IMS
Version 15.1.0

Release Planning

(March 1, 2018 edition)

IBM

IMS
Version 15.1.0

Release Planning
(March 1, 2018 edition)

IBM

Contents

About this information

These topics provide general information to help you evaluate and plan for IMS™ 15. The topics describe the new features and enhancements for IMS 15, the hardware and software requirements for these new features and enhancements, considerations for migration and coexistence for IMS 15, and an overview of the IMS Tools that are enhanced to work with IMS 15.

This information is available in IBM® Knowledge Center.

For information about IMS 14, including its enhancements and considerations for migration and coexistence, see *IMS 14 Release Planning,* .

When you install IMS, you will also need information from the following information units:
* *IMS Version 15 Installation*
* *IMS Version 15 System Definition*
* *IMS 15: Program Directory for Information Management System Transaction and Database Servers*

How to use the Release Planning information

Use the IMS 15 Release Planning information to learn about the new enhancements in IMS 15 and to plan your migration.

The Release Planning information is organized into the following parts:
* Part 1, "General planning information for IMS 15," on page 1, which contains general information about the new release, including the following information:
   – A summary list of the enhancements included in IMS 15
   – An overview of the hardware and software requirements
   – Migration considerations for IMS 15
   – Coexistence considerations between the supported versions of IMS
   – A listing of the changed, new, and deleted messages and abends for IMS 15
   – A listing of the new and changed log records for IMS 15
*
* Part 3, "IBM IMS Tools support for IMS 15," on page 137, which describes IBM IMS Tools and products that support IMS 15.
* Part 4, "Featured integration solutions for IMS 15," on page 177, which contains overviews of some of the new ways that you can further integrate IMS 15 and the rest of your IT architecture.

Note: If you are migrating to IMS 15 from IMS Version 13 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.

```
►►──required_item─────────────────────────────────────────────►◄
            └─optional_item─┘
```
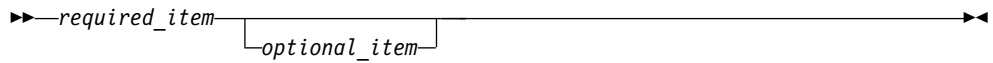
If an optional item appears above the main path, that item has no effect on the execution of the syntax element and is used only for readability.

```
              ┌─optional_item─┐
►►──required_item─────────────────────────────────────────────►◄
```
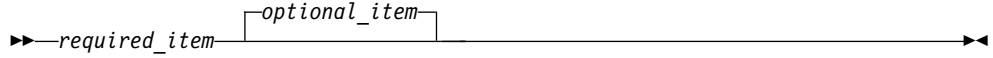
- If you can choose from two or more items, they appear vertically, in a stack.

  If you *must* choose one of the items, one item of the stack appears on the main path.

```
►►──required_item──┬─required_choice1─┬─────────────────────────►◄
                   └─required_choice2─┘
```
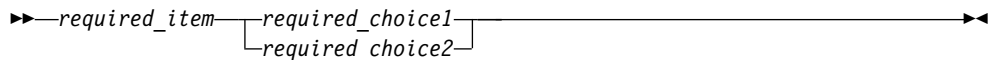
  If choosing one of the items is optional, the entire stack appears below the main path.

```
►►──required_item──────────────────────────────────────────────►◄
              ├─optional_choice1─┤
              └─optional_choice2─┘
```
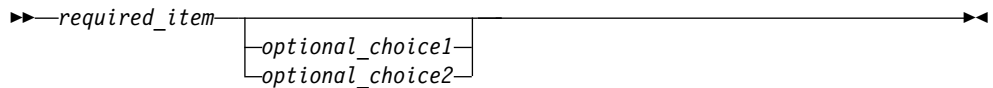
  If one of the items is the default, it appears above the main path, and the remaining choices are shown below.

```
              ┌─default_choice──┐
►►──required_item─────────────────────────────────────────────►◄
              ├─optional_choice─┤
              └─optional_choice─┘
```
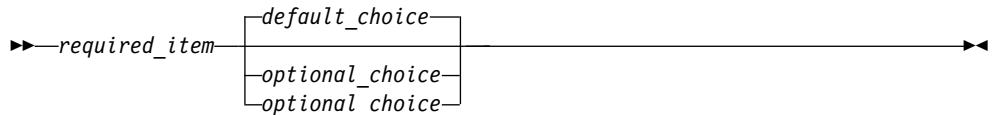
- An arrow returning to the left, above the main line, indicates an item that can be repeated.

```
              ┌───────────────┐
►►──required_item──▼─repeatable_item─┴─────────────────────────►◄
```

