes to troubleshooting for IMS

For a list of the messages and codes that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Database Image Copy 2 utility (DFSUDMT0) enhancement for zEnterprise Data Compression

In IMS 14, the IMS Database Image Copy 2 utility (DFSUDMT0) is enhanced to indicate a preference for zEnterprise Data Compression (zEDC) services, so that if zEDC is available in z/OS when the utility creates an image copy, zEDC is used for compression.

The zEnterprise Data Compression support is a function of z/OS that provides better compression ratios compared to traditional software-based compression services and uses fewer CPU cycles.

The Database Image Copy 2 utility requests compression services from DFSMSdss by issuing the DUMP command with the COMPRESS ZCOMPRESS(PREFERRED) options indicated. If zEDC is available, image copies are compressed by using zEDC services. If zEDC is not available or the request is rejected for any reason, the standard DFSMSdss compression services are used.

Requirements

The DFSMSdss APAR OA42238 (PTF UA74782) must be installed to exploit hardware compression and decompression.

Changes to utilities

The Database Image Copy 2 utility (DFSUDMT0) is enhanced to indicate COMPRESS ZCOMPRESS(PREFERRED) on the DFSMSdss DUMP command when the utility requests compression services from DFSMSdss.

DBRC migration and coexistence

DBRC enhancements include changes to the RECON data set format for IMS 14 and support that allows IMS Version 12 and IMS Version 13 systems to coexist until all of the systems are upgraded to IMS 14.

The RECON data set changes for IMS 14 are as follows:

- The header version indicator is set to V14R1.
- The default value for MINVERS is set to "12.1."
- The RECON level is set to "14.1" (X'E0).

- The log release level in the PRILOG, SECLOG, PRISLD, SECSLD, PRIOLD, and SECOLD (and interim) records is set to X'37', but is listed as "14.1".
- The version and coexistence level in the subsystem record is set to "14.1" (X'E0').

To support the HALDB support for 8-GB OSAM data sets enhancement:

- The INIT.DB and CHANGE.DB commands are enhanced with the OSAM8G and NOOSAM8G keywords. These keywords indicate whether the maximum data set size of OSAM database data sets in a PHDAM or PHIDAM database is 4 or 8 gigabytes.
- The TYPE=HALDB DB record is changed so that the DSORG field indicates the maximum size of the OSAM database data sets.
- The output for the LIST commands is also changed, where appropriate, to show the maximum size of the OSAM database data sets.

To support the IMS management of ACBs:

- The INIT.RECON and CHANGE.RECON commands are enhanced with the new CATALOG(*name*) and NOCATALG keywords. In an IMS system that manages ACBs, DBRC retrieves database definitions from the IMS catalog. The IMS catalog can be identified either in the RECON header or the DBRC command that is being issued (CHANGE.DBDS, CHANGE.PART, INIT.DB, INIT.DBDS, INIT.PART, and NOTIFY.REORG).
- The RECON record is enhanced with a new catalog field that identifies the name of the IMS catalog that the DBRC commands use for database definitions.
- The output of the LIST.RECON command is also changed to include the name of the IMS catalog or CATALOG=**NULL** if DBRC commands use a DBD library for database definitions.

Certain DBRC API requests are enhanced to show the new information that is in the RECON data sets.

The output block version number of the DBRC API is as follows:

- 6.0 for the IMS 14 API.
- 5.0 for the IMS Version 13 API.
- 4.0 for the IMS Version 12 API.

DBRC REPAIR.RECON command enhancement

In IMS 14, the DBRC REPAIR.RECON command enhancement provides an easy way to fix errors and inconsistencies in the RECON data set, which simplifies upgrading the RECON data set between versions of IMS.

In the RECON data set, the DMB Table record keeps track of which DMB (data management block) numbers are in use. The record is created when the first database record is registered in the RECON data set. The record is updated as new databases are registered and existing databases are deleted from the RECON data set. The DMB Table record also keeps track of the total number of databases that are registered.

A DMB number is assigned when a database is registered in the RECON data set. The DMB number is used to identify the database. The DMB number assigned to HALDB, DEDB, and IMS (non-HALDB) full function databases is a global DMB number and is used in data sharing. The global DMB number is a two-byte field, and its high-order bit is always on.

The following problems, which are typically created outside of DBRC control, can occur in the RECON data set:

- A missing or corrupted DMB Table record
- Inconsistencies in the DMB numbers that are recorded in the RECON header, database, and Fast path area records

In the past, the DMB Table record could be built or rebuilt and the DMB numbers could be fixed only during the RECON upgrade process when upgrading a lower release RECON to either IMS Version 12 or IMS Version 13. In IMS 14, the new DBRC REPAIR.RECON command is added to fix these errors and inconsistencies. No longer fixing these types of problems during the RECON upgrade process makes migration to new versions of IMS faster. In addition, being able to detect and fix problems that occur in the RECON data set after the upgrade process has been completed improves the integrity of the RECON data set.

Migration considerations

After the RECON data set has been upgraded to IMS 14, consider issuing the REPAIR.RECON DMBNUM CHECKUP command to verify that all of the DMB numbers in the RECON header, database, and Fast Path area records are consistent. If problems are found when you verify the DMB numbers, issue the REPAIR.RECON DMBNUM UPDATE command to fix inconsistent DMB numbers in the records.

If you use security procedures to verify the authority of a user to issue DBRC commands, update the DBRC Command Authorization exit routine (DSPDCAX0) or RACF command authorization definitions to authorize the use of the REPAIR.RECON command. As with other DBRC commands, consider restricting access to only those users who must issue the command.

Restrictions

The DBRC REPAIR.RECON command is supported only in batch using the Database Recovery Control utility (DSPURX00) or the DBRC API.

When inconsistencies are found in the DMB numbers, the DMB numbers can be changed only if the database or area is unauthorized.

Impact to administering IMS

Specific problems, which are most often created outside the control of DBRC, can arise in the RECON data sets. You can use the new REPAIR.RECON command to find and fix problems with corrupted or missing DMB Table records or inconsistencies with the DMB number in the RECON header, database, and Fast Path area records.

The two main options on the REPAIR.RECON command are DMBTABLE and DMBNUM. The DMBTABLE option rebuilds the DMB Table record and fixes inconsistent DMB number entries found elsewhere in the RECON data set. If the DMB Table record is missing, it is created. The DMBNUM option provides the ability to fix just the DMB number inconsistencies and allows you to limit the scope of the command to all database records, certain types of database records, a specific database record, or only the RECON header record.

The REPAIR.RECON command can read and change a large number of records, impacting the availability of the RECON data sets. Therefore, consider running the REPAIR.RECON command on a backup copy of the RECON data sets to assess the impact. Then, run the command when access to the actual RECON data sets is minimal. For REPAIR.RECON with the DMBNUM option, first run the command with the CHECK option on the backup copy of the RECON, and, if any DMB numbers need to be fixed, then run the command with the UPGRADE option on the actual RECON.

Impact to troubleshooting for IMS

For a list of the messages and codes that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Impact to commands

This enhancement introduces the new DBRC REPAIR.RECON command.

For a list of the commands that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

Impact to exit routines

If you use the DBRC Command Authorization exit routine (DSPDCAX0), update the routine to authorize users to issue the REPAIR.RECON command.

DEDB Alter function enhancements

In IMS 14, the DEDB Alter function enhancements introduce several new capabilities for altering the physical attributes of a DEDB area while the DEDB area or the DEDB database remains accessible online.

You can now use the DEDB Alter utility (DBFUDA00) to make the following changes without taking the area offline:

- Alter the SIZE, UOW, or ROOT specifications in the definition of a DEDB area with or without sequential dependent (SDEP) segments defined
- Add one or more DEDB areas at the end of a DEDB database with the new DEDB Alter ADDAREA function
- Replace the randomizer that is used for the DEDB database by using the DEDB Alter ALTERAREA, REPLRAND, or ADDAREA function
- Add a Segment Edit/Compression exit routine as part of the ALTERAREA, REPLRAND, or ADDAREA function.

If you are adding a Segment Edit/Compression exit routine, during DEDB Alter utility execution the new Segment Edit/Compression exit is not invoked for any areas. The Segment Edit/Compression exit routine is not enabled during DEDB Alter processing. Instead, it is enabled for all DEDB areas after successful DEDB Alter utility completion.

You can now also use the DEDB Alter utility in FDBR and XRF environments when ACBSHR=N. The IMS 14 DEDB Alter functions support all IMS environments including FDBR and XRF for ACBSHR=Y and ACBSHR=N.

If you enable ACB libraries by specifying ACBMGMT=ACBLIB in the DFSDFxxx member of the IMS PROCLIB data set, the DEDB Alter utility supports both FDBR and XRF for ACBSHR=Y and ACBSHR=N.

If you enable IMS to manage ACBs by specifying ACBMGMT=CATALOG in the DFSDFxxx member, the DEDB Alter utility supports both ACBSHR=Y and ACBSHR=N. When IMS manages ACBs, the DEDB Alter utility supports only FDBR.

If the IMS management of ACBs is enabled and the DEDB Alter utility fails when you are altering a DEDB database or area, the DEDB definitions that are used to alter the DEDB remain in the IMS catalog.

To run the DEDB Alter utility in IMS systems that manage ACBs in the IMS catalog, install the following APARs/PTFs:

- PI50757/UI32921 DEDB Alter Catalog Infrastructure
- PI52007/UI33824 DEDB Alter Catalog Infrastructure
- PI53603/UI34358 DEDB Alter Catalog Infrastructure
- PI54901/UI35596 DEDB Alter Catalog Online Support for ACBSHR=Y

If you use the IMS generation utilities to alter a DEDB database and the IMS management of ACBs is enabled, until APAR PI55681 is installed, the DEDB Initialization utility (DBFUMIN0) still requires the staging ACB library. The staging ACB library is used as input to properly format the shadow data sets for the DEDB Alter utility.

The DEDB Alter function is invoked using the DEDB Alter utility (DBFUDA00). The DEDB Alter utility supports three functions: ALTERAREA, ADDAREA, and REPLRAND. Only one function can be invoked in a single DEDB Alter utility execution.

Requirements:

- A two-stage randomizer must be used, which enables areas to be processed individually.
- All IMS data sharing systems must be at IMS Version 13 or later. The MINVERS value in the RECON data set must be 13.1 or later.
If you are using function of the DEDB Alter utility that was added in IMS 14 or later, such as ADDAREA, all IMS data-sharing systems must be at the IMS 14 or later level. The MINVERS value in the RECON data set must be 13.1 or later. If the MINVERS value in the RECON data set is not set to 13.1 or later, the DEDB Alter utility terminates without performing the requested function.
- The DEDB databases must be registered to DBRC.
- If you are using the DEDB Alter utility to add a Segment Edit/Compression exit routine, the Segment Edit/Compression exit routine must be able to handle mixed compressed and non-compressed data in a DEDB database. If data is non-compressed, on read access, it does not expand the non-compressed data, and on write access, it compresses the data and writes out as compressed data. If data is compressed, on read access, it expands the compressed data, and on write access, it compresses the data and writes out as compressed data.

Restrictions:

- You can alter only one area in a DEDB database at a time with the DEDB Alter utility. You can have multiple instances of the DEDB Alter utility running concurrently, but each instance must be running against a different DEDB database.
- Only one DEDB Alter utility with the ADDAREA function can be executed in an IMSplex at a time. However, concurrent DEDB Alter utilities with the ALTERAREA or REPLRAND functions can be executed for different DEDB databases. You can add 1 to 100 DEDB areas to the end of a DEDB database in a DEDB Alter utility execution.
- The DEDB Alter function does not support DEDB databases that are in Virtual Storage Option (VSO) or Shared Virtual Storage Option (SVSO) mode. DEDB databases with the VSO or SVSO option must first be unloaded with the /VUNLOAD command before running the DEDB Alter utility.
- Because the IMS management of ACBs does not support XRF environments, the DEDB Alter utility does not support XRF when IMS manages ACBs.

Migration considerations

To run the DEDB Alter utility, all IMS data-sharing systems must be at the IMS Version 13 or later level. The MINVERS value in the RECON data set must be 13.1 or later. The default value for MINVERS in IMS 14 is 12.1.

If you are invoking the DEDB Alter utility for an IMS 14 function (such as ADDAREA), the DEDB Alter utility detects the version of the IMS system and terminates if the IMS version does not support the particular change.

Coexistence considerations

To run the DEDB Alter utility, all IMS data-sharing systems must be at the IMS Version 13 or later level. The MINVERS value in the RECON data set must be 13.1 or later. The default value for MINVERS in IMS 14 is 12.1.

If you are using function of the DEDB Alter utility that was added in IMS 14 or later, such as ADDAREA, all IMS data-sharing systems must be at the IMS 14 or later level. The MINVERS value in the RECON data set must be 13.1 or later. If the MINVERS value in the RECON data set is not set to 13.1 or later, the DEDB Alter utility terminates without performing the requested function.

You must install IMS Version 13 APAR/PTF PI29543/UI24793 so that the DEDB Alter utility terminates with a non-zero return code when the ADDAREA function is attempted on a DEDB area that is shared by an IMS Version 13 system.

Log record changes

The following additional sub codes were added to the X'5961' log record for this enhancement in IMS 14:

X'02' For adding a compression exit routine

X'0A' For adding an area with ADDAREA

Impact to installing and defining IMS

Impact to administering IMS

For a list of topics for administering IMS that are changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

Programming for IMS

Impact to troubleshooting for IMS

For a list of the messages and codes that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Impact to commands

For a list of the commands that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

Impact to exit routines

If you are using the DEDB Alter utility to add a Segment Edit/Compression exit routine, the Segment Edit/Compression exit routine must be able to handle mixed compressed and non-compressed data in a DEDB database.

Impact to utilities

For a list of the utilities that are changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

Dynamic database definition enhancements

IMS 14 introduces several enhancements that support dynamic database definition: the IMS management of ACBs, IMS support for the SQL data definition language (DDL), and the ability to record the DDL statements that are submitted to IMS.

Used together, the IMS management of ACBs and the IMS DDL interface provide an alternative to the cumbersome PSBGEN, DBDGEN, ACBGEN, and Online Change processes that were previously required to create or modify databases (DBDs) and the application program views (PSBs) of those databases before IMS 14.

When the IMS management of ACBs is enabled, the online IMS system no longer uses the ACB, DBD, and PSB libraries as the definitive repository of the runtime definitions of databases and program views. Instead, IMS loads the runtime database and program view definitions from the IMS catalog.

To support tools, utilities, and processes that might continue to require the libraries, you can still use the generation utilities and ACB, DBD, and PSB libraries,

even after the IMS management of ACBs is enabled. However, you do need to make sure that the resources in the ACB, DBD, and PSB libraries match the active resources in the IMS catalog.

To submit DDL to IMS, you can use a tool like the IMS Enterprise Suite Explorer for Development or a Java batch utility that is provided by IMS.

The DDL statements that are submitted to IMS can be captured by using the new X'03 subtype of the IMS-specific X'29' z/OS System Management Facility (SMF) record.

Data Definition Language (DDL) for IMS

In IMS 14, you can use industry standard Data Definition Language (DDL) statements to create, modify, and delete IMS databases and application program resources. The definitions that are either added or modified in IMS by using DDL are no longer stored in the DBD, PSB, or ACB libraries. Instead, they are stored in the IMS catalog and its associated data sets.

Creating or changing a database or application resource with DDL statements allows you to change the active definitions in the target IMS system, without requiring a system generation process or online change. Instead, you submit the DDL to IMS, and IMS automatically stores the definitions in the IMS catalog. In most cases, after IMS receives the DDL changes, the changes must be activated in an online IMS system by issuing the IMS type-2 command IMPORT DEFN SOURCE(CATALOG). If you are creating a new database, additional steps might be required in the IMS system before the database can be activated.

When the IMS management of ACBs is enabled, the IMS catalog manages the database and application metadata for all IMS users. The DBD and PSB macro statements, DBD, PSB, and ACB generation, and the DBD, PSB, and ACB libraries are no longer needed. Instead, IMS manages the DBD and PSB definitions and the associated ACBs in the IMS catalog.

Mapping DDL terms to IMS terms

The terms that DDL uses to refer to database concepts differ from some of the terms that are used by IMS.

For example, in a DDL context, an IMS *program specification block (PSB)* and its *program communication blocks (PCBs)* are referred to as a *program view* and its *schemas* is synonymous with the term *program specification block (PSB)*, which is used in an IMS context. Similarly, in a DDL context, an IMS *database descriptor (DBD)* might be referred to as a *database definition*.

Terms in the DDL syntax are primarily based on a relational database model. Consequently, DDL does not contain many of the terms that IMS uses to describe concepts that are specific to the IMS hierarchical database model. In the IMS implementation of DDL, some of the relational-database terms in the DDL syntax have been preserved, but they define structures that in IMS are known by different terms. For example, the terms *table* and *column* from DDL map to the IMS terms *segment* and *field*. In many cases, where IMS terms do not map to existing DDL terms, IMS extends standard DDL to include IMS-specific terms.

Automation options for DDL CREATE statements

DDL ALTER and DDL CREATE prepare resources as pending definition in the catalog. Subsequently, a separate IMPORT DEFN SOURCE(CATALOG) would be issued in order to bring the DDL defined resources online and as active definition in the catalog.

The automation options for DDL CREATE statements provides the capability to prepare and bring DDL CREATE databases and programs resources online and as active definitions in the catalog without an IMPORT DEFN SOURCE(CATALOG) command for one or more IMS systems.

The automation options for DDL CREATE statements require the installation of the following APARs:

- PI67302: Include IGD messages before IGD17219I for extended format data set allocation failure
- PI64211: NEW FUNCTION: DDL EXTENSION CREATE DATABASE | PROGRAMVIEW
- PI69585: DDL Extensions CREATE SPE data sharing support

For **ACBMGMT=CATALOG** specified in <SECTION=CATALOG> in DFSDFxxx PROCLIB:

- Provides DDL Auto Import Create Database and Create Program View support for DEDB databases, full function databases, HALDB databases, and their new program views for **ACBSHR=N** for one or more IMS systems in an IMSplex and **ACBSHR=Y** for single IMS.
- Provide Dynamic Resource Definition (DRD) support for newly defined DDL CREATE resources when **MODBLKS=DYN** is enabled.
- Provide Auto Create DEDB Area Data Sets:
 - Allocates and formats DEDB Area data sets.
 - Registers DEDB database to DBRC.

If **MODBLKS=DYN** specified in <SECTION=COMMON_SERVICE_LAYER> in DFSDFxxx PROCLIB:

- Provides Dynamic Resource Definition support to dynamically create database and program runtime attributes if they do not exist for these new resources.

Default descriptors are used to create the associated runtime attributes with overrides, for example:

- DB attribute overrides
 - Access: Update
 - If DEDB, Resident: Yes
- PGM attribute override
 - Program Type: BMP

With APAR/PTF PI71632/UI44375, you can use the following parameters in the DDL section of the DFSDFxxx member of the IMS.PROCLIB data set. If a DEDB database is created by using the CREATE DATABASE DDL statement and a DEDB area data set is automatically allocated by IMS for the database, you can use the parameters to define the DEDB area data set.

DSN= Defines the data set name.

DSNDATA=

Defines the data component name.

With APAR/PTF PI71632/UI44375, you can also use the **VOLUME=** parameter in the DDL section of the DFSDFxxx member. If a DEDB database is created by using the CREATE DATABASE DDL statement and a DEDB area data set is automatically allocated by IMS for the database, you can use the **VOLUME=** parameter to specify multiple volumes for the DEDB area data set.

Migration considerations

Changes that are submitted to IMS by DDL are not available in DBD, PSB, or ACB libraries. Until all applications and utilities are configured to retrieve DBD, PSB, and ACB information from the IMS catalog, you can retrieve DBD and PSB changes that are submitted by DDL from the IMS catalog by using the new IMS Catalog Library Builder utility (DFS3LU00).

Coexistence considerations

You cannot submit DDL to an IMS Version 13 system or earlier.

All IMS systems that share an ACB library (ACBLIB) must be migrated to IMS 14 at the same time. After the IMS systems that shared the ACBLIB are migrated to IMS 14, some of the systems in the sharing group can continue to use ACBLIBs, while other IMS systems in the group are migrated to IMS-managed ACBs, as long as the ACBs in the ACBLIB and the ACBs in the IMS catalog are the same.

Until all of the IMS systems that share ACBs are migrated to use IMS-managed ACBs, avoid using either DDL or the IMS generation utilities to make changes to the active ACBs. If changes to ACBs are required, make sure that the changes are activated in both the IMS catalog and the ACBLIB. If ACB changes that are submitted by DDL must be activated, use the IMS Catalog Library Builder utility (DFS3LU00) to copy the pending ACB changes from the IMS catalog to a staging ACBLIB data set. You can then use the Online Change process to activate the ACB changes. The pending changes in the IMS catalog are activated by issuing the IMPORT DEFN SOURCE(CATALOG) command.

Requirements

IMS support for DDL requires that the IMS management of ABCs is enabled. For more information, see “IMS management of ACBs enhancement” on page 127.

Restrictions

DDL is supported only through a Java client using the IMS Universal drivers, through IMS Enterprise Suite Explorer for Development, or through tools that support the IMS Universal drivers.

IMS DDL does not support Fast Path secondary indexes. Do not issue CREATE or ALTER statements for Fast Path secondary indexes.

If you specify any of the following keywords on the CREATE TABLE statement, you cannot change the keyword and keyword value by using the ALTER TABLE statement. To change the keyword and keyword value, you must first use the

DROP TABLE statement to delete the table. Then, you must re-create the table by using the CREATE TABLE statement and specify the keyword and keyword value again.

- The INTERNALNAME`internalname` keyword to specify the internal name of the segment type being defined.
- DIRECT DEPENDENT | SEQUENTIAL DEPENDENT

Impact to installing and defining IMS

To enable support for DDL, your IMS system must be configured to use the IMS catalog and to manage ACBs. The IMS management of ACBs replaces DBD, PSB, and ACB libraries.

For more information, see “IMS management of ACBs enhancement”

Impact to administering IMS

You can use IMS DDL to create, modify, and delete IMS databases and application resources. DDL commands can be submitted with the IMS Universal Drivers, through tools that support them, and the SQL Batch utility. For more information about the new statements that are defined to support DDL, see Statements.

Programming for IMS

New programming information is added about DDL support for type-4 connections using the IMS Universal JDBC driver. For more information about the new statements that are defined to support DDL, see Statements.

Impact to troubleshooting for IMS

Several SQL completion codes have been added for DDL support.

With APAR/PTF PI46915/UI43180, you can use the DDL trace to write DDL events in IMS to an internal, type-2 trace table.

For a list of the messages and codes that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, “Message and code changes in IMS 14,” on page 95.

Impact to utilities

The SQL Batch utility has been added for DDL support. This utility is delivered with APAR PI30848.

IMS management of ACBs enhancement

IMS can manage the runtime *application control blocks* (ACBs) for databases and program views for you. When IMS manages ACBs, IMS no longer requires DBD, PSB, and ACB libraries. You can define databases and program views by using SQL DDL statements or by using the IMS generation utilities.

ACBs are the runtime blocks that represent the active databases and program views in online and batch IMS environments. They are created from the databases and program views that you define to the IMS system.

Most ACBs are pre-built, stored in a data set in binary format, and loaded into memory by the online IMS system or by batch application programs. Some application programs and some utilities, such as those that run in an offline DL/I batch region, build ACBs dynamically during run time.

When IMS manages ACBs, as indicated by `ACBMGMT=CATALOG` in the `<CATALOG>` section of the `DFSDFxxx` member, IMS stores ACBs in the IMS directory, a collection of system-managed data sets that are associated with an IMS catalog. The IMS directory data sets include data sets for the active ACBs, a staging data set for ACBs that are pending activation, and a boot strap data set that IMS uses to manage the IMS directory.

When IMS managed ACBs, IMS can build, activate, and load ACBs into memory dynamically when database and program view definitions are submitted to IMS by using SQL DDL statements. Upon receiving the SQL statements, IMS automatically updates the IMS catalog and can, if directed to do so, activate certain changes to database or program view definitions automatically.

Changes that are not activated automatically or that are saved in the staging data set for later activation, can be activated by issuing the IMS type-2 `IMPORT DEFN SOURCE(CATALOG)` command. IMS type-2 command support requires the Operations Manager (OM) and Structured Call Interface (SCI) components of the IMS Common Service Layer (CSL). In data sharing environments, the CSL Resource Manager (RM) component is also required.

Defining databases and program views with SQL DDL statements is an alternative to the process of coding DBD and PSB source, generating DBDs, PSBs, and ACBs with utilities, and performing an Online Change process to activate the resulting ACBs. The DDL statements can be submitted to IMS through a separate product, such as the IMS Enterprise Suite Explorer for Development.

When IMS manages ACBs, you can also still define databases and program views by using the DBD and PSB generation utilities and build ACBs by using either the ACB Maintenance utility or the ACB Generation and Catalog Populate utility (`DFS3UACB`).

If you use the `DFS3UACB` utility, in addition to building the ACBs, the utility can also update the IMS catalog and activate the ACBs by loading them into the IMS directory. If you do not use the `DFS3UACB` utility, you can achieve the same results by using the ACB Maintenance utility and the IMS Catalog Populate utility (`DFS3PU00`).

The IMS Catalog Directory Recovery utility (`DFS3RU00`) can also be used to rebuild the IMS directory and write the online resources into the IMS directory data sets.

Both the `DFS3UACB` and the `DFS3PU00` utilities require exclusive access when they update the IMS directory directly. So, if the utilities are run in the `UPDATE` mode, the IMS systems that use the IMS catalog that is being updated must be shut down. To avoid shutting down the IMS systems, run the utilities in `STAGE` mode and add them to the IMS directory by using the `IMPORT DEFN SOURCE(CATALOG)` command.

In many cases, whether you use DDL or the generation utilities, new or changed ACBs are placed in the staging data set as pending changes and you must activate the ACBs by issuing the `IMPORT DEFN SOURCE(CATALOG)` command. The Online Change process is not used. In other cases, such as when you delete a resource, the changes to the ACBs are activated automatically.

When you enable IMS management of ACBs, IMS performance is similar to what it is when an ACB library is used. IMS uses the same amount of I/O to access IMS-managed ACBs as it does to access ACBs in an ACB library.

Before IMS can manage ACBs, IMS and the IMS catalog must be set up to support ACB management by running the `DFS3PU00` utility with the `MANAGEDACBS=SETUP` control statement that you provide as input. After IMS is set up for ACB management, you specify `ACBMGMT=CATALOG` in the `<CATALOG>` section of the `DFSDFxxx PROCLIB` member.

Migration considerations

When you enable the IMS management of ACBs, the new system data sets that are required to manage the ACBs can use slightly more storage than the ACB libraries. However, IMS manages these system data sets, so you do not need to allocate the storage or the data sets.

In IMS 14, the IMS management of application control blocks (ACBs) introduces migration considerations for users of the IMS catalog.

When IMS manages ACBs, which are the runtime control blocks for active databases and program views, the IMS catalog replaces ACB libraries as the repository of the active ACBs. Consequently, backup and recovery procedures for the IMS catalog are critical. Create image copies of the IMS catalog and use them with the logs to recover the IMS catalog. In previous releases of IMS, you might have been able to rebuild the IMS catalog from your ACB libraries. However, when IMS manages ACBs, you can use SQL DDL to add or modify database and program views. Changes made by using DDL would not be contained in an ACB library.

The default access level of the IMS catalog changes from `ACCESS=READ` to `ACCESS=UPDATE` when the IMS management of ACBs is enabled.

When the IMS management of ACBs is enabled, an IMS catalog functions like an ACB library for all of the IMS systems that use that IMS catalog. So, if you are migrating a multi-system environment in which the IMS systems share an IMS catalog but each system has its own ACB library, enabling the IMS management of ACBs effectively converts your IMS systems from using multiple, independent ACB libraries to sharing a single set of ACBs.

An IMS catalog supports only a single active instance of an ACB for each version of IMS that uses the IMS catalog. In the IMS catalog, each instance of an ACB is identified by the time stamp that is taken when the ACB was created. When you set up an IMS catalog to manage ACBs for multiple IMS systems that each used their own ACB library, if different instances of the same ACB appear in different ACB libraries, only one instance of the ACB is loaded into the IMS directory data set, a system-managed data set that is an extension of the IMS catalog. Usually, the instance that is loaded into the IMS directory is the instance with the most recent time stamp.

Before you enable the IMS management of ACBs, if multiple IMS systems use separate ACB libraries that contain common ACBs, make sure that all of the common ACBs in the separate libraries were built from the same DBD and PSB libraries.

After the IMS management of ACBs is enabled, any changes to an active ACB in the IMS catalog potentially impacts all IMS systems that share the IMS catalog.

Although it is not recommended to do so, you can enable a dedicated IMS catalog for each IMS system that needs separately maintained ACBs. However, the benefits of sharing an IMS catalog would be lost.

When the IMS management of ACBs is enabled and DDL statements are used to define or modify databases or program views, the online IMS system processes the DDL statements, updates the IMS catalog, and builds the runtime control blocks. So, the costs that are associated with this processing are incurred by the online IMS system.

This processing cost is avoided during the initial migration to IMS-managed ACBs because the batch DBD and PSB generation utilities are used to create the IMS 14 DBD and PSB control blocks. Also the IMS Catalog Populate utility can be run in batch mode when it updates the IMS catalog and sets up the IMS system to manage ACBs.

After the IMS management of ACBs is enabled, if your installation uses DDL heavily to define or modify databases and program views, you might see an increase in DL/I processing over the processing in IMS systems that are not enabled for the IMS management of ACBs.

Coexistence considerations

Before you enable the IMS management of ACBs for the first time in a data sharing environment, migrate all IMS Version 13 and earlier systems in the data sharing group to the new IMS release.

Requirements

In IMS 14, if you are running IMS on z/OS 2.1, the IMS management of ACBs function requires z/OS 2.1 DFSMS APAR OA45400 applied via PTF UA73855.

The Operations Manager (OM) and Structured Call Interface (SCI) components of the IMS Common Service Layer (CSL) must be enabled for the IMS type-2 command `IMPORT DEFN SOURCE(CATALOG)` command.

In data sharing environments, the CSL Resource Manager (RM) component is required.

Restrictions

The following restrictions currently apply to the IMS management of ACBs function.

Change coordination restrictions with the IMS management of ACBs

In data sharing environments, if `ACBSHR=Y` is specified, the activation of resource changes in IMS systems that share an IMS catalog is always coordinated globally by the `IMPORT DEFN SOURCE(CATALOG)`

command. Activating resources locally or separately in the individual IMS systems that share the IMS catalog is not supported. If you require more control over the activation of resource changes in IMS systems that share ACBs, use a separate aliased IMS catalog to manage the ACBs for each IMS system.

Database utilities and region type restrictions

Some IMS database-related utilities, and some vendor products that use the UDR or ULU region type, might still require access to DBDLIB and PSBLIB data sets even when the IMS managed ACBs function is enabled.

The following IMS utilities still require the use of DBDLIB or PSBLIB data sets:

- DEDB Area Data Set Compare utility (DBFUMMH0)
- DEDB Area Data Set Create utility (DBFUMRI0)
- DEDB Sequential Dependent Delete utility (DBFUMDL0)
- DEDB Sequential Dependent Scan utility (DBFUMSC0)
- Database Preorganization utility (DFSURPR0)
- Database Prefix Resolution utility (DFSURG10)
- Database Prefix Update utility (DFSURGP0)
- Database Scan utility (DFSURGS0)
- Database Surveyor utility (DFSPRSUR)
- HALDB Index/ILDS Rebuild utility (DFSPREC0)
- HALDB Migration Aid utility (DFSMAID0)
- HISAM Reorganization Reload utility (DFSURRL0)
- HISAM Reorganization Unload utility (DFSURUL0)
- High-Speed DEDB Direct Reorganization utility (DBFUHDR0)
- Partial Database Reorganization utility (DFSPRCT1 and DFSPRCT2)
- Utility Control Facility (DFSUCF00)

IMS catalog population restrictions

If IMS manages ACBs, you must shut down IMS when you run one of the following utilities with the SETUP or the UPDATE parameter specified on the MANAGEDACBS= control statement:

- The IMS Catalog Populate utility (DFS3PU00)
- The ACB Generation and Catalog Populate utility (DFS3UACB)

However, you can run the DFS3PU00 utility with MANAGEDACBS=STAGE specified without shutting down IMS, even if IMS manages ACBs.

XRF does not support the IMS management of ACBs

The Extended Recovery Facility (XRF) function does not support the IMS management of ACBs.

RSR does not support the IMS management of ACBs

The Remote Site Recovery function does not support the IMS management of ACBs.

Impact to installing and defining IMS

The IMS management of ACBs requires the IMS catalog. If your IMS system does not already use the IMS catalog, you must enable the IMS catalog when you enable

IMS management of ACBs. The IMS Installation Verification Program includes steps for enabling the IMS catalog, but not for enabling the IMS management of ACBs.

Use the DFS3PU00 utility to set up IMS to manage ACBs. Using your existing ACB libraries as input, the DFS3PU00 utility updates the records in the IMS catalog to indicate the active instances of the databases and program views and loads the ACBs into the IMS directory system data set.

By default, the high-level qualifier of the IMS directory data sets is derived from the data set name prefix of the IMS catalog database data sets. You can use a different high-level qualifier by specifying the SYSDSNHLQ= parameter in one of the following places, which are listed in the order in which IMS uses them:

1. A TYPE=CATDSDLQ DFSMDA macro statement
2. The Catalog Definition exit routine (DFS3CDX0)
3. The CATALOG section of the DFSDFxxx PROCLIB member

After IMS is set up to manage ACBs, ACBMGMT=CATALOG must be specified in the CATALOG section of the DFSDFxxx member of the PROCLIB data set and the IMS systems that use the IMS catalog must be restarted. If they are not specified already, CATALOG=Y and ALIAS=*name* must also be specified.

Impact to administering IMS

When the IMS management of ACBs is enabled, IMS no longer uses the ACBLIB, DBDLIB, and PSBLIB data sets. Instead, IMS uses the IMS catalog. The system-managed directory data sets of the IMS catalog functionally replaces the active ACBLIB data set. The IMS system loads the runtime control blocks from the directory data sets at startup or when the IMPORT DEFN SOURCE(CATALOG) command is issued.

Previously, the only way to define DBDs and PSBs was with generation utility macro statements. When the IMS management of ACBs is enabled, you can still use the generation utility definitions, but you can also define DBDs and PSBs by using the SQL Data Definition Language (DDL). DDL can be submitted to IMS from IMS Enterprise Suite Explorer for Development or by using the new SQL Batch utility. Upon receiving the DDL, IMS generates the ACBS for you, stores them in the IMS catalog, and, in some cases, automatically activates the resources in the IMS system.

When IMS manages ACBs, you can no longer activate ACBs by using the online change process. Instead, you activate most changes by issuing the IMPORT DEFN SOURCE(CATALOG) command. In some cases, if you define your DBDs and PSBs by using DDL, IMS can activate certain changes automatically when you submit the DDL to IMS.

This change in ACB management potentially impacts any process, application program, tool, or product that previously required access to the ACB, DBD, and PSB libraries. In an online IMS 14 system, when the IMS management of ACBs is enabled, IMS switches IMS application programs, utilities, and other functions to the use of the IMS catalog automatically. However, for batch environments, you might need to update JCL or use the IMS Catalog Definition exit routine to have application programs switch to using the IMS catalog.

Additionally, any procedures, tools, or products that are provided by your company or other product vendors might need to be updated.

You can still use the ACB, DBD, and PSB generation utilities after the IMS management of ACBs is enabled, or you can create the libraries from the IMS catalog by using the IMS Catalog Library Builder utility (DFS3LU00). The IMS Catalog Library Builder utility creates complete libraries in either control block format or generation utility macro instruction format. You can also list or retrieve one or more individual resources from the IMS catalog by using the IMS Catalog API (DFS3CATQ). The IMS Catalog API returns the resources in control block format.

During a recovery process when the IMS management of ACBs is enabled, if you need to recover the online resources that are created by DDL, use the IMS Catalog Directory Recovery utility (DFS3RU00).

Programming for IMS

When the IMS management of ACBs is enabled, either the JCL for batch application programs needs to be updated to indicate that the IMS catalog is enabled and that IMS is managing ACBs or if the JCL cannot be modified, the Catalog Definition exit routine (DFS3CDX0) can be used instead.

Any application programs that previously referenced the ACB, DBD, or PSB libraries reference the IMS catalog instead when IMS manages ACBs.

Impact to troubleshooting for IMS

The IMS management of ACBs enhancement introduces many new DFS messages that are issued by IMS and some IMS utilities, such as the IMS Catalog Populate utility (DFS3PU00).

For a list of the messages and codes that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Impact to commands

The IMS management of ACBs enhancement updates several IMS and DBRC commands.

The following DBRC commands are enhanced to either specify the name of an IMS catalog to use or to display the current default IMS catalog:

- CHANGE.DBDS
- CHANGE.PART
- CHANGE.RECON
- INIT.DB
- INIT.DBDS
- INIT.PART
- INIT.RECON

- LIST.RECON
- NOTIFY.REORG

IMPORT DEFN SOURCE(CATALOG)

A new CATALOG option can be specified on the SOURCE keyword of the IMPORT DEFN command to support the activation of resources that are saved in the IMS catalog by importing the ACBs of those resources into the system-managed IMS directory.

A new OPTIONS() keyword allows you to specify options when you issue the IMPORT DEFN SOURCE(CATALOG) command. The options that you can specify include:

- DELPENDERR, which cleans up the staging data set if errors occur during the prepare phase of import processing. Requires APAR PI82099 (PTF UI49647).
- REFRESHPSB, which activates PSBs that were previously flagged as refresh pending by the UPDATEPSB option. Requires APAR PI51217 (PTF UI41861).
- UPDATEPSB, which activates modified PSBs in a subset of sharing IMS systems in an IMSplex and flags the PSBs as refresh pending in the rest of the systems in the IMSplex. Requires APAR PI51217 (PTF UI41861).

For a list of the commands that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

Impact to exit routines

Existing IMS Catalog Definition exit routines (DFS3CDX0) must be updated when the IMS management of ACBs is enabled so that the specification for ACBMGMT in the exit routine matches the specification for ACBMGMT in the CATALOG section of the DFSDFxxx PROCLIB member.

IMS 14 updates the sample Catalog Definition exit routine with a flag that indicates whether the IMS management of ACBs is enabled.

Impact to utilities

The IMS management of ACBs enhancement introduces some new IMS utilities and enhances many others.

The IMS Catalog Populate utility (DFS3PU00) or the ACB Generation and Populate utility (DFS3UACB) is required to set up the IMS management of ACBs for the first time. Running one or the other of these utilities is also required to make utility-generated ACBs available to the online IMS systems that manage ACBs. To support these tasks, the utilities support a new MANAGEDACBS control statement on the SYSINP DD statement. The MANAGEDACBS control statement has the following purposes:

MANAGEDACBS=SETUP

Required to set up the IMS catalog to manage ACBs. Creates the IMS directory system data sets that are required by IMS to manage ACBs. Any existing instances of the IMS directory data sets are replaced.

MANAGEDACBS=STAGE

Saves the ACBs from the input ACB libraries to a staging data set. ACBs

saved in the staging data set are not activated until you add them to the IMS system by issuing the IMPORT DEFN SOURCE(CATALOG) command.

After APAR PI76125 (PTF UI46904) is installed, the IMS Catalog Populate utility no longer scratches and re-creates the staging data set when MANAGEDACBS=STAGE is specified. This change in behavior preserves any existing resources that might be in the staging data set before the utility runs. After the APAR is installed, if you need the utility to scratch and re-create the staging data set, specify the new DELETE parameter. For example, MANAGEDACBS=(STAGE,DELETE).

MANAGEDACBS=UPDATE

Updates existing IMS directory system data sets in exclusive mode.

With APAR PI46909 (PTF UI35217), the IMS Catalog Populate utility is enhanced to support adding and activating dynamic option (DOPT) PSBs without taking IMS or the IMS catalog offline or requiring a subsequent step to activate the DOPT PSBs. You can also perform both of these actions with the ACB Generation and Catalog Population utility (DFS3UACB).

With APAR PI58722 (PTF UI40956), the IMS Catalog Copy utility (DFS3CCE0, DFS3CCI0) is enhanced to support the IMS catalog directory data set. You can copy the active ACBs in an IMS-managed ACB environment by specifying the CCUDIREX DD statement in the JCL of the IMS Catalog Export utility (DFS3CCE0). The CCUDIREX DD statement must point to the data set that will contain the information about the active ACBs that is required to populate a target IMS system with the ACBs. To install the copied ACBs into the target IMS system, specify the CCUDIRIM DD statement in the JCL of the IMS Catalog Import utility. The CCUDIRIM DD statement must point to the data set that contains the information about the active ACBs that are from the originating IMS-managed ACB environment.

Some IMS utilities that run in a ULU or a UDR region still require access to the DBD and PSB libraries. For these utilities, after you enable the IMS management of ACBs function, retain the DBDLIB and PSBLIB data sets. These utilities are listed in the preceding "Restrictions" on page 130 section.

When IMS manages ACBs, after the required service is installed, the following database batch utilities do not require the DBD and PSB libraries:

- DEDB Initialization utility (DBFUMIN0). Requires APAR PI55596 (PTF UI39974).
- Database Batch Backout utility (DFSBB000). Requires APAR PI63855 (PTF UI38936).
- Database Image Copy 2 utility (DFSUDMT0). Requires APAR PI67618 (PTF UI40743).
- Database Image Copy utility (DFSUDMP0). Requires APAR PI61703 (PTF UI39976).
- Database Recovery utility (DFSURDB0). Requires APAR PI66598 (PTF UI40944).
- HD Reorganization Reload utility (DFSURGL0). Requires APAR PI46912 (PTF UI39920).
- HD Reorganization Unload utility (DFSURGU0). Requires APAR PI46912 (PTF UI39920).

For tools, products, or processes that require DBD and PSB libraries or the macro source statements, IMS 14 introduces the Database Library Builder utility

(DFS3LU00). The Database Library Builder utility creates DBD and PSB source statements, control blocks, or both by reading the DBD and PSB records in the IMS catalog.

When the IMS management of ACBs function is enabled, the IMS Catalog Record Purge utility deletes from the IMS directory the DBDs and PSBs that are deleted from the IMS catalog. ACBs that are deleted from the IMS directory are also deleted from the directory staging data set.

The IMS Catalog Directory Recovery utility (DFS3RU00) can be used to rebuild the IMS directory data sets and recover the ACBs of IMS databases and programs to the directory by reading active resources from the IMS catalog, or clean up the IMS directory data sets.

PSB activation control

In IMS 14, when the IMS management of ACBs is enabled, you can use the `OPTION(UPDATEPSB)` keyword of the `IMPORT DEFN` command to control the activation of program views (PSBs) in an IMSplex. The PSBs can then be activated in one of two ways; in the IMS systems that are specified on the command or in the IMS systems not specified on the command.

IMS systems that are not specified on the command, continue to use the old PSB definitions, if any, until a second `IMPORT DEFN` command is issued for the IMS systems with the `OPTION(REFRESHPSB)` option specified.

Requirements

APAR PI51217 enhances the `IMPORT DEFN SOURCE(CATALOG)` command by adding an `OPTION()` and `FOR()` keyword to target the import of a subset of systems within an IMSplex. This APAR must be applied on each system in the IMSplex in order to use this enhancement.

Changes to commands

For a list of the commands that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

IMS catalog activity record

In IMS 14, the IMS catalog activity record enhancement logs certain types of activities that affect the IMS catalog as type-29, subtype-3 records in z/OS System Management Facility (SMF).

System programmers and administrators can use the activity record in SMF to review IMS catalog activities.

The type-29 record is exclusive to IMS, and its subtype-3 record is exclusive to the IMS catalog activity record feature.

Migration considerations

This enhancement delivers the new SMF record type-29, subtype-3.

Products or programs that read SMF records should be tolerant of new record types and subtypes.

Prerequisites

To use the IMS catalog activity record, the following SMF prerequisites must be met:

- SMF must be installed in z/OS.
- SMF must be configured to accept type-29, subtype-3 records in the settings for SMF parameter SYS1.PARMLIB(SMFPRMxx) in z/OS.

Impact to troubleshooting for IMS

New abend 3001 is added to IMS 14 for this enhancement.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Related reference:

 [z/OS: SMFPRMxx SMF parameters](#)

Fast Path control block changes

In IMS 14, certain Fast Path control blocks changed. These changes are not related to any specific enhancement.

Load module DBFFATC0

Load module DBFFATC0, which resided in PVT storage, was split into three modules: DBFFAT0C, DBFDDBDP0, and DBFDBTC0.

In prior versions, DBFFATC0 was 42C0 bytes. DBFFATC0 is now only 1D88 bytes and was moved from PVT to ECSA storage.

DBFDDBDP0 is 16A8 bytes and remains in PVT storage.

DBFDBTC0 is 430 bytes and is in EPVT storage.

Control block length changes

The lengths of the following control blocks changed in IMS 14.

- DBFLSRT
- DBFEPST
- DBFESCD
- DBFUHSW
- DBFDMCB (upon migrating to IMS 14, an ACB generation is required to account for his change)
- DBFDMAC (upon migrating to IMS 14, an ACB generation is required to account for his change)

Changes to storage blocks that are used to track IMS storage

In IMS 14, the following storage tracking blocks (CDEs) moved from LSQA (24-bit) to 64-bit storage (STEs):

Parts

DBFDRIS0 DBFICI10 DBFINI10 DBFINI2M DBFINI20 DBFINI21 DBFINI29 DBFINI30
DBFMLOP3 DBFOLC01 DBFPALC0 DBFVTSK0 DBFVXAS0 DBFVXPL1

ITASKS

HRTREPST EMREPST LIRLEPST NALCEPST CST0EPST CSTSEPST VXCSEPST PVTSEPST
CSNIEPST HCASEPST HXCSEPST HCTREPST VOCIEPST HMISEPST MLBIEPST MLBREPST
MLBDEPST CST2EPST MER0PST VXGREPST ST02EPST FP64EPST FPBKEPST FPD1EPST
FPD5EPST FPD6EPST

Others

FPIEEQE MSDBTABL FPITT FPMMPR DBFMSDSN DBFDMB DBFMSDWA DBFMBR AREALIST
DBFFPSRB GENCBST TOPUDMHR VSOPECB
DDIRNAME AREANAME ACBAIO ACBBIO AREADLET TEMPLIST

FDBR in-doubt thread support for ESAF

The External Subsystem Attach Facility (ESAF) supports the resolution of in-doubt work on an external subsystem by a Fast Database Recovery (FDBR) region.

IMS now supplies a sample ESAF In-Doubt Notification exit routine (DFSFDN0) that issues messages with the following information about in-doubt units of work on external subsystems:

- Subsystem name
- Subsystem type
- Resolution action
- Origin application schedule number (OASN)

When FDBR begins processing the in-doubt IMS work after a failure, it calls the DFSFDN0 exit routine to indicate whether a given UOW was committed or aborted. User-provided automated recovery programs can use the information from the sample DFSFDN0 exit routine that is provided with IMS to commit or abort the unit of work on the external subsystem when the external subsystem becomes available again. FDBR calls the exit routine for each in-doubt unit of work it resolves.

Migration considerations

The method that you use to define the DFSFDN0 exit routine to IMS has changed. See “Impact to installing and defining IMS.”

Restrictions

This enhancement does not affect RRS-managed units of recovery.

Impact to installing and defining IMS

To use this enhancement, you must link the ESAF In-Doubt Notification exit routine named DFSFDN0 into the IMS.SDFSRESL concatenation of the FDBR procedure. You can use either the IMS-supplied exit routine or your own exit routine module, but the module must be named DFSFDN0.

In a member specified with the SSM= parameter for the IMS startup procedure, you can specify the type of an external subsystem with the SST= parameter. The supported ESS types you can specify on the SST= parameter are:

DB2

Db2 for z/OS

MQ IBM MQ

WOLA

WebSphere Optimized Local Adapter

SST= is a keyword-only parameter. It cannot be specified as a positional parameter.

Impact to administering IMS

This enhancement simplifies administration for IMS environments that include external subsystems such as Db2 for z/OS databases or IBM MQ systems. You can use the DFSFIDN0 exit routine to resolve work on an external subsystem after an IMS failure.

Impact to troubleshooting for IMS

This enhancement adds new IMS messages that are issued by the supported, IMS-supplied DFSFIDN0 exit routine.

For a list of the messages and codes that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Impact to commands

For a list of the commands that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

Impact to exit routines

A new IMS-supplied sample exit routine is included for the ESAF In-Doubt Notification exit routine (DFSFIDN0). You can use this sample exit routine as provided, modify it for your environment, or replace it with a user-supplied exit routine.

GSAM metadata enhancement

In IMS 14, you can optionally define a single virtual segment with multiple non-searchable fields in a GSAM DBD to add field metadata from COBOL copybooks to the GSAM DBD record in the IMS catalog.

IMS ignores the virtual segment and the non-searchable fields, but application programs and products, such as IMS Enterprise Suite Explorer for Development, can retrieve the metadata from the IMS catalog when they access the GSAM database through the IMS Universal drivers.

Changes to administering IMS

Database administrators can define a virtual segment in the definition of a GSAM database to map fields in the DBD that were previously mapped only in COBOL

copybooks. The fields in a virtual segment are unknown to IMS and cannot be specified in an SSA.

Changes to programming for IMS

The field metadata that is defined the DBD is stored in the IMS catalog for use by application programs and products that access the GSAM database through the IMS Universal drivers.

Changes to utilities

The DBD Generation utility is updated in IMS 14 to support the specification of a virtual segment and fields in the DBD of a GSAM database.

The fields that are defined in a GSAM database must be named by the EXTERNALNAME parameter. GSAM field definitions do not support the NAME parameter.

HALDB support for 8-GB OSAM data sets enhancement

In IMS 14, IMS HALDB databases (partitioned, full-function databases) that are not reorganized by the HALDB Online Reorganization function support 8-GB OSAM data sets.

Unlike VSAM database data sets, which can have a maximum size of 4 gigabytes, OSAM database data sets can have a maximum size of 8 gigabytes. However, before IMS 14, if OSAM was used for HALDB databases, the maximum data set capacity was limited to 4 gigabytes. Database administrators can now increase the maximum size of HALDB OSAM database data sets to 8 gigabytes if DBRC is being used. When an HALDB OSAM database data set is increased to 8 gigabytes, the HALDB Online Reorganization (OLR) function cannot be run against that database. To continue to be able to use OLR for HALDB OSAM databases, the maximum size of the OSAM database data sets must be kept to 4 gigabytes.

Taking advantage of HALDB support for 8-GB OSAM data sets means increased database scalability. Defining larger database data sets increases overall database capacity, reduces the number of database data sets that need to be defined, and reduces the risk of exceeding the maximum number of database data sets that can be open at the same time.

The DBRC commands INIT.DB and CHANGE.DB are enhanced with the OSAM8G and NOOSAM8G keywords. OSAM8G specifies that the maximum data set size of OSAM database data sets in a PHDAM or PHIDAM database is 8 gigabytes. NOOSAM8G specifies that the maximum data set size is 4 gigabytes.

Migration considerations

Unless you have IMS Version 13 APAR PI23918 applied to your IMS Version 13 systems, all IMS systems must be at the IMS 14 level and the MINVERS value in the RECON data set must be 14.1.

If you are using HALDB support for 8-GB OSAM data sets on IMS Version 13, IMS systems in a data sharing environment can be at the IMS Version 13 level. The appropriate maintenance (SPE APAR PI23918) must be applied to all the IMS

Version 13 resident libraries (RESLIBS) that access the RECON data set. The MINVERS value in the RECON data set must be 14.1 or it must be 13.1 with the CDSLID value set to 2.

The default value for MINVERS in IMS 14 is 12.1.

Coexistence considerations

The coexistence considerations for this enhancement are the same as the migration considerations.

Restrictions

The HALDB Online Reorganization function does not support HALDB 8-GB OSAM database data sets. HALDB databases that use 8-GB OSAM data sets can be reorganized by using the offline utilities HD Reorganization Unload utility (DFSURGU0) and HD Reorganization Reload utility (DFSURGL0).

Impact to administering IMS

Database administrators have more flexibility in implementing and managing their HALDB databases. Large HALDB databases are more easily accommodated by using 8-GB OSAM data sets instead of 4-GB data sets. The maximum OSAM data set capacity for new and existing HALDB databases is recorded in the RECON data set.

When the maximum data capacity of OSAM data sets is reduced and any data set in a partition contains more than 4 GB of data, the partition might need to be split.

Impact to troubleshooting for IMS

If the INITIATE OLREORG command is issued for a HALDB database that supports 8-GB OSAM data sets to start the HALDB online reorganization process, the command fails with completion code 1EF.

For a list of the messages and codes that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Impact to commands

In addition to changing the DBRC INIT.DB and CHANGE.DB commands, this enhancement changes the LIST output for OSAM HALDBs to indicate the maximum data set capacity. The master database listing includes either DSORG=OSAM to indicate 4-GB data sets or DSORG=OSAM8G to indicate 8-GB data sets. The partition database listing does not include these fields.

For a list of the commands that are changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

ODBM accounting enhancement

The IMS 14 ODBM accounting enhancement provides an option to record information about the processing of database access requests by ODBM. It provides a log of all work handled by the ODBM address space so that accounting can issue charge back to its users.

ODBM leverages the z/OS System Management Facility (SMF) to perform logging of ODBM accounting information, such as CPU usage, and its retrieval. The logging of the ODBM address space is activated when the optional parameter LOGOPT=ACCOUNTING is specified in the ODBM initialization member, CSLDIxxx.

Performance considerations

When the ODBM audit log is enabled, there is an increased code path for logging activities.

Log record changes

The log records are SMF-compliant. The log record layout is defined in SMF log records for ODBM accounting (System Administration).

Concurrent active thread enhancement

In IMS 14, the concurrent active thread enhancement provides a new Open Database Manager (ODBM) configuration parameter **MINTHRDS** that specifies the minimum number of concurrent active threads for an individual IMS data store.

A new parameter **MINTHRDS** is added to the CSLDCxxx member of the IMS PROCLIB data set, QUERY ODBM TYPE(DATASTORE) command, and QUERY ODBM TYPE(CONFIG) command. In the CSLDCxxx member of the IMS PROCLIB data set, this parameter specifies the minimum number of concurrent active threads for an individual IMS data store. In the QUERY ODBM TYPE(DATASTORE) command and QUERY ODBM TYPE(CONFIG) command, this parameter returns the minimum number of concurrent active threads to an individual IMS data store.

OSAM 24-bit storage constraint relief enhancement

In IMS 14, OSAM database data sets use less 24-bit storage.

As the number of open database data sets that use the OSAM access method increases, so does the use of 24-bit storage. Consequently, a lack of 24-bit storage can limit the number of OSAM database data sets that can be opened.

OSAM database data sets use a DFSMSdfp control block called a *data extent block* (DEB). The DEB is stored in 24-bit storage.

Prior to IMS 14, IMS stored extra information in the DEB that is unique to the OSAM access method.

IMS 14 reduces the amount of 24-bit storage that is used by OSAM database data sets by storing the extra OSAM information in 31-bit storage, thereby reducing the size of the DEBs and the amount of 24-bit storage that is used by OSAM database

data sets. This provides some 24-bit storage relief to users of OSAM database data sets, and potentially allows more OSAM database data sets to be opened in IMS.

Impact to troubleshooting for IMS

The documentation for abend U0403 was updated to reflect the change to OSAM information in the DEB.

For a list of the messages and codes that are changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

SQL support enhancements

SQL support for COBOL application programs was added in IMS Version 13. IMS 14 further enhances the functionality by supporting grouping (GROUP BY) and providing aggregate functions for SELECT statements.

IMS Database Manager can process SQL calls in the native host environment, enabling SQL programmers to access IMS databases in a similar manner as they use for COBOL applications in relational databases. In this release, aggregate functions, AVG, COUNT, MAX, MIN, and SUM, are added.

The SQL support also provides support for the IBM IMS Data Provider for Microsoft .NET. The added aggregate functions and all SQL statements and data types supported for COBOL application programs also apply to .NET applications.

Requirements

This enhancement requires COBOL Version 5 with IMS coprocessor support. With COBOL Version 5, all load modules must reside within a partitioned data set extended (PDSE).

Restrictions

The following restrictions are in place for this function:

- Only a subset of SQL 2008 is supported.
- The SQL syntax is limited to what IMS DBMS can process natively.
- Only SELECT, INSERT, UPDATE, DELETE, predicate, aggregation, and ordering support is provided.
- The AS clause is not supported.
- XML DB support is not provided.
- Static SQL is not supported.

Programming for IMS

New programming information is added on COBOL support for SQL and the existing SQL information regarding the IMS Universal JDBC driver has been moved to form a new section of information about application programming for SQL.

Impact to troubleshooting for IMS

New SQL error codes have been added for SQL support for COBOL.

For a list of the messages and codes that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Chapter 11. IMS Transaction Manager enhancements

The enhancements to IMS Transaction Manager (IMS TM) in IMS 14 include enhancements to Multiple Systems Coupling (MSC), Open Transaction Manager Access (OTMA), IMS Connect, synchronous callout, and more.

The Chapter 12, “IMS system enhancements,” on page 183 in IMS 14 might also impact IMS TM.

APPC flood control enhancement

In IMS 14, the optional APPC flood control enhancement helps avoid IMS abends caused by the exhaustion of private 31-bit storage by moving APPC z/OS requests that are waiting to be processed by IMS into 64-bit storage after the number of requests reaches a certain threshold.

Without APPC/IMS flood control enabled, if IMS cannot process the messages received from APPC quickly enough or at all, your IMS system can run out of private storage, which can lead to an IMS abend.

The APPC flood control enhancement is delivered with APAR/PTFs PI68466/UI47190/UI47191.

Migration considerations

The APPC/IMS flood control option is enabled by default in IMS 14. If you need to disable this function, specify `APPCMAXC=0` in `DFSDCxxx` member of the IMS PROCLB data set.

The output of the `/DISPLAY ACT` command is modified to include information about the current maximum number of active APPC conversations.

Changes to installing and defining IMS

By default, IMS initiates flood control measures when the number of active APPC conversations reaches the default maximum of 5000. IMS stops all input from APPC z/OS when the number of APPC messages waiting in 64-bit storage to be accepted by IMS for processing reaches the default maximum of 1000000.

You can modify these maximums or disable APPC flood control by specifying the `APPCMAXC=` parameter in the `DFSDCxxx` member in the IMS PROCLIB data set.

You can also define a threshold for stopping all APPC/IMS input by specifying the `64_bit_max` value in the `APPCMAXC=(31_bit_max,64_bit_max)` startup parameter. After the threshold is reached, all input to APPC/IMS is stopped until 70% of the waiting APPC/IMS request messages have been processed.

Changes to troubleshooting for IMS

IMS 14 includes several new messages related to APPC flood control.

For a list of the messages and codes that are new for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Changes to commands

The information returned by the /DISPLAY ACT command is expanded to include a MAXC= output field that displays the maximum number of APPC conversations that can be active before IMS starts queueing incoming APPC requests in 64-bit storage. If the displayed value is 0, APPC flood control is disabled.

The /PURGE command is enhanced to reject any APPC requests that are queued in 64-bit storage with the sense code TP_Not_Available_No_Retry.

For a list of the commands that are changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

IMS Connect support for cascading global transactions to IMS TM on different z/OS images

In IMS 14, IMS Connect and IMS TM are enhanced to support two-phase commit processing for cascaded global z/OS Resource Recovery Services transactions that are received from the IMS TM resource adapter via a TCP/IP connection when IMS Connect and IMS TM are located on different z/OS images.

IMS Connect support for cascading global RRS transactions to IMS TM data stores that are on different LPARs increases availability by allowing two-phase commit transactions to be routed to any available IMS. This allows for better workload balancing, and simplifies network configuration because a dedicated IMS and IMS Connect pair is no longer required to support two-phase commit processing.

The support for cascading global transactions across LPARs for is best used as failover protection that is used only when the normal processing of global transactions by an IMS system that is on the same LPAR as IMS Connect is interrupted. Limiting the processing of global transaction to an IMS system that is on the same LPAR as IMS Connect is more efficient and requires fewer calls to RRS.

Migration considerations

Support for cascading global transactions across z/OS images is disabled by default. Do not enable the support until both IMS Connect and IMS are migrated to IMS 14.

If IMS TM resource adapter and IMS Connect are currently connected by using the local option (the z/OS Program Call interface), you must reconfigure IMS TM resource adapter and IMS Connect to use a TCP/IP connection before you can enable support for cascading transactions from IMS Connect to IMS TM systems on a different z/OS LPAR.

The `CASCADE=` keyword is not recognized in an IMS Version 13 IMS Connect configuration and, if specified, prevents IMS Connect from starting. Therefore, avoid specifying the `CASCADE=` keyword until migration to IMS 14 is complete and there is little chance of falling back to Version 13 using the same IMS Connect configuration.

When you are ready to enable support for cascaded transactions, you can use the `UPDATE IMSCON` command to enable cascaded transaction without stopping IMS Connect; however, you must restart data store connections to apply the new `CASCADE=` option to the data store connection.

To save changes to the `CASCADE=` keyword that are made by the `UPDATE IMSCON` command across restarts of IMS Connect, you must specify the `CASCADE=` option in the IMS Connect `HWSCFGxx` configuration member.

Coexistence considerations

Support for cascading global transactions from the IMS TM resource adapter to an IMS TM system on a different LPAR is supported only after both IMS Connect and the IMS control region are migrated to IMS 14.

If IMS Connect receives a global transaction from IMS TM resource adapter when IMS Connect and IMS are not on the same LPAR and either IMS Connect, IMS, or both, are not at IMS 14, the transaction is rejected with a NAK message that includes a sense code of X'2F'.

Log record changes

This enhancement doesn't change the format of any log records, however in IMS 14 the information in type 01 log records might have slightly different implications depending on whether support for cascading global transactions to different LPARs is enabled.

Requirements

Support for cascading global transaction from IMS Connect to an IMS TM system on another z/OS image requires the following software:

- IMS TM resource adapter
- z/OS Resource Recovery Services (RRS) running on each LPAR with a coupling facility for multiple LPAR access
- IMS and IMS Connect must both be IMS 14

This enhancement has the same hardware requirements as the base IMS 14 product.

Restrictions

The local option, which uses the z/OS Program Call function and an IMS Connect port defined as `LOCAL`, does not support cascading transactions. Support for cascading transactions from IMS Connect to an IMS TM system on another z/OS LPAR is available only when IMS TM resource adapter connects to IMS Connect via TCP/IP.

Changes to installing and defining IMS

Support for cascading transactions across z/OS images is enabled in IMS Connect by specifying `CASCADE=Y` in either the IMS Connect system definition or the definition of individual data store connections.

The `CASCADE` parameter can be specified in the IMS Connect `HWS` and `DATASTORE` configuration statements and IMS type-2 `CREATE` and `UPDATE` commands for IMS Connect system and data store definitions.

Optionally, IMS Connect user message exit routines can be modified to check the `XIBDS` exit interface block to determine if IMS Connect is on the same LPAR as IMS and if support for cascading global transactions to different LPARs is enabled.

Changes to troubleshooting for IMS

Two new reason codes, `X'01'` and `X'02'` can be returned with the `X'2F'` sense code in `NAK` messages for errors that occur during the processing of global `RRS` transactions.

You can display an entire unit of recovery (UOR) family by using `TSO SPOC`.

For a list of the messages and codes that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Changes to commands

In IMS 14, you can use IMS type-2 commands to set or change the `CASCADE=` parameter in the IMS Connect configuration and data store definitions.

You can also use the type-2 `QUERY IMSCON` command with either `TYPE(DATASTORE)` or `TYPE(CONFIG)` specified to display the current `CASCADE` value, the status of the support for particular data store connections, and which LPARs IMS and IMS Connect are running on.

The IMS Connect `WTOR` and z/OS `MODIFY` commands display only the current `CASCADE` value.

For a list of the commands that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

Changes to exit routines

The `XIBDS_FLAG` field in the IMS Connect `XIBDS` exit interface block can contain two new flags that indicate whether an IMS data store is on a different LPAR than IMS Connect and whether support for cascading global transactions to different LPARs is enabled.

The IMS TM Resource Adapter user message exit routine (`HWSJAVA0`), as well as other IMS Connect user message exit routines, can be modified to check the

XIBDS_FLAG field of the XIBDS exit interface block to determine if IMS Connect is on a different LPAR than IMS (X'40') and if support for cascading global transactions to different LPARs is enabled (X'20').

ISC VTAM enhancement for ERP messages

The IMS 14 ISC VTAM enhancement for ERP messages provides a new keyword option for specifying how IMS handles error recovery procedure (ERP) messages when they are received from the other session half via an ISC session that is on a VTAM connection.

An ERP message is an ERP type-7 function management header (FMH7) followed by an error message. The error message is an error message issued by the subsystem that is sending the ERP message. For example, if an IMS subsystem is sending the ERP message, the ERP message includes the FMH7 header followed by a DFSnnnnn error message. If a CICS subsystem is sending the ERP message, the error message is a DFHnnnnn message.

IMS receives an ERP message when, after IMS sends a message to another subsystem (IMS, CICS, or other subsystem type), the subsystem encounters an error processing the message, and sends back an ERP message.

Prior to IMS 14, when an ERP message was received by an IMS system via an ISC VTAM session, IMS kept the original message on the queue, closed the ISC session, and passed the ERP message as a DFS2073I message to the master terminal operator (MTO).

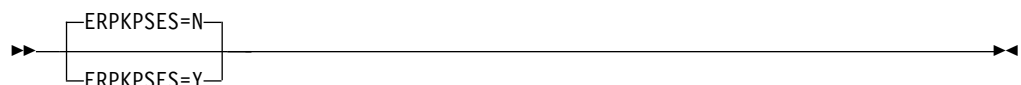
IMS 14 introduces a new ERPKPSES parameter on the DFSDCxxx PROCLIB member, which provides greater control over what IMS does with ISC sessions and sense code X'0846xxxx' ERP messages when an ERP message is received from another subsystem on a VTAM connection. This control can improve the availability of ISC sessions on VTAM connections and can aid the diagnosis of ISC message errors by routing the error messages to wherever it is most convenient for IMS operators to receive them.

When ERPKPSES=Y is specified, the original message is initiated from the secondary side of the session with COMPT1|2|3|4=SINGLE2 | MULT1 | MULT2 specified, and an ERP message is received on an ISC VTAM session, IMS keeps the ISC session active, dequeues the original message, and passes the ERP message to the original inputting terminal. This behavior is similar to how IMS handles ERP messages received for message switches.

This enhancement allows customers of ISC VTAM to be able to handle FMH7 ERP messages more gracefully.

Impact to installing and defining IMS

A new parameter, ERPKPSES, is added to the DFSDCxxx PROCLIB member with the following syntax:



ERPKPSES=

Specifies whether or not IMS keeps the ISC VTAM session open after an error recovery procedure (ERP) message is received from the partner subsystem with sender ERP sense code X'08460000'. Depending on the bracket protocols that were used in the ISC output message in error, ERPKPSES also determines whether the ERP message is passed to the master terminal operator (MTO) or to the original inputting terminal.

N Upon receiving an ERP message, IMS terminates the ISC VTAM session, keeps the original message on the queue, and passes the ERP message to the master terminal operator (MTO). ERPKPSES=N is the default.

Y Upon receiving an ERP message, IMS keeps the ISC VTAM session open, dequeues the original message, and passes the ERP message to either the MTO or the original input terminal, depending on the bracket protocols that were used in the original message.

Impact to administering IMS

IMS operators and system administrators should be aware that when ERPKPSES=Y is specified in the DFSDCxxx PROCLIB member and a X'0846xxxx' selective receiver ERP message is returned, in certain circumstances the original input message will be discarded, the ISC session will remain active, and the ERP message will be returned to the inputting terminal instead of the MTO.

Impact to troubleshooting for IMS

When troubleshooting ISC VTAM communication errors, troubleshooters should be aware that when ERPKPSES=Y is specified and the other subsystem encounters an error processing a message that was initiated from the secondary side of the session with COMPT n =SINGLE2 | MULT1 | MULT2 specified, upon receiving the X'0846xxxx' ERP messages, IMS keeps the ISC session active, dequeues the original message, and passes the ERP message to the original inputting terminal. This behavior is similar to how IMS handles ERP messages received for message switches.

The documentation for message DFS2073I is updated to reflect the optional change to IMS processing for ERP messages.

For a list of the messages and codes that are changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Java environment dynamic scripting enhancement

In IMS 14, you can improve the security, flexibility, and control of your Java environment settings in IMS by moving the configuration parameters from the DFSJVMEV and DFSJVMMS members of the PROCLIB data set into several data sets that are referenced by the new //STDENV DD statement in your dependent region JCL.

The IMS Java environment dynamic scripting enhancement provides the possibility to dynamically configure the Java environment variables and options by providing

partitioned data sets (PDS) or physical sequential (PS) data sets under the new `//STDENV DD JCL` statement. The `//STDENV DD` statement points to files that contain the Java environment definitions in a Shell syntax. Files are read in order. The parameters that are referenced by the `//STDENV DD` statement are used as input to the `BPXBATCH` utility for launching the JVM environment. This feature is supported for all the IMS dependent regions that run Java virtual machine.

You can organize the parameters that define your Java environment into several PDS or PS data sets based on the aspect of the Java environment that they define and the roles that are allowed to modify them. Multiple scripts can then be concatenated into one `//STDENV DD` statement. The shell variables that are provided in the scripts can be further used within the shell.

The `//STDENV DD` statement can be used to replace the `DFSJVMEV` and `DFSJVMMS` members of the IMS `PROCLIB` data set that were used to define the Java environment variables. When the `//STDENV DD` statement is used, both the `DFSJVMEV` and `DFSJVMMS` members are ignored, so you must move all of the contents of both members into the shell script.

When the `//STDENV DD` statement is used, you can configure the code page format for `stdout` and `stderr` by setting the `JZOS_ENABLE_OUTPUT_TRANSCODING` and `JZOS_OUTPUT_ENCODING` variables under the statement.

Using the dynamic scripts allows the system administrator to separate the responsibilities between the system programmers and application developers that use PDS or PS data sets. The system programmer can restrict the update access on some data sets to the application developer and provide the update access on the files that will be required by the developers. It provides greater security by allowing the IMS system programmer to prevent application developers that use the scripts from overriding the environment variable definitions.

Migration considerations

If the `//STDENV DD` statement is present, both the `DFSJVMEV` and `DFSJVMMS` `PROCLIB` members are ignored.

Before you use the `//STDENV DD` statement, ensure that you move the parameter configurations that were specified in the `DFSJVMEV` and `DFSJVMMS` members of the `IMS.PROCLIB` data set to the shell scripts that are referenced by the `DD` statement.

Requirements

APAR [PI68127](#) is required to enable the Java environment dynamic scripting enhancement.

APAR [PI82480](#) is required to enable the encoding support for Java environment dynamic scripting and to increase the classpath length limit to 150K.

Restrictions

- The shell scripts support only the export statements and variable substitution in the shell syntax.
- The shell scripts cannot issue the `exit` shell command.
- The input to the `//STDENV DD` statement cannot contain line numbers.

- The shell scripts run under a non-login shell, so the /etc/profile script and the user's default .profile login script are not executed automatically. If they are needed, you can explicitly execute these scripts in the shell.
- The shell script cannot have any limitations on the logical record length (LRECL).
- The length of the classpath provided under the //STDENV DD statement is limited to 16K. With APAR PI82480, the classpath length limit is extended to 150K.

Changes to installing and defining IMS

When the //STDENV DD statement is present, the IMS procedure parameter **ENVIRON=** that specifies the DFSJVMMS member and **JVMOPMAS=** that specifies the DFSJVMMS member are both ignored by the IMS system. Only the parameters that are referenced by the //STDENV DD statement are used.

The //STDENV DD statement follows the regular JCL rules. It can be a concatenation of several DSNs, instream, and DATA statements.

You can set the variable **JZOS_OUTPUT_ENCODING** under the //STDENV DD statement to specify the code page that is used to convert the raw bytes for stdout and stderr. If no code page is provided in JZOS_OUTPUT_ENCODING, the default code page in the current locale is used for stdout and stderr. You can use the variable **JZOS_ENABLE_OUTPUT_TRANSCODING** to enable or disable the code page that is specified in JZOS_OUTPUT_ENCODING. The encoding option is enabled by default. If **JZOS_ENALBE_OUTPUT_TRANSCODING** is set to false, then the code page specified in JZOS_OUTPUT_ENCODING is ignored and raw bytes are written to stdout and stderr.

Changes to administering IMS

You can use the //STDENV DD statement instead of the DFSJVMEV and DFSJVMMS members to provide java environment variables and options to load the JVM. The method of using the DFSJVMEV and DFSJVMMS members to define JVM is still valid when the //STDENV DD statement is not used.

By using the Java environment dynamic scripting enhancement, a system programmer can dynamically manage the permissions of IMS data sets that an application developer has access to.

Changes to commands

When you issue the UPDATE PGM START(REFRESH) command on JVM regions, if an //STDENV DD statement is present, the JVM is reloaded based on the configurations in the shell scripts that are referenced by the DD statement, instead of refreshing the DFSJVMMS, DFSJVMEV, and DFSJVMAP members of the PROCLIB data set.

JVM 64-bit support in Java dependent regions

In IMS 14, a Java Virtual Machine (JVM) in either a Java message processing (JMP) or a Java batch processing (JBP) dependent region can use 64-bit storage.

You can use the new JVM= parameter to switch between the 64-bit and 31-bit addressing mode. The DFSJBP and DFSJMP procedures are changed to support the JVM= parameter.