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

```
              ┌──────,────────┐
►►──required_item──▼─repeatable_item─┴─────────────────────────►◄
```

  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.

```
►►──required_item──┤ fragment-name ├───────────────────────────►◄
```

fragment-name:

```
├──required_item──────────────────────────────────────────┤
         └─optional_item─┘
```
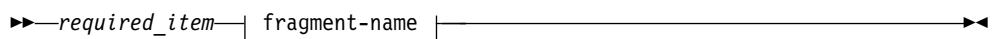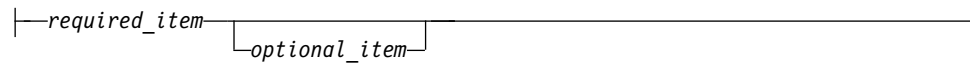
- In IMS, a b symbol indicates one blank position.
- Keywords, and their minimum abbreviations if applicable, appear in uppercase. They must be spelled exactly as shown. Variables appear in all lowercase italic letters (for example, *column-name*). They represent user-supplied names or values.
- Separate 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 15

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 15. 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 15 ISPF panel functions by using a keyboard or keyboard shortcut keys.

For information about navigating the IMS 15 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 15 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 15

In addition to the new functions and enhancements that are available, IMS 15 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 15 functions might affect your installation, list new, changed, and deleted messages and codes, and describe the IBM IMS Tools that support IMS 15.

Chapter 1. Hardware requirements

IMS 15 has base hardware requirements. Some individual functions have additional hardware requirements.

Processor requirements

IMS 15 runs only in z/Architecture® mode on an IBM System z10® processor or later.

The following table lists the processors supported by IMS 15.

Table 1. Supported IBM processors for IMS 15

Machine name	Machine type
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 2 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 15 has the following coupling facility requirements.

A coupling facility level of 15 or later is required for the following IMS 15 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 15 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).

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 V15.01.00*:
- 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 (DFSMStvs). Therefore, parallel RECON access requires Coupling Facility (CF) hardware in the System z sysplex.

Terminals and equipment supported by IMS 15

IMS 15 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 15.

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 15

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 15, 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

Table 3. Terminals that are supported by IMS 15, but withdrawn from IBM Marketing (continued)

IMS-supported product	Compatible product	Switched mode	Polled mode	Local mode	SNA	Notes
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 15

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

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 15 has base software requirements. Some individual functions have additional software requirements.

Operating software requirements

IMS 15 and its various functions have specific operating software requirements.

Before you install IMS 15, 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 15 is IMS1500.

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 V15.01.00*.

IMS 15 base software requirements

The base IMS 15 system runs on z/OS Version 2 Release 2 or later. Certain features and functions have additional software requirements.

IMS 15 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 2 (5650-ZOS) or later
  - The following z/OS V2R2 APARs must be installed:
    - OA47042
    - OA49728
    - OA51189
  - RACF®, or an equivalent product, if security is used. RACF is available with the IBM Security Server for z/OS (a separately orderable feature of z/OS V2R2).
  - IBM High-Level Assembler Toolkit, a separately orderable feature of z/OS V2R2.
  - DFSMS 2.2 or 2.3 APAR OA51385 for OLDS zHyperWrite support.
  - For z/OS data set encryption support, one of the following DFSMS versions:
    - DFSMS 2.2 with APAR OA50569 and its prerequisite and corequisite APARs installed.
    - DFSMS 2.3.

IMS 15 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 15 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.

Coexistence APARs

Certain functions of IMS 15 can coexist with IMS Version 13 and IMS 14 with the appropriate coexistence APARs applied.

See "Overview of coexistence APARs" on page 75 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 15. 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)
Related concepts:

⟶ IMS Enterprise Suite SOAP Gateway overview

⟶ IMS TM Resource Adapter overview

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 V2R2.0 or later (TCP/IP).
- To implement security, z/OS Security Server RACF or an equivalent product.
- 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 15, 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 15