Migration considerations

By default, Java dependent regions load and run with a 31-bit Java virtual machine (JVM). You can modify the JVM addressing mode to 64-bit by specifying the JVM=64 parameter in the DFSJBP and DFSJMP procedure EXEC statement. The new JVM= parameter is valid only for the JBP and JMP regions.

When you specify the JVM= parameter on the EXEC PGM=DFSRRC00 card for the JMP region, you must specify all positional parameters, PRLD=, SSM=, PARDLI=, MINTHRD= and MAXTHRD=, before the JVM= parameter. MAXTHRD= and MINTHRD= are two positional parameters that are hidden when the JVM= parameter is not provided.

By specifying JVM=64, Language Environment, the infrastructure that supports a 64-bit JVM, is also changed to the 64-bit addressing mode.

Restrictions

In the default 31-bit addressing mode, the IMS system performs an implicit check for Java exceptions before executing a SYNC call. When JVM=64 is used, the IMS system does not check for any Java exceptions before executing the call.

In the default 31-bit addressing mode, the Language Environment error recovery routines can catch application errors and issue return codes. When JVM=64 is used, Language Environment does not support error recovery for the IMS system. Some existing error scenarios might abend the application.

Language Environment supports C, C++, and assembly language interoperability in a 64-bit addressing mode, but does not support COBOL and PL/I interoperability in a 64-bit addressing mode. Do not switch to JVM=64 if your Java application invokes either COBOL or PL/I. If the regions are switched to use JVM=64 inadvertently, and incompatible interoperable applications are running, the application might receive system or user abends.

Changes to installing and defining IMS

The JVM= parameter is added to the DFSRRC00 EXEC card. The DFSJBP and DFSJMP procedures are enhanced to support the JVM= parameter.

JVM=64 introduces a new AMODE 64 load module, DFSJVM64. The DFSJVM64 module is included in the SDSFJLIB data set.

Changes to troubleshooting for IMS

In the 64-bit addressing mode, diagnostic dumps that involve the Java dependent regions are different than in the 31-bit addressing mode. Language Environment provides IPCS commands to implicitly support 64-bit address spaces and the Language Environment control block structures and linkages are changed.

For a list of the messages and codes that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

| For a complete list of all of the new, changed, and deleted messages, and abend
| codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page
| 95.

JVM usage statistics

In IMS 14, you can enable IMS to capture the usage statistics that are generated by a persistent Java virtual machine (JVM) running in an IMS dependent region.

You can capture the following usage statistics:

- The elapsed running time of the JVM
- The garbage collection policy of the JVM
- Enhanced memory statistics, including the number of garbage collections, garbage collection elapsed time, and the amount of heap memory used
- The number of threads processed by the JVM

The JVM usage statistics are written by the IMS Java dependent region resource adapter to type-29, subtype-2 records in z/OS System Management Facility (SMF).

System programmers and administrators can enable IMS to capture the usage statistics for JVMs that run in IMS dependent regions at specified time intervals and then view the statistics in SMF.

Migration considerations

This enhancement delivers the new SMF record type-29, subtype-2.

Products or programs that read SMF records should be tolerant of new record types and subtypes.

Prerequisites

Before you enable IMS to capture JVM usage statistics, ensure that all of the following prerequisites are met:

- Both IMS APAR/PTF PI51753/UI34117, which is an SMP/E service, and IMS APAR/PTF PI30848/UI34793 are installed.
- The latest version of the IMS Universal JDBC driver is downloaded and all of your existing IMS Universal JDBC driver files are replaced with the files of the latest version. In particular, you must use the latest version of the `imsudb.jar` and `imsutm.jar` files.
- SMF is installed in z/OS.
- SMF is configured to accept type-29, subtype-2 records in the settings for SMF parameter `SYS1.PARMLIB(SMFPRMxx)` in z/OS.

Mobile workload reporting enhancement

IMS 14 is enhanced to work with the z/OS Workload Manager to track and report on mobile transactions and the amount of processing that they use. Mobile workloads are defined as the transactions processed by a named mobile application, or transactions that can be identified as originating from a mobile device.

You are responsible to track mobile workload transactions and produce a file showing mobile CPU consumption each month. Reports are submitted monthly to

set monthly licence charges. The mobile transactions data must consist of general purpose processor CPU seconds for each mobile transaction program summarized by hour by LPAR for all machines processing mobile transactions. The Sub-capacity Reporting Tool (SCRT) calculates a rolling 4 hour average utilization for each LPAR in each hourly interval. SCRT determines peak utilization across all the LPARs in which a program runs.

New attributes are added to classification rules to allow WLM to accumulate the mobile workload transaction CPU time in the LPAR. As a result, you can generate mobile workload reports with SCRT Version 23.13.0 or later in one of the following methods:

- Generating reports with the WLM classification rules.
- Generating reports with the original CSV files.

The functionality of the Mobile/zCAP Workload Reporting Tool (MWRT) is added to the Sub-Capacity Reporting Tool (SCRT).

Log record changes

The following log records are new or changed by this enhancement in IMS 14:

- X'56FA'

Requirements

Mobile workload reporting enhancement requires the following IMS 14 APARs :

- PI46933 (PTF numbers UI32215, UI32216)
- PI51948 (PTF number UI42428)

This enhancement also requires z/OS 2.1 with the following APARs installed in z/OS:

- WLM APAR OA47042 (PTF numbers UA79989, UA79990)
- WLM APAR OA49728 (PTF numbers UA80803, UA80804)
- WLM APAR OA51189

This enhancement is available for mobile workloads running on a zEC12 or zBC12 server with AWLC or AEWLC sub-capacity pricing.

Restrictions

- In flows where processor usage data is reported to Workload Manager (WLM) on the IWM4RPT (WLM Report) service, IMS ensures that the information reported to WLM is consistent with the information reported on the log record. As a side effect of this, the CPU execution times recorded in TPEXTIME, TPEZAAP, TPENDS, and TPENDU are computed slightly earlier in the internal IMS processing, possibly resulting in very small reductions to the amount of CPU time reported for each transaction.

Changes to administering IMS

You can configure your IMS system to use one of the following tracking mechanisms:

- Use an individual LPAR for mobile-only workloads. All defining programs can report the General Capacity Processor (GCP) CPUs that run on the LPAR as mobile CPUs.

- Build individual subsystems of the defining programs for mobile-only workloads. The defining programs can then report the GCP CPUs that run on the subsystems as mobile CPUs.
- Use the same subsystems for mobile and non-mobile workloads, but create different CPU reports for mobile CPU.

If the same subsystem is used for mobile and non-mobile workloads, IMS uses the following values to configure the IMS system to differentiate the two workload types:

- The TCP/IP port number
- The LTERM override name, if applicable, or the TPIPE name if no LTERM override value is specified

To indicate which TCP/IP port numbers, LTERM override names, and TPIPE names are associated with mobile workloads, create new Classification Rules for mobile workloads and set the Reporting Attribute option to MOBILE.

Changes to troubleshooting for IMS

A new control block table (CBT) pool is added in IMS 14:

- WLMB (Work area for parameter list for Workload Manager Classify service).

MSC enhancements

IMS 14 introduces several enhancements to the Multiple Systems Coupling (MSC) function, including a reduction in the amount of 24-bit storage that IMS uses, simplified enablement of MSC in IMS systems, and the dynamic definition of MSC resources.

MSC 24-bit storage reduction

In IMS 14, MSC-enabled IMS systems use less 24-bit storage, because MSC DDM modules, most MSC control blocks, and some MSC command modules are now stored in 31-bit storage.

The MSC control blocks are stored in a new DFSCLL3x member of the IMS.SDFSRESL data set. The DCBs for CTC links remain in 24-bit storage.

Before IMS 14, all of the MSC control blocks were stored in 24-bit storage in the DFSVNUCx member of the IMS.SDFSRESL data set.

In the DFSCLL3x member, the MSC control blocks are formatted differently but otherwise the same as in earlier IMS releases. IMS loads the MSC resources from the DFSCLL3x member into new MSCP and MSCL storage pools.

Migration considerations

Requirements:

- Exit routines that use information from the MSC logical and physical link blocks, such as the TM and MSC Message Routing and Control User exit routine (DFSMSCE0) and the Message Control/Error exit routine (DFSCMUX0), must be changed to either use callable services or use the FIND/SCAN function of the DFSCBTS macro to find the logical link blocks (LLBs) and the physical link control blocks (LCBs). The exit routines can no longer use the MSCDLLB and MSCDLCB fields in the MSCD block.

- Some of the MSC modules that ran in 24-bit storage in earlier versions of IMS, such as those related to logical links and physical links, are now loaded into new 31-bit storage pools. As a result of this change, MSC control blocks can no longer be sequentially scanned or their addresses calculated by multiplying the block length by the link number. The new storage pools (MSCP and MSCL) must be scanned using the DFSCBTS macro.
- User exit routines that reference MSC control blocks must run in 31-bit mode.
- User exit routines in the IMS nucleus can no longer use a VCON to reference MSC DDM modules or command modules and will need to be changed.

The /DISPLAY ASMT MSPLINK ALL command now displays physical links in alphabetic order instead of in system-generated order.

The QUERY MSLINK and QUERY MSPLINK commands now display the resources in alphabetic order instead of in system-generated order. This might affect existing automation if the automation looks for QUERY output in system-generated order.

MSC modules in the nucleus are moved from the nucleus to 31-bit storage, and their addresses are now saved in the MSCD.

The following table is a list of the changed modules:

MSC module addresses that are now in MSCD			MSC module addresses that are now in CMDE
DFSCMC10	DFSCMSH0	DFSCMS30	DFSCMLA0
DFSCMC20	DFSCMSI0	DFSCMS60	DFSCML70
DFSCMC40	DFSCMSP0	DFSCMS80	DFSIDPC0
DFSCMC50	DFSCMSQ0	DFSCMS81	DFSIDPD0
DFSCMR00	DFSCMSR0	DFSCMS82	
DFSCMSA0	DFSCMSS0	DFSCM4X0	
DFSCMSB0	DFSCMSU0	DFSCM7V0	
DFSCMSC0	DFSCMSV0	DFSCM7W0	
DFSCMSE0	DFSCMSX0	DFSCM7X0	
DFSCMSF0	DFSCMSY0	DFSVBLK0	
DFSCMSG0	DFSCMSZ0	DFSXC7D0	

Note: User exits in the IMS nucleus that reference MSC command modules with a VCON will no longer work. MSC command modules in 24-bit nucleus are moved to 31-bit storage. MSC module addresses are now in DFSCMDE (pointed to by SCDCMDE).

Coexistence considerations

For an IMS that is defined with MSC resources to coexist with IMS 14, any MSC user exits that access MSC LLB control blocks (logical links) or MSC LCB control blocks (physical links) must be changed to access those blocks by using DFSCBTS services. This change is required regardless of whether dynamic definition of MSC is enabled.

Log record changes

The log records that are changed by this enhancement in IMS 14 include:

- X'4036' – New for MSC logical link blocks (LLB, LTB1, LTB2, CRB). Replaces X'4008' for LLB, X'4005' for LTB, and X'4010' for CRB.
- X'4037' – New for MSC physical link blocks (LCB, CTT). Replaces X'400F' for LCB. CTT logging is new.
- X'4038' – New for MSC PCB for bandwidth.
- X'4039' – New for remote LTERMs.

Impact to administering IMS

Some of the MSC control blocks that were stored in 24-bit storage in earlier versions of IMS, such as those related to logical links and physical links, are now loaded into new 31-bit storage pools. As a result of this change, MSC control blocks can no longer be sequentially scanned or their addresses calculated by multiplying the block length by the link number. The new pools (MSCP and MSCL) must be scanned using the DFSCBTS macro.

Automation might need to be changed if it issues any of the following commands:

Commands	Reason automation might need to be changed
/DISPLAY ASMT MSPLINK, QUERY MSLINK, or QUERY MSPLINK commands	The resources are now displayed in alphabetic order instead of in system definition order.
QUERY LTERM, QUERY MSLINK, QUERY MSNAME, or QUERY MSPLINK commands that parse the output headers	All local headers now have an "L" prefix, to identify them as local values.
QUERY MSLINK SHOW(ALL)	Now displays the column for definitional attribute BACKUP.
QUERY MSPLINK SHOW(ALL)	Now displays columns for definitional attributes ASR, BACKUP, BUFSIZE, MAXSESS, and MODETBL.

Changes to commands

This enhancement changes the output or behavior of the following commands:

- /CHANGE LINK updates the last update timestamp.
- /DIAGNOSE SNAP LINK displays a 31-bit LLB address instead of a 24-bit address.
- /DISPLAY ASMT MSPLINK ALL displays MSC physical links in alphabetic order instead of system definition order.
- /DISPLAY POOL CBT displays information about the new MSCP and MSCL storage pools.
- /MSASSING LINK | MSNAME | SYSID updates the last update timestamp for the MSC logical link, MSC logical link path, or system ID.
- QUERY LTERM, QUERY MSLINK, QUERY MSNAME, and QUERY MSPLINK commands are changed as follows:
 - New SHOW keywords DEFNTYPE and TIMESTAMP are supported.
 - SHOW(ALL) keyword shows new definition type and timestamp columns.

- Show an "L" before each local header.
- UPDATE MSLINK, UPDATE MSNAME, and UPDATE MSPLINK commands record the timestamp when MSC resources are updated.

Changes to exit routines

Some of the MSC modules that ran in 24-bit storage in earlier versions of IMS, such as those related to logical links and physical links, are now loaded into new 31-bit storage pools. As a result of this change, MSC control blocks can no longer be sequentially scanned or their addresses calculated by multiplying the block length by the link number. The new pools (MSCP and MSCL) must be scanned using the DFSCBTS macro.

Addresses in the registers are now 31-bit addresses.

User exits in the IMS nucleus that reference MSC DDM or command modules with a VCON must be changed. Examples include the TM and MSC Message Routing and Control user exit routine (DFSMSCE0) and the Message Control/Error exit routine (DFSCMUX0). MSC DDM modules are removed from the nucleus and now reside in 31-bit storage. Their addresses can be accessed from pointers in other control blocks.

Alternatively, user exits that access MSC physical link (LCB) and logical link (LLB) control blocks can now access these blocks by using DFSCBTS FUNC / SCAN services. User exits that access MSC physical link (LCB) and logical link (LLB) control blocks must be changed to access these blocks by using callable services or DFSCBTS FUNC / SCAN services.

The MSC feature flag (SCDPRDEF,SCDPDMUL) still means that MSC is enabled, but it could be enabled with no MSC resources generated. In this case, fields MSCDLLB, MSCDLLBN, MSCDLCB, and MSCDLCBN are all zero.

MSC enablement enhancements

IMS 14 simplifies the enablement of Multiple Systems Coupling (MSC) with a new MSC= execution parameter and the consolidation of MSC parameters in a new MSC section of the DFSDFxxx member of the IMS PROCLIB data set.

The new MSC= execution parameter simplifies the initial enablement of MSC in an IMS system. Before IMS 14, you could not enable MSC without first defining an MSC link with stage-1 system definition macros. In IMS 14, MSC can be enabled without defining any MSC resources. This simplified enablement can be helpful in a shared-queues environment where, if any IMS system in the shared-queues group uses MSC, MSC must be enabled in all IMS systems in the shared-queues group, even if they do not use MSC.

Cloning back-end IMS systems in a shared-queues environment is simpler with the MSC= parameter because defining a unique MSC link is no longer required to enable MSC in each back-end IMS system that processes MSC messages on the shared queue.

Local system IDs (SYSIDs) can now be defined for an IMS system by specifying the new SYSID parameter in the new MSC section of the DFSDFxxx PROCLIB member. In an IMSplex with shared-queues environment, specifying the SYSID parameter initiates the building of the common SYSID (or SID) table.

The new MSC section of the DFSDFxxx member can also be used to consolidate all MSC PROCLIB member parameters into a single PROCLIB member. The MSC parameters that can now be specified in the new MSC section of the DFSDFxxx member include:

- GENIMSID (from the DFSDCxxx PROCLIB member)
- MSCSEC (from the DFSDCxxx PROCLIB member)
- MSCVGR (from the DFSDCxxx PROCLIB member)
- MSCRSCS=DYN | NODYN
- SYSID

Migration considerations

If you intend to enable dynamic definition of MSC later, specify the new execution parameter MSC=Y, because it is required when dynamic definition of MSC is enabled and no MSC resources are defined during IMS system definition.

If you are migrating an IMS system that already contains MSC resources that are defined during IMS system definition, you are not required to specify the MSC= parameter when you migrate to IMS 14 with the same system definition.

Impact to installing and defining IMS

IMS 14 introduces significant new options for enabling MSC and defining MSC resources.

MSC can be enabled by using the new MSC= execution parameter.

The DFSDFxxx member of the PROCLIB data set supports a new MSC section where new and existing MSC parameters can be specified. All of the MSC parameters that were previously specified only in the DFSDCxxx PROCLIB member can also be specified in the MSC section of the DFSDFxxx member. The existing MSC parameters can still be specified in the DFSDCxxx member. If the MSC parameters are specified in both the DFSDCxxx and DFSDFxxx member, the specifications in the DFSDCxxx member take precedence.

The following MSC parameters are new in IMS 14:

- MSC=Y | N, an execution parameter
- SYSID=(*sysid1,sysid2*), a new parameter in the new MSC section of the DFSDFxxx member

MSC= is an execution parameter that can be specified in the DFSPBxxx PROCLIB member. MSC=N is the default, but IMS overrides the default if any MSC resources were defined during stage-1 system definition.

The SYSID parameter defines local MSC system identifiers for an IMS system and is specified in the new MSC section of the DFSDFxxx PROCLIB member.

Before IMS 14, a SYSID value was specified on the MSNAME macro and required a logical link to which the SYSID was assigned.

Impact to troubleshooting for IMS

IMS can now issue several new and changed messages and abend codes to indicate the value of the MSC= parameter and any identify problems that might occur with either its specification or with the enablement of MSC.

For a list of the messages and codes that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

MSC type-2 command enhancements

In IMS 14, the UPDATE MSPLINK command is enhanced to support modification of the ADDR, BACKUP, BUFSIZE, and SESSION attributes of a Multiple Systems Coupling (MSC) physical link.

Changes to installing and defining IMS

You can now modify the following attributes of an MSC physical link in an online IMS system by using the IMS type-2 command UPDATE MSPLINK:

ADDR

The address of the channel-to-channel adapter.

BACKUP

The automatic restart priority for TCP/IP and VTAM physical links after an XRF takeover.

BUFSIZE

The input and output buffer sizes for logical links that are assigned to the physical link.

SESSION

For TCP/IP and VTAM physical links, the number of parallel sessions that can be active.

Changes to troubleshooting for IMS

For a list of the messages and codes that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Changes to commands

The UPDATE MSPLINK command is enhanced to allow a systems programmer to update the ADDR, BACKUP, BUFSIZE, and SESSION attributes of an MSC physical link.

For a list of the commands that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

Dynamic definition of MSC resources

IMS 14 supports the dynamic definition of Multiple Systems Coupling (MSC) links, link paths, and remote logical terminals (LTERMs).

With the introduction of support for the dynamic definition of MSC resources such as physical and logical links, link paths, and remote LTERMs, you can create and delete new MSC resources without coding stage-1 system definition macros and without taking IMS offline. The MSC resources are created and deleted by using new IMS type-2 CREATE and DELETE commands.

Additionally, with the IMS type-2 commands that are used to create and update MSC resources, you can sort and scroll the MSC resource definitions, and use wildcard characters to select sets of MSC resources.

Each time an MSC resource is created, modified, or deleted dynamically, IMS logs the event in a X'22' or X'02' log record. During a warm or emergency restart, IMS restores the MSC resources to their state before termination by reading the log records. The log records are also useful for diagnosing problems. For newly created MSC resources, the X'22' log records include the time when the MSC resource was created.

Migration considerations

To migrate to using dynamic definition of MSC resources, evaluate existing transaction, LTERM, and MSNAME names for conflicts with dynamic MSC. Dynamic MSC reserves additional name prefixes, including MSNS and MSNI. If you already have existing transaction, LTERM, or MSNAMEs defined that start with the prefix MSNS or MSNI, the CREATE MSNAME command might not complete successfully. The CREATE MSNAME command defines an internal system MSNAME with the naming convention MSNIxxxx, where xxxx is the link number. If MSNIxxxx already exists as an LTERM, transaction, or MSNAME, the CREATE MSNAME command will fail.

| The QUERY LTERM, QUERY MSLINK, QUERY MSNAME, and QUERY MSPLINK
| commands are changed to include L before all local headers in command output,
| as preconditioning to distinguish from the global output fields for the IMSRSC
| repository. If you have automation that looks for the headers, it might need to
| change to look for the new local headers that have L at the beginning.

Automation might need to be changed if it issues the following commands:

- /DISPLAY ASMT MSPLINK, QUERY MSLINK, or QUERY MSPLINK commands, because the resources are now displayed in alphabetic order instead of in system definition order.
- QUERY LTERM, QUERY MSLINK, QUERY MSNAME, or QUERY MSPLINK commands that parse the output headers, because all local headers now have an L prefix, to identify them as local values.
- QUERY MSLINK specified with SHOW(ALL), because SHOW(ALL) now displays the column for definitional attribute BACKUP.

- QUERY MSPLINK specified with SHOW(ALL), because SHOW(ALL) now displays columns for definitional attributes ASR, BACKUP, BUFSIZE, MAXSESS, and MODETBL.

IMS 14 manages link numbers differently after a cold start when logical links are deleted and re-created. Update your operating procedures to use link names instead of link numbers. The numbers for links remain the same across warm and emergency restarts, even if there are number gaps caused by deleted links. If links are created and deleted dynamically, resulting in gaps in link numbers, when an IMS system is cold started, the gaps are removed and the link numbers of the logical links that followed the gap are adjusted.

After migration, perform the following procedure to clean up system generation definitions:

1. Remove the MSC definitions from the stage 1 system definition, which include the following items:
 - MSPLINK, MSLINK, MSNAME, and NAME macros
 - MSVID and SYSTEM=(MSVERIFY) parameters in the IMSCTRL macro
 - SYSID parameter in the APPLCTN and TRANSACT macros
2. Run system generation.

Coexistence considerations

For an IMS generated with MSC resources to coexist with IMS 14, any MSC user exits that access MSC LLB control blocks (logical links) or MSC LCB control blocks (physical links) must be changed to access those blocks using IMS callable services. This change is required regardless of whether dynamic MSC is enabled. See the DFSMSCEO user exit for examples of using callable services for FIND and SCAN control blocks LCB (MSPLINKs), LLB (MSLINKs), LNB (MSNAMEs), and RCNT (remote LTERMs).

Log record changes

The following log records are changed by this enhancement in IMS 14:

- X'22' - Updated for the dynamic creation and deletion of MSC resources
- X'40' - Updated for the dynamic creation and deletion of MSC resources

Requirements

To dynamically define MSC resources, the Common Service Layer (CSL) with at least the Structured Call Interface (SCI) and the Operations Manager (OM) enabled is required.

Restrictions

A resource definition data set (RDDS) cannot be used with dynamically defined MSC resources.

Impact to installing and defining IMS

IMS 14 introduces significant new options for enabling MSC and defining MSC resources dynamically.

MSC resources can now be defined and deleted by using new type-2 CREATE and DELETE commands with new MSC keywords.

To use the new type-2 CREATE and DELETE commands to define MSC resources online, MSC dynamic resource definition must be enabled by specifying the new MSCRSCS=DYN parameter in the MSC section of the DFSDFxxx PROCLIB member.

MSC MSPLINKs and MSLINKs are now packaged in a new DFSCLL3x member, and are removed from the IMS nucleus.

Impact to administering IMS

As in earlier releases, IMS stage-1 system generation assigns numbers to links in the order the links are generated in. However, in IMS 14, if dynamic resource definition is enabled for MSC and a logical link is deleted dynamically, the link numbers of the remaining logical links can change after a cold start, even if the MSC resources are imported from the repository, system definition macros are used to keep the dynamic and static MSC definitions synchronized. Therefore, operational procedures must reference the logical links by link name instead of by link number.

When a logical link is defined by using the CREATE MSLINK command, IMS assigns the next available link number to the logical link. IMS manages the link number locally.

When a logical link is deleted by using the DELETE MSLINK command, IMS deletes the link number along with the logical link, and the link number is available for reuse. If a logical link that is deleted was re-created, the logical link might be assigned a different link number.

All links have a link name. For static links created by system generation, where a *linkname* (the label name on the MSLINK macro) is not specified, IMS internally generates a link name of DFSLxxxx, where *xxxx* represents the link number. For example, the link name for link 5 would be DFSL0005. This convention enables links to be referenced by IMS type-2 commands.

Recommendation: Operate and administer MSC logical links by using link names instead of link numbers. If the operating procedures that your installation uses for MSC links are based on link numbers, update them to use link names instead.

IMS type-2 MSC commands require link names instead of link numbers. For example, instead of starting a link by using the link number, as in the command /RSTART LINK 10, start the link by using the link name, as in the command UPDATE MSLINK NAME(*logicallinkname*) START(COMM).

Additional operational considerations:

- The CREATE MSNAME command must be issued before the CREATE LTERM command is issued.
- The DELETE commands must be issued in this order: DELETE LTERM, DELETE MSNAME, DELETE MSLINK, DELETE MSPLINK.
- Resources might need to be stopped for a DELETE command to succeed.
- The DELETE MSPLINK command is rejected if any MSLINK references it.
- The DELETE MSLINK command is rejected if any MSNAME references it.

- The DELETE MSNAME command is rejected if any LTERM or transaction references it.

Programming for IMS

Applications might need to be changed because of the changes to the QUERY commands for this enhancement. System programmers might need to change automation because of the QUERY command changes, and any automation that accesses MSC control blocks will need to be changed because the MSC control blocks are packaged differently.

Impact to troubleshooting for IMS

For a list of the messages and codes that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Changes to commands

This enhancement introduces new type-2 CREATE and DELETE commands for MSC resources.

The existing UPDATE MSLINK, UPDATE MSNAME, and UPDATE MSPLINK commands are enhanced to record the timestamp when MSC resources are updated, which can be displayed with the QUERY MSLINK, QUERY MSNAME, or QUERY MSPLINK commands.

The existing QUERY LTERM, QUERY MEMBER, QUERY MSLINK, QUERY MSNAME, and QUERY MSPLINK commands are enhanced to display timestamps and definition types. These commands are changed to include L before all local headers in command output, as preconditioning to distinguish from the global output fields for the IMSRSC repository.

The QUERY MEMBER TYPE(IMS) command is enhanced to display whether dynamic MSC is enabled.

For a list of the commands that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

Changes to exit routines

The IMS initialization exit (DFSINTX0) is no longer able to access MSC control blocks during IMS initialization, because the MSC control blocks are not built until restart. If the DFSINTX0 exit tries to access MSC control blocks, it will not find any and might need to be changed. The MSC control blocks that cannot be found are LLB, LCB, LNB, and RCNT. The DFSMSCE0 user exit initialization entry point, which is called at IMS restart, can access those control blocks with FIND/SCAN control block callable services. See the prolog of that user exit for details and samples of those services.

IMSRSC repository support for dynamically defined MSC resources

IMS 14 enhances the IMSRSC repository to optionally support Multiple Systems Coupling (MSC) resources.

With APAR PI50129 (PTF UI44232) and all of its prerequisite PTFs, MSC resources that are created or modified dynamically with the type-2 CREATE and UPDATE commands can be automatically exported to the IMSRSC repository at IMS checkpoint time and before IMS shutdown. After the MSC resources are stored in the IMSRSC repository, the resources are saved permanently, even across cold starts. Before IMS 14, MSC resources that were created or modified dynamically were saved across an IMS cold start only if stage-1 system definition macros were used and IMS was taken offline. After MSC resources are exported to the IMSRSC repository, IMS can automatically retrieve the dynamically defined MSC resources from the repository at cold start instead of from the resources generated during an IMS system definition.

With the IMSRSC repository, all MSC definitions are stored in a single centralized location.

Migration considerations

- If you use the IMSRSC repository to store dynamically defined MSC resources, ensure that automation and operational procedures that issue commands for MSC resources use type-2 commands, which specify link names, instead of type-1 commands, which specify link numbers. For example, instead of using the /RSTART LINK 10 command to start a link, use the UPDATE MSLINK NAME(*logicallinkname*) START(COMM) command. During stage-1 system generation, the IMS system assigns numbers to logical links in the order in which the links are generated. However, the numbers for links are not stored in the IMSRSC repository. If logical links are referenced by using link numbers and are automatically imported from the IMSRSC repository, the numbers of the links are likely to change at the next IMS cold start.
- If you are using channel-to-channel (CTC) links, consider removing the DD definitions for the CTC links from the IMS JCL before you import the MSC resources from the IMSRSC repository. This allows the CTC addresses that are defined to be used for the CTC links that are imported from the IMSRSC repository.
- If you use the IMSRSC repository to store dynamically defined MSC resources, the DFSCLL3x and DFSCLR0x members of the IMS.SDFSRESL data set are no longer required. Because the DFSCLC0x member might contain non-MSC resources, the DFSCLC0x might still be required. After you are satisfied with the setup of your DRD environment for MSC resources, the DRD environment is running successfully, and your MSC resources are exported to the IMSRSC repository, you can remove the DFSCLL3x and DFSCLR0x members. For the DFSCLC0x member, update the member to remove MSC logical link path definitions. If automatic import is enabled, MSC resource definitions are imported during IMS cold start from the repository that contains the most current data. However, you can continue to use the DFSCLL3x, DFSCLR0x, and DFSCLC0x members as the source for your MSC resource definitions, instead of a repository. If you continue to use the DFSCLL3x, DFSCLR0x, and DFSCLC0x members, keep your system definition macros synchronized with the changes you make dynamically using DRD commands.
- If you migrate to using the IMSRSC repository for dynamically defined MSC resources but continue to use the DFSCLL3x, DFSCLR0x, and DFSCLC0x

members of the IMS.SDFSRESL data set, keep resource definitions that are in the members synchronized with the resource definitions that are in the repository. This synchronization enables you to maintain viable DFSCLL3x, DFSCLR0x, and DFSCLC0x members if you must disable DRD and fall back to using the system generation process for MSC resources. To keep your DFSCLL3x, DFSCLR0x, and DFSCLC0x members synchronized with your online definitions, update your static macro definitions with the changes that you make dynamically using type-2 commands. When changes are made dynamically, perform a MSC system definition to add, change, or delete resources from the DFSCLL3x, DFSCLR0x, and DFSCLC0x members.

Recommendation: It is recommended that you enable automatic export to the IMSRSC repository only after both of the following tasks are completed:

- All RM systems are migrated to IMS 14 or later.
- For IMS 14 systems, the enabling APAR for IMSRSC repository for dynamically defined MSC resources, APAR PI50129, is applied.
- For IMS 15 systems, the enabling APAR for IMSRSC repository for dynamically defined MSC resources, APAR PI74957, is applied.

Coexistence considerations

You might need to update an MSC resource definition in an IMSplex in which an IMS 14 system that uses an IMSRSC repository coexists with an IMS Version 13 or earlier IMS system. In this scenario, the MSC resource updates that are made on the IMS 14 system by using the UPDATE commands must be made on the IMS Version 13 or earlier IMS systems by coding stage-1 system definition macros.

If the IMSRSC repository support for dynamically defined MSC resources is enabled for IMS 14, apply the following coexistence APARs to the RM systems in the IMSplex:

- IMS 14: APAR/PTF PI49208/UI32429
- IMS Version 13: APAR/PTF PI49334/UI32565
- IMS Version 12: APAR/PTF PI49329/UI32564

Use one of the following methods to install the coexistence APARs that are listed in the preceding list:

- First, install IMS Version 12 APAR/PTF PI49329/UI32564, IMS Version 13 APAR/PTF PI49334/UI32565, and the IMS 14 preconditioning APAR/PTF PI49208/UI32429 on all RM systems in the IMSplex. Then, install the IMS 14 preconditioning APAR/PTF PI45186.
- Install all of the following APARs on all RM systems in the IMSplex at the same time:
 - IMS Version 12 APAR/PTF PI49329/UI32564
 - IMS Version 13 APAR/PTF PI49334/UI32565
 - IMS 14 preconditioning APAR/PTF PI49208/UI32429
 - IMS 14 preconditioning APAR/PTF PI45186

Log record changes

After dynamically defined MSC resources are automatically exported to the IMSRSC repository, a X'22' subtype X'51' record is written to the IMS log. The record contains the IDs of the IMS systems that the MSC definitions were written

for and the number of MSC definitions that were exported to the repository for each resource type.

Impact to defining IMS

The MSCREPO parameter is introduced in the MSC section of the DFSDFxxx member of the IMS PROCLIB data set to define whether MSC resources are stored in the IMSRSC repository.

In the DYNAMIC_RESOURCES section of the DFSDFxxx member, the AUTOIMPORT parameter is modified to also apply to MSC resources, if dynamic resource definition for MSC resources is enabled. If AUTOIMPORT=AUTO is defined and the IMSRSC repository does not contain MSC resources, the IMS system reads and imports MSC resources from the DFSCLL3x member of the IMS.SDFSRESL data set if the member is not empty.

Impact to administering IMS

As in earlier releases, IMS stage-1 system generation assigns numbers to links in the order in which the links are generated. However, in IMS 14, if dynamic resource definition and the IMSRSC repository are enabled for MSC resources, the link numbers are likely to change at the next IMS cold start when the MSC resources are automatically imported from the repository. Link numbers are not stored in the repository. Consequently, operational procedures should reference the logical links by link name instead of link number.

Impact to troubleshooting for IMS

For a list of the messages and codes that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Changes to commands

This enhancement updates the following existing type-2 commands:

Table 16. Type-2 command enhancements

Command	How the command is enhanced
/CHANGE LINK	Marks the logical link as needing to be exported.
CREATE LTERM	Marks the MSC resources as needing to be exported.
CREATE MSLINK	
CREATE MSNAME	
CREATE MSPLINK	
DELETE DEFN TARGET(REPO)	Supports the following keywords for the TYPE parameter: LTERM, MSLINK, MSNAME, MSPLINK.

Table 16. Type-2 command enhancements (continued)

Command	How the command is enhanced
/MSASSIGN LINK TO MSPLINK	Marks the logical link or logical link path (MSNAME) as needing to be exported.
/MSASSIGN MSNAME TO LINK	
/MSASSIGN SYSID TO LINK	
QUERY LTERM	Supports the SHOW(EXPORTNEEDED) keyword, to display the MSC resources that need to be exported, and displays repository data.
QUERY MSLINK	
QUERY MSNAME	
QUERY MSPLINK	
UPDATE MSLINK	Marks the MSC resource as needing to be exported, if one or more of the attributes are changed. Changes can now be retrieved at IMS cold start, if the resources are imported from the IMSRSC repository.
UPDATE MSNAME	
UPDATE MSPLINK	

MSC resources that you create and update by using the IMS type-2 CREATE and UPDATE commands can now be saved across cold starts by exporting the online resource definitions to the IMSRSC repository.

For a list of the commands that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

OTMA enhancements

IMS 14 introduces a variety of enhancements to OTMA.

Conversation ID enhancement for QUERY OTMATI command

In IMS 14, the output of the QUERY OTMATI command can include the ID of the conversation that a message belongs to. After identifying orphaned messages on the tpipe, you can now note the conversation ID of the orphaned messages and cancel the conversation by specifying the conversation IDs on the /EXIT CONV command.

Conversational messages can become orphaned on a tpipe if the OTMA client terminates without IMS detecting the termination.

Changes to commands

IMS 14 enhances the QUERY OTMATI command by adding a new CONVID option to the SHOW() keyword and a new ConVID field in the command output.

For a list of the commands that are changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

Dynamic storage for OTMA descriptors

In IMS 14, IMS allocates the storage for OTMA descriptors dynamically, which provides two potential benefits. For installations that require more descriptors than were allowed in previous versions of IMS, you can increase the maximum allowed

number of descriptors that can be defined. For installations that use fewer descriptors than the maximum allowable number, the OTMA descriptors use less ECSA storage in IMS 14.

In earlier releases of IMS, IMS allocated fixed amounts of storage for OTMA descriptors regardless of how many descriptors were defined. If you defined only a few descriptors, the rest of the fixed amount of storage was wasted storage. If you used many descriptors, the fixed amount of storage limited the number of descriptors that you could define.

In IMS 14, because IMS allocates the storage for OTMA descriptors dynamically, if you define only a few descriptors, IMS uses only the amount of ECSA storage that is required for those descriptors. And if you need to define more descriptors that were allowed in previous versions, you can increase the limit and IMS allocates only as much more storage as is needed for current number of defined descriptors.

By default, the maximum number of OTMA descriptors that you can define in IMS 14 is the same as the maximum number of descriptors that you could define in previous releases of IMS: 255 for OTMA client descriptors and 510 for OTMA destination descriptors.

Now, you can increase the maximum number for both types of descriptor limits up to 4095 by using the DDESCMAX and MDESCMAX parameters in the DFSOTMA descriptor in the DFSYDTx member of the IMS PROCLIB data set.