Java™ applications that run in or access IMS 15 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 15 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 15, 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 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 8.0.2.10 or later (31-bit or 64-bit), available from IBM Support Fix Central.
- 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	*pathprefix*/usr/lpp/ims/ims15/imsjava/ imsudb.jar
IBM IMS Universal JDBC driver	*pathprefix*/usr/lpp/ims/ims15/imsjava/ imsudb.jar
IBM IMS Universal Database resource adapter	For use within WebSphere Application Server (both z/OS and distributed platforms): *pathprefix*/usr/lpp/ims/ims15/imsjava/ rar/imsudbJLocal.rar for JCA/JDBC local transaction processing only *pathprefix*/usr/lpp/ims/ims15/imsjava/ rar/imsudbJXA.rar for JCA/JDBC two-phase (XA) commit processing or local transaction processing *pathprefix*/usr/lpp/ims/ims15/imsjava/ rar/imsudbLocal.rar for CCI local transaction support *pathprefix*/usr/lpp/ims/ims15/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	*pathprefix*/usr/lpp/ims/ims15/imsjava/ imsudb.jar
IMS Universal JDBC driver	*pathprefix*/usr/lpp/ims/ims15/imsjava/ imsudb.jar
IMS Universal Database resource adapter	For use within WebSphere Application Server (both z/OS and distributed platforms): *pathprefix*/usr/lpp/ims/ims15/imsjava/ rar/imsudbLocal.rar for CCI programming interface to perform SQL or DL/I data operations *pathprefix*/usr/lpp/ims/ims15/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 Java Development Kit (JDK) 8.0.2.10 or later (31-bit or 64-bit), available from IBM Support Fix Central.
- 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.

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 CSLDC*xxx* 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 (DFSMStvs), 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 DFSDF*xxx* 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 15, 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 15 can be used with the following versions of IMS Enterprise Suite, although some components or functions might have specific IMS requirements.
- IMS Enterprise Suite for Distributed Systems V3.3
- IMS Enterprise Suite V3.2

Related information:

IMS Enterprise Suite Distributed Systems V3.3 overview

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 15 functions might include additional version requirements for CICS.

CICS Transaction Server for z/OS Version 4.2 (5655-S97) or later can connect to either the IMS 15 Database Manager (DB) or, using the appropriate TM interface, the IMS 15 Transaction Manager.

Db2 for z/OS subsystems supported

IMS 15 Transaction Manager can connect with Db2 for z/OS.

The IMS 15 Transaction Manager can be connected to any of the following Db2 products:
- Db2 11 for z/OS (5615-DB2) or later

IBM MQ subsystems supported

IMS 15 supports IBM MQ.

Intersystem Communication (ISC) subsystems supported

Using Intersystem Communication (ISC), the IMS 15 Transaction Manager can be connected to IMS 15 and earlier systems, to IBM CICS Transaction Server for z/OS, and to user-written software.

The IMS 15 Transaction Manager can be connected to the following products by using ISC:
- IMS 14 (5635-A05)
- IMS Version 13 (5635-A04)
- IBM CICS Transaction Server for z/OS Version 4.2 (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 15

IMS 15 is written in High Level Assembler Release 6, PL/X, C, C++, and JDK Version 8.

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 SDK for z/OS, Java Technology Edition, V8.0.2.10 (31-bit or 64-bit), 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 15 supports application programs that are supported by IMS Version 13 and IMS 14.

All application programs that are supported under IMS Version 13 and IMS 14 are still supported under IMS 15. In general, you should not have to recompile, reassemble, or rebind an IMS application program to run under IMS 15.

Chapter 3. Packaging for IMS 15

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 V15.01.00* for installation instructions.

Table 7. FMID requirements in IMS 15

FMID	Description	DB batch	DBCTL	DB/DC	DB/DC with XRF (¹)	DCCTL
HIR2230	Internal Resource Lock Manager V2R3	O	O	O	O	N
HMK1500 (²)	System Services component IVP component Database Recovery Control Logging component IMS Connect	R	R	R	R	R
JMK1501	Database Manager function	R	R	R	R	N
JMK1502	Transaction Manager function	N	N	R	R	R
JMK1503	Extended Terminal Option feature	N	N	O	O	O
JMK1506	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

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 15 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 15 Installation*, before proceeding with the migration and preparation of your own systems.

The following table lists the sample jobs that are shipped with IMS 15.

Table 8. Sample installation jobs that are shipped with IMS 15

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
DFSPROCB	COPY	Sample job to copy IMS sample JCL procedures to site library

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 15.

Chapter 5. Migration to IMS 15

The objective of migration support is to enable an orderly migration from your current release of IMS to IMS 15.

These topics describe general migration considerations for IMS 15 and migration considerations for specific new features and functions of IMS 15.

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 V15.01.00* for the most current hardware requirements, software requirements, prerequisites, and installation information. For general installation information, see *IMS Version 15 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 15.