Changes to DFSTSCD and DFSYDES control blocks

To support allocating storage for OTMA descriptors dynamically, the OTMA DFSTSCD and DFSYDES control blocks were modified in IMS 14.

In the DFSTSCD control block, the following changes were made for destination descriptors:

- The TSCD_DR_DESC_TABLE field is zero. In prior versions of IMS, this field pointed to the first destination descriptor entry that was defined in contiguous storage.
- The TSCD_DR_BEGCHAIN field points to the first destination descriptor entry in the chain. In prior versions of IMS, this field pointed to the first destination descriptor entry that was defined in contiguous storage.
- The TSCD_DR_ENDCHAIN field points to the last destination descriptor entry in the chain.

The TSCD_DR_DESC_COUNT field in the DFSTSCD control block, which contains the count of destination descriptor entries, is unchanged.

In the DFSTSCD control block, the following changes were made for member descriptors:

- The TSCD_DESC_TABLE field points to the first member descriptor entry in the chain. In prior versions of IMS, this field pointed to the first member descriptor entry that was defined in contiguous storage.

The TSCD_DESC_COUNT field in the DFSTSCD control block, which contains the count of member descriptor entries, is unchanged.

In the DFSYDES control block, the following changes were made for destination descriptors:

- For destination descriptors, the YDRDESC DSECT defines the mapping of an OTMA destination descriptor entry. The YDRDESC_FWDPTR field points to the next destination descriptor entry in the chain. The YDRDESC_BWDPTR field points to the previous destination descriptor entry in the chain.
- For member descriptors, the YDESC DSECT defines the mapping of an OTMA member descriptor entry. The YDESC_NEXT field points to the next member descriptor entry in the chain.

Migration considerations

Before IMS 14, only a value of 255 or 510 was displayed in messages DFS2384W and DFS3678E, which are the warning and error messages that are issued for approaching or exceeding the maximum allowable number of OTMA descriptors. In IMS 14, the value that is displayed in the error messages can be 255, 510, or another value that is specified on the DDESCMAX or MDESCMAX keywords in the DFSOTMA descriptor.

Log record changes

The descriptor mapping changed the contents of the following log records in IMS 14:

- X'4035', the Destination descriptor checkpoint log record
- X'221B', the log record for OTMADESC type-2 commands

Requirements

If either the DDESCMAX parameter, the MDESCMAX parameter, or both are defined in the DFSOTMA descriptor, the DFSOTMA descriptor must be specified in the first 255 client (type-M) descriptor entries in the DFSYDTx member of the IMS PROCLIB data set.

This enhancement has the same hardware and software requirements as the base IMS 14 product.

Changes to installing and defining IMS

You can now adjust the maximum number of OTMA descriptors that you can define by specifying the DDESCMAX and MDESCMAX parameters in the DFSOTMA descriptor in the DFSYDTx member of the IMS PROCLIB data set.

Changes to troubleshooting for IMS

The IMS control block table (CBT) contains two new storage pools for OTMA descriptors: OTDD for OTMA destination descriptors and OTMD for OTMA member descriptors. You can display these pools by issuing the /DISPLAY POOL command with either the DCC, CBT, or ALL keyword specified.

Before IMS 14, only a value of 255 or 510 was displayed in messages DFS2384W and DFS3678E, which are the warning and error messages that are issued for approaching or exceeding the maximum allowable number of OTMA descriptors. In IMS 14, the value that is displayed in the error messages can be 255, 510, or another value that is specified on the DDESCMAX or MDESCMAX keywords in the DFSOTMA descriptor.

For a list of the messages and codes that are changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

OTMA dynamic SAPs enhancement

In IMS 14, the OTMA dynamic SAPs enhancement enables you to specify a minimum and a maximum number of save area prefixes (SAPs) that IMS allocates for OTMA client input messages to maintain performance and limit the usage of ECSA storage.

IMS uses the SAPs with the OTMA input message (OIM) task control block (TCBs) of an OTMA client to process OTMA input messages. IMS requires one SAP for each input message. After IMS finishes processing a message, IMS reuses the SAP for another message.

By default, IMS pre-allocates 18 SAPs for an OTMA client when the client connects. The pre-allocated SAPs remain allocated to the OTMA client until IMS is restarted.

If all SAPs are in use when new input messages are received from an OTMA client, IMS dynamically allocates more SAPs as needed until the maximum number of SAPs is reached. By default, the maximum number of SAPs that can be dynamically allocated to an OTMA client is 500. After dynamically allocated SAPs are no longer needed, IMS gradually cleans them up, until only the pre-allocated SAPs remain.

If IMS cannot dynamically allocate more SAPs quickly enough or the maximum number of SAPs are already allocated, IMS uses selective dispatching to prioritize processing so that higher priority work is processed before lower priority work.

During IMS system checkpoints, if IMS used selective dispatching for OTMA input messages for any length of time since the last system checkpoint, IMS issues message DFS0769I, OIM SELECTIVE DISPATCHING - SAPS.

In most cases, selective dispatching is active only momentarily until IMS allocates more SAPs and does not indicate a problem nor significantly impact performance. However, if IMS uses selective dispatching frequently or you expect regular high volumes of critical work, you might consider adjusting the number of SAPs that IMS pre-allocates to ensure that there are enough to process all incoming OTMA messages. You might also consider increasing the maximum number of SAPs that IMS can allocate to OTMA clients.

Statistics for the usage of SAPs by OIM TCBs are written at an IMS level in the x'450F' dispatcher statistics log record. They are written at an individual OTMA member level in the X'4518' individual TCB dispatcher statistics log record. Examining the statistics in these log records can provide useful information about the occurrences of selective dispatching.

IMS stores SAPs in the extended common storage area (ECSA). Allocating more SAPs uses more ECSA storage.

Log record changes

Statistics for the usage of SAPs by OIM TCBs are written at an IMS level in the X'450F' dispatcher statistics log record. They are written at an individual OTMA member level in the new X'4518' individual TCB dispatcher statistics log record.

Changes to installing and defining IMS

The OTMA dynamic SAP enhancement adds the DSAP and DSAPMAX parameters to both the OTMA client descriptors and the DFSOTMA descriptor in the DFSYDTx member of the IMS.PROCLIB data set. DSAP defines the number of pre-allocated SAPs that IMS allocates during initialization. DSAPMAX defines the maximum number of SAPs that IMS can allocate dynamically when more SAPs are needed.

Changes to administering IMS

If necessary for performance reasons or for adjusting the usage of ECSA storage, IMS system administrators can adjust the number of SAPs that IMS allocates for OTMA clients by specifying the DSAP and DSAPMAX parameters in OTMA client descriptors or the DFSOTMA descriptor in the DFSYDTx member of the IMS.PROCLIB data set. DSAP defines the number of pre-allocated SAPs that IMS allocates during initialization. DSAPMAX defines the maximum number of SAPs that IMS can allocate dynamically when more SAPs are needed.

Changes to troubleshooting for IMS

During a system checkpoint, if OTMA entered into selective dispatching mode at any time since the last system checkpoint, IMS issues message DFS0769I with OIM displayed as the TCB type. For information about the occurrences of selective dispatching, review the statistics in the X'450F' and X'4518' log records.

If errors occur coding the new DSAP and DSAPMAX parameters in the OTMA descriptors, they are indicated by new error text that is displayed by message DFS2385E.

For a list of the messages and codes that are changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

OTMA security enhancements

IMS 14 introduces several security enhancements for OTMA.

Automatic refresh of cached ACEEs when RACF definitions change

In IMS 14, IMS automatically listens for notification from z/OS of changes to the RACF security credentials of OTMA users. When a notification is received, OTMA refreshes the impacted ACEEs.

During IMS initialization, IMS automatically registers with the z/OS event notification facility (ENF) to listen for notifications of changes to the OTMA security definitions in RACF. No IMS configuration is required.

When IMS is registered with ENF for OTMA, IMS listens for ENF event code 71, which indicates that a users security definitions were modified in RACF. Upon receiving the ENF notification, OTMA refreshes the cached ACEEs automatically. You no longer have to issue the /SECURE OTMA REFRESH command after changes are made in RACF security definitions.

This support is enabled only when OTMA security is CHECK, FULL, or PROFILE.

Changes to troubleshooting for IMS

IMS 14 introduces a new error message DFS3525E, which is issued if IMS cannot register with the z/OS Event Notification Facility during startup.

For a list of the messages and codes that are new for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

OTMA client bid security enhancement

In IMS 14, you can enforce security on OTMA client bid requests without enabling transaction and command security by specifying the new JOIN level of OTMA security.

You can enable the JOIN level of OTMA security either by specifying OTMASE=J in your startup procedure or by issuing the IMS type-1 command /SECURE OTMA JOIN.

Changes to installing and defining IMS

You can now specify J on the OTMASE startup parameter to enable the JOIN level of OTMA security. When the JOIN level is enabled, IMS authorizes the OTMA client during the client bid request, but does not enable transaction or command security.

Changes to commands

You can now enable the JOIN level of security for OTMA clients by issuing the IMS type-1 command /SECURE OTMA JOIN. You can enable JOIN security for individual OTMA clients by specifying the tmember name on the TMEMBER keyword of the command.

For a list of the commands that are changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

OTMA ACEE flood control enhancement

In IMS 14, OTMA is enhanced with the accessor environment element (ACEE) flood control function to prevent an excessive number of RACF ACEEs from being stored in cache.

Previously, the number of ACEEs that were stored in cache could not be limited. An unlimited number of ACEEs stored in cache by OTMA and a high aging value for the ACEEs caused virtual storage in the IMS control region to run out.

With APAR/PTFs PI68466/UI47190/UI47191, you can enable OTMA ACEE flood control in the DFSOTMA descriptor of the DFSYDTx member of the IMS PROCLIB data set. When you enable OTMA ACEE flood control, you limit the number of RACF user IDs, which in turn limits the number of ACEEs that are cached by OTMA. This in turn prevents virtual storage in the IMS control region from running out.

With OTMA ACEE flood control enabled, OTMA checks the expiration values of the least recently used ACEEs first to speed up the ACEE cleanup process.

With APAR/PTFs PI68466/UI47190/UI47191, you can also specify the **ACEEAGE** parameter on the /SECURE OTMA command to define an OTMA ACEE aging value that overrides the aging value passed by OTMA clients. You can use the **ACEEAGE** parameter to define an aging value that is lower than the value passed by OTMA clients, thereby expediting the cleanup of OTMA ACEEs that are stored in cache.

Changes defining IMS

The DFSOTMA descriptor of the DFSYDTx member is enhanced with the following parameters:

TOACEE=NO|YES

Enables or disables the OTMA ACEE flood control function.

ACEEUSR=

When OTMA ACEE flood control is enabled, allows you to define the number of RACF user profiles that are stored in ACEEs.

Changes to commands

The following commands are enhanced:

- The /DISPLAY OTMA command is enhanced with the following output fields:

ACEECT

Displays the total number of cached OTMA ACEEs for the OTMA server.

TOACEE

Displays whether OTMA ACEE flood control is enabled.

- The /SECURE OTMA command is enhanced with the **ACEEAGE** parameter to allow you to define an ACEE aging value for OTMA clients.

For a list of the commands that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

OTMA tpipe parallelism enhancement

In IMS 14, the OTMA tpipe parallelism enhancement, which enables OTMA tpipes in IMS to support multiple active RESUME TPIPE requests on the same tpipe, can significantly increase the throughput on OTMA tpipe for output messages, particularly those for synchronous or asynchronous callout requests, and can significantly improve failover protection for OTMA tpipes.

Enterprise architects, system programmers, or anyone responsible for designing a system that processes a high volume of synchronous and asynchronous IMS callout messages via OTMA Resume TPIPE requests are likely to find this enhancement beneficial.

When an OTMA tpipe supports multiple active RESUME TPIPE requests, OTMA clients can pull output messages from the tpipe by using multiple active RESUME TPIPE requests, which the tpipe processes in parallel. If the processing for any one RESUME TPIPE request becomes impaired, OTMA continues to deliver the output messages on the tpipe through the other active RESUME TPIPE requests, which prevents the RESUME TPIPE request, or the tpipe itself, from becoming a bottleneck for output messages from IMS.

OTMA tpipe support for multiple active RESUME TPIPE requests also improves failover protection for OTMA tpipes by eliminating the need to switch to a back up OTMA client if the active OTMA client terminates. When an OTMA tpipe supports multiple active RESUME TPIPE requests from multiple clients, if any one of the OTMA clients fail or lose their connection, the others can continue processing the output from the tpipe without any lapse in processing.

Support for multiple active RESUME TPIPE requests can also decrease the complexity and cost of routing output from multiple IMS application programs through OTMA tpipes to the same final destination. Without the parallel processing of RESUME TPIPE requests, to achieve optimum performance, as well as to avoid a potential bottleneck, the output from the IMS application programs is typically routed through multiple OTMA destination descriptors or OTMA tpipes; however, such a configuration usually requires some combination of unique coding in each IMS application program, in multiple OTMA destination descriptors, and in the OTMA clients. With support for multiple active RESUME TPIPE requests, you can realize similar performance benefits by routing the output from the multiple application programs through a single OTMA destination descriptor and a single OTMA tpipe. Multiple OTMA clients can then retrieve the output by issuing the same RESUME TPIPE requests with the OTMA tpipe name specified as an alternate client ID.

For diagnostic purposes, when support for multiple active RESUME TPIPE requests is enabled, the RESUME TPIPE token can be used to correlate each RESUME TPIPE request with the client that issued it. The RESUME TPIPE token, as well as the ID of any tpipe to which undelivered output is rerouted, can be displayed by issuing the existing commands that support OTMA and IMS Connect.

Migration considerations

By default, support for parallel RESUME TPIPE requests is disabled, so the OTMA tpipe parallelism enhancement is not expected to affect migration.

Coexistence considerations

IMS Version 12 and IMS Version 13 instances of IMS Connect can connect to OTMA tpipes that support parallel processing of RESUME TPIPE requests after the appropriate coexistence APARs are applied. The APARs are required only for IMS Version 12 and IMS Version 13 instances of IMS Connect that connect to tpipes in which tpipe parallelism is enabled.

For connections from IMS Version 12 and IMS Version 13 instances of IMS Connect, support for the parallel processing of RESUME TPIPE requests can be enabled only from the OTMA client descriptors.

Previous versions of IMS Connect require these compatibility fixes:

- IMS Connect Version 13 APAR/PTF PM93878/UK98632
- IMS Connect Version 12 APAR/PTF PM93880/UK98633

Log record changes

The log records that are changed by this enhancement in IMS 14 include:

- X'31' - Queue Manager Get Unique
- X'35' - Queue Manager Enqueue
- X'36' - Queue Manager Dequeue
- X'38' - Queue Manager Re-enqueue
- X'4033' - Checkpoint MTEs
- X'4032' - Checkpoint YQABs

Impact to installing and defining IMS

In IMS OTMA, TPIPE support for multiple active resume TPIPE requests can be enabled by specifying MULTIRTP=Y in an OTMA client descriptor in the DFSYDTx member of the IMS PROCLIB data set.

You can define an IMS system default for all OTMA clients by specifying the MULTIRTP parameter in the DFSOTMA system client descriptor. The MULTIRTP value in the DFSOTMA client descriptor can be overridden by individual client descriptors and by OTMA clients in their client bid requests.

For IMS Connect connections to IMS OTMA, you can specify TPIPE support for multiple active resume TPIPE requests by specifying MULTIRTP=Y when you define the connection with either the CREATE IMSCON TYPE(DATASTORE) command or the IMS Connect DATASTORE configuration statement.

You can also define an IMS Connect default for MULTIRTP for all connections by specifying the MULTIRTP parameter in either the UPDATE IMSCON TYPE(CONFIG) command or in the IMS Connect HWS configuration statement.

Impact to administering IMS

You can check the number of output messages that are currently queued on a TPIPE by issuing the IMS command /DISPLAY TMEMBER TPIPE with either OUTPUT or SYNC specified. In a shared queues environment, you can see the number of messages that are queued by issuing the /DISPLAY TMEMBER TPIPE QCNT command.

To diagnose problems, you can correlate a specific RESUME TPIPE request with the IMS Connect client that issued it from the information returned by the /DISPLAY TMEMBER command in IMS and the QUERY IMSCON TYPE(CLIENT) in IMS Connect.

Programming for IMS

Although the OTMA tpipe parallelism enhancement does not directly impact application programming, because of the nature of parallel processing, make sure that the application programs that send messages to or receive messages from a tpipe that supports parallel RESUME tpipe requests can handle messages that are not processed in the order in which they were sent.

Impact to troubleshooting for IMS

To diagnose problems, you can correlate a specific RESUME TPIPE request with the IMS Connect client that issued it from the information returned by the /DISPLAY TMEMBER command in IMS and the QUERY IMSCON TYPE(CLIENT) in IMS Connect.

A new NAK code is introduced: X'0037 (NACK_WRONG_MULTIRTP).

The OTMA tpipe parallelism enhancement introduces changes to the following IMS messages:

- DFS2385E
- DFS3494E

For a list of the messages and codes that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Changes to commands

The MULTIRTP keyword is added to the following IMS commands that create or update IMS Connect online resources:

- CREATE IMSCON TYPE(DATASTORE)
- UPDATE IMSCON TYPE(CONFIG)

The current settings for MULTIRTP are now included in the following commands that display IMS Connect resource definitions:

- QUERY IMSCON TYPE(CONFIG)
- QUERY IMSCON TYPE(DATASTORE)
- VIEWHWS
- VIEWDS
- QUERY DATASTORE
- QUERY MEMBER TYPE(IMSCON)

The IMS command /DISPLAY TMEMBER is enhanced in IMS 14 to show the number of resume TPIPE requests that are currently active on a TPIPE when the TPIPE SYNC keywords are specified. The number of active resume TPIPE requests appears under the RTA output heading.

When support for multiple active RESUME TPIPE requests is enabled and /DISPLAY TMEMBER TPIPE OUTPUT is issued, the command output also

displays the RESUME TPIPE token of each RESUME TPIPE request that is currently active on a given OTMA tpipe, along with the RESUME TPIPE option and mode.

The IMS Connect command QUERY IMSCON TYPE(CLIENT) also displays the RESUME TPIPE tokens.

For a list of the commands that are changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

Shared queues buffer pool enhancement

In IMS 14, the shared queues buffer pool enhancement increases the maximum number of message pool buffers that you can specify on the QBUFMAX execution parameter to 9,999,999.

The QBUFMAX parameter limits the number of message queue buffers for the queue pool.

Migration considerations

Check for any user exit routines or usermods that reference the QSCDQBUL field that is defined by macro DFSQSCD. The maximum value that the QSCDQBUL field can contain has increased from 9999 to 9999999.

Changes to troubleshooting for IMS

The value field for the QBUFMAX parameter that is displayed in DFS1929I can now contain a 1- to 7-digit number. Previously, the field could only contain a 1- to 4-digit number.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Changes to exit routines

Field QSCDQBUL that is defined in macro DFSQSCD can now contain a value larger than decimal 9999. If you have user exit routine or usermod code that refers to this field, ensure that the code can handle values larger than 9999.

Shared queues overflow feedback enhancement

In IMS 14, the IMS Common Queue Server (CQS) can provide usage feedback about the shared queues structure to the Queue Space Notification exit routine (DFSQSSP0) when a CQS client adds or removes a message from the shared queue.

This usage feedback information indicates how much of the message queue structure is used for both the primary structure and overflow structure:

- Number of structure entries in total and in use
- Number of structure elements in total and in use

This enhancement helps you make decisions to prevent the message queue structure from becoming full, and thus ensure system availability.

Changes to administering IMS

The CQS client requests, CQSPUT and CQSDEL, can provide information about current structure utilization through the FEEDBACK= and FEEDBACKLEN= parameters. IMS uses this capability to provide structure utilization information about the IMS shared message queue structure to the Queue Space Notification exit routine for shared queues (DFSQSSP0). The structure utilization information contains the total number of allocated and in-use entries and elements in the primary and overflow structures. This information helps you prevent the message queue structure from becoming full. Note that DFSQSSP0 provides only information about the IMS shared message queue structure, not the IMS shared EMH structure.

Changes to programming for IMS

The CQS client requests CQSPUT and CQSDEL provide two new parameters, FEEDBACK and FEEDBACKLEN, which allow you to request feedback information about the shared queue structure usage. This information includes the number of entries and elements that are allocated in the structure and the number currently in use. If the structure is in overflow mode, information about both the primary and overflow structure is returned.

Changes to exit routines

The Queue Space Notification exit routine (DFSQSPC0/DFSQSSP0) parameter list DFSPARM provides new fields QSPCFBKP and QSPCFBKL. These fields are used to receive feedback information about CQS structure usage.

Synchronous callout enhancements

IMS 14 enhances tracing and the ICAL call for synchronous callout processing.

ICAL call enhancement to support control data for synchronous callout requests

In IMS 14, the ICAL call is enhanced to support the inclusion of control data in synchronous callout requests.

IMS application programs can make a synchronous callout request by using the ICAL DL/I call. The ICAL call sends a callout request to an IMS Connect client application and receives a response. The routing specification of a callout message is defined in the message's application interface block (AIB) and an OTMA destination descriptor. You can define up to 4095 destination descriptors in a DFSYDTx member of the IMS PROCLIB data set. When there are thousands of callout end points for the messages, using a limited number of OTMA destination descriptor entries for routing specification creates a challenge. Additionally, there is no easy way to include multiple Universally Unique Identifiers (UUID), SOAP headers, security token, or even user-provided routing information in the callout message.

To solve this problem, the ICAL call format accepts optional *control data*, which can be the URL for a port, UUID, user token, security information, or any other information. By using the control data field, IMS application programs can specify the routing information or other control data at run time when they issue the ICAL call.

The control data can consist of 1 to many control data items so that a number of services or operations can be specified on the same synchronous callout call. Each control data item starts with 4 bytes length field followed by a tag, data, and an end tag.

If you are using SOAP Gateway, you can use control data to specify the name of the XML converter you would like to use to process the request. In the control data, use the <DFSCNVTR> tags to specify an XML converter name.

The IMS Java dependent region (JDR) support provides an API for control data of the ICAL call.

The control data is designed for a callout request for an outbound message. The control data in a response message is not supported.

Coexistence considerations

IMS Connect Version 13 does not support control data, a feature of a higher-level IMS system, and that unpredictable results are to be expected in a coexistence situation where IMS Connect is at Version 13 and IMS is at Version 14. IMS 14 and IMS Connect Version 13 can coexist if the IMS Connect client does not request a callout message that contains control data.

Impact to installing and defining IMS

For SOAP Gateway messages, the XML converter name can be specified in the control data area to override the current converter name that IMS Connect would have used to process the message.

Impact to administering IMS

OTMA message prefix, IMS Connect message structures, and OTMA header fields used by IMS Connect are changed to support control data.

Impact to programming for IMS

When the DL/I ICAL call is used for synchronous callout requests, IMS application programs can specify the endpoint information or other routing specification for the callout message in the control data area of the ICAL call during run time when they issue the call.

IMS transactions that are written in Java and leverage the IMS Java dependent region resource adapter can also issue ICAL calls that include control data.

Trace enhancement for synchronous callout processing

IMS 14 makes diagnosing problems with OTMA synchronous callout processing easier by introducing new message ID types in the X'6701' log records that OTMA writes.

OTMA writes X'6701' log records for every type of message that is exchanged between OTMA and an OTMA client during the processing of synchronous callout requests and their associated responses. The types of message that are used during synchronous callout include the RESUME TPIPE request, the synchronous callout request, the acknowledgement from the client, the synchronous callout response, the acknowledgement from OTMA, and more.

In IMS 14, IMS introduces two new X'6701' record types for OTMA synchronous callout processing: YAKO and YNKO, to distinguish ACK and NAK messages that are sent by OTMA from ACK and NAK messages that are received by OTMA. Before IMS 14, IMS used YACK and YNAK for ACK and NAK messages, regardless of whether they were sent or received by OTMA.

Log record changes

The X'6701' log records are changed by this enhancement. The ID= field in the formatted log record for an ACK message that is sent by OTMA now displays a value of YAKO. The ID= field in the formatted log record for a NAK message that is sent by OTMA now displays a value of YNKO.

The ID values for log records for ACK and NAK messages that are received by OTMA remain unchanged.

Changes to troubleshooting for IMS

The usability of the X'6701' log records for OTMA synchronous callout processing is improved by the introduction of the YAKO and YNKO record types, which enable users to distinguish ACK and NAK messages that are sent by OTMA from ACK and NAK messages that are received by OTMA.

Chapter 12. IMS system enhancements

The enhancements to the IMS system in IMS 14 include enhancements to the IMS repository, user exits, the total cost of ownership, dependent regions, commands, and more.

The enhancements to the IMS 14 system are not specific to either the IMS Database Manager or the IMS Transaction Manager and might impact both.

Command enhancements overview

To support new and enhanced functions and improve the overall usability and serviceability of IMS, IMS 14 introduces new commands and enhances existing commands.

Most of the command enhancements are to type-2 commands in support the IMS strategy of enhancing the capability of single point of control (SPOC) applications that issue type-2 commands through the Operations Manager (OM) API or the REXX SPOC API.

Subsections:

- “APPC/IMS flood control command enhancements”
- “DBRC command enhancements” on page 184
- “DEDB alter command enhancements” on page 184
- “/DISPLAY command enhancements for ESAF external subsystem types” on page 184
- “Dynamic refresh enhancement for PWFI and WFI regions command enhancements” on page 184
- “FDBR in-doubt thread support for ESAF command enhancements” on page 184
- “HALDB support for 8 GB OSAM data sets enhancement” on page 185
- “IMS Connect command enhancements” on page 185
- “IMS management of ACBs” on page 186
- “IMS type-2 trace command enhancements” on page 187
- “MSC command enhancements” on page 187
- “OTMA command enhancements” on page 188
- “IMSRSC repository command enhancements” on page 189
- “PSB activation control ” on page 189
- “User exit command enhancements” on page 189

APPC/IMS flood control command enhancements

The new APPC/IMS flood control function enhances two commands:

- /DISPLAY ACT command (Commands)
- /PURGE command (Commands)

The information that is returned by the /DISPLAY ACT command (Commands) command is expanded to include a MAXC= output field that displays the

maximum number of APPC conversations that can be active before IMS starts queuing incoming APPC requests in 64-bit storage. If the displayed value is 0, APPC/IMS flood control is disabled.

The /PURGE command (Commands) command is enhanced to reject any APPC requests that are queued in 64-bit storage with the sense code TP_Not_Available_No_Retry.

DBRC command enhancements

The CHANGE.RECON command (Commands) is enhanced for DBRC migration.

The GENJCL.IC command (Commands) is enhanced to support the Database Image Copy 2 utility (DFSUDMT0) enhancement for zEnterprise Data Compression.

The following new DBRC command rebuilds the RECON DMB (data management block) Table record and fixes inconsistencies of the DMB numbers in various records in the RECON data set.

- REPAIR.RECON command (Commands)

DEDB alter command enhancements

The following DBRC commands are enhanced for IMS 14 to support the DEDB alter function:

- /DBRECOVERY command (Commands)
- DBDS (Fast Path) record fields (Commands)
- INIT.ADS command (Commands)
- INIT.DBDS command (Commands)
- UPDATE DB command (Commands)

/DISPLAY command enhancements for ESAF external subsystem types

In IMS 14, the output of the following /DISPLAY commands were modified to include ESSTYPE, which shows the type of external subsystem with which the External Subsystem Attach Facility (ESAF) is connected.

- /DISPLAY OASN SUBSYS command (Commands)
- /DISPLAY SUBSYS command (Commands)

Dynamic refresh enhancement for PWFI and WFI regions command enhancements

The following commands are enhanced to support dynamic refresh of PWFI and WFI regions.

- /PSTOP command (Commands)
- UPDATE PGM command (Commands)

FDBR in-doubt thread support for ESAF command enhancements

The following commands were modified to support the FDBR in-doubt thread support enhancement:

- /DISPLAY SUBSYS command (Commands)
- /DISPLAY OASN SUBSYS command (Commands)

HALDB support for 8 GB OSAM data sets enhancement

The following IMS and DBRC commands are enhanced for the HALDB support for 8-GB OSAM data sets enhancement:

- CHANGE.DB command (Commands)
- INITIATE OLREORG command (Commands)
- INIT.DB command (Commands)
- LIST.DB command (Commands) (For HALDB master records, the command output shows the maximum data capacity of OSAM data sets. See DB (HALDB) record fields (Commands)).

IMS Connect command enhancements

IMS 14 introduces a number of new and changed commands for IMS Connect.

The following type-2 commands are new for IMS Connect in IMS 14:

- CREATE IMSCON TYPE(IMSPLEX) command (Commands)
- DELETE IMSCON commands (Commands)
- DELETE IMSCON TYPE(DATASTORE) command (Commands)
- DELETE IMSCON TYPE(IMSPLEX) command (Commands)
- DELETE IMSCON TYPE(PORT) command (Commands)

The following type-2 commands are enhanced for IMS Connect in IMS 14:

- CREATE IMSCON TYPE(DATASTORE) command (Commands)
- CREATE IMSCON TYPE(PORT) command (Commands)
- QUERY IMSCON TYPE(CONFIG) command (Commands)
- UPDATE IMSCON TYPE(CONFIG) command (Commands)
- UPDATE IMSCON TYPE(DATASTORE) command (Commands)
- UPDATE IMSCON TYPE(IMSPLEX) command (Commands)
- UPDATE IMSCON TYPE(MSC) command (Commands)
- UPDATE IMSCON TYPE(ODBM) command (Commands)
- UPDATE IMSCON TYPE(PORT) command (Commands)
- UPDATE IMSCON TYPE(RMTIMSCON) command (Commands)
- IMS Connect type-2 commands (Commands)

The following IMS Connect commands are enhanced to support the setting and displaying of the timeout value for idle client connections (IDLETO):

- CREATE IMSCON TYPE(PORT) command (Commands)
- QUERY IMSCON TYPE(CONFIG) command (Commands)
- QUERY IMSCON TYPE(PORT) command (Commands)
- UPDATE IMSCON TYPE(CONFIG) command (Commands)
- UPDATE IMSCON TYPE(PORT) command (Commands)

The following commands are changed in IMS 14 to support cascading global RRS transactions when IMS Connect and IMS are on different LPARs:

Table 17. IMS Connect commands updated for cascaded transaction support

Type-2 commands	WTOR commands	z/OS modify commands
<ul style="list-style-type: none"> • CREATE IMSCON TYPE(DATASTORE) command (Commands) • QUERY IMSCON TYPE(DATASTORE) command (Commands) • QUERY IMSCON TYPE(CONFIG) command (Commands) • UPDATE IMSCON TYPE(CONFIG) command (Commands) • UPDATE IMSCON TYPE(DATASTORE) command (Commands) 	<ul style="list-style-type: none"> • VIEWDS command (Commands) • VIEWHWS command (Commands) 	<ul style="list-style-type: none"> • IMS Connect QUERY DATASTORE command (Commands) • IMS Connect QUERY MEMBER command (Commands)

IMS management of ACBs

Both IMS and DBRC commands are enhanced in IMS 14 to support the IMS management of ACBs.

Most notably, the IMPORT DEFN command is enhanced to support the activation of new and changed ACBs when the keyword SOURCE(CATALOG) is specified. IMS commands that support Online Change are modified to indicate that they do not support changing ACBs when ACBs are managed by IMS. Certain DBRC commands are enhanced to specify or display the IMS catalog that is managing the ACBs.

The following IMS commands were enhanced to support the IMS management of ACBs:

- CREATE PGM command (Commands)
- CREATE PGMDESC command (Commands)
- /DISPLAY ACT command (Commands)
- IMPORT DEFN SOURCE(CATALOG) command (Commands)
- INITIATE OLC command (Commands)
- INITIATE OLREORG command (Commands)
- /MODIFY command (Commands)
- QUERY DB command (Commands)
- QUERY MEMBER command (Commands)
- QUERY PGM command (Commands)
- QUERY PGMDESC command (Commands)
- /STOP REGION command (Commands)
- UPDATE PGM command (Commands)
- UPDATE PGMDESC command (Commands)

The following DBRC commands were enhanced to support the IMS management of ACBs:

- CHANGE.DBDS command (Commands)

- CHANGE.PART command (Commands)
- CHANGE.RECON command (Commands)
- INIT.DB command (Commands)
- INIT.DBDS command (Commands)
- INIT.PART command (Commands)
- INIT.RECON command (Commands)
- LIST.RECON command (Commands)
- NOTIFY.REORG command (Commands)
- RECON record fields (Commands)

IMS type-2 trace command enhancements

The following type-2 commands are added to support IMS type-2 traces:

- QUERY TRACE command (Commands)
- UPDATE TRACE command (Commands)

MSC command enhancements

The IMS 14 MSC enhancements introduce new and changed IMS commands.

The following IMS type-2 commands are new in IMS 14 for dynamic definition of MSC resources:

- CREATE LTERM command (Commands)
- CREATE MSPLINK command (Commands)
- CREATE MSLINK command (Commands)
- CREATE MSNAME command (Commands)
- DELETE LTERM command (Commands)
- DELETE MSPLINK command (Commands)
- DELETE MSLINK command (Commands)
- DELETE MSNAME command (Commands)

The following IMS commands are changed in IMS 14 for dynamic definition of MSC resources:

- QUERY LTERM command (Commands)
- QUERY MEMBER command (Commands)
- QUERY MSLINK command (Commands)
- QUERY MSNAME command (Commands)
- QUERY MSPLINK command (Commands)
- UPDATE MSPLINK command (Commands)

The following IMS commands are changed in IMS 14 for IMSRSC repository support for dynamically defined MSC resources:

- /CHANGE LINK command (Commands)
- CREATE LTERM command (Commands)
- CREATE MSLINK command (Commands)
- CREATE MSNAME command (Commands)
- CREATE MSPLINK command (Commands)
- DELETE DEFN command (Commands)

- /MSASSIGN command (Commands)
- QUERY LTERM command (Commands)
- QUERY MSLINK command (Commands)
- QUERY MSNAME command (Commands)
- QUERY MSPLINK command (Commands)
- UPDATE MSLINK command (Commands)
- UPDATE MSPLINK command (Commands)
- UPDATE MSNAME command (Commands)

OTMA command enhancements

Multiple OTMA enhancements in IMS 14 introduce changes to existing IMS and IMS Connect commands.

The QUERY OTMATI command (Commands) is enhanced to display conversation IDs.

The OTMA ACEE flood control enhancement introduces changes into the following commands and command topics:

- /DISPLAY OTMA command (Commands)
- /SECURE command (Commands)

The OTMA client bid security enhancement introduces changes into the following commands and command topics:

- Status and attributes for the /DISPLAY command (Commands)
- /DISPLAY OTMA command (Commands)
- /DISPLAY TMEMBER command (Commands)
- /SECURE command (Commands)

The following IMS commands were enhanced for OTMA tpipe parallelism in IMS 14:

- /DISPLAY TMEMBER command (Commands)
- CREATE IMSCON TYPE(DATASTORE) command (Commands)
- QUERY IMSCON TYPE(CLIENT) command (Commands)
- QUERY IMSCON TYPE(CONFIG) command (Commands)
- QUERY IMSCON TYPE(DATASTORE) command (Commands)
- UPDATE IMSCON TYPE(CONFIG) command (Commands)

The following IMS Connect WTOR commands were enhanced for OTMA tpipe parallelism in IMS 14:

- VIEWWDS command (Commands)
- VIEWHWS command (Commands)

The following IMS Connect z/OS MODIFY commands were enhanced for OTMA tpipe parallelism in IMS 14:

- IMS Connect QUERY DATASTORE command (Commands)
- IMS Connect QUERY MEMBER command (Commands)

IMSRSC repository command enhancements

The IMS 14 IMSRSC repository enhancements introduce various changed IMS commands:

The following IMS type-1 and type-2 commands are changed in IMS 14:

- /CHECKPOINT command (Commands)
- DELETE DEFN command (Commands)
- EXPORT command (Commands)
- IMPORT DEFN SOURCE(REPO | RDDS) command (Commands)
- QUERY DB command (Commands)
- QUERY DBDESC command (Commands)
- QUERY IMS command (Commands)
- QUERY PGM command (Commands)
- QUERY PGMDESC command (Commands)
- QUERY RTC command (Commands)
- QUERY RTCDESC command (Commands)
- QUERY TRAN command (Commands)
- QUERY TRANDESC command (Commands)
- UPDATE IMS command (Commands)

PSB activation control

The following commands are enhanced to support the new OPTION(UPDATEPSB) and OPTION(REFRESHPSB) for the IMPORT DEFN SOURCE(CATALOG) command.

- IMPORT DEFN SOURCE(CATALOG) command (Commands)
- QUERY MEMBER command (Commands)

User exit command enhancements

The following commands are enhanced to support the new IMSMON and AOIE exit types.

- QUERY USEREXIT command (Commands)
- REFRESH USEREXIT command (Commands)

Dynamic refresh enhancement for PWFI and WFI regions

In IMS 14, the dynamic refresh enhancement for PWFI and WFI regions enables you to refresh all instances of pseudo-wait-for-input (PWFI) or wait-for-input (WFI) application programs that are running in a dependent region by issuing a single UPDATE PGM command.

Before IMS 14, when a change was made to an application program, and the program was scheduled and waiting for work in one or more PWFI (or WFI) regions, the regions had to be stopped in order to reload the program. It was difficult to find all the regions to stop and it was time consuming to restart each region.

IMS 14 introduces a new START(REFRESH) filter for the UPDATE PGM command. By using the UPDATE PGM START(REFRESH) command, all IMS users of PWFI regions or non-PWFI regions running WFI=YES transactions will receive the following benefits:

- Operational considerations are reduced due to dynamically refreshing PWFI (or WFI) regions
- Regions need not be terminated and are available to do other work
- Program gets rescheduled when a new work is available

Changes to administering IMS

Before IMS 14, when application program changes were made in IMS PGMLIB, you had to manually stop all PWFI regions or regions running WFI=YES transactions where the program was scheduled so that the new changes could be obtained when the program was scheduled again. It was difficult to find all the regions to stop, and it was time consuming to restart each region.

In IMS 14, you can use the UPDATE PGM START(REFRESH) command to *post* all the regions where the program is scheduled. When a region is posted, a QC status will be returned to the application program so that the program will stop processing any more messages.

The UPDATE PGM START(REFRESH) command is supported for programs scheduled in the following region types:

- MPP pseudo-wait-for-input (PWFI) regions in which the program is scheduled and the program is not preloaded by the DFSMPLxx PROCLIB member
- JMP PWFI regions in which the specified program name is scheduled
- MPP, JMP, and message-driven BMP regions in which the program is scheduled and that are running a transaction defined as WFI=YES

If the program is waiting for work in the dependent region, the UPDATE PGM START(REFRESH) command returns a QC status to the application program when the command is processed. If the program is scheduled and processing work, the QC status code is returned to the application program when the current work is completed and before the next message is processed. When the program is rescheduled in a region when new work is available, the region will pick up any program changes from PGMLIB.

The UPDATE PGM START(REFRESH) command is not supported for the following region types:

- MPP regions where the program is loaded by the DFSMPLxx PROCLIB member
- IFP regions
- JBP regions
- Non-message-driven BMP regions

Changes to troubleshooting for IMS

This enhancement adds new abend code 0330 to IMS 14.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Changes to commands

The UPDATE PGM command is enhanced to add the START(REFRESH) keyword to post all PWFI regions and regions running WFI=YES transactions and have the specified program name scheduled.

For a list of the commands that are changed for this enhancement, see the online version of this information under *Release Planning* in the IBM Knowledge Center.

External Subsystem Attach Facility enhancements

In IMS 14, the External Subsystem Attach Facility (ESAF) is enhanced to support the specification of additional subsystem types and a new exit routine that external subsystems can use to improve thread processing.

Also, for users of the Fast Database Recovery facility, IMS 14 provides a new sample exit routine that facilitates the clean up of in-doubt units of work in the external subsystem. For more information about the in-doubt thread support, see “FDBR in-doubt thread support for ESAF” on page 138.

ESAF Associate Thread exit routine enhancement

In IMS 14, the External Subsystem Attach Facility is enhanced to support the ESAF Associate Thread exit routine. This exit routine can be used by external subsystems to improve processing during the creation and termination of threads by pooling dependent region threads.

The optional ESAF Associate Thread exit routine is activated during signon processing and is called prior to the Signon exit routine. The Associate Thread exit routine is not called if it is not specified by the external subsystem.

Migration considerations

To use the ESAF Associate Thread exit routine, the external subsystem must provide it and specify the name of the exit routine in the External Subsystem Module Table (ESMT).

When the exit routine is used, IMS loads it in the dependent region.

Changes to installing and defining IMS

To use the Associate Thread exit routine, the external subsystem must provide the exit routine in the DFSESL library and define the exit routine in an ESMT. If the exit routine is not used, the ESMT is not changed.

Changes to exit routines

The Associate Thread exit routine can affect the processing of the Create Thread and Terminate Thread exit routines, depending on the implementation by the external subsystem.

ESAF subsystem type enhancement

In IMS 14, you can now identify the attached subsystem as IBM MQ, WebSphere Optimized Local Adapter, or Db2 for z/OS by using the SST= keyword parameter in the SSM member of the IMS PROCLIB data set.

In earlier versions of IMS, although you could attach other types of subsystems to IMS by using ESAF, the only subsystem identifier that IMS supported was DB2 for z/OS.

Additionally, commands and log records now show SST values for improved ESAF usability.

Log record changes

The following log records are new or changed by this enhancement in IMS 14:

- The type X'56' log record with subcode X'000009' now includes the type of external subsystem that connected or disconnected.
- The type X'40' log record with subcode X'31' for a subsystem index (SIDX) checkpoint now includes the type of external subsystem.

Impact to installing and defining IMS

In the PROCLIB member that is specified with the SSM= parameter for the IMS startup procedure, you can specify the type of an external subsystem with the SST= parameter. The supported ESS types you can specify on the SST= parameter are:

DB2

Db2 for z/OS

MQ IBM MQ

WOLA

WebSphere Optimized Local Adapter

Restriction: SST= is a keyword-only parameter. It cannot be specified as a positional parameter.

Recommendation: If you are currently using ESAF to connect to an MQ or WebSphere Optimized Local Adapter subsystem, update your SSM definitions to include SST=MQ or SST=WOLA.

Attention: If you define the subsystem in positional format, IMS displays DB2 as the ESS type in both command responses and DFS messages even if you are using MQ or WOLA.

Impact to administering IMS

The /DISPLAY SUBSYS and /DISPLAY OASN SUBSYS commands are enhanced to return the external subsystem type. You can use this information to automate in-doubt work handling during recovery.

Impact to troubleshooting for IMS

For a list of the messages and codes that are changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Impact to commands

The /DISPLAY SUBSYS and /DISPLAY OASN SUBSYS are enhanced with new output information. Both commands now display the type of external subsystem (ESSTYPE) in addition to the information that was returned in previous versions.

The command syntax and input values that are allowed for these commands are not affected. Only the command output is enhanced.

For a list of the commands that are changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

ESAF connection pooling

In IMS 14, the **DB2JCC_ESAF_THREAD_NOTIFICATION= YES | NO** parameter is added to the DFSJVMEV member of the IMS PROCLIB data set.

ESAF provides support for additional subsystems and exit routines. Previously, for every new connection from an IMS program, ESAF connections were created and then deleted. Therefore, an overhead was caused in the DB2 ESAF connection process. This enhancement reduces connection overhead by enabling ESAF to support DB2 connection pooling for up to 50 pool entries from a persistent JVM, MPP, BMP, or IFP region. The **DB2JCC_ESAF_THREAD_NOTIFICATION= YES | NO** parameter is also added to the DFSJVMEV member of the IMS PROCLIB data set. This parameter improves the efficiency of DB2 JCC driver in dynamically preparing for SQL statement execution.

This enhancement is available with DB2 for z/OS Version 11 APARs PI61982 and PI61983, DB2 for z/OS JCC driver Version 4.19.66 APAR PI59547, and Version 3.69.66 APAR PI59538.

To enable this enhancement, set the external subsystem module table (ESMT) to DSNMIN20 in the **SSM=** parameter.

Changes to installing and defining IMS

The **DB2JCC_ESAF_THREAD_NOTIFICATION= YES | NO** parameter is added to the DFSJVMEV member of the IMS PROCLIB data set. For more information, see DFSJVMEV (JVM environment settings member) (System Definition).

IMS 64-bit storage manager enhancement

IMS 14 has a new internal 64-bit storage manager to help reduce below-the-bar storage constraints and facilitate the use of 64-bit storage by IMS components. The IMS 64-bit storage manager makes it easier for IMS components to move storage above the bar for storage constraint relief.

Migration considerations

In IMS 14, the QUERY POOL command is enhanced to provide more accurate feedback for syntax checking for the TYPE and SHOW keywords.

In earlier versions of IMS, when building QUERY POOL output with completion code text, the output was built with the CCText output header right-justified instead of left-justified. In IMS 14 output, the CCText header is now left-justified as it is with other type-2 commands.

Coexistence considerations

If the QUERY POOL command is routed to an IMS 14 system and an IMS 13 or IMS 12 system at the same time, the QUERY POOL command will have mixed results.

Log record changes

The X'4519' statistics log record includes information about the pools that are managed by the IMS 64-bit storage manager. The X'4519' log record is mapped by the DFSL4519 macro.

IMS Connect enhancements

In IMS 14, enhancements are made to IMS Connect to improve reliability, availability, serviceability, and security. New type-2 IMS Connect commands are also added.

IMS Connect command enhancements

In IMS 14, new type-2 IMS Connect commands are added and existing IMS Connect type-2 commands are enhanced so that users can change configuration definitions during IMS Connect runtime. These commands make it possible to change IMS Connect definitions and put them in effect without having to restart IMS Connect.

IMS Connect type-2 command enhancements improve the task of managing IMS Connect resources and support the IMS strategy of providing consistent new functionality through type-2 commands.

Migration considerations

Automated programs that issue IMS Connect type-2 commands might need to be modified to handle certain type-2 commands that are now processed synchronously in IMS 14. IMS Connect type-2 commands that are processed synchronously return the full results of the command in the command response, instead of returning part of the response to the system console. The commands that are processed synchronously also include new completion codes. The following commands are processed synchronously in IMS 14:

- CREATE IMSCON TYPE(DATASTORE) command
- CREATE IMSCON TYPE(PORT) command
- UPDATE IMSCON TYPE(DATASTORE) START(COMM) command
- UPDATE IMSCON TYPE(IMSPLEX) START(COMM) command
- UPDATE IMSCON TYPE(MSC) START(COMM) command
- UPDATE IMSCON TYPE(ODBM) START(COMM) command
- UPDATE IMSCON TYPE(PORT) START(COMM) command
- UPDATE IMSCON TYPE(RMTIMSCON) START(COMM) command

Coexistence considerations

IMS 14 introduces new and changed IMS type-2 commands for IMS Connect.

If a type-2 command is broadcast to an IMSplex in which multiple versions of IMS Connect coexist, instances of IMS Connect from earlier versions reject commands

that include keywords that they do not support and ignore commands that they do not recognize. If the type-2 command is routed specifically to an IMS Version 12 instance of IMS Connect, the Operations Manager (OM) rejects the command.

Requirements

Although no special hardware or software is required to use the IMS Connect type-2 commands, IMS Connect must be configured with an Operations Manager (OM) and a Structured Call Interface (SCI) because all type-2 commands must come into IMS Connect through the OM API or the REXX SPOC API.

To issue IMS Connect type-2 commands, a Common Service Layer with OM and SCI is required.

Impact to commands

The new type-2 commands enable the following tasks:

- Dynamically create IMSplex definitions with the CREATE IMSCON TYPE(IMSPLEX) command
- Dynamically delete data store definitions with the DELETE IMSCON TYPE(DATASTORE) command
- Dynamically delete IMSplex definitions with the DELETE IMSCON TYPE(IMSPLEX) command
- Dynamically delete port definitions with the DELETE IMSCON TYPE(PORT) command

The following type-2 commands are enhanced to process synchronously to improve usability:

- CREATE IMSCON TYPE(DATASTORE) command
- CREATE IMSCON TYPE(PORT) command
- UPDATE IMSCON TYPE(DATASTORE) START(COMM) command
- UPDATE IMSCON TYPE(IMSPLEX) START(COMM) command
- UPDATE IMSCON TYPE(MSC) START(COMM) command
- UPDATE IMSCON TYPE(ODBM) START(COMM) command
- UPDATE IMSCON TYPE(PORT) START(COMM) command
- UPDATE IMSCON TYPE(RMTIMSCON) START(COMM) command

The following type-2 commands are enhanced to enable the updating of resource definitions:

- UPDATE IMSCON TYPE(DATASTORE) SET() command
- UPDATE IMSCON TYPE(PORT) SET() command

Modifications that are made to IMS Connect resources by using these commands are not saved across restarts of IMS Connect.

For a list of the commands that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

Impact to exit routines

In IMS 14, IMS Connect continues to add and enhance commands to allow users to create, update, and delete resource definitions during IMS Connect run time.

New events are added for the IMS Connect Event Recorder exit routine (HWSTECL0) to record the changes to these resources. Only changes to resources that can be defined in the IMS Connect configuration have an event. For example, a change to the keep alive value of a port resource will trigger a resource change event, but stopping the port resource will not trigger a resource change event, although it will still trigger other existing events.

An event is issued for a resource change as a result of one of the following:

- Resource is created by the CREATE command
- Resource is deleted by the DELETE command
- Resource is updated by the UPDATE command

For a specific resource change event, the resource and its attribute changes are set in a new resource information block (HWSRSIB), which is passed to the event recorder exit in the event recorder parameter list (ERPL). The new RSIB is similar to the existing TCPIB and DSIB, and it is passed to the event recorder exit in the same way.

Recording IMS Connect RACF statistics for IMS DB connections

In IMS 14 with APAR/PTF PI80202/UI47653, you can enable RACF statistics to be recorded when IMS Connect issues the RACF call RACROUTE REQUEST=VERIFY to authenticate Open Database Manager (ODBM) client connections to IMS DB. You can enable RACF statistics by specifying ODRACFST=Y in the ODACCESS statement of the HWSCFGxx configuration member.

After you enable RACF statistics, the statistics, including the last date and time a user was authenticated to access data in IMS DB, are recorded by RACF no more than once per day to a system management facility (SMF) data set or log stream. The SMF data set or log stream that is used to record the RACF statistics is specified in the RACF configuration.

You can use the RACF statistics to enforce security policies, such as defining how frequently user passwords need to be changed or enabling access to be automatically revoked for inactive users.

If you enable RACF statistics, the STAT=ASIS parameter is used for the RACROUTE REQUEST=VERIFY call to enable the options that are specified on the RACF command SETROPTS to take effect.

Changes to installing and defining IMS

The ODRACFST= parameter is added to the ODACCESS statement of the HWSCFGxx member of the IMS PROCLIB data set. You can use this parameter to define whether RACF statistics and messages are enabled and updated for the execution of the RACF call RACROUTE REQUEST=VERIFY.

Changes to administering IMS

IMS Connect administrators can use the `QUERY IMSCON TYPE(CONFIG)` and `UPDATE IMSCON TYPE(CONFIG)` commands to query and update the value of the RACF statistics option.

Changes to commands

The `SHOW(ODRACFST)` keyword is added to the `QUERY IMSCON TYPE(CONFIG)` command to allow you to query whether RACF statistics are enabled.

The `SET(ODRACFST(ON))` and `SET(ODRACFST(OFF))` options are added to the `UPDATE IMSCON TYPE(CONFIG)` command to allow you to update the value of the RACF statistics option.

For a list of the commands that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

Recording IMS Connect RACF statistics for IMS TM connections

In IMS 14 with APAR/PTF PI89512/UI52498, you can enable RACF statistics to be recorded when IMS Connect issues the RACF call `RACROUTE REQUEST=VERIFY` to authenticate Open Transaction Manager Access (OTMA) client connections to IMS TM. You can enable RACF statistics by specifying `TMRACFST=Y` in the `HWS` statement of the `HWSCFGxx` configuration member.

After you enable RACF statistics, the statistics, including the last date and time a user was authenticated to access data in IMS TM, are recorded by RACF no more than once per day to a system management facility (SMF) data set or log stream. The SMF data set or log stream that is used to record the RACF statistics is specified in the RACF configuration.

You can use the RACF statistics to enforce security policies, such as defining how frequently user passwords need to be changed or enabling access to be automatically revoked for inactive users.

If you enable RACF statistics, the `STAT=ASIS` parameter is used for the `RACROUTE REQUEST=VERIFY` call to enable the options that are specified on the RACF command `SETROPTS` to take effect.

Changes to installing and defining IMS

The `TMRACFST=` parameter is added to the `HWS` statement of the `HWSCFGxx` member of the `IMS PROCLIB` data set. You can use this parameter to define whether RACF statistics and messages are enabled and updated for the execution of the RACF call `RACROUTE REQUEST=VERIFY` for OTMA client connections to IMS TM.

Changes to administering IMS

IMS Connect administrators can use the `QUERY IMSCON TYPE(CONFIG)` and `UPDATE IMSCON TYPE(CONFIG)` commands to query and update the value of the `TMRACFST=` parameter.

Changes to commands

The SHOW(TMRACFST) keyword is added to the QUERY IMSCON TYPE(CONFIG) command to allow you to query whether RACF statistics are enabled.

The SET(TMRACFST(ON)) and SET(TMRACFST(OFF)) options are added to the UPDATE IMSCON TYPE(CONFIG) command to allow you to update the value of the RACF statistics option.

For a list of the commands that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

Timeout option for idle connections

In IMS 14, you can configure IMS Connect to close an IMS TM client connection if it remains idle for a specified period of time after the prior client interaction completes. You specify the time period on the IDLETO parameter in the TCPIP configuration statement.

When the IDLETO parameter is specified, the timer for idle connections activates only after the first message is received on a client connection. After the IDLETO timer is activated, it resets each time a new message is received on the connection before the timeout interval expires.

You can define both a global idle timeout value and a port-specific timeout value. The global idle timeout, which is specified as IMS Connect configuration attribute, functions as a default for connections on all ports that are used for IMS TM communications. The port-specific idle timeout value is specified in the definition of a port and overrides the global default.

You can set or update the IDLETO values by using the appropriate CREATE IMSCON or UPDATE IMSCON commands or by using the TCPIP statement in the IMS Connect HWSCFGxx configuration member.

You can view current IDLETO specifications by using the QUERY IMSCON commands.

The timeout option provided by the IDLETO parameter is in addition to the timeout options provided by the existing TIMEOUT parameter. The TIMEOUT parameter defines both the timeout interval between the creation of a new socket connection and the first message, and the timeout interval between the receipt of a message and the response to that message from IMS.

The IMS Connect Event Recorder exit routine (HWSTECL0) is also updated to include the IDLETO value when port resources are created.

Changes to installing and defining IMS Connect

The IDLETO parameter provides a new timeout option for socket connections that you can specify when configuring an IMS Connect instance.

Changes to administering IMS

IMS Connect administrators can review the traffic on socket connections and adjust the IDLETO timeout value to minimize the amount of IMS Connect resources that idle socket connections use.

Changes to commands

The following commands were updated to support the setting and displaying of IDLETO values:

- CREATE IMSCON TYPE(PORT)
- QUERY IMSCON TYPE(PORT)
- QUERY IMSCON TYPE(CONFIG)
- UPDATE IMSCON TYPE(PORT)
- UPDATE IMSCON TYPE(CONFIG)

For a list of the commands that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

Changes to exit routines

IMS Connect adds a new VAR_IDLETO field to event X'03' for the IMS Connect Event Recorder exit routine (HWSTECL0).

IMS Connect WTOR command interface option enhancement

In IMS 14 with APAR/PTF PI85790/UI50870, if you do not use WTOR commands with IMS Connect, you can disable the WTOR command interface for IMS Connect so that the WTOR input messages no longer take up space in the system console.

You can disable or enable the WTOR command interface by specifying the **WTORCMD** parameter in the HWS statement of the IMS Connect HWSCFGxx configuration member.

Changes to installing and defining IMS

The HWS statement of the HWSCFGxx configuration member is enhanced with the **WTORCMD** parameter to allow you to disable or enable the WTOR command interface for IMS Connect.

Changes to administering IMS

To be able to monitor the system console more easily, system administrators can disable the WTOR command interface for IMS Connect instances that do not issue WTOR commands by specifying **WTORCMD=N** in the HWS statement of the HWSCFGxx member. After the WTOR command interface is disabled, messages from the IMS Connect instances for which **WTORCMD=N** is specified are not displayed on the system console.

IMSRSC repository enhancements

IMS 14 introduces a variety of enhancements to the IMSRSC repository and the management of IMS resource definitions in general.

Automatic export to the IMSRSC repository at IMS checkpoints

In IMS 14, when the IMSRSC repository is enabled and either `AUTOEXPORT=AUTO` is explicitly specified and not defaulted to or `AUTOEXPORT=REPO` is specified in the `DYNAMIC_RESOURCES` section of the `DFSDFxxx PROCLIB` member, then at each IMS checkpoint, IMS automatically exports any online resource definitions that have been created or updated since the last successful export (manual or autoexport) to the IMSRSC repository.

Migration considerations

If `AUTOEXPORT=AUTO` is specified in the `DFSDFxxx` member and IMS is enabled with the IMSRSC repository, autoexport to the IMSRSC repository is enabled at system checkpoints. To disable the autoexport to the IMSRSC repository function during migration to IMS 14, modify the `DFSDFxxx` member in one of the following ways:

- Remove the `AUTOEXPORT=` specification so that `AUTOEXPORT` will default to `AUTO` and not trigger the autoexport to the IMSRSC repository. If IMS also has system RDDS, then autoexport will be done to the RDDS.
- Modify to `AUTOEXPORT=NO` so that no autoexport is enabled.
- Modify to `AUTOEXPORT=RDDS` so that autoexport is only to the system RDDS and not the IMSRSC repository.

Remember: Autoexport to the IMSRSC repository is not enabled if `AUTOEXPORT=AUTO` is defaulted to. You must explicitly specify autoexport in the `DFSDFxxx` member.

If you have `AUTOEXPORT=AUTO` defined with IMSRSC repository enabled and no system RDDS defined, IMS Version 13 and earlier will disable autoexport (`AUTOEXPORT=N`) since no system RDDS data sets are defined. A `DFS3374W` message is issued to indicate that autoexport is disabled.

If you have `AUTOEXPORT=AUTO` explicitly specified and not defaulted to with IMSRSC repository enabled and no system RDDS defined, IMS 14 will, at the end of the next checkpoint, export the changed definitions automatically since the last export to the IMSRSC repository.

If you have `AUTOEXPORT=AUTO` defined with IMSRSC repository enabled and system RDDS defined, IMS Version 13 and earlier will export the changed definitions automatically to the system RDDS at the end of the next checkpoint.

If you have `AUTOEXPORT=AUTO` defined with IMSRSC repository defined and system RDDS defined, IMS 14 will, at the end of the next checkpoint, export the changed definitions automatically since the last export to the IMSRSC repository. IMS will also export the changed definitions automatically to the system RDDS. The automatic export to the system RDDS will occur even if the automatic export to the IMSRSC repository did not succeed.

Coexistence considerations

Automatic export to the IMSRSC repository is supported only by an IMS 14 Resource Manager (RM) or later. Any lower-level RM systems that receive a request to update the repository for an automatic export will fail. IMS Version 12 APAR PI27283 and IMS Version 13 APAR PI27285 are open for the coexistence to

| not support RM to handle the AUTOEXPORT parameter. It is recommended that
| you enable automatic export to the IMSRSC repository only after all RM systems
| have been migrated to IMS 14.

Log record changes

The following log records are changed by this enhancement:

X'22' Modified to add subtype X'51'.

X'4098'

| With APAR PI48036 (PTFs UI40046 and UI40047), modified to add one byte
| to the end of the record. This byte is added to the record only if Extended
| Recovery Facility (XRF) and dynamic definition for Multiple Systems
| Coupling (MSC) resources are enabled. This byte contains MSC resource
| bits that indicate MSC resources were created or changed since the last
| automatic export. This information is used in an XRF complex if the
| alternate IMS system takes over before the active IMS system automatically
| exported MSC resources to the MSC repository. After the alternate IMS
| system completes takeover, the bits are used by the alternate IMS system to
| automatically export the MSC resources to the MSC repository.

Restrictions

The **QRY SHOW(EXPORTNEEDED)** command is supported only when IMS is enabled with the IMSRSC repository.

The **QRY SHOW(EXPORTNEEDED)** command is not supported with other SHOW parameters.

Changes to installing and defining IMS

Automatic export to the IMSRSC repository can be enabled by explicitly specifying (not by defaulting to) **AUTOEXPORT=AUTO** or **AUTOEXPORT=REPO** in the **DFSDFxxx** member of the IMS PROCLIB data set.

Changes to troubleshooting for IMS

The following messages are new for this enhancement:

- DFS4370I
- DFS4390I
- DFS4391E

For a list of the messages and codes that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Changes to commands

The **QUERY IMS** command will show the following values for **AUTOEXPORT** status:

- *N* if autoexport is disabled
- *Y* if autoexport is enabled to RDDDS, the IMSRSC repository, or both

The UPDATE IMS command is enhanced to turn off automatic export to both the IMSRSC repository and RDDS if both are enabled.

For a list of the commands that are changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

Changes to utilities

The DFSURCL0 utility with **EXPORTNEEDED** specified will process any X'22' subtype X'51' autoexport log records in the IMS log before creating the non-system RDDS with the resource definitions that need to be exported to the IMSRSC repository.

Create RDDS from Log Records utility (DFSURCL0) enhancement

In IMS 14, the Create RDDS from Log Records utility (DFSURCL0) is enhanced to create a non-system RDDS that contains only the resource and descriptor definitions that have not been exported to the IMSRSC repository.

This capability is useful when you must cold start IMS from the IMSRSC repository after a shutdown or abnormal termination and not all the created or modified resource definitions since the last export have been exported to the repository.

The non-system RDDS that is generated from the DFSURCL0 utility with the **EXPORTNEEDED** parameter can be used, before or after the IMS cold start, as input to:

- The RDDS to Repository utility (CSLURP10) to write the definitions to the repository before IMS is cold started.
- The **IMPORT** command after IMS cold starts from the repository so that IMS has the same resource definitions as before the cold start.

Migration considerations

Run the same version of the DFSURCL0 utility as the version of the IMS that produced the IMS logs being used as input.

For example, use the IMS 14 DFSURCL0 utility with IMS 14 log data sets. If you run with mixed versions, the results can be unpredictable.

Coexistence considerations

When you specify the **EXPORTNEEDED** control statement on the Create RDDS from Log Records utility (DFSURCL0), it is recommended that you run the same version of the DFSURCL0 utility as the version of the IMS that produced the IMS logs being used as input. For example, use the IMS 14 DFSURCL0 utility with IMS 14 log data sets. Otherwise, the results can be unpredictable.

Log record changes

No log records are introduced or changed by this enhancement in IMS 14. The DFSURCL0 utility with the **EXPORTNEEDED** parameter will process the checkpoint log records and any X'22' log records in the IMS log before creating the non-system RDDS with the resource definitions that need to be exported to the IMSRSC repository. Any X'22' log records in the IMS log that result from automatic export to the repository will also be processed.

Changes to administering IMS

By using the DFSURCL0 utility, you can create a non-system RDDS data set that contains only the resource definitions that have not been exported to the repository.

If an RDDS was created by the Create RDDS from Log Records utility (DFSURCL0) to hold any new or modified resources that were not exported to the IMSRSC repository before IMS was shut down, you can either specify that RDDS on the IMPORT command after IMS cold starts from the IMSRSC repository, or use that RDDS as input to the CSLURP10 utility.

Changes to utilities

The Create RDDS from Log Records utility (DFSURCL0) is enhanced to add the optional keyword parameter EXPORTNEEDED on the utility control statement.

DELETE DEFN command enhancement for IMS IDs

In IMS 14, the DELETE DEFN command is enhanced to support deleting an IMS ID from the IMSRSC repository.

With APAR PI55356 (PTF UI39067), when an IMS is leaving the IMSplex or when an IMS ID is added to the IMSRSC repository in error, the IMS ID information can be deleted from the IMSRSC repository using the DELETE DEFN TARGET(REPO) TYPE(IMSIDMBR) command.

Coexistence considerations

The DELETE DEFN command is an existing type-2 command that can be issued to one or more IMS systems from OM API and is processed by the command master IMS. The DELETE DEFN TYPE(IMSIDMBR) FOR(IMSID(imsid)) command is only supported by a V14 or higher IMS systems. If the DELETE DEFN TYPE(IMSIDMBR) FOR(IMSID(imsid)) command is routed to a lower level IMS system the command will fail with an error.

Changes to commands

For a list of the commands that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

QUERY command enhancement for unexported resources

In IMS 14 with DRD enabled with IMSRSC repository, you can use the new EXPORTNEEDED keyword on the QUERY *rsc_type* commands, where *rsc_type* can be DB, DBSESC, PGM, PGMDESC, RTC, RTCDESC, TRAN, or TRANDESC, to identify the runtime resource definitions that have been created or updated since the last successful export to the IMSRSC repository. This function can be issued periodically to determine whether any new or modified resource definitions need to be written to the repository so that an EXPORT DEFN command can then be issued to save the changes to the repository.

The EXPORTNEEDED keyword is supported on the QUERY command for the following resource and descriptor types:

- Databases (QUERY DB)
- Database descriptors (QUERY DBDESC)

- Programs (QUERY PGM)
- Program descriptors (QUERY PGMDESC)
- Routing codes (QUERY RTC)
- Routing code descriptors (QUERY RTCDESC)
- Transactions (QUERY TRAN)
- Transaction descriptors (QUERY TRANDESC)

Changes to administering IMS

This enhancement provides system administrators and operators the ability to identify any new or modified runtime resource definitions that have not been exported to the IMSRSC repository.

Changes to commands

This enhancement changes the behavior of the following type-2 commands

- QUERY DB
- QUERY DBDESC
- QUERY PGM
- QUERY PGMDESC
- QUERY RTC
- QUERY RTCDESC
- QUERY TRAN
- QUERY TRANDESC

For a list of the commands that are changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

IMS type-2 trace enhancement

In IMS 14, the IMS type-2 trace enhancement provides a new type-2 table trace that can be activated at IMS startup or while IMS is active. The trace tables can be updated with a different volume level while IMS is up, and they can be queried to gather diagnostic information.

Changes to installing and defining IMS

A new TRCLEV statement can be specified in the <DIAGNOSTICS_STATISTICS> section of the DFSDFxxx PROCLIB member to set up the type-2 trace tables and specify the trace volume level and the size.

Changes to troubleshooting for IMS

For a list of the messages and codes that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Changes to commands

The type-2 QUERY TRACE and UPDATE TRACE commands are provided to display information from the trace tables and to update the tables.

| For a list of the commands that are new or changed for this enhancement, see the
| online version of this information under Release Planning in the IBM Knowledge
| Center.

Reduced total cost of ownership enhancements

Internal changes made in IMS 14 can improve efficiency and reduce MIPS usage.

IMS 14 includes the following internal enhancements:

- Exploit z9 instructions where appropriate
- Allow program-to-program switch messages inserted from BMPs to be selected for shared queues local-first processing
- Add 32K and 48K buffer sizes in CQS interface buffer manager
- BPETIMER CANCEL service optimization
- Misc Logger Performance Fixes
- BPESVC service “quick path” for OTMA Callable Interface
- Internal optimization to OTMA “find destination” service
- OTMA send-only with reroute optimization

zIIP utilization enhancement

Note:

This information provides only general descriptions of the types and portions of workloads that are eligible for execution on Specialty Engines (for example, zIIPs, System z Application Assist Processors (zAAPs), and Integrated Facility for Linux (IFL) (“SEs”). IBM authorizes customers to use IBM SEs only to execute the processing of Eligible Workloads of specific Programs expressly authorized by IBM as specified in the “Authorized Use Table for IBM Machines” provided at www.ibm.com/systems/support/machine_warranties/machine_code/aut.html (“AUT”). No other workload processing is authorized for execution on an SE. IBM offers SEs at a lower price than General Processors/Central Processors because customers are authorized to use SEs only to process certain types and/or amounts of workloads as specified by IBM in the AUT.

In IMS Version 14 (“V14”), the IMS Common Queue Server (“CQS”) is added to the set of IMS address spaces that can utilize IBM System z zIIP specialty engines. In V14, certain processing in the IMS CQS address space as specified below may, when enclave SRB execution is enabled by a user, be executed under enclave service request blocks (“SRBs”). Enclave SRB execution may be enabled as unconditional;¹ or it may be enabled as conditional, based on the presence of at least one online zIIP during address space initialization. When enclave SRB execution is enabled, V14 will direct z/OS to authorize such work to be processed on an available zIIP.

Except as otherwise described below, the following IMS Connect, ODBM, and CQS processing can execute under an enclave SRB in V14:

1. In an unconditional status, the user may enable such processing to be executed in enclave SRB mode when no zIIP is online or part of the user's System z environment. But, of course, no zIIP utilization could result. A user might request enclave SRB execution even in the absence of zIIPs, to allow the system to project how much work is eligible to run on zIIP and probably would if there were zIIP capacity available. See the PROJECTCPU parameter of the IEAOPTxx SYS1.PARMLIB member in z/OS *MVS Initialization and Tuning Reference* (SA23-1380) for further information.

- Processing of IMS Connect address space SOAP message threads for SOAP messages arriving via TCP/IP
- Processing of IMS Connect and ODBM address space Distributed Relational Database Architecture™ (“DRDA”) threads for DRDA requests arriving via TCP/IP
- Processing of IMS Connect address space Multiple System Coupling (“MSC”) threads for MSC messages arriving via TCP/IP
- Processing of IMS Connect address space Intersystem Communication (“ISC”) threads for ISC messages arriving via TCP/IP
- Processing of ODBM address space threads for requests arriving through the CSLDMI API.
- Processing of CQS address space threads that process CQSREAD and CQSBrowse requests (CQSREAD and CQSBrowse are requests used by CQS clients such as the IMS control region to read and browse data in the CQS message queue and resource structures)

Note that any user exits called by the above processing will not execute under an enclave SRB. User exits are always given control in task control block (TCB) mode, and such exit instructions are not authorized to be processed on a zIIP. Also note that certain processing cannot, due to technical restrictions, execute under enclave SRBs. Such processing includes calling z/OS Resource Recovery Services (“RRS”), IMS DL/I call processing, and z/OS supervisor calls (“SVCs”). IMS switches from SRB mode into TCB mode to perform such processing, and thus such processing will not execute on a zIIP.

Documentation changes

The following publications contain new or changed topics for this enhancement.

Table 18. Links to new or changed topics for this enhancement.

Publication	Links to topics
<i>Release planning</i>	• zIIP utilization hardware requirements (Release Planning)
<i>System definition</i>	• BPE configuration parameter member of the IMS PROCLIB data set (System Definition)

User exit enhancements

The user exit enhancements in IMS 14 include a new IMS Monitor exit, user exits that are now refreshable, and the ability to call multiple exit routines from a single exit.

IMS Monitor (IMSMON) user exit enhancement

IMS 14 introduces a new exit, the IMS Monitor (IMSMON) user exit, which provides a clearly differentiated programming interface (CDPI) for customers and vendors to capture the same data that is available to the IMS Monitor.

The IMS Monitor user exit provides access to the IMS Monitor data without the need to modify IMS code. The exit is called during IMS initialization, IMS termination, and anytime an IMS Monitor log record is written, regardless whether the IMS Monitor is on or off

Restrictions

IMSMON exit routines must register interest in the SLOG codes (field name SL_CODE) that they are to be called for. However, because multiple user exit routines might be defined for the IMSMON user exit, and other routines could register interest in more SLOG codes, a user exit routine might be driven for SLOG codes for which it does not register interest. The exit routines perform this registration when called with the INIT function code.

Changes to installing and defining IMS

A new user exit type, IMSMON, can be specified in the EXITDEF parameter in the USER_EXITS section of the DFSDfxxx member. When the EXITDEF=(TYPE=IMSMON,EXITS=(*exit_name(s)*) statement is specified, the exit is called during IMS initialization, IMS termination, and anytime a Monitor log record is written to the IMS Monitor data set. The exit is called regardless whether the Monitor is on or off

Changes to administering IMS

The IMS Monitor user exit provides a standardized framework and exit point that gives you access to IMS Monitor data for real-time information for the entire lifetime of an IMS. The user exit could be utilized to monitor IMS and to collect performance data.

Changes to troubleshooting for IMS

This enhancement introduces new and changed IMS messages.

For a list of the messages and codes that are new or changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Changes to commands

Support for the IMSMON exit type is added to the QUERY USEREXIT and REFRESH USEREXIT commands. The names of the IMS Monitor user exit routines that are defined to the IMS system are returned to the QUERY USEREXIT command and refreshed by the REFRESH USEREXIT command.

Changes to exit routines

A new IMS Monitor user exit (IMSMON) is added.

Refreshable type-2 Automated Operator exit routine

In IMS 14, you can use the new user exit services with a type-2 Automated Operator exit routine. The type-2 Automated Operator exit routine is now associated with a new exit routine type (AOIE) rather than a specific exit routine, so IMS can call one or more type-2 Automated Operator exit routines. You can refresh AOIE type exit routines without restarting IMS, define multiple user exit

routines at the AOIE exit point, and query AOIE type exit routines. DFSAOE00 is now designated as an AOIE type exit routine.

When the type-2 Automated Operator exit routine is enabled to use the new user exit services, you can reload AOIE type exit routines without restarting the IMS control region by issuing the IMS type-2 command REFRESH USEREXIT TYPE(AOIE). You can also query the type-2 Automated Operator exit routine by issuing the IMS type-2 command QUERY USEREXIT TYPE(AOIE).