  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 66 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 75.
- 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 (SDFS*xxxx*) and distribution (ADFS*xxxx*) 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 15 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 15.
- 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 15 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
      - 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 IMS1500 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 24.
   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 66.
      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 8,
      "Message and code changes in IMS 15," on page 91. 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 15 from IMS Version 13 or earlier

If you are migrating to IMS 15 from IMS Version 13 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 15 from IMS Version 13, in addition to the requirements and enhancements of IMS 15, your migration plan must also account for the requirements and enhancements that are introduced into IMS in IMS 14.

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.

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 was discontinued in IMS 14.

Remote Site Recovery (RSR)

IMS 15 and later systems do not support Remote Site Recovery (RSR). Installations that use RSR can instead use a multisystem IMSplex that includes geographically disperse cloned IMS systems for seamless failover protection and disaster recovery.

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 15 and later subsystems do no 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 15 and later systems do not 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.

MFS SOA support

IMS Message Format Service support for service-oriented architecture (MFS SOA) is discontinued in IMS 15 and later systems.

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 13 or IMS 14 to IMS 15.

The information about the IMS features and enhancements that were introduced in IMS 14 are not documented in the IMS 15 Release Planning information. If you are skipping one or more IMS releases when you migrate to IMS 15, 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 15 from IMS Version 12, also review the migration considerations in the following information:

IMS Version 14: Release Planning, GC19-4224

IMS Version 13: Release Planning, GC19-3658

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 15: DB

Specific migration considerations apply when you are migrating from the IMS Version 13 or IMS 14 Database Manager to the IMS 15 Database Manager.

These topics describe the IMS considerations for migrating to the IMS 15 Database Manager.

Database recovery utilities migration considerations

The IMS 15 Database Recovery (DFSURDB0) and Database Change Accumulation (DFSUCUM0) utilities accept as input log, image copy, and change accumulation data sets created by IMS Version 13 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 15, 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 14, 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 15:
- You regenerate version 1 of DBD1A by using the IMS 15 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 14. The IMS 15 instance of version 1 now becomes the active version of the DBD and the IMS 14 instance of version 1 is no longer used.
- You do not regenerate version 0 of DBD1A. Instead, you preserve the IMS 14 instance of the DBD1A version 0 in the IMS catalog. Application programs that use version 0 in IMS 15 continue to use the segment instances that were generated in IMS 14.

DEDB Alter enhancement migration considerations

The DEDB Alter utility has requirements for migration to IMS 15.

If you are invoking the DEDB Alter utility for an IMS 15 function, such as ALTERDB, the DEDB Alter utility detects the version of the IMS system and terminates if the IMS version does not support the particular change.

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. If you are migrating from IMS Version 13, the ACBs for the catalog must be generated from the DBDs and PSBs that IMS 15 supports. If you are migrating from IMS 14, you do not need to re-generate the catalog ACBs. 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 15 set of the existing DBD and PSB resources.When migrating from IMS Version 13, the IMS catalog holds a complete set of your existing resources and a complete set of IMS 15 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.

If you are migrating from IMS Version 13, you must generate the catalog ACBs from the DBDs and PSBs that IMS 15 supports by using the IMS 15 ACB Generation and Catalog Populate utility (DFS3UACB). The DBDs and PSBs for the catalog changed between IMS Version 13 and IMS 15. Step 3 is required.

If you are migrating from IMS 14, you do not need to generate new catalog ACBs from the DBDs and PSBs. The DBDs and PSBs for the catalog did not change between IMS 14 and IMS 15. Skip step 3.

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 15 if it is running.
3. If you are migrating from IMS Version 13, install the IMS 15 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 15 ACB Generation and Populate utility (DFS3UACB) to generate the ACBs for 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 run 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 15 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.
6. Restart the IMS 15 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)

Migrating an existing IMS catalog in systems that manage ACBs:

If you are migrating from IMS 14 with managed ACBs enabled to IMS 15, you can enable IMS 15 to support IMS directory ACBs that are currently being used by an IMS 14 system and load ACBs from the same IMS directory that the IMS 14 system is using. Install APAR PI79314 for IMS 15 to allow an IMS 15 system to support directory ACBs generated from an IMS 14 system. After installing both IMS 14 APAR PI82331 and IMS 15 APAR PI79314, IMS 14 and IMS 15 systems can share the same directory.

ACBSHR=Y must be specified for coexistence between IMS 14 and IMS 15 in a managed ACB environment, even if you are migrating a single IMS 14 system. If the IMS 14 directory was created in an ACBSHR=N environment, you must specify ACBSHR=Y in the IMS 14 system and re-create the IMS directory before the IMS 14 and IMS 15 systems can coexist with managed ACBs enabled.