The use of the enhanced user exit services with the type-2 Automated Operator exit routine is optional. You can continue to manage the DFSAOE00 exit routine as you do today with no changes required.

Enabling the new user exit services for the type-2 Automated Operated exit routine requires only a minor change.

Migration considerations

This enhancement does not affect migration. However, you can optionally migrate your exits to use the new user exit services by defining your current exit routines in the EXITDEF parameter of the USER_EXITS section of the DFSDFxxx member. The current exit routine can then be used along with information on the exit being returned by the QRY USEREXIT command. The exit is also refreshable by using the REFRESH USEREXIT command.

Coexistence considerations

The DFSAOE00 exit routine is not refreshable in versions of IMS earlier than IMS 14. To enable DFSAOE00 as a refreshable exit routine, define it in an EXITDEF parameter in the USER_EXITS section of the DFSDFxxx member.

Impact to installing and defining IMS

A new user exit type, AOIE, can be specified in the EXITDEF parameter in the USER_EXITS section of the DFSDFxxx member. No additional steps are necessary to use a single exit routine that is named DFSAOE00. If you use multiple exit routines, specify EXITDEF=(TYPE= AOIE,EXIT=(*exit_names*)) in the EXITDEF parameter of the DFSDFxxx member of the IMS.PROCLIB data set.

The following example statement defines one AOIE type exit routine:
EXITDEF=(TYPE= AOIE,EXIT=(UEXIT1))

The following example statement defines multiple AOIE type exit routines:
EXITDEF=(TYPE= AOIE,EXIT=(UEXIT1,UEXIT2,UEXIT3,UEXIT4))

Impact to administering IMS

You can define the routines for the AOIE exit routine type in the EXITDEF parameter in the USER_EXITS section of the DFSDFxxx member. The QRY USEREXIT command is used to query information about the routines for the user exit types. The REFRESH USEREXIT command is used to dynamically refresh the user exit types.

Programming for IMS

You can use the GMSG DL/I call function with the AOIE exit routine type.

Impact to troubleshooting for IMS

The messages that pertain to the DFSAOE00 exit routine now apply to all AOIE type exit routines.

For a list of the messages and codes that are changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

For a complete list of all of the new, changed, and deleted messages, and abend codes in IMS 14, see Chapter 7, "Message and code changes in IMS 14," on page 95.

Changes to commands

Support for the AOIE exit routine type is added to the QUERY USEREXIT and REFRESH USEREXIT commands. The names of the AOIE type exit routines that are defined to the IMS system are returned to the QUERY USEREXIT command and refreshed by the REFRESH USEREXIT command.

For a list of the commands that are changed for this enhancement, see the online version of this information under Release Planning in the IBM Knowledge Center.

Changes to exit routines

A new type-2 Automated Operator exit routine type (AOIE) is added.

ACEE creation and management enhancement

In IMS 14, the **ESAF_SIGNON_ACEE=YES | NO** is added to the DFSJVMEV member of the IMS PROCLIB data set.

The ACEE is needed for external subsystem (ESS) thread processing. Previously, an ESS created and managed the ACEE for every ESAF call whenever the ACEE was needed. As a result, much workload was caused in ACEE creation and deletion for every ESAF call. To improve the ESS performance in ACEE creation and management, a parameter is added to the DFSJVMEV member of the IMS PROCLIB data set. You can use this parameter to enable the IMS system to create and manage the ACEE in a persistent JVM dependent region for ESAF threads. This enhancement reduces the overhead of ACEE creation and management because the ACEE is created only once and available for future use.

Changes to installing and defining IMS

The **ESAF_SIGNON_ACEE=YES | NO** is added to the DFSJVMEV member of the IMS PROCLIB data set. For more information, see .

Part 3. IBM IMS Tools support for IMS 14

The IBM IMS Tools products are designed to enhance the performance and operation of IMS, and are upgraded and enhanced to work with IMS 14.

The IMS Tools products are categorized by function. The information describes the minimum version and release levels of the IBM IMS Tools products that support IMS 14.

For more information about these tools, go to www.ibm.com/software/data/db2imstools .

For a list of current PTFs for these tools, go to IBM Db2 and IMS Tools PTF Listing.

Chapter 13. IBM IMS Tools Solution Packs for z/OS

The IMS Tools Solution Packs consolidate and integrate related tools, utilities, and functions into single, lower-cost offerings to provide complete solutions that improve systems and data management, enhance performance, and support increased availability of IMS DB and TM systems.

IBM IMS Database Solution Pack for z/OS, V2.1

The IBM IMS Database Solution Pack for z/OS is a utilities management tool that combines the entire set of IBM IMS Tools database products that are needed to manage IMS Full Function and HALDB databases into a single, consolidated solution.

IMS Database Solution Pack for z/OS provides helps keep databases operational and tuned. It helps reduce the operational complexity and the impact of database reorganization on system resources.

It provides a comprehensive set of high-performance utilities to unload, load, index build, reorganize, backup, verify, and report on full-function databases.

The IMS Database Solution Pack for z/OS includes the following tools:

- “IBM IMS Database Reorganization Expert for z/OS, V4.1” on page 242
- “IBM IMS High Performance Image Copy for z/OS, V4.2” on page 221
- “IBM IMS High Performance Load for z/OS, V2.1” on page 242
- “IBM IMS High Performance Pointer Checker for z/OS, V3.1” on page 243
- “IBM IMS High Performance Prefix Resolution for z/OS, V3.1” on page 243
- “IBM IMS High Performance Unload for z/OS, V1.2” on page 243
- “IBM IMS Index Builder for z/OS, V3.1” on page 221
- “IBM IMS Library Integrity Utilities for z/OS, V2.2” on page 244

The elements of both IBM IMS Online Reorganization Facility for z/OS and IBM IMS High Availability Large Database (HALDB) Toolkit for z/OS are also assimilated into IMS Database Solution Pack for z/OS , V2.1 and are new components of the Solution Pack.

Program number: 5655-DSP

IBM IMS Fast Path Solution Pack for z/OS, V1.3

The IBM IMS Fast Path Solution Pack for z/OS delivers extensive utilities that include functions a database administrator can use to analyze, maintain, and tune IMS Fast Path databases. Included are features that can help boost system availability by enabling the operations of a variety of key functions without taking the IMS database offline.

The IBM IMS Fast Path Solution Pack for z/OS includes:

- IMS Database Repair Facility
- IMS Fast Path Advanced Tool
- IMS Fast Path Basic Tools

- IMS Fast Path Online Tools
- “IBM IMS High Performance Image Copy for z/OS, V4.2” on page 221
- “IBM IMS Library Integrity Utilities for z/OS, V2.2” on page 244

Program number: 5655-W14

IBM IMS Performance Solution Pack for z/OS, V1.3

The IBM IMS Performance Solution Pack for z/OS delivers a more affordable, comprehensive portfolio of IBM database management tools. The product combines the features and functions several separate tools. The products are tightly integrated, making the end-to-end analysis of IMS transactions faster and easier than ever, supporting improved productivity for problem analysts, improved IMS application performance, more efficient IMS resource utilization, and higher system availability.

The IBM IMS Performance Solution Pack for z/OS includes the following tools:

- “IBM IMS Connect Extensions for z/OS, V2.4” on page 229
- “IBM IMS Performance Analyzer for z/OS, V4.4” on page 230
- “IMS Problem Investigator for z/OS, V2.4” on page 230

Program number: 5655-S42

IBM IMS Recovery Solution Pack for z/OS, V2.1

IMS Recovery Solution Pack for z/OS, V2.1 combines all the features, functions, and processes to support efforts to implement best practices backup and recovery scenarios. It also allows for the simultaneous backup and recovery of multiple data sets and Fast Path areas. This solution can help reduce the operational complexity and the impact of database backup and recovery on system resources.

The IBM IMS Recovery Solution Pack for z/OS includes the following tools:

- IMS Database Recovery Facility
- IMS Database Recovery Facility: Extended Functions
- IMS High Performance Change Accumulation Utility
- “IBM IMS High Performance Image Copy for z/OS, V4.2” on page 221
- “IBM IMS Index Builder for z/OS, V3.1” on page 221

Program number: 5655-ISR

IBM Tools Base for z/OS, V1.5

IBM Tools Base for z/OS consists of various components that support key strategic architectures, technologies, and services that are utilized by IMS tools and Db2 for z/OS. In V1.5, the Administration Console component and the embedded IMS Explorer for Administration function that are in V1.4 are moved to IBM Management Console for IMS and Db2 for z/OS.

IBM Tools Base provides common functions that are required in all IMS Tools solution pack environments. Tools Base must either already be installed or installed at the same time as a Solution Pack.

Tools Base is a no-charge PID that you must order from ShopZ when you order other IMS Tools solution packs. Tools Base requires a license and must be installed before installing the first IMS Tools solution pack.

For a list of current PTFs for IBM Tools Base, go to IBM Db2 and IMS Tools PTF Listing.

Tools Base components provide the infrastructure that supports key solution strategies for IMS Tools.

The Tools Base V1.5 includes the following categories and components:

Autonomics

The Autonomics component tools simplify common database maintenance tasks by collecting database state information, analyzing this data, and providing passive or active responses to conditions that exceed specified threshold values. The tools that come in the Autonomics component are:

- IBM Tools Base Autonomics Director for z/OS

Autonomics Director is a core solution for scheduling and automating common database maintenance tasks. Sensor-enabled IMS Tools products capture the state of specific database conditions and store this data in a centralized repository. Autonomics Director uses Policy Services to evaluate this data and then makes recommendations, such as indicating a need to reorganize a database.

- IBM Tools Base Policy Services for z/OS

Policy Services use policies and rules to evaluate the sensor data that is collected by IMS Tools products, and then provides a response to any condition that exceeds the specified threshold values.

- IBM Tools Base IMS Tools Knowledge Base for z/OS

IMS Tools Knowledge Base manages centralized repositories that are created to store and view a variety of data including reports, policies, rules, notification lists, database state (sensor) data, exceptions, recommendations, and evaluation schedules.

Common Services

The Common Services provide functions that facilitate the interactions among tools, and between tools and IMS. The tools that come in the Common Services component are:

- IBM Tools Base Distributed Access Infrastructure for z/OS

Distributed Access Infrastructure enables distributed clients to access IMS Tools through TCP/IP socket communication.

- IBM Tools Base IMS Tools Common Services for z/OS

IMS Tools Common Services includes the Generic Exits for calling multiple exit routines from a single exit point, and the Tools Online System Interface that provides a command interface between IMS and IMS Tools.

- IBM Tools Base Connection Server for z/OS

Connection Server extends ISPF functions for specific IMS Tools to Eclipse-based applications and the IMS Records User Data Utility for scanning and removing from IMS log records any sensitive or confidential user data, such as customer business information. .

Supplementary tools

The Tools Base includes the following Supplementary tools:

- IBM Tools Base IMS Hardware Data Compression Extended for z/OS
IMS Hardware Data Compression Extended provides functions for compressing IMS data by using the z/OS hardware data compression (HDC) that is available on IBM processors.
- IBM Tools Base IMS Batch Terminal Simulator GUI plug-in
The BTS GUI plug-in provides a graphical user interface that allows distributed Eclipse-based clients to access IMS Batch Terminal Simulator in an Eclipse-integrated development environment.
- IBM Tools Base IMS Batch Terminal Simulator Resource Adapter
The BTS Resource Adapter allows you to use IMS Batch Terminal Simulator to test J2C applications that drive IMS Transactions from the WebSphere Application Server environment. Customers use the BTS Resource Adapter to allow their J2C applications to be tested as an inexpensive alternative to testing on a real IMS system.

Program number: 5655-V93

Related concepts:

Chapter 14, "IBM Management Console for IMS and Db2 for z/OS, V1.1," on page 217

Chapter 14. IBM Management Console for IMS and Db2 for z/OS, V1.1

IBM Management Console for IMS and Db2 for z/OS (Management Console) is a lightweight web server that consolidates and simplifies information from IMS and Db2 for z/OS into a single, holistic, web-based interface that is accessible from a standard web browser.

The Administration Console component and its embedded IMS Explorer for Administration extension in IMS Tools Base for z/OS, V1.4 are no longer delivered as part of Tools Base for z/OS, V1.5. They are integrated into Management Console with additional functions.

Management Console, a replacement for IMS Control Center, connects to the IMS Operations Manager through IMS Connect. If you are using the IMS Control Center function, use Management Console instead.

Use of Management Console can help simplify and consolidate presentation of complex information that is gathered across all of the IMS and Db2 for z/OS systems.

- With a single installation of Management Console, you can get an enterprise-wide presentation of system health data, including autonomic symptoms, exceptions, and recommended actions through a standard web browser.
- You can progressively drill down from the enterprise or system level down to individual objects.
- The enterprise-wide view displays the resources with the highest severity symptoms and exceptions and provides the ability to navigate directly to those resources with recommended actions.
- Embedded help provides details and concepts to reduce the learning curve and build the essential foundation for new IMS and Db2 for z/OS system programmers and database administrators.
- You can also view, filter, and search IMS Tools reports that are stored within IMS Tools Knowledge Base repositories.
- You can define and optionally schedule active autonomic maintenance windows to automatically perform recommended actions through the graphical calendaring support.


For IMS, with a defined IMS Connect connection and a specified IMSplex name, related resources are discovered.

- Individual resource dashboards are provided for viewing and accessing each IMSplex, IMS subsystem, IMS Connect, IMS database (full function or Fast Path), IMS transaction, IMS program, or IMS routing code.
- You can search for IMS resources, or query, start, and stop IMS resources, individually or in groups.
- You can update transactions, databases, Fast Path routing codes, and application programs.
- You can issue IMS commands from the embedded IMS command console. The command result is displayed in text, and results from type-2 commands can also be viewed in grid mode (table format). Results in both modes can be printed.

| Program number: 5655-TAC

| **Related information:**

|  [Configuring Management Console for IMSplex management](#)

|  [Configuring Management Console for autonomic maintenance of IMS databases](#)

Chapter 15. IMS Application Management tools

IBM tools provide the reliability and affordability you need to maximize the value of your IMS application management.

IBM IMS Batch Terminal Simulator for z/OS, V4.1

The IBM IMS Batch Terminal Simulator for z/OS tool:

- Provides a comprehensive way to test and check IMS application program logic, IMS application interfaces, teleprocessing activity, 3270 format control blocks, and database activity.
- Simulates the operation of IMS applications in TSO and batch environments.
- Operates transparently to the applications, requiring no changes to IMS code, control blocks, libraries, or application load modules.
- Provides a trace of all DL/I calls, SQL calls and IBM MQ calls with related details.
- Provides a stable online system in test and production environments to execute applications properly before they are put online.
- Provides Java language application support under JBP regions.
- Provides a Playback function which creates a file of screen images that correspond to the transaction activities captured in an IMS OLDS or SLDS.
- Can include or exclude by LTERM, transaction, and MODname.
- Include by user ID and time range.
- Output listings include a list of the transaction codes that have been executed, user IDs, transaction code/user ID mapping, listing of the MFS MODnames used, and the transaction screen images.

Program number: 5655-BT4

IBM IMS Program Restart Facility for z/OS, V2.2

The IBM IMS Program Restart Facility for z/OS helps to automate the backout and restart of abended IMS Batch Message Processing (BMP) regions and Data Language/I (DL/I) batch jobs.

The IMS Program Restart Facility for z/OS assists with the following actions:

- A batch job that requires a restart is restarted in accordance with proper restart procedures.
- Identifies and uses the correct log data set and the correct checkpoint ID.
- Automates the backout of abended DL/I batch jobs without the need to manually code and execute multiple IMS utility jobs.
- Automates IMS DL/I batch backout processing at the time an abend occurs (or when the job is restarted if backout cannot be done at abend time). The backout process releases database locks and makes database segments available to other tasks.

In a data sharing IMSplex environment, IMS Program Restart Facility for z/OS provides the additional feature of allowing a BMP to be restarted using a different IMS system. Taking advantage of this feature helps to provide improved restart times in the event of a system outage.

In addition, IMS Program Restart Facility for z/OS provides the ability to restart an IMS job on a different version of IMS than was in use when the job abended. This allows the implementation of an IMS version upgrade by simply abending running IMS jobs before the upgrade, and restarting the jobs after the upgrade is completed.

IMS Program Restart Facility for z/OS is easily implemented and typically requires no application changes. While an IMS batch job is running, IMS Program Restart Facility for z/OS intercepts and writes checkpoint records to two data sets. These data sets are deleted when the batch job completes successfully. In the event of an abend, however, the data sets remain in place and are used to restart the job.

Version 2.2 of IMS Program Restart Facility for z/OS offers:

- New ISPF interface.
- Significant reduction of contention when options are updated.
- Support for restarting jobs across different versions of IMS.
- Capability to specify options that are based on job name, step name, and IMSID instead of only by job name.

All functions of IBM IMS Batch Backout Manager for z/OS (5697-H75) are now incorporated into IMS Program Restart Facility for z/OS , V2.2.

Program number: 5655-E14

Chapter 16. IMS Backup and Recovery Management tools

The following sections describe the IBM IMS backup and recovery management tools.

IBM IMS DEDB Fast Recovery for z/OS, V2.2

The IBM IMS DEDB Fast Recovery for z/OS tool:

- Assists in the operation and maintenance of data integrity of IMS databases.
- Is designed as a fast alternative to emergency restart (ERE) failure recovery.
- Corrects online log data sets (OLDSs) by invalidating logging for transactions that did not reach the synch point.
- Significantly reduces the amount of time needed to recover DEDBs after an IMS failure.
- Generates the JCL for the MSDB Dump Recovery utility to be processed before an IMS cold start for MSDB (main storage database) recovery.
- Shortens the recovery time of an unscheduled IMS cold start while maintaining the integrity of the IMS databases.
- Supports multiple IMS releases from a single Load Library.

Program number: 5655-E32

IBM IMS High Performance Image Copy for z/OS, V4.2

The IBM IMS High Performance Image Copy for z/OS tool functions are:

- Running an image copy function with the hash checking of IMS HP Pointer Checker V2.
- Performing accuracy checking of an image copy.
- Allocating all input and output data sets dynamically.
- Optionally compressing output image copies.
- Creating image copies.
- Providing advanced copy services; concurrent copy, FlashCopy and SnapShot copies.
- Reducing image copy and recovery time. Reduces elapsed time and CPU utilization.
- Providing automatic checkpoint and restart.
- Stopping and starting databases automatically.
- Integrating with other IMS Tools utilities.

Program number: 5655-N45

IBM IMS Index Builder for z/OS, V3.1

The IBM IMS Index Builder for z/OS:

- Offers several features that improve overall performance and enhance ease of use.
- Builds or rebuilds primary and secondary indexes quickly.

- Allows the user to specify an optional output file where records that are needed for prefix resolution can be split off and written as they are read in.
- Eliminates the need to image copy indexes.
- Recognizes index records that have duplicate keys and writes the duplicate keys to a SYSOUT data set.
- Supports building IMS HALDB primary indexes.
- Integrates with the Database Recovery Facility, which enables the building of primary and secondary indexes during a database recovery.

Program number: 5655-R01

IBM IMS Recovery Expert for z/OS, V2.2

The IBM IMS Recovery Expert for z/OS is a storage-aware backup and recovery solution that integrates storage processor fast-replication facilities with IMS backup and recovery operations to allow instantaneous backups with no application downtime and to help reduce recovery time, and simplify disaster recovery procedures while using fewer processor I/O and storage resources. The IMS Recovery Expert V2.2 product belongs to the family of IMS Tools that provides backup and recovery solutions.

Program number: 5655-S98

IBM IMS Recovery Solution Pack for z/OS, V2.1

IMS Recovery Solution Pack for z/OS, V2.1 combines all the features, functions, and processes to support efforts to implement best practices backup and recovery scenarios. It also allows for the simultaneous backup and recovery of multiple data sets and Fast Path areas. This solution can help reduce the operational complexity and the impact of database backup and recovery on system resources.

The IBM IMS Recovery Solution Pack for z/OS includes the following tools:

- IMS Database Recovery Facility
- IMS Database Recovery Facility: Extended Functions
- IMS High Performance Change Accumulation Utility
- “IBM IMS High Performance Image Copy for z/OS, V4.2” on page 221
- “IBM IMS Index Builder for z/OS, V3.1” on page 221

Program number: 5655-ISR

Chapter 17. IMS Database Administration tools

The following sections describe the IBM IMS database administration tools.

IBM IMS Sequential Randomizer Generator for OS/390, V1.1

The IBM IMS Sequential Randomizer Generator for OS/390 tool:

- Creates a randomizer that enables the user to access HDAM and DEDB database segments either directly or sequentially.
- Allows access to HDAM and DEDB databases in logical key sequence without sacrificing their efficient direct access capabilities.
- Optimizes the distribution of database records by adjusting the randomizing module if the number of synonyms or the control interval (CI) or block utilization exceeds the user-specified value.

Program number: 5655-E11

IBM IMS Database Solution Pack for z/OS, V2.1

The IBM IMS Database Solution Pack for z/OS is a utilities management tool that combines the entire set of IBM IMS Tools database products that are needed to manage IMS Full Function and HALDB databases into a single, consolidated solution.

IMS Database Solution Pack for z/OS provides helps keep databases operational and tuned. It helps reduce the operational complexity and the impact of database reorganization on system resources.

It provides a comprehensive set of high-performance utilities to unload, load, index build, reorganize, backup, verify, and report on full-function databases.

The IMS Database Solution Pack for z/OS includes the following tools:

- “IBM IMS Database Reorganization Expert for z/OS, V4.1” on page 242
- “IBM IMS High Performance Image Copy for z/OS, V4.2” on page 221
- “IBM IMS High Performance Load for z/OS, V2.1” on page 242
- “IBM IMS High Performance Pointer Checker for z/OS, V3.1” on page 243
- “IBM IMS High Performance Prefix Resolution for z/OS, V3.1” on page 243
- “IBM IMS High Performance Unload for z/OS, V1.2” on page 243
- “IBM IMS Index Builder for z/OS, V3.1” on page 221
- “IBM IMS Library Integrity Utilities for z/OS, V2.2” on page 244

The elements of both IBM IMS Online Reorganization Facility for z/OS and IBM IMS High Availability Large Database (HALDB) Toolkit for z/OS are also assimilated into IMS Database Solution Pack for z/OS , V2.1 and are new components of the Solution Pack.

Program number: 5655-DSP

Chapter 18. IMS Data Replication tools

The following sections describe the IBM IMS data replication tools.

IMS Cloning Tool for z/OS, V1.2

The IMS Cloning Tool for z/OS provides a faster, simpler cloning solution that can help improve productivity and supports efforts to reduce total cost. It automates the cloning process to provide usable IMS clones within minutes, helping to boost efficiency and to free up DBA time.

Program number: 5655-U91

InfoSphere Data Replication for IMS for z/OS, V11.3

IBM InfoSphere® Data Replication for IMS for z/OS helps increase the availability of IMS data whether it is for a continuous availability strategy or ensuring the timely distribution of critical enterprise information.

This tool:

- Helps eliminate geographic limitations
- Helps achieve minimal recovery time objectives (RTOs)
- Provides support for mixed IMS workloads
- Provides flexible, unified monitoring
- Can import or export metadata
- Provides high scalability and performance
- Supports the IBM Change Data Capture Solutions

Program number: 5655-IM1

Chapter 19. InfoSphere IMS tools

The following sections describe the IBM InfoSphere IMS tools.

IBM InfoSphere Classic Change Data Capture for z/OS, V11.3

IBM InfoSphere Classic Change Data Capture for z/OS (InfoSphere Classic CDC for z/OS) is a replication solution that captures changes to non-relational mainframe data and delivers them to relational databases, producing an accurate relational replica of your mainframe data on supported target databases in near-real time.

This tool:

- Helps eliminate geographic limitations
- Helps achieve minimal recovery time objectives (RTOs)
- Provides support for mixed IMS workloads
- Provides flexible, unified monitoring
- Can import or export metadata
- Provides high scalability and performance
- Supports the IBM Change Data Capture Solutions

Program number: 5655-IM5

IBM InfoSphere Classic Federation Server for z/OS, V11.3

The IBM InfoSphere Classic Federation Server for z/OS tool provides the following functions:

- Provides SQL access to mainframe databases and files with transactional speed and enterprise scale without mainframe programming.
- Allows applications and tools to issue SQL SELECT, INSERT, UPDATE, and DELETE commands using ODBC, JDBC, or a CLI (Command Level Interface) to access System z data.
- Stores System z accessed data in VSAM, IAM, and sequential files, as well as Db2 for z/OS, IMS, Software AG Adabas, and CA-Datacom and CA-IDMS databases all without mainframe programming.

Program number 5655-IM4

IBM InfoSphere Optim Test Data Management Solution for z/OS, 11.3

The IBM InfoSphere Optim™ Test Data Management Solution for z/OS tool:

- Creates test databases that are relationally intact subsets of an existing production database.
- Migrates subsets of data that require data transformations as part of the migration. This feature is especially useful for masking data used in testing applications or to introduce altered data to the production database.
- Provides for aging all types of date columns, regardless of initial format, and adjusting the resulting dates to suit site-specific business rules. An extensive set

of sample definitions to handle these business rules is distributed with Move. Definitions can be customized to accommodate organization-specific business rules.

Program number: 5655-O21

InfoSphere Data Replication for IMS for z/OS, V11.3

IBM InfoSphere Data Replication for IMS for z/OS helps increase the availability of IMS data whether it is for a continuous availability strategy or ensuring the timely distribution of critical enterprise information.

This tool:

- Helps eliminate geographic limitations
- Helps achieve minimal recovery time objectives (RTOs)
- Provides support for mixed IMS workloads
- Provides flexible, unified monitoring
- Can import or export metadata
- Provides high scalability and performance
- Supports the IBM Change Data Capture Solutions

Program number: 5655-IM1

Chapter 20. IMS Performance Management tools

The following sections describe the IBM IMS performance management tools.

IBM IMS Buffer Pool Analyzer for z/OS, V1.4

The IBM IMS Buffer Pool Analyzer for z/OS tool:

- Provides modeling facilities to assist with making informed decisions about the addition of buffers to an existing pool, or sizing requirements for a new buffer pool.
- Helps you determine the impact of buffer pool changes before they are made to take the guess work out of the process.
- Analyzes IMS database buffer pools (OSAM and VSAM) to provide statistical analysis of the impact of changes that affect the buffer pools.
- Provides I/O rates and buffering requirements for a specific database.
- Allows for better allocation of real memory resources.
- Identifies databases that most heavily use each database subpool.
- Performs “what if” scenario analysis, such as identifying the impact of splitting a specific database into a new buffer pool.
- Determines the performance effects for a given buffer pool when you add or reduce the number of buffer pools.

Program number: 5697-H77

IBM IMS Connect Extensions for z/OS, V2.4

IBM IMS Connect Extensions for z/OS tool:

- Improves the availability, reliability, and performance of TCP/IP access to IMS through IMS Connect.
- Dynamically manages workload through rules-based routing of TCP/IP OTMA transactions or TCP/IP DRDA requests.
- Records and reports IMS Connect events, activities, and utilization in real time.
- Provides a single point of control for multiple IMS Connect systems.
- Includes an REXX interface for automated operations.
- Dynamically responds to changes in data store availability and flood conditions.
- Improves system security with flexible access control.
- Complements IBM IMS Performance Analyzer for z/OS and IMS Problem Investigator for z/OS tools to aid reporting and accelerate problem determination.

Program number: 5655-S56

IBM IMS Network Compression Facility for z/OS, V1.2

The IBM IMS Network Compression Facility for z/OS:

- Provides several new options for compression of 3270 data streams:
 - Compresses all repeated characters, not just spaces, nulls, asterisks, and dashes

- Allows 3270 Field Merge, blank elimination, and elimination of non-display fields
- Eliminates redundant 3270 Set Buffer Address (SBA) commands
- Makes installation easier by eliminating the need for the XCM address space and subsystem, and for SYS1.PARMLIB updates, except APF authorization of one data set.
- Includes a utility to convert current options to the new PROCLIB member format.
- Improves flexibility by allowing cross-MVS system inquiries and updates to Network Compression Facility information and options.

Program number: 5655-E41

IBM IMS Performance Analyzer for z/OS, V4.4

IBM IMS Performance Analyzer for z/OS tool:

- Delivers end-to-end transit analysis for all types of transaction workloads, including shared queues, across the IMS enterprise.
- Provides comprehensive reporting for IMS Log and IMS Monitor, IMS Connect Extensions for z/OS, and OMEGAMON[®] ATF records.
- Provides a wide variety of reports at various levels of detail, from high-level management summaries and graphical reports to detailed traces for in-depth analysis. These reports can help analyze transaction response time and measure the usage and availability of important resources.
- Provides a complete end-to-end picture of the transaction lifecycle by combining reports from IMS logs and IMS Connect Extensions journals.
- Allows you to design your own form-based transit reports and export these reports to Db2 or in CSV format for input to a business analytics tool such as IBM Cognos[®] Insight.
- Automatically selects log file using Database Recovery Control (DBRC) for quick and easy report requests.
- Includes comprehensive specialized reporting of Database Control (DBCTL) and Fast Path.

Program number: 5655-R03

IMS Problem Investigator for z/OS, V2.4

IBM IMS Problem Investigator for z/OS tool:

- Enhances problem-determination and diagnostic capabilities for IMS TM and IMS DB systems.
- Use the information about IMS and the following related subsystems:
 - IMS Log and IMS Monitor data sets
 - Common Queue Server (CQS) log stream and extracts
 - OMEGAMON[®] ATF
 - Db2 and IBM MQ
 - IMS and IMS Connect transaction indexes that IMS Performance Analyzer creates
 - IMS Connect event data from IMS Connect Extensions journals
- Determines transaction times and event latencies to help identify bottlenecks.

- Provides an end-to-end replay of an IMS transaction on a single screen, including Db2 and MQ events.
- Connects records that are associated with the same transaction across all logs, allowing tracking of all records associated with the transaction while hiding unrelated records.
- Drills down to field level to help analyze the causes of problems that are highlighted in the high-level transaction response time and resource utilization. These problems are reported by IMS Performance Analyzer.
- Provides a REXX command interface for you to customize log record analysis and extracts.

Program number: 5655-R02

IBM Transaction Analysis Workbench, V1.3

The IBM Transaction Analysis Workbench helps you analyze problems with the performance or behavior of transactions on z/OS and includes the following features:

Coverage across z/OS subsystems

Uses the logs and other historical data generated by each subsystem during normal transaction processing and system operations. No special agents are required.

Supported data sources include CICS, Db2, IMS, IBM MQ, WebSphere Application Server for z/OS, various SMF records, and OPERLOG.

Consolidation of different subsystem logs in a single interface

Tracks individual transactions across multiple subsystems and analyzes different logs in the same consistent user interface.

Collaboration between users

Saves information about a problem, such as locations of log files and log records of interest, and then shares that information with other users, enabling collaborative analysis without rework.

Log forwarding

Converts log data to CSV or JSON format and forwards the data to analytics platforms such as Splunk, Elastic, or Hadoop.

Mobile Workload Pricing for z/OS (MWP)

Creates the CSV and SMF files that are required by the Mobile Workload Reporting Tool (MWRT).

Program number: 5697-P37

IBM IMS Performance Solution Pack for z/OS, V1.3

The IBM IMS Performance Solution Pack for z/OS delivers a more affordable, comprehensive portfolio of IBM database management tools. The product combines the features and functions several separate tools. The products are tightly integrated, making the end-to-end analysis of IMS transactions faster and easier than ever, supporting improved productivity for problem analysts, improved IMS application performance, more efficient IMS resource utilization, and higher system availability.

The IBM IMS Performance Solution Pack for z/OS includes the following tools:

- “IBM IMS Connect Extensions for z/OS, V2.4” on page 229
- “IBM IMS Performance Analyzer for z/OS, V4.4” on page 230
- “IMS Problem Investigator for z/OS, V2.4” on page 230

Program number: 5655-S42

Chapter 21. IMS Regulatory Compliance tools

The following sections describe the IBM IMS regulatory compliance tools.

IBM InfoSphere Guardium Data Encryption for Db2 and IMS Databases, V1.2

Formerly known as IBM Data Encryption for IMS and Db2 Databases, InfoSphere Guardium® Data Encryption leverages the System z Crypto Hardware to efficiently secure sensitive and private data at the Db2 row level and the IMS segment level.

The IBM InfoSphere Guardium Data Encryption for Db2 and IMS Databases tool:

- Provides Db2 Edit routines and IMS Exit routines that invoke the z/OS Integrated Cryptographic Service Facility (ICSF) which exploits the Crypto Hardware for data encryption and decryption.
- Contains sample implementation jobs.
- Includes an ISPF front end to build implementation jobs.
- Provides the capability to specify unique encryption keys.

Program number: 5655-P03

IBM Security Guardium S-TAP for IMS

The IBM Security Guardium S-TAP® for IMS tool (formally known as IBM InfoSphere Guardium S-TAP for IMS) includes support for:

- Capture of database and segment reads and changes (insert, update, delete)
- Capture of segment concatenated key and segment data on request to provide before and after images of updated segments
- Capture of access to IMS data sets outside the control of IMS services (database data sets, image copy data sets, IMS log data sets, RECON data sets)
- Direct streaming of audit data from z/OS process to a networked Guardium appliance to support near real-time reporting
- System STOP and START activity as recorded in the IMS log

Program number: 5655-ST9

Chapter 22. IMS System management tools

The following sections describe the IBM IMS System management tools.

IBM IMS Command Control Facility for z/OS, V2.2

The IBM IMS Command Control Facility for z/OS:

- Issues commands for DBCTL, DCCTL, or DB/DC regions.
- Issues IMS commands from a batch utility or from a TSO session via an ISPF Interface, or from a Callable Application Program Interface (API).
- Ensures successful processing of database START, STOP, DBR, and DBD commands.
- Synchronizes online change and database commands across all regions in a sysplex using the batch interface or callable API.
- Keeps members of an IMSplex synchronized by storing commands that failed due to an IMS being unavailable, and issuing the stored commands at IMS startup.
- Enhanced CCF Message Log improved problem determination and system operation whether running a single IMS or multi-IMSplex.
- Contains added IMS Operations Manager command support that provides more flexibility by enabling users that do not want to use APPC to be able to use the tool.
- Contains message disposition determination which enables users to improve or eliminate user written code.

Program number: 5655-R58

IBM IMS Extended Terminal Option Support for z/OS, V3.2

The IBM IMS Extended Terminal Option Support for z/OS tool:

- Provides a front-end to the IMS Extended Terminal Option (ETO) feature.
- Offers capabilities to help manage, implement, customize, and exploit the benefits of ETO in your systems environment to manage resources effectively.
- Supports tailoring of all parts of ETO, including sign-on processing.
- Lets you set global options for the entire user community and override options for specific terminals or user IDs.
- Allows LTERM names that start with a numeric value.
- Extends shared queues support to issue SYSTEMS level ENQ for all LTERM names that are associated with a user at sign-on.
- Offers SLU type P support and SLU 1 console support that allow you to supply a variety of options on an LU-by-LU basis (for example, Logmode, Logon Description, ASOT, ALOT).

Program number: 5655-L61

IBM IMS High Performance System Generation (SYSGEN) Tools for z/OS, V2.3

The IBM IMS High Performance System Generation (SYSGEN) Tools for z/OS:

- Offers an integrated solution that helps reduce the requirement for IMS SYSGENs.
- Provides DBAs and System Programmers with a reliable, easy-to-use IMS management tools to make application resource changes faster using fewer resources.
- Includes IMSplex support which provides the capability to simultaneously update multiple IMS control regions when executing a resource update list.
- Provides additional options for reverse sysgen, including the capability to select either in-core control blocks or MODBLKS data set resource definitions.
- Contains IMS storage display and zap which provides the capability to display and alter IMS control blocks and storage. A storage map provides z/OS virtual storage boundaries and percent utilization for common storage areas.
- Reloads ACBLIB definitions for database descriptions (DBDs) and program specifications blocks (PSBs) without using online change for ACBLIB.
- Generates a resource update list that will make IMS control blocks match IMS sysgen source macros.
- Manages your IMS SYSGEN definitions using ISPF panels (database, program, transaction, and route code definitions).
- Enables changes, additions, or deletions of resource definitions.
- Updates IMS security definitions, reload ACBs, and issue IMS commands.
- Dynamically makes changes to definitions.
- One user can define what changes are required, and have another user implement the change at a later time, via ISPF or batch.
- Back out changes installed by HP Sysgen Tools.

Program number: 5655-P43

IBM IMS Workload Router for z/OS, V2.7

The IBM IMS Workload Router for z/OS tool:

- Works with IMS TM to provide transparent routing or balancing of a transaction workload among two or more IBM systems.
- Uses IMS Multiple Systems Coupling (MSC).
- Is adaptable to a variety of system configurations.

Program number: 5697-B87

Chapter 23. Tivoli IMS tools

The following sections describe the IBM Tivoli® IMS tools.

Tivoli Decision Support for z/OS V1.8 or later

IBM Tivoli Decision Support for z/OS is designed to help you understand your performance challenges by collecting raw systems metrics data, consolidating them in a repository, providing tools to help you improve operational planning, cost management, responsiveness, and decision making.

Program number: 5698-B06

Tivoli Monitoring, V6.3.0

IBM Tivoli Monitoring can monitor and manage system and network applications on a variety of operating systems, track the availability and performance of your enterprise system, and get reports to track trends and troubleshoot problems.

Program number: 5724-C04

Tivoli AF/OPERATOR on z/OS V3.4.1

IBM Tivoli AF/OPERATOR on z/OS is a robust console automation solution for z/OS environments. Use it to streamline common tasks, optimize system performance and reduce avoidable errors, so that your staff only focuses on events or tasks that require human intervention.

Program number: 5608-C03

Tivoli AF/Integrated Resource Manager, V5.0.0

IBM Tivoli AF/Integrated Resource Manager (AF/IRM) simplifies system automation and resource management through an intuitive console-driven interface that provides comprehensive subsystem management without additional coding.

AF/IRM represents a streamlined paradigm in automation implementation and management that leverages packaged automation objects based on IBM Tivoli best practices. AF/IRM eases automation implementation and maintenance for z/OS and Parallel Sysplex environments, applications and network resources.

Program number: 5608-AFI

Tivoli OMEGACENTER Gateway on z/OS V3.4.1

IBM Tivoli OMEGACENTER Gateway on z/OS

IBM Tivoli OMEGACENTER Gateway on z/OS is a system automation tool for z/OS environments. It integrates your console automation solutions, availability monitors and Tivoli OMEGAMON solutions to provide automated responses to system events.

Program number: 5608-C04

Tivoli OMEGAMON XE for Db2 Performance Expert on z/OS, V5.2 or later

IBM Tivoli OMEGAMON XE for Db2 Performance Expert on z/OS combines the sophisticated reporting, monitoring and buffer pool analysis features of the IBM Tivoli OMEGAMON XE for Db2 Performance Monitor on z/OS and IBM Db2 Buffer Pool Analyzer products.

Tivoli OMEGAMON XE for Db2 Performance Expert on z/OS also adds expert database analysis functions to help you maximize performance and enhance productivity.

Program number: 5655-W37

Tivoli OMEGAMON XE for Db2 Performance Monitor on z/OS, V5.2 or later

Tivoli OMEGAMON XE for Db2 Performance Monitor on z/OS enables you to monitor, analyze and optimize the performance of Db2 for z/OS and Db2 applications online in real time and in batch reports.

Program number: 5655-W38

Tivoli OMEGAMON XE for IMS on z/OS, V5.3

IBM Tivoli OMEGAMON XE for IMS on z/OS monitors and manages the availability, performance, and resource utilization of your IMS systems, either at a system level or within an IMSplex.

Program number: 5698-T02

Tivoli System Automation for z/OS, V3.5

IBM Tivoli System Automation for z/OS is an IBM Tivoli NetView® for z/OS base software product that provides a single point of control for a various range of systems management functionality.

Tivoli System Automation for z/OS is a policy-based, self-healing, high-availability solution to maximize efficiency and availability of critical systems and applications. IBM Tivoli System Automation for z/OS plays a key role in supplying high end-to-end automation solutions.

Program number: 5698-SA3

Chapter 24. IMS TM management tools

The following sections describe the IBM IMS TM management tools.

IBM IMS Configuration Manager for z/OS, V2.2

You can use IBM® IMS™ Configuration Manager for z/OS® to analyze, modify, and deploy IMS resources and parameters.

IMS Configuration Manager helps you to achieve the following benefits:


- Automated mapping of your entire IMS environment, including resources and parameter configurations.
- Create a common, consistent configuration across your enterprise by identifying inconsistencies.
- Simpler transition to DRD. Integrate DRD with your current change control practices.
- Optimized online installation of resources.
- A unified, structured, and secure change management system and associated processes.
- A faster time-to-production for new IMS applications.
- A better understanding of your parameter configurations across all the IMS systems in your enterprise.

Program number: 5655-WR2

Related concepts:

 [IMS Configuration Manager for z/OS V2.2](#)

Related information:

 http://www.ibm.com/support/knowledgecenter/SSF2ZH_2.2.0/gplu-overview.html

IBM IMS Queue Control Facility for z/OS, V3.2

The IBM IMS Queue Control Facility (QCF) for z/OS tool:

- Manages IMS message queues in both shared and non-shared queue environments.
- Defines up to ten areas of the total queue space to monitor for small or large messages, using the new Queue Space Utilization Notification mechanism.
- Detects (automatically) an IMS cold start and initiates the requeue of messages that were in the queue before the cold start.
- Detects (automatically) an IMS warm start and initiates the requeue of messages (or offloads the messages) that were in the dead letter queue before warm start.
- Offloads (automatically) any messages that were on the queue during message overflow.
- Select messages based on a data string for faster problem determination.
- Contains a new filter for the dead letter queue.
- Contains multiple new console commands to display the top number of destinations that are using the message queues. These commands can display

the destinations using the message queues over last number minutes and initiate the requeue or offload of the messages that were in the queue.

Program number: 5697-N50

IBM IMS Sysplex Manager for z/OS, V1.3

The IBM IMS Sysplex Manager for z/OS tool provides:

- Real-time management of the IMS sysplex Environment.
- Single point of control.
- Single system image through local and aggregate view of data.
- Simplified user interface (TSO/ISPF).
- Structured displays of IMS resources and CF structures.
- Global support of type-1 commands, OM type-2 commands and the IMS TSO SPOC.
- Basic z/OS performance information and SVC dump capture.
- Statistics for CSL (OM, RM and SCI), IRLM, and CQS.
- Dashboard with key system indicators and threshold monitoring.
- Management functions.
- Intercept of system exceptions and generates console alerts.
- Real-time IRLM Long Lock Report.
- Automatic real-time recognition when IRLM detects long locks.
- Consolidated and analyzed information for the top blocker, which is recorded in an exceptions file and sent to the z/OS console for automated operations.
- Browse, delete, and recover capability for messages on shared queues.
- Delete capability for RM resource structure entries.
- Assign affinity capability for transactions in shared-queues environment.
- Support for IMS DB/TM, DBCTL, and DCCTL for IMS Version 8 and later.

Program number: 5655-P01

Chapter 25. IMS Utilities Management tools

The following sections describe the IBM IMS utilities management tools.

IMS Cloning Tool for z/OS, V1.2

The IMS Cloning Tool for z/OS provides a faster, simpler cloning solution that can help improve productivity and supports efforts to reduce total cost. It automates the cloning process to provide usable IMS clones within minutes, helping to boost efficiency and to free up DBA time.

Program number: 5655-U91

IBM IMS Database Control Suite for z/OS, V3.2

The IBM IMS Database Control Suite for z/OS:

- Provides a day-to-day IMS database environment management tool for DBAs, system programmers, and application support specialists.
- Enables you to collect and build on the data that is required to generate job control language (JCL) for database backup/recovery/reorganization management using IMS Base Utilities and IBM High Performance Tools.
- Significant improvements to DBDSGRPs that include:
 - Loading pre-existing DBDSGRPs into IMS DBCS from the RECON.
 - Using your naming conventions rather than generated names.
 - Creating DBDSGRPs with a mix of HALDB and full-function databases.
- Has a powerful batch collection function that enables you to enter a database description library (DBDLIB), which then triggers the tool to build the jobs for all logically related members in just one pass, saving time and steps and avoiding data redundancy errors.
- Integrates IBM IMS tools and utilities that automate database backup, recovery, reorganization and monitoring.
- Facilitates database maintenance efficiency with single-step job control language (JCL) conversion.
- Improves DBA productivity with an intuitive interface and ease-of-use enhancements.
- Provides an ISPF interface for the IMS Library Integrity Utilities.
- Includes a simplified user interface; reduction and consolidation of the number of data sets, ISPF panels, and process steps; and more flexibility for local standards.
- Automates tedious manual IMS maintenance functions to make database management easier.
- Is a functional replacement for the IMS DBICF (Database Integrity Control Facility) tool.
- Provides improved usability and productivity because the build process, during which JCL skeletons for utilities are built, is no longer required.
- Provides improved product panels and panel navigation; field level help for all help panels; and fuller integration with DBRC terminology.

Program number: 5655-L08

IBM IMS Database Reorganization Expert for z/OS, V4.1

The IBM IMS Database Reorganization Expert for z/OS tool:

- Supports the creation, customization, and centralized storage of policies on database exception detection and conditional reorganization.
- Provides a Smart Reorganization utility with the capabilities of the exception notification and conditional reorganization based on policies stored in a centralized policy repository.
- Collects statistical data about databases and determines the reorganization needs of the databases based on the reorganization policy selected for the database.
- Detects database exceptions and notifies the TSO users or z/OS operators that are specified in the relevant policy of the exceptional state.

This tool is included in the “IBM Tools Base for z/OS, V1.5” on page 214 (5655-V93).

Program number: 5655-S35

IBM IMS High Performance Image Copy for z/OS, V4.2

The IBM IMS High Performance Image Copy for z/OS tool functions are:

- Running an image copy function with the hash checking of IMS HP Pointer Checker V2.
- Performing accuracy checking of an image copy.
- Allocating all input and output data sets dynamically.
- Optionally compressing output image copies.
- Creating image copies.
- Providing advanced copy services; concurrent copy, FlashCopy and SnapShot copies.
- Reducing image copy and recovery time. Reduces elapsed time and CPU utilization.
- Providing automatic checkpoint and restart.
- Stopping and starting databases automatically.
- Integrating with other IMS Tools utilities.

Program number: 5655-N45

IBM IMS High Performance Load for z/OS, V2.1

The IBM IMS High Performance Load for z/OS tool:

- Provides a high performance database reloading capability for IMS full-function databases.
- Initializes empty HDAM and HIDAM databases.
- Supports IMS Parallel Reorganization for z/OS V3 capabilities, such as image copy creation during database reorganization.
- Supports reorganization reload of HALDB partitions, including online-reorganization-capable HALDB partitions of IMS Version 9 and later.
- Automatically initializes HALDB partition data set before reload.
- Provides a performance replacement for IMS Partition Initialization utility.
- Creates ILDSs.

- Supports various formats of the unloaded data sets.
- Includes the Physical Sequence Sort for Reload (PSSR) utility.
- Includes the Bitmap Resetter utility.
- Gives you the option to load compressed data that was previously unloaded by IBM IMS High Performance Unload for OS/390 in a compressed format.

Program number: 5655-M26

IBM IMS High Performance Unload for z/OS, V1.2

The IBM IMS High Performance Unload for z/OS tool:

- Unloads HALDB, HDAM, HIDAM, HISAM, and SHISAM databases.
- Allows you to unload broken data sets.
- Gives you the option to unload compressed data without decompression overhead.
- Provides multiple standard formats for unload data sets.
- Includes a variety of statistical reports for improved tuning.
- Provides an API that enables application programs to leverage the tool's high performance retrieval techniques efficiently.
- Provides a user exit facility for additional processing of each segment.
- Provides a Sequential Subset Randomizer utility,

Program number: 5655-E06

IBM IMS High Performance Prefix Resolution for z/OS, V3.1

The IBM IMS High Performance Prefix Resolution for z/OS tool:

- Enables you to resolve and update prefixes of IMS databases involved in logical relationships as a single job step.
- Eliminates the intermediate Work File 2 (WF2) and Work File 3 (WF3) data sets.
- Helps you avoid much of the I/O, tape handling, and DASD requirements that are often associated with prefix resolution and prefix update.
- Executes the prefix resolution and prefix update functions as replacements for the IMS Prefix Resolution and IMS Prefix Update utilities.
- Supports IMS Parallel Reorganization for z/OS, V3 single job step execution of database reorganization, prefix resolution, and prefix update tasks.

Program number: 5655-M27

IBM IMS High Performance Pointer Checker for z/OS, V3.1

The IBM IMS High Performance Pointer Checker for z/OS:

- Enables you to analyze corrupt databases quickly and reduce the amount of time spent to diagnose and repair them.
- Generates reports that facilitate system tuning, report space utilization, and detect and report problems in primary and secondary indexes.
- Provides the ability to set several new thresholds, including available extents, CA and CI splits, and database and data set last extents.
- Has significant improvements in performance to Full Checking capabilities: improved usability with simplified setup and operation; parallel processing of databases; and improved reporting.

- Provide ease of use and fast, easy detection of database status.
- Interactively and in batch mode repairs VSAM- and OSAM-organized IMS databases that contain pointer or data errors.
- Performs repairs quickly, thereby reducing the amount of time that the affected database is taken offline.
- Features VSAM and OSAM pointer repair, pointer navigation, a backout safety feature, and an ISPF front end.
- Runs in interactive mode, in which you can view entire blocks of data or individual IMS segments and you can navigate to other segments. Any changes made are tracked and can be undone.
- Runs in batch mode, in which you can dump blocks from the data set or submit changes to the block data.
- Integrates with IMS Parallel Reorg and HP Image Copy.

Program number: 5655-U09

IBM IMS Index Builder for z/OS, V3.1

The IBM IMS Index Builder for z/OS:

- Offers several features that improve overall performance and enhance ease of use.
- Builds or rebuilds primary and secondary indexes quickly.
- Allows the user to specify an optional output file where records that are needed for prefix resolution can be split off and written as they are read in.
- Eliminates the need to image copy indexes.
- Recognizes index records that have duplicate keys and writes the duplicate keys to a SYSOUT data set.
- Supports building IMS HALDB primary indexes.
- Integrates with the Database Recovery Facility, which enables the building of primary and secondary indexes during a database recovery.

Program number: 5655-R01

IBM IMS Library Integrity Utilities for z/OS, V2.2

The IBM IMS Library Integrity Utilities for z/OS:

- Manages IMS ACB, PSB, and DBD libraries:
 - DBD/PSB/ACB Compare, DBD/PSB/ACB Reversal, and DBD/PSB/ACB Mapper
 - Reporting information in a tabular form
 - Checking results from your desktop and ensuring that all PSBs and DBDs were processed as expected
 - Advanced ACBGen
 - Providing a high-speed generation process that greatly reduces the time needed to process large volumes of IMS ACBs
- Includes an Integrity Checker to help prevent system outages caused by databases corrupted by using the wrong DBD.
- Includes a Consistency Checker function to ensure that all the necessary definitions have been created for a database.
- Includes the MFS Reversal and MFS Compare utilities.

- Converts Message Format Services MID, MOD, DIF, and DOF control blocks back into Message Format Services utility control statements.
- Helps you recover the source and compare deltas if you lose your MFS source library or suspect a difference between the generated control blocks and the source.
- Provides useful summary reports of the IMS FORMAT library that show the relationships among the members.
- Cross tool integration provides LIU reporting in IMS Tools utilities and a LIU ISPF available in DB Control Suite.

Program number: 5655-U08

IBM IMS Database Solution Pack for z/OS, V2.1

The IBM IMS Database Solution Pack for z/OS is a utilities management tool that combines the entire set of IBM IMS Tools database products that are needed to manage IMS Full Function and HALDB databases into a single, consolidated solution.

IMS Database Solution Pack for z/OS provides helps keep databases operational and tuned. It helps reduce the operational complexity and the impact of database reorganization on system resources.

It provides a comprehensive set of high-performance utilities to unload, load, index build, reorganize, backup, verify, and report on full-function databases.

The IMS Database Solution Pack for z/OS includes the following tools:

- “IBM IMS Database Reorganization Expert for z/OS, V4.1” on page 242
- “IBM IMS High Performance Image Copy for z/OS, V4.2” on page 221
- “IBM IMS High Performance Load for z/OS, V2.1” on page 242
- “IBM IMS High Performance Pointer Checker for z/OS, V3.1” on page 243
- “IBM IMS High Performance Prefix Resolution for z/OS, V3.1” on page 243
- “IBM IMS High Performance Unload for z/OS, V1.2” on page 243
- “IBM IMS Index Builder for z/OS, V3.1” on page 221
- “IBM IMS Library Integrity Utilities for z/OS, V2.2” on page 244

The elements of both IBM IMS Online Reorganization Facility for z/OS and IBM IMS High Availability Large Database (HALDB) Toolkit for z/OS are also assimilated into IMS Database Solution Pack for z/OS , V2.1 and are new components of the Solution Pack.

Program number: 5655-DSP

IBM IMS Fast Path Solution Pack for z/OS, V1.3

The IBM IMS Fast Path Solution Pack for z/OS delivers extensive utilities that include functions a database administrator can use to analyze, maintain, and tune IMS Fast Path databases. Included are features that can help boost system availability by enabling the operations of a variety of key functions without taking the IMS database offline.

The IBM IMS Fast Path Solution Pack for z/OS includes:

- IMS Database Repair Facility

- IMS Fast Path Advanced Tool
- IMS Fast Path Basic Tools
- IMS Fast Path Online Tools
- “IBM IMS High Performance Image Copy for z/OS, V4.2” on page 221
- “IBM IMS Library Integrity Utilities for z/OS, V2.2” on page 244

Program number: 5655-W14

Chapter 26. Miscellaneous IBM tools that support IMS

The following IMS tools also support IMS 14.

IMS Application Development Facility (ADF), V2.2

IMS Application Development Facility is an architectural framework within which IMS application development and processing is accomplished.

IBM Application Performance Analyzer for z/OS, V13.1

The IBM Application Performance Analyzer for z/OS measures and reports how your applications use resources. The tool:

- Helps your business maximize the performance of your existing applications and improve the response time of your online transactions and batch turnaround times.
- Gives you the information you need to isolate performance problems in applications and test the effect of increased workloads on your systems.
- Monitors, analyzes and reports the performance of CICS, Assembler, COBOL, PL/I, C/C++, Db2, IMS and IBM MQ applications.
- Collects samples from the monitored address space and analyzes the system or resource application usage of CPU, DASD, I/O or the total address space.
- Features online analysis and reports that can be created as PDF or XML files, so that you can view them on workstations or transfer easily to other applications.
- Integrates with Fault Analyzer for z/OS and Debug Tool for z/OS.

Program number: 5697-Q09

IBM Application Recovery Tool for IMS and Db2 Databases

The IBM Application Recovery Tool for IMS and Db2 Databases minimizes recovery time, cost, and errors by creating a common point-in-time for data recovery and by synchronizing IMS and Db2 logs. The tool:

- Enables a range of database recovery functions in e-business transaction environments.
- Simplifies and coordinates batch application recovery of IBM IMS and IBM Db2 data to a common point, reducing the time and cost of data recovery.
- Recovers IMS and Db2 applications individually.
- Eases log, utility and process management for database recovery.
- Supports IMS High Availability Large Databases (HALDBs) and IMS SYSPLEX data sharing.
- Supports the latest versions of IMS and Db2.

Program number: 5697-F56

IBM DataRefresher™, Version 1.1

IBM DataRefresher provides you with facilities for copying, refining, and manipulating data from a source database or file on one system, and formatting it for a target database or file on the same, or another, system.

Program number: 5696-703

IBM IMS DataPropagator for z/OS, V3.1

The IBM IMS DataPropagator for z/OS® tool:

- Supports IBM MQ-based, asynchronous near real-time propagation that allows:
 - The Capture Component to capture IMS database changes performed by IMS Batch, IMS BMP, and IMS TM application programs
 - The Apply Component to transform the IMS database changes into relational format and apply them to Db2 tables on the same or on different systems from the IMS databases
 - IMS-to-IMS propagation
 - The transmission of the database changes from the Capture to the Target Component through the recoverable, buffered, and queued asynchronous messaging services of IBM MQ
- Reduces elapsed time between the IMS database updates and the Db2 for z/OS table updates to as little as a few seconds.
- Supports point-in-time propagation, allowing the content of the Db2 for z/OS target tables to reflect clearly identified logical points in time of the IMS source databases, for example, the logical end of a business day.
- Exploits the well established, high-performance, reliable queued messaging services of IBM MQ and allows enterprises to leverage their IBM MQ administration and operations skills.
- Improves performance by transmitting multiple IMS database changes within the same IBM MQ message, by compressing IBM MQ message data, and by exploiting the caching services of the Virtual Lookaside Facility (VLF) of z/OS.

Program number: 5655-E52

IBM Debug Tool for z/OS, V12.1 or V13.1

IBM Debug Tool for z/OS is an interactive source-level debugging tool for compiled applications in a variety of environments. The tool can:

- Debug an application interactively as it runs
- Perform seamless debugging of mixed-language applications
- Adjust an application while debugging
- Display, monitor, and alter program variables

Program number: 5655-W70, 5655-Q10

IBM Fault Analyzer for z/OS, 12.1 or 13.1

The IBM Fault Analyzer for z/OS assists developers in analyzing and fixing application and system failures. It offers developers information to help determine the cause of failure and assist in resolving the problem.

Program number: 5655-W69, 5655-Q11

IBM File Manager for z/OS for IMS Data, V12.1 or V13.1

The IMS component of File Manager (FM/IMS) is an ISPF application with which you can display, edit, update, create, copy, compare, print and erase your data files. This tool is a member of the IBM Problem Determination Tools suite.

Program Number: 5655-W68, 5655-Q12

IBM Hourglass, V7.1

HourGlass is a z/OS system application, allowing sites to alter the Date/Time returned to a z/OS application when a time request is made (SVC 11 or PC Time Requests). The tool:

- Patterns or specific applications, transactions, users, and address space names can be specified to limit the scope at which HourGlass will alter returned date/time information.
- Accurately simulates date and time processing for thorough application testing.
- Coordinates reporting and data transfers across time zones.
- Identifies applications that request the system date and time, and resolve potential problems before they occur.

Program Number: 5655-U59

IBM MQ for z/OS V8.0.0 and V9.0.0

IBM MQ for z/OS V8.0.0 and V9.0.0 offers a messaging powerhouse for the IBM System z platform as part of the universal messaging backbone. It delivers robust connectivity for flexible, reliable messaging for applications including Web services and Web 2.0. It includes market-leading JMS and now also offers publish and subscribe messaging.

Program Number 5655-W97, 5655-MQ9

WebSphere MQ for z/OS, V7.1.0

WebSphere MQ for z/OS V7.1.0 offers a messaging powerhouse for the IBM System z platform as part of the universal messaging backbone. It delivers robust connectivity for flexible, reliable messaging for applications including Web services and Web 2.0. It includes market-leading JMS and now also offers publish and subscribe messaging.

Program Number 5655-R36

Part 4. Featured integration solutions for IMS 14

IMS provides a wide variety of options for other software, middleware, and hardware products to connect to and integrate with both IMS transactions and data. Many of these products provide integrated connectivity features specific to IMS. A select few of the integration solutions provided by these products in coordination with IMS are featured here.

Chapter 27. IBM IMS Enterprise Suite for z/OS

The IMS Enterprise Suite for z/OS features independent components that facilitate application development and extends access to IMS transactions and data through use of industry standard tools, programming interfaces, and data protocols.

The components of the IMS Enterprise Suite enhance connectivity, expand application development, and extend standards and tools to facilitate integration with existing assets or other IBM solutions. The latest version is V3.2. V3.2 adds the support for IMS V14.

IMS Explorer for Development (IMS Explorer)

IMS Explorer is an Eclipse-based graphical tool that simplifies IMS application development tasks by displaying and enabling editing of IMS databases, segments, fields, and more, from an industry-standard IDE. IMS Explorer also provides the support for creating REST services from IMS transactions to run on IBM z[®]/OS Connect Enterprise Edition. You can also deploy and test the REST services directly from IMS Explorer.

IMS Explorer enables IMS application developers and database architects and developers to:

- Perform common and essential tasks in an end-to-end application development lifecycle
- Simplify the development and visualization of database description (DBD) and program specification block (PSB) resource definitions
- Import COBOL and PL/I data structures to an IMS database by using the importers of, and shell-sharing with, IBM Developer for System z, to generate PSB source, and to import and export DBD and PSB source from or to a z/OS remote system
- Leverage the IMS Universal drivers, thus offering a relational view of IMS data and offering new function, such as graphical assistance to build SQL statements
- Populate the IMS catalog
- Model, create, test, and publish mobile services to access IMS transactions

V3.2 adds the support for generating and editing the DDL for an existing PCB or table in the SQL editor.

IMS Explorer for Development provides enhancements to most functions provided by the IMS Enterprise Suite DLIModel utility plug-in, which is no longer supported.

If you use the IMS Enterprise Suite DLIModel utility plug-in, migrate to the IMS Enterprise Suite Explorer for Development instead.

Although IMS Explorer for Development includes enhancements to most IMS Enterprise Suite DLIModel utility plug-in functions, it does not provide support for IMS database web services or IMS XML DB.

- For IMS database web services, you can generate web-enabled IMS database queries by shell-sharing IMS Explorer for Development with IBM Data Studio.

- The XML data support in IMS databases is discontinued in IMS 14. For earlier versions of IMS, the support is discontinued when IMS Version 12 goes out of service.

IMS Connect API for Java

IMS Connect API for Java provides programming control of connections to IMS Connect, of interactions with IMS, and of the data that is sent to IMS for those interactions in the Java programming language. The API provides a simple way to describe TCP/IP socket connections, interaction protocols, message headers, and data through the concepts of reusable profiles. More granular, lower-level calls are provided for more granular controls. The APIs can be used to develop custom IMS Connect TCP/IP client applications in Windows and z/OS environments.

Support for the C API is discontinued in V3.2.

IMS Data Provider for Microsoft .NET

The IMS Data Provider for Microsoft .NET provides the solution for Microsoft .NET-based applications to access and manipulate IMS data. .NET application developers can use their preferred development environment, such as Microsoft Visual Studio, to call the provided APIs.

The IMS Data Provider for Microsoft .NET requires IMS 13 or later.

IMS SOAP Gateway

IMS SOAP Gateway enables IMS applications to interoperate outside of the IMS environment through the SOAP protocol to provide and request services that are independent of platform, environment, application language, or programming model. IMS applications can become web services or send callout requests to external web services.

The enhancements for V3.2 include:

- Dynamic specification of target destination and XML converter for synchronous callout requests
- Installation process simplification through reduction of manual JCL job modifications and submission
- Support of IBM z/OS Communications Server AT-TLS feature for the callout scenario

IMS Mobile Feature Pack

IMS Mobile Feature Pack provides a solution to create, test, publish, and manage RESTful mobile services to access IMS transactions through the *IMS Mobile feature*, a feature that is installed into either IBM z/OS Connect EE V2.0 (imsmobile-2.0 feature), or WebSphere Application Server Liberty Profile z/OS Connect (imsmobile-1.0 feature).

Important:

- The latest IMS Mobile feature for z/OS Connect EE (imsmobile-2.0 feature) is now made available as the *IMS service provider* that is included directly in z/OS Connect EE V2.0.5 (APAR PI70432) or later. This change significantly improves the installation experience by reducing the steps required to obtain and install the code and to start using the IMS mobile feature. The IMS Mobile Feature Pack component for z/OS Connect EE in IMS Enterprise Suite is removed by APAR PI70548.
- The IMS Mobile feature for WebSphere Liberty Profile z/OS Connect (imsmobile-1.0 feature) is deprecated. IMS Enterprise Suite V3.2 is the

| last release to include this component. Customers should use the IMS
| service provider that is bundled with z/OS Connect EE V2.0.5 or later.

| For the latest information about the IMS Mobile feature in z/OS Connect
| EE, see Using the IMS service provider in the IBM Knowledge Center.

Java Message Service (JMS) API

The JMS API enables IMS applications that run in Java message processing (JMP) or Java batch processing (JBP) regions to issue synchronous callout requests to external services. To use the JMP and JBP support for synchronous callout, the JMS version 1.1 interface classes are required.

Some of the IMS Enterprise Suite components include the requisite open source code.

All of the IMS Enterprise Suite components support standard installation processes. The components that run on z/OS support SMP/E and the components that run on distributed platforms support or use the IBM Installation Manager.

For more information about the versions of IMS Enterprise Suite that works with IMS V14, see the IMS Enterprise Suite software requirement topic.

Related reference:

“IMS Enterprise Suite software requirements” on page 16

Chapter 28. Mobile and REST API solution with IBM z/OS Connect Enterprise Edition

You can build REST services and APIs to access your IMS assets by using the IMS service provider in IBM z/OS Connect Enterprise Edition (z/OS Connect EE). Your IMS™ applications can also make REST API calls through z/OS® Connect EE.

z/OS Connect EE provides a framework that enables z/OS-based programs and data to participate fully in the new API economy for mobile and cloud applications.

z/OS Connect EE is designed to provide a fast, secure, and reliable connector that accepts standard protocols and can interact with multiple z/OS backend programs and systems. z/OS Connect EE provides a standard interface for these assets to be identified and reached by using the Representational State Transfer (REST) technology, and services are represented in the JavaScript Object Notation (JSON) format. This interface also handles data conversion as needed and provides authorization services to allow or deny users that are based on roles. In other words, z/OS Connect EE is the API gateway into z/OS, with a single, configurable, high throughput REST/JSON interface into CICS, IMS, Db2, and WebSphere Application Server.

In the latest release, z/OS Connect EE V3, tooling support for creating IMS services and developing, deploying, and testing REST APIs to access the services is provided through z/OS Connect EE V3 API toolkit, an Eclipse-based workstation tool that is installed into IBM Explorer for z/OS. After an API is tested and deployed, it can be published to a developer portal. API subscription and management can be handled products such as IBM API Connect™.

z/OS Connect EE V3.0.1 adds the support for z/OS applications to act as an API consumer, issuing RESTful API calls to utilize the underlying services and data from external request endpoints.

IMS as an API provider

IMS service provider that is included in z/OS Connect EE is known as the *IMS Mobile feature* for building mobile services to access your IMS assets. The IMS service provider handles the communications with IMS and data transformation between JSON and byte arrays.

For more information about z/OS Connect EE and the IMS service provider, see *Using the IMS service provider* (in z/OS Connect EE documentation in IBM Knowledge Center).

IMS as an API consumer

z/OS Connect EE provides a Build toolkit that generates, from the API documentation (Swagger document), a set of files to enable an IMS application to call a REST API through z/OS Connect EE. These files include what the z/OS Connect EE server needs to handle data mapping and data transformation, and what the IMS application needs to send the API request and process the response.

| For more information, see z/OS applications to call REST APIs (in z/OS Connect EE
| documentation in IBM Knowledge Center).

| For more information about API economy and IMS, see the API economy solution
| adoption kit.

| IBM z/OS Connect Enterprise Edition V3.0 Program number: 5655-CE3

Chapter 29. IBM Cognos Business Intelligence solution


You can connect an IBM Cognos® Business Intelligence server directly to IMS operational data to create drill-to-detail reports with online access to your critical business data.

An IMS database typically contains operational data for high-volume transactional workloads. These workloads include finance, insurance, and retail applications that contain a wealth of valuable data about customers, interactions, and business trends. In a typical business intelligence scenario, the operational data must be warehoused and pre-processed before it is accessible to the IBM Cognos Business Intelligence (BI) report author. However, relying only on warehoused data adds constraints to the flexibility of a business reporting application. Because the report author has access only to the warehoused data and not the online transactional system, the most up-to-date detail information is not available. IBM Cognos BI 10.2 offers a data connection type for direct connections to IMS databases.

This solution requires the following IMS components:

- IMS Catalog
- IMS Common Service Layer (CSL)
- IMS Open Database Manager (ODBM)
- IMS Connect

Related information:

 [IBM Business Analytics Proven Practices: Best Practices for Using IMS Data in IBM Cognos BI Reports](#)

Chapter 30. Data discovery with IBM Watson Explorer

You can connect IBM Watson™ Explorer directly to IMS operational data to explore your critical business data.

An IMS database typically contains operational data for high-volume transactional workloads. These workloads include finance, insurance, and retail applications that contain a wealth of valuable data about customers, interactions, and business trends. IBM Watson Explorer provides users with a 360 degree view of their enterprise data.

This solution requires the following IMS components:

- IMS Catalog
- IMS Common Service Layer (CSL)
- IMS Open Database Manager (ODBM)
- IMS Connect

Program number: 5725-I17

Related information:

 [IBM Watson Explorer V11](#)

Chapter 31. Integrated access with IBM DataPower Gateway

The IBM DataPower Gateway provides integrated support for IMS 14. The DataPower Gateway supports both access to IMS databases and transactions from web-based clients, and synchronous callout requests to web services from IMS application programs.



The DataPower Gateway provides a single point of control to simplify, govern, and optimize the delivery of services and applications and enhance the security of XML and IT services. They extend and strengthen the capabilities of an infrastructure by providing a multitude of functions that can be managed through a web browser interface.

The DataPower Gateway provides three types of support for IMS: Access to databases in IMS DB.

- Access to IMS DB allows an external application to issue SQL calls against IMS databases by using the integrated IMS Universal JDBC driver that is delivered with DataPower Gateway.
- Access to IMS transactions that are running in IMS TM. Access to IMS TM through DataPower allows an external application to initiate a transaction request to an application program that is running in an IMS dependent region and fetch data back.
- Support for synchronous callout requests from application programs that are running in IMS to data or service providers that are running on the DataPower backend.

DataPower Gateway provides plug-in usability with little to no changes to an existing network or application software. No proprietary schemas, coding, or APIs are required to install or manage the device. DataPower Gateway supports XML integrated development environments to help reduce the number of hours that are spent in developing and debugging XML applications.

Related concepts:

-  [DataPower for IMS Implementation Guide](#)
-  [IBM DataPower Gateway Version 7.5](#)

Part 5. Appendixes

Notices

This information was developed for products and services offered in the US. This material might 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 grant you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US*

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

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

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 jurisdictions 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 websites are provided for convenience only and do not in any manner serve as an endorsement of those

websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide 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, MD-NC119
Armonk, NY 10504-1785
US*

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 document 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.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

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 actual people or business enterprises 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.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as follows:

© (your company name) (year).

Portions of this code are derived from IBM Corp. Sample Programs.

© Copyright IBM Corp. _enter the year or years_.

Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com)[®] are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at <http://www.ibm.com/legal/copytrade.shtml>.

Adobe, the Adobe logo, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

UNIX is a registered trademark of The Open Group in the United States and other countries.

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.

IBM Online Privacy Statement

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 IBM's Online Privacy Statement at <http://www.ibm.com/privacy/details> the section entitled "Cookies, Web Beacons and Other Technologies" and the "IBM Software Products and Software-as-a-Service Privacy Statement" at <http://www.ibm.com/software/info/product-privacy>.

Bibliography

This bibliography lists all of the publications in the IMS 14 library.

Title	Acronym	Order number
<i>IMS Version 14 Application Programming</i>	APG	SC19-4208
<i>IMS Version 14 Application Programming APIs</i>	APR	SC19-4209
<i>IMS Version 14 Commands, Volume 1: IMS Commands A-M</i>	CR1	SC19-4210
<i>IMS Version 14 Commands, Volume 2: IMS Commands N-V</i>	CR2	SC19-4211
<i>IMS Version 14 Commands, Volume 3: IMS Component and z/OS Commands</i>	CR3	SC19-4212
<i>IMS Version 14 Communications and Connections</i>	CCG	SC19-4213
<i>IMS Version 14 Database Administration</i>	DAG	SC19-4214
<i>IMS Version 14 Database Utilities</i>	DUR	SC19-4215
<i>IMS Version 14 Diagnosis</i>	DGR	GC19-4216
<i>IMS Version 14 Exit Routines</i>	ERR	SC19-4217
<i>IMS Version 14 Installation</i>	INS	GC19-4218
<i>IMS Version 14 Licensed Program Specifications</i>	LPS	GC19-4231
<i>IMS Version 14 Messages and Codes, Volume 1: DFS Messages</i>	MC1	GC18-4219
<i>IMS Version 14 Messages and Codes, Volume 2: Non-DFS Messages</i>	MC2	GC18-4220
<i>IMS Version 14 Messages and Codes, Volume 3: IMS Abend Codes</i>	MC3	GC18-4221
<i>IMS Version 14 Messages and Codes, Volume 4: IMS Component Codes</i>	MC4	GC18-4222
<i>IMS Version 14 Operations and Automation</i>	OAG	SC19-4223
<i>IMS Version 14 Release Planning</i>	RPG	GC19-4224
<i>IMS Version 14 System Administration</i>	SAG	SC19-4225
<i>IMS Version 14 System Definition</i>	SDG	GC19-4226
<i>IMS Version 14 System Programming APIs</i>	SPR	SC19-4227
<i>IMS Version 14 System Utilities</i>	SUR	SC19-4228
<i>Program Directory for Information Management System Transaction and Database Servers V14.01.00</i>		GI10-8988
<i>Program Directory for Information Management System Database Value Unit Edition V14.01.00</i>		GI13-4602
<i>Program Directory for Information Management System Transaction Manager Value Unit Edition V14.01.00</i>		GI13-4601

Index

Numerics

- 24-bit storage
 - OSAM 24-bit storage constraint relief enhancement 142
- 64-bit storage
 - IMS 64-bit storage manager enhancement 193
 - JVM support in Java dependent regions 153

A

- ACB (application control block)
 - IMS management of ACBs
 - coexistence considerations 84
 - IMS management of ACBs function software requirements 11
 - IMS-managed ACBs
 - enhancement 128
 - load from IMS catalog enhancement 128
- ACB management function software requirements 11
- ACBMGMT
 - IMS-managed ACBs enhancement 128
- ACBs (application control blocks)
 - fallback
 - IMS management of ACBs 73, 74
 - falling back from the IMS management of ACBs 73, 74
 - migration considerations 35
- accessibility
 - features x
 - keyboard shortcuts x
- ACEE
 - ACEE creation and management 209
- ACEE creation and management 209
- administration
 - IMS Explorer for Administration 214
- AOIE-type user exit routine enhancements 208
- APPC (advanced program-to-program communication)/IMS
 - flood control enhancement 145
- APPC flood control enhancement 145
- application control block (ACB)
 - IMS management of ACBs
 - coexistence considerations 84
 - IMS management of ACBs function software requirements 11
 - IMS-managed ACBs
 - enhancement 128
 - load from IMS catalog enhancement 128
- application control blocks (ACBs)
 - fallback
 - IMS management of ACBs 73, 74
 - falling back from the IMS management of ACBs 73, 74

- application control blocks (ACBs)
 - (continued)
 - migration considerations 35
- application management tools 219
- Application Performance Analyzer for z/OS 247
- application programs
 - PWFI
 - Dynamic refresh enhancement for PWFI and WFI regions 189
 - refreshing
 - Dynamic refresh enhancement for PWFI and WFI regions 189
 - WFI
 - Dynamic refresh enhancement for PWFI and WFI regions 189
- application programs supported 18
- Associate Thread exit routine enhancement 191
- attachment facility
 - ESAF subsystem type enhancement 192
- Audits
 - IMS catalog activity record enhancement 136
- AUTOEXPORT
 - Automatic export to the IMSRSC repository at IMS checkpoints 200
 - Automatic export to the IMSRSC repository at IMS checkpoints 200

B

- buffer pools
 - Shared queues buffer pool enhancement 179

C

- callout
 - synchronous
 - enhancements 180
 - ICAL call enhancement to support control data 180
- callout function software requirements 10
- catalog
 - fallback considerations for IMS-managed ACBs 73, 74
 - fallback considerations for the IMS catalog 71
- catalog, IMS
 - migration considerations 33
 - migration steps 33
- charge back
 - ODBM accounting enhancement 142

- checkpoint
 - IMSRSC repository
 - Automatic export to the IMSRSC repository at IMS checkpoints 200
- CICS subsystems supported 16
- codes
 - new, changed, and deleted 95
- coexistence
 - APAR summary 77
 - DBRC
 - small programming enhancements 80
 - exit routines 83
 - Fast Database Recovery (FDBR) 83
 - functional areas 79
 - IMS Universal drivers 89
 - Open Database enhancement 89
 - PTF summary 77
 - PTFs
 - unprocessed coexistence service, identifying 78
 - restrictions for IMS 14 93
 - unprocessed coexistence service, identifying 78
- coexistence considerations
 - ACBs, IMS management of 84
 - CQS 79
 - database recovery utilities 82
 - DEDB Alter enhancement 81
 - DRA 82
 - dynamic resource definition 82
 - general 78
 - HALDB alter enhancements 83
 - HALDB support for 8-GB OSAM data sets enhancement 83
 - IMS 64-bit storage manager 84
 - IMS catalog 84
 - IMS Connect 87
 - IMS management of ACBs 84
 - IMS repository function 85
 - MSC 88
 - number of PSTs 89
 - OTMA 89
 - Remote Site Recovery (RSR) 89
 - security exit routines 92
 - SECURITY macro 92
 - shared message queues 93
 - synchronous program switch 93
 - Syntax Checker 93
 - utilities 82
- Cognos Business Intelligence solution 259
- Command Control Facility for z/OS, IMS 235
- commands
 - enhancements 183
- Communication Controller for Linux on System z (CCL)
 - withdrawal from support 44

- compression
 - Image Copy 2 utility (DFSUDMT0) enhancement for zEnterprise Data Compression 117
 - zEnterprise Data Compression
 - Image Copy 2 utility (DFSUDMT0) enhancement 117
- concurrent active thread enhancement 142
- Configuration Manager for z/OS, IMS 239
- continuous delivery enhancements
 - 2017, February 112
 - 2017, January 112
 - 2017, July 110
 - 2017, June 110
 - 2017, March 111
 - 2017, May 111
 - 2017, October 110
- explanation 109
- overview 109
- PTF enhancements
 - 2017, February 112
 - 2017, January 112
 - 2017, July 110
 - 2017, June 110
 - 2017, March 111
 - 2017, May 111
 - 2017, October 110
- control blocks
 - Fast Path
 - changes in IMS 14 137
- Controlled activation of changed PSBs 136
- conversational transactions
 - OTMA conversation ID enhancement for QUERY OTMATI command 169
- cost of ownership, total (TCO)
 - reduced total cost of ownership enhancements 205
- coupling facility requirements 3
- CQS
 - shared queues overflow protection enhancement 179
 - usage feedback enhancement for shared queues 179
- Create RDDS from Log Records utility (DFSURCL0) enhancement 202

D

- DASD requirements 4
- data definition language (DDL)
 - Dynamic database definition enhancement 123
- Data Definition Language (DDL)
 - DDL for IMS enhancement 124
- Data Encryption for Db2 and IMS Databases, InfoSphere Guardium 233
- data entry database (DEDB)
 - buffers
 - SDEP, automatic management enhancement 116
- Data Explorer
 - See Watson Explorer

- data sets
 - OSAM
 - HALDB support for 8-GB OSAM data sets enhancement 140
- data sharing
 - requirements 3, 10
- Database Change Accumulation utility (DFSUCUM0)
 - fallback considerations 69
- Database Control Suite for z/OS, IMS 241
- Database Recovery Control (DBRC)
 - coexistence requirements 80
 - fallback considerations 69
 - migration to IMS 14 44
 - RECON data set
 - migrating 46
- Database Reorganization Expert for z/OS, IMS 242
- Database Resource Adapter (DRA)
 - coexistence considerations 82
 - software requirements 10
- Database Solution Pack for z/OS, IMS 213, 223, 245
- database versioning
 - migration considerations 31
- databases
 - defining
 - Dynamic database definition enhancements 123
 - Dynamic database definition enhancements 123
 - GSAM
 - metadata enhancement 139
- DataPower Gateway
 - IMS support 263
- Db2 for z/OS
 - subsystem supported 17
- DBFUHDR0 (High-Speed DEDB Direct Reorganization utility)
 - 64-bit buffer enhancement for Fast Path high speed processing 115
- DBRC
 - migration
 - changes to the RECON data set in IMS 14 45
 - RECON data set
 - changes in IMS 14 45
 - REPAIR.RECON command enhancement 118
- DBRC (Database Recovery Control)
 - coexistence requirements 80
 - fallback considerations 69
 - migration to IMS 14 44
 - RECON data set
 - migrating 46
- DDL (data definition language)
 - Dynamic database definition enhancement 123
- DDL (Data Definition Language)
 - DDL for IMS enhancement 124
- DEDB (data entry database)
 - buffers
 - SDEP, automatic management enhancement 116
- DEDB Alter enhancement
 - coexistence considerations 81

- DEDB Alter enhancement (*continued*)
 - migration considerations 32
- DEDB Alter function enhancements
 - ABC 120
- DEF
 - concurrent active thread enhancement 142
 - DEDB Alter function enhancements 120
 - Mobile workload reporting enhancement 154
 - MSC type-2 command enhancements 161
- DELETE DEFN command
 - IMS ID enhancement 203
- Deleted messages and codes for IMS 14 104
- dependent regions
 - Java
 - 64-bit support 153
 - JVM 64-bit support 153
- DFSUCUM0 (Database Change Accumulation utility)
 - fallback considerations 69
- DFSURCL0
 - Create RDDS from Log Records utility (DFSURCL0) enhancement 202
- diagnoses
 - tracing
 - IMS type-2 trace enhancements 204
- diagnosis
 - IMS trace enhancements 194
- discontinuance of support 28
- dispatching
 - OTMA dynamic SAPs enhancement 172
- DL/I calls
 - ICAL call
 - control data enhancement 180
- DRD (dynamic resource definition)
 - migration considerations 47, 55
 - migrating from non-DRD to DRD and the IMSRSC repository 48
 - migrating from non-DRD to DRD with RDDS 50
 - migrating from RDDS to the IMSRSC repository 51
 - migrating MSC resources to the IMSRSC repository 52
- Dynamic database definition
 - enhancement 123
- dynamic definition
 - MSC resources
 - fallback considerations 71
- Dynamic definition
 - MSC enhancement 162, 166
- Dynamic refresh enhancement for PWFI and WFI regions 189
- dynamic resource definition
 - Create RDDS from Log Records utility (DFSURCL0) enhancement 202
 - QUERY command enhancement for unexported resources 203
- dynamic resource definition (DRD)
 - MODBLKS resources
 - fallback considerations 70

E

enhancements

- 64-bit buffer enhancement for Fast Path high speed processing 115
- ACB management 128
- ACEE creation and management 209
- APPC flood control enhancement 145
- application control block (ACB) management 128
- application development 143
- Automatic export to the IMSRSC repository at IMS checkpoints 200
- base level 112
- buffer management, Fast Path DEDB SDEP 116
- command 183
- concurrent active thread enhancement 142
- continuous delivery, overview 109
- Controlled activation of altered PSBs 136
- CQS
 - shared queues overflow protection 179
 - shared queues usage feedback 179
- Create RDDS from Log Records utility (DFSURCL0) 202
- Data Definition Language (DDL) for IMS 124
- DBRC REPAIR.RECON command 118
- DEDB Alter 120
- DEDB SDEP buffer management 116
- DELETE DEFN enhancement for IMS IDs 203
- Dynamic database definition 123
- dynamic definition, MSC 162, 166
- Dynamic refresh enhancement for PWFI and WFI regions 189
- dynamic storage enhancement for OTMA descriptors 170
- ESAF Associate Thread exit routine 191
- ESAF connection pooling 193
- ESAF subsystem type 192
- External Subsystem Attach Facility (ESAF) 191
- Fast Path control block changes in IMS 14 137
- Fast Path DEDB SDEP buffer management 116
- FDBR in-doubt thread support 138
- GSAM metadata enhancement 139
- HALDB support for 8-GB OSAM data sets 140
- ICAL call enhancement to support control data for synchronous callout requests 180
- Image Copy 2 utility (DFSUDMT0) enhancement 117
- IMS 14 Database Manager overview 115
- IMS 14 system overview 183

enhancements (continued)

- IMS 14 Transaction Manager overview 145
- IMS 64-bit storage manager enhancement 193
- IMS catalog activity record enhancement 136
- IMS Connect and IMS TM enhancement for cascaded global transactions 146
- IMS Connect commands 194
- IMS Connect enhancements 194
- IMS Connect idle connection timeout (IDLETO) 198
- IMS Connect WTOR command interface option enhancement 199
- IMS Enterprise Suite 253
- IMS Monitor (IMSMON) user exit 206
- IMS Tools support 211
- IMS type-2 trace enhancement 204
- IMS-managed ACBs 128
- IMSRSC repository 200
- ISC enhancement for processing ERP messages 149
- Java environment dynamic scripting enhancement 150
- JVM 64-bit support in Java dependent regions 153
- JVM usage statistics 154
- load from IMS catalog 128
- Mobile workload reporting enhancement 154
- MSC 156
- MSC 24-bit storage reduction 156
- MSC dynamic definition 162, 166
- MSC simplified enablement 159
- MSC type-2 command enhancements 161
- ODBM accounting 142
- OSAM 24-bit storage constraint relief 142
- OTMA 169
- OTMA ACEE flood control 174
- OTMA automatic refresh of cached ACEEs when RACF definitions change 173
- OTMA client bid security enhancement 174
- OTMA conversation ID enhancement for QUERY OTMATI command 169
- OTMA dynamic SAPs 172
- OTMA security enhancements 173
- OTMA TPIPE parallelism 176
- OTMA trace enhancement for synchronous callout processing 181
- QUERY command enhancement for unexported resources 203
- QUERY OTMATI command 169
- recording IMS Connect RACF statistics for IMS DB connections 196
- recording IMS Connect RACF statistics for IMS TM connections 197
- reduced total cost of ownership 205

enhancements (continued)

- SDEP, Fast Path DEDB buffer management 116
- Shared queues buffer pool enhancement 179
- shared queues overflow protection 179
- shared queues usage feedback 179
- SQL support 143
- synchronous callout 180
- TCO (total cost of ownership), reduced 205
- total cost of ownership (TCO), reduced 205
- trace enhancement for OTMA synchronous callout processing 181
- user exit enhancements in IMS 14 206
- User exits enhancement
 - AOIE-type user exit routine 208
 - Automated Operator user exit routine, type-2 208
 - DFS AOEO0 user exit routine 208
 - Type-2 Automated Operator user exit routine 208
- ERP messages
 - ISC enhancements 149
- ESAF
 - ACEE creation and management 209
 - ESAF connection pooling 193
- ESAF (External Subsystem Attach Facility)
 - enhancements 191
 - ESAF Associate Thread exit routine enhancement 191
 - subsystem type enhancement 192
- ESAF Associate Thread exit routine enhancement 191
- ESAF connection pooling 193
- ETO Support for z/OS, IMS 235
- exit routines
 - coexistence 83
 - enhancements
 - AOIE-type user exit routine 208
 - Automated Operator user exit routine, type-2 208
 - DFS AOEO0 user exit routine 208
 - Type-2 Automated Operator user exit routine 208
 - ESAF Associate Thread exit routine enhancement 191
 - migration 55
 - migration considerations 55
 - software requirements 15
 - type-2 Automated Operator user exit routine
 - migration considerations 55
- exits
 - enhancements
 - AOIE-type user exit routine 208
 - Automated Operator user exit routine, type-2 208
 - DFS AOEO0 user exit routine 208
 - Type-2 Automated Operator user exit routine 208
 - IMS Monitor (IMSMON) user exit enhancement 206

- Explorer for Administration
 - See IMS Explorer for Administration
- Extended Terminal Option Support for z/OS, IMS 235
- External Subsystem Attach Facility (ESAF)
 - enhancements 191
 - ESAF Associate Thread exit routine enhancement 191
 - subsystem type enhancement 192

F

- fallback
 - considerations 68
- fallback considerations
 - Database Change Accumulation utility (DFSUCUM0) 69
 - DBRC 69
 - dynamic definition
 - MSC resources 71
 - dynamic resource definition (DRD)
 - MODBLKS resources 70
 - IMS catalog 71
 - IMS management of ACBs 73, 74
- Fast Database Recovery (FDBR)
 - coexistence 83
- Fast Path
 - control block changes in IMS 14 137
- Fast Path DEDB SDEP buffers
 - automatic management enhancement 116
- Fast Path high speed processing
 - 64-bit buffer enhancement 115
- FDBR in-doubt thread support
 - enhancements 138
- fix category for IMS 78
- FIXCAT 78
- FMH7 messages
 - ISC enhancements 149
- FMID (function modification identifier)
 - requirements for IMS 14 19
- function modification identifiers (FMIDs)
 - requirements for IMS 14 19

G

- GHI
 - concurrent active thread enhancement 142
 - DEDB Alter function enhancements 120
 - Mobile workload reporting enhancement 154
 - MSC type-2 command enhancements 161
- GSAM metadata enhancement 139

H

- HALDB alter enhancements
 - coexistence considerations 83
 - migration considerations 32
- HALDB support for 8-GB OSAM data sets enhancement 140
 - coexistence considerations 83

- HALDB support for 8-GB OSAM data sets enhancement (*continued*)
 - migration considerations 32
- hardware requirements
 - large sequential data set support 5
- High Performance Load for z/OS, IMS 242
- High Performance Pointer Checker for z/OS, IMS 243
- High Performance Prefix Resolution for z/OS, IMS 243
- High Performance System Generation (SYSGEN) Tools for z/OS, IMS 235
- High Performance Unload for z/OS, IMS 243
- High-Speed DEDB Direct Reorganization utility (DBFUHDR0)
 - 64-bit buffer enhancement for Fast Path high speed processing 115
- high-speed sequential processing (HSSP)
 - 64-bit buffer enhancement for Fast Path high speed processing 115
- HSSP (high-speed sequential processing)
 - 64-bit buffer enhancement for Fast Path high speed processing 115

I

- IBM API Connect 257
- IBM Cognos Business Intelligence solution 259
- IBM Management Console for IMS and Db2 for z/OS 214
- IBM MQ
 - subsystem supported 17
- ICAL DL/I call
 - control data enhancement 180
- IDLTO parameter
 - IMS Connect idle connection timeout (IDLETO) 198
- Image Copy 2 utility (DFSUDMT0)
 - enhancement for zEnterprise Data Compression 117
- image copy enhancements
 - hardware requirements 4
- IMS 14
 - base-level enhancements 112
 - enhancements
 - base level 112
 - FDBR in-doubt thread support 138
 - FMIDs 19
 - IMS Tools support 211
 - IRLM 9
 - packaging
 - FMIDs 19
 - programming languages written in 17
 - terminals supported 6
- IMS 14 changed messages and codes 101
- IMS 14 new messages 95
- IMS 64-bit storage manager enhancement 193
- IMS application management tools 219
- IMS catalog
 - coexistence considerations 84

- IMS catalog (*continued*)
 - fallback considerations 71
 - fallback considerations for IMS-managed ACBs 73, 74
 - migration considerations 33
 - migration steps 33
- IMS catalog activity record
 - enhancement 136
- IMS Cloning Tool for z/OS 225, 241
- IMS Configuration Manager for z/OS 239
- IMS Connect
 - coexistence considerations 87
 - IMS TM enhancement for cascaded global transactions 146
 - LOCAL option
 - discontinuance of support 30
 - end of support 30
 - support, end of 30
 - migration considerations 61
 - requirements 12
 - software requirements 12
 - SSL
 - discontinuance of support 30
 - end of support 30
 - support, end of 30
 - Timeout option for idle connections 198
- IMS Connect enhancements 194
 - commands 194
- IMS Connect Extensions for z/OS 229
- IMS Connect WTOR command interface
 - option enhancement 199
- IMS Database Control Suite for z/OS 241
- IMS Database Reorganization Expert for z/OS 242
- IMS Database Solution Pack for z/OS 213, 223, 245
- IMS DB
 - recording IMS Connect RACF statistics for IMS DB connections 196
 - recording IMS Connect RACF statistics for IMS TM connections 197
- IMS Enterprise Suite
 - components 253
 - software requirements 16
- IMS Explorer for Administration 214
- IMS High Performance Load for z/OS 242
- IMS High Performance Pointer Checker for z/OS 243
- IMS High Performance Prefix Resolution for z/OS 243
- IMS High Performance System Generation (SYSGEN) Tools for z/OS 235
- IMS High Performance Unload for z/OS 243
- IMS ID
 - DELETE DEFN enhancement 203
- IMS Library Integrity Utilities for z/OS 244
- IMS Mobile feature 257
- IMS mobile solution 257

- IMS Monitor (IMSMON) user exit enhancement 206
- IMS Program Restart Facility for z/OS 219
- IMS Queue Control Facility for z/OS 239
- IMS repository
 - migration considerations 62
- IMS repository function
 - coexistence considerations 85
- IMS service provider 257
- IMS Solutions
 - DataPower Gateway 263
 - IBM Cognos Business Intelligence 259
- IMS Sysplex Manager for z/OS 240
- IMS Tools
 - IBM Tools Base for z/OS 214
- IMS Tools support for IMS 14 211
- IMS Universal drivers
 - coexistence 89
 - software requirements 12
- IMS Version 10
 - programming considerations 21
- IMS Version 12
 - migrating to IMS 14 28
- IMS Version 14
 - base-level enhancements 112
 - enhancements
 - base level 112
- IMS Workload Router for z/OS 236
- IMSID
 - DELETE DEFN enhancement 203
- IMSplex
 - migration considerations 56
- IMSRSC repository
 - Automatic export to the IMSRSC repository at IMS checkpoints 200
 - IMS 14 enhancements 200
 - migration considerations
 - migrating from non-DRD to DRD and the IMSRSC repository 48
 - migrating from RDDS to the IMSRSC repository 51
 - migrating MSC resources to the IMSRSC repository 52
- InfoSphere Classic Change Data Capture for z/OS 227
- InfoSphere Data Explorer
 - See Watson Explorer
- InfoSphere Guardium Data Encryption for Db2 and IMS Databases 233
- InfoSphere Guardium S-TAP for IMS 233
- InfoSphere Optim Test Data Management Solution for z/OS 227
- INQY call
 - migration considerations 60
- installation
 - considerations
 - FMIDs 19
 - sample jobs for IMS 14 21
- Integration
 - DataPower support for IMS 263
 - z/OS Connect EE support for IMS 257
- integration solutions 253

- Intersystem Communication (ISC) supported subsystems 17
- IRLM
 - packaging 9
 - support for IMS 14 9
- ISC
 - enhancements
 - ERP messages, processing 149
 - FMH7 messages, processing 149
- ISC TCP/IP
 - software requirements 11
- IVP (installation verification program) migration considerations 61

J

- Java
 - dependent regions
 - 64-bit support, JVM 153
 - JVM 64-bit support, JVM 153
 - Java environment dynamic scripting 150
 - JVM 64-bit support in Java dependent regions 153
 - JVM usage statistics 154
- Java application
 - requirements 12
- Java environment dynamic scripting enhancement 150
- Java environment scripting
 - migration considerations 38
- JVM
 - Java environment scripting 150
 - usage statistics 154
- JVM 64-bit support
 - migration considerations 37
- JVM 64-bit support in Java dependent regions 153
- JVM usage statistics 154

K

- KBLA
 - discontinuance of support 28
 - IMS Records User Data Scrub utility (DFSKSCRO) 28
- KBLA utilities 28
- keyboard shortcuts x

L

- large sequential data set support
 - hardware requirements 5
- legal notices
 - notices 267
 - trademarks 267, 269
- Library Integrity Utilities for z/OS, IMS 244
- Load for z/OS, IMS High Performance 242
- load from IMS catalog
 - migration considerations 35
- load from IMS catalog enhancement 128
- LOCAL option
 - discontinuance of support 30
 - end of support 30

- LOCAL option (*continued*)
 - support, end of 30
- log records
 - new or changed 105

M

- messages
 - new, changed, and deleted 95
- metadata
 - GSAM metadata enhancement 139
- migration
 - catalog, IMS 33
 - discontinuance of support 28
 - IMS Version 12, migrating from 28
 - migrating IMS 14 from IMS Version 12 or earlier 28
 - MSC migration considerations 40, 41
 - procedure 26
 - RECON data set
 - changes in IMS 14 45
 - RSR (Remote Site Recovery) 63
 - skip-release migrations 28
 - steps 26
 - to IMS 14 DB 31
 - to IMS 14, DBRC 44
 - to IMS 14, general
 - recommendations 23
 - migration considerations
 - ACBs, IMS management of 35
 - APPC enhancements 37
 - CQS 44
 - database recovery utilities 31
 - database versioning 31
 - DEDB Alter enhancement 32
 - DRD (dynamic resource definition) 47, 55
 - migrating from non-DRD to DRD and the IMSRSC repository 48
 - migrating from non-DRD to DRD with RDDS 50
 - migrating from RDDS to the IMSRSC repository 51
 - migrating MSC resources to the IMSRSC repository 52
 - exit routines 55
 - fallback 68
 - HALDB alter enhancements 32
 - HALDB support for 8-GB OSAM data sets enhancement 32
 - IMS 64-bit storage manager 56
 - IMS Connect 61
 - IMS-managed ACBs 35
 - IMSplex 56
 - INQY call 60
 - IVP (installation verification program) 61
 - Java environment scripting 38
 - JVM 64-bit support 37
 - OTMA 43
 - REPAIR.RECON 45
 - security 66
 - synchronous program switch 44
 - Syntax Checker 68
 - type-2 Automated Operator user exit routine 55
 - XRF 44

- Mobile workload reporting
 - enhancement 154
- monitoring
 - IMS Monitor (IMSMON) user exit enhancement 206
- MSC (Multiple Systems Coupling)
 - coexistence considerations 88
 - dynamic definition
 - enhancements 162, 166
 - enhancements 156
 - 24-bit storage reduction 156
 - simplified enablement 159
 - migration considerations 40, 41
 - requirements, hardware 6
- MSC type-2 command
 - enhancements 161
- Multiple Systems Coupling (MSC)
 - coexistence considerations 88
 - dynamic definition
 - enhancements 162, 166
 - enhancements 156
 - 24-bit storage reduction 156
 - simplified enablement 159
 - migration considerations 40, 41
 - requirements, hardware 6

N

- Network Compression Facility for z/OS, IMS 229

O

- ODBM
 - accounting enhancement 142
 - recording IMS Connect RACF
 - statistics for IMS DB connections 196
 - recording IMS Connect RACF
 - statistics for IMS TM connections 197
- ODBM accounting enhancement 142
- OMEGACENTER Gateway on z/OS 237
- Open Database enhancement
 - coexistence 89
- Open Database solution
 - software requirements 15
- Open Transaction Manager Access (OTMA)
 - automatic refresh of cached ACEEs
 - when RACF definitions change 173
 - client bid security enhancement 174
 - conversation ID enhancement for QUERY OTMATI command 169
 - descriptors
 - dynamic storage enhancement 170
 - dynamic storage enhancement for OTMA descriptors 170
 - IMS 14 enhancements 169
 - QUERY OTMATI command enhancement 169
 - security enhancements 173
 - trace enhancement for synchronous callout processing 181

- Open Transaction Manager Access (OTMA) dynamic SAPs
 - enhancement 172
- operating system requirements
 - IMS 14 9
- OSAM
 - 24-bit storage constraint relief enhancement 142
 - HALDB support for 8 GB-OSAM data sets enhancement 140
- OTMA
 - ACEE flood control enhancement 174
 - coexistence considerations 89
 - migration considerations 43
 - TPIPE parallelism enhancement 176
 - tpipes
 - TPIPE parallelism enhancement 176
- OTMA (Open Transaction Manager Access)
 - automatic refresh of cached ACEEs
 - when RACF definitions change 173
 - client bid security enhancement 174
 - conversation ID enhancement for QUERY OTMATI command 169
 - descriptors
 - dynamic storage enhancement 170
 - dynamic storage enhancement for OTMA descriptors 170
 - IMS 14 enhancements 169
 - QUERY OTMATI command enhancement 169
 - security enhancements 173
 - trace enhancement for synchronous callout processing 181
- OTMA (Open Transaction Manager Access) dynamic SAPs
 - enhancement 172
- OTMA ACEE flood control
 - enhancement 174
- OTMA ACEE flood control
 - enhancement 174
- OTMA automatic refresh of cached ACEEs when RACF definitions change 173
- OTMA client bid security
 - enhancement 174
- OTMA conversation ID enhancement for QUERY OTMATI command 169
- OTMA security enhancements 173
- OTMA TPIPE parallelism
 - enhancement 176

P

- packaging
 - product 19
- parallel RECON access
 - requirements
 - hardware 6
 - software requirements 15
- Performance Analyzer for z/OS, IMS 230
- Pointer Checker for z/OS, IMS High Performance 243

- pooling
 - ESAF connection pooling 193
- Prefix Resolution for z/OS, IMS High Performance 243
- Problem Investigator for z/OS, IMS 230
- processor requirements 3
- Program Restart Facility for z/OS, IMS 219
- program specification blocks (PSBs)
 - activation of altered PSBs,
 - controlling 136
- program views
 - activation of altered PSBs,
 - controlling 136
- programming considerations for IMS 14 21
- programming languages
 - supported by IMS 14 17
- PSBs (program specification blocks)
 - activation of altered PSBs,
 - controlling 136
- PSTs
 - coexistence considerations 89
- PTFs
 - enhancements
 - 2017, February 112
 - 2017, January 112
 - 2017, July 110
 - 2017, June 110
 - 2017, March 111
 - 2017, May 111
 - 2017, October 110
- PWFI regions
 - Dynamic refresh enhancement for PWFI and WFI regions 189

Q

- QUERY command
 - enhancement to display unexported resources 203
- QUERY command enhancement for unexported resources 203
- Queue Control Facility for z/OS, IMS 239

R

- RACF
 - recording statistics for IMS DB connections 196
 - recording statistics for IMS TM connections 197
- RDDS (resource definition data set)
 - Create RDDS from Log Records utility (DFSURCL0) enhancement 202
 - migration considerations
 - migrating from non-DRD to DRD with RDDS 50
 - migrating from RDDS to the IMSRSC repository 51
 - migrating MSC resources to the IMSRSC repository 52
- RECON data set
 - changes in IMS 14 45
 - migrating 46

- RECON data set (*continued*)
 - migration
 - changes in IMS 14 45
 - REPAIR.RECON command
 - enhancement 118
- recording IMS Connect RACF statistics for IMS DB connections 196
- recording IMS Connect RACF statistics for IMS TM connections 197
- Recovery Solution Pack for z/OS 214, 222
- reduced total cost of ownership
 - enhancements 205
- release planning
 - how to use the Release Planning information vii
- Remote Site Recovery (RSR)
 - coexistence with IMS 14 89
 - discontinuance of support 29
 - end of support 29
 - hardware requirements for IMS 14 7
 - migration to IMS 14 63
 - support, end of 29
- REPAIR.RECON
 - migration considerations 45
- REPAIR.RECON command
 - enhancement 118
- repository
 - IMSRSC repository IMS 14
 - enhancements 200
- requirements
 - DASD 4
 - hardware
 - coupling facility 3
 - DASD 4
 - Multiple Systems Coupling (MSC) 6
 - parallel RECON access 6
 - processors 3
 - Remote Site Recovery (RSR) 7
 - shared message queues 3
 - system console 3
 - tape units 3
 - zIIP utilization 8
 - hardware and software for IMS 14 3
- Java applications
 - software 15
- software
 - application programs
 - supported 18
 - callout function 10
 - CICS subsystems supported 16
 - data sharing 10
 - Db2 for z/OS 17
 - DBRC 80
 - DRA 10
 - for IMS 14 9
 - IMS Enterprise Suite 16
 - IMS management of ACBs
 - function 11
 - IMS Universal drivers 12
 - IRLM 9
 - ISC TCP/IP 11
 - Java application 12
 - operating system requirements 9
 - programming languages
 - supported 17

- requirements (*continued*)
 - software (*continued*)
 - SQL support 15
 - software support
 - Intersystem Communication (ISC)
 - subsystems 17
 - softwareIBM MQ 17
- resource definition data set (RDDS)
 - Create RDDS from Log Records utility (DFSURCL0) enhancement 202
- resource definitions
 - exporting
 - IMSRSC repository at IMS
 - checkpoints 200
 - IMSRSC repository
 - Automatic export to the IMSRSC repository at IMS
 - checkpoints 200
- resources
 - exporting to a resource repository
 - Create RDDS from Log Records utility (DFSURCL0)
 - enhancement 202
 - QUERY command enhancement to show unexported runtime
 - resources 203
- REST services 257
- RSR (Remote Site Recovery)
 - coexistence with IMS 14 89
 - hardware requirements for IMS 14 7
 - migration to IMS 14 63

S

- sample installation jobs for IMS 14 21
- SAPs (save area prefixes)
 - OTMA dynamic SAPs
 - enhancement 172
- save area prefixes (SAPs)
 - OTMA dynamic SAPs
 - enhancement 172
- SCD control block
 - address of 83
- SDEP (sequential dependent segments)
 - buffers, automatic management
 - enhancement 116
- security
 - migration considerations 66
 - OTMA
 - automatic refresh of cached ACEEs
 - when RACF definitions
 - change 173
 - OTMA security enhancements 173, 174
 - security exit routines
 - coexistence considerations 92
- Security Guardium S-TAP for IMS 233
- SECURITY macro 28
 - coexistence considerations 92
 - discontinuance of support 28
- Security Reverification exit routine (DFSCTSE0)
 - migration considerations 55
- selective dispatching
 - OTMA dynamic SAPs
 - enhancement 172

- sequential dependent segments (SDEP)
 - buffers, automatic management
 - enhancement 116
- shared message queue requirements 3
- shared queues
 - CQS, shared queues usage feedback
 - enhancement 179
 - migration considerations 37
 - usage feedback enhancement for CQS
 - clients 179
- Shared queues
 - buffer pool enhancement 179
- Shared queues buffer pool
 - enhancement 179
- Signon/off security exit routine (DFSCSGN0)
 - migration considerations 55
- SMF records
 - IMS catalog activity record
 - enhancement 136
- SOA 253
- software
 - coexistence
 - DBRC small programming
 - enhancements 80
 - software requirements
 - callout function 10
 - DBRC 15
 - DFSPREC0 (HALDB Index/ILDS
 - Rebuild utility) 10
 - DRA 10
 - HALDB Index/ILDS Rebuild utility (DFSPREC0) 10
 - IMS Connect 12
 - IMS Enterprise Suite 16
 - IMS management of ACBs
 - function 11
 - IMS Universal drivers 12
 - Intersystem Communication (ISC) 17
 - ISC TCP/IP 11
 - Java applications 15
 - Open Database APIs
 - user exit enhancements 15
 - software requirements for IMS 14 9
- SQL
 - Data Definition Language (DDL) for
 - IMS enhancement 124
 - DDL (Data Definition Language) for
 - IMS enhancement 124
- SQL support
 - enhancements 143
 - requirements 15
- SSL
 - IMS Connect
 - discontinuance of support for
 - SSL 30
 - end of support for SSL 30
 - support for SSL, end of 30
- storage management
 - IMS 64-bit storage manager
 - enhancement 193
- subsystem attach
 - ESAF subsystem type
 - enhancement 192
- synchronous callout
 - enhancements 180

- synchronous callout (*continued*)
 - ICAL call enhancement to support control data 180
- synchronous program switch
 - coexistence considerations 93
 - migration considerations 44
- Syntax Checker
 - migration considerations 68
- syntax diagram
 - how to read viii
- Sysplex Manager for z/OS, IMS 240
- system console requirements 3
- System Generation (SYSGEN) Tools for z/OS, IMS High Performance 235

T

- tape-unit requirements 3
- TCO (total cost of ownership)
 - reduced total cost of ownership enhancements 205
- TCP/IP
 - requirements for IMS Connect 12
- terminals supported by IMS 14 6
- Timeout option for idle connections 198
- Tivoli Decision Support for z/OS 237
- Tivoli OMEGAMON XE for Db2
 - Performance Expert on z/OS 238
- Tivoli OMEGAMON XE for Db2
 - Performance Monitor on z/OS 238
- Tivoli OMEGAMON XE for IMS on z/OS 238
- Tivoli System Automation for z/OS 238
- TM resource adapter
 - LOCAL option
 - discontinuance of support 30
 - end of support 30
 - support, end of 30
- Tools
 - Application Performance Analyzer for z/OS 247
- tools support for IMS 14 211
- total cost of ownership (TCO)
 - reduced total cost of ownership enhancements 205
- tpipes
 - OTMA TPIPE parallelism enhancement 176
- trace enhancement for OTMA
 - synchronous callout processing 181
- trace enhancement, IMS type-2 204
- trace tables
 - IMS type-2 trace enhancement 204
- tracing
 - IMS trace enhancements 194
 - IMS type-2 trace enhancements 204
- trademarks 267, 269
- Transaction Analysis Workbench 231
- Transaction Authorization exit routine (DFSCSTRN0)
 - migration considerations 55
- transactions
 - cascaded global transactions, IMS Connect and IMS TM enhancement for 146

- Type 2 Automated Operator exit routine (DFSABOE00)
 - enhancements 208
- type-2 Automated Operator user exit routine
 - migration considerations 55

U

- Unload for z/OS, IMS High Performance 243
- user exit enhancements
 - AOIE-type user exit routine 208
 - Automated Operator user exit routine, type-2 208
 - DFSABOE00 user exit routine 208
 - Type-2 Automated Operator user exit routine 208
- user exit enhancements in IMS 14 206
- user exits
 - software requirements 15
- utilities
 - coexistence considerations 82

W

- Watson
 - IBM Watson Explorer 261
- Watson Explorer 261
- WFI regions
 - Dynamic refresh enhancement for PWFI and WFI regions 189
- WLM
 - Mobile workload reporting enhancement 154
- Workload Router for z/OS, IMS 236

X

- XRF
 - Communication Controller for Linux on System z (CCL)
 - withdrawal from support 44
 - full-function databases, support for
 - discontinuance of support 30
 - end of support 30

Z

- z/OS Connect EE 257
- zEnterprise Data Compression (zEDC)
 - Image Copy 2 utility (DFSUDMT0) enhancement 117
- zIIP utilization
 - requirements
 - hardware 8



Product Number: 5635-A05
5655-DSE
5655-TM3

Printed in USA

GC19-4224-02



Spine information:

IMS Version 14

Release Planning