The migration procedure for the IMS catalog differs depending on whether you are migrating a single IMS system or multiple IMS systems that share a catalog.

If you are migrating a single IMS system to IMS 15, step 4 is not required.

If you are migrating multiple IMS systems, or if you plan to have IMS 14 and IMS 15 systems coexist in a data sharing environment. Step 4 is required.

The following APARs do not apply to ACBs that are in an ACB library. If you require an ACBLIB in your IMS 15 managed ACB environment for the use of tools, utilities, batch jobs, or for any other reason, you must regenerate the ACBs in the ACBLIB by using the IMS 15 Catalog Library Builder utility (DFS3LU00).

1. If IMS 15 is running, shut it down.
2. Install the IMS 15 APAR PI79314 on your system to allow to use an IMS directory that contains ACBs generated by an IMS 14 system.
3. Restart the IMS 15 system with the same catalog and ACB management definitions as your IMS 14 system.
4. Install the IMS 14 APAR PI82331 on your IMS 14 system to allow IMS 14 to support directory ACBs generated by an IMS 15 system, and to make changes to the IMS directory from IMS 15. If you do not need to use your IMS 15 system to create or modify any ACBs while the IMS directory is shared with an IMS 14 system, APAR PI82331 is not required.

Related reference:

⮕ IMS Catalog Library Builder utility (DFS3LU00) (System Utilities)

Related information:

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 DFSDF*xxx* 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 15: TM

IMS considerations for migrating from IMS Version 13 or IMS 14 Transaction Manager to IMS 15 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 13 or IMS 14 Transaction Manager to IMS 15 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=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.

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 PI84302 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 PI84302 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_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.

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=DFSRRC00,
//            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.IMD0.SDFSJLIB,DISP=SHR
//         DD DSN=IMSVS.IMD0.SDFSRESL,DISP=SHR
//         DD DSN=CEE.SCEERUN,DISP=SHR
//         DD DSN=SYS1.CSSLIB,DISP=SHR
//STDENV DD DSN=hlq1.IMSCONF(IMS1ENV)
//       DD DSN=hlq2.IMSCONF(IMS1OPT)
//       DD DSN=hlq2.IMSCONF(IMS1MAIN)
//       DD DSN=hlq2.IMSCONF(IMS1DBUG)
//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":
```

```
# IMS1DBUG 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 15 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 MSNI*xxxx*, where *xxxx* is the link number. If MSNI*xxxx* 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 68

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

There are several considerations for migrating to IMS 15 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(*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.

OTMA migration considerations

IMS 15 includes migration considerations for OTMA.

Migration considerations introduced in IMS 15

Network security credential propagation enhancement

> If the security-data section of the OTMA message prefix contains network security credentials, the size of the OTMA message can increase by up to 504 bytes. Therefore, consider increasing the size of the SHMSG and LGMSG message queue data sets and the size of the message queue pool.

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.

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 15: System

IMS considerations for migrating from IMS Version 13 or IMS 14 systems to IMS 15 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 13 or IMS 14 systems to IMS 15 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.

Any customer- or vendor-written CQS client that processes CQS return, reason, or completion codes should be evaluated to see if it should be updated for the new codes.

See "Common Queue Server coexistence considerations" on page 76 for information about CQS coexistence rules.

DBRC migration considerations

Migrating DBRC to IMS 15 includes several tasks, including upgrading the RECON data set.

These topics describe the considerations and tasks for migrating DBRC to IMS 15.

Changes to the RECON data set:

Certain records in the RECON data set are new or changed from the records in
IMS Version 13 and IMS 14.

Changes introduced in IMS 15

In IMS 15, the following RECON records have changed fields:
- DSPRCNRC:
  - The RCNDATA flag within the RCNFIXED field is changed to indicate the
    new RECON header name.
  - The RCNMVERS flag within the RCNFIXED field is changed to indicate the
    current minimum version number during upgrade. The version number
    cannot be earlier than 13.1 (X'D1').
- DSPRCR1: The RCR1LVL flag within the RCR1VERS field is changed to indicate
  the RECON header extension record.
- DSPLOGRC: The LOGRELVL flag within the LOGFIXED field is changed to
  indicate the log release level, which is 15.1 in IMS 15 for all log record types.
- DSPOLDRC: The OLDRELVL flag within the OLDDENT(*) field is changed to
  indicate the OLDS release level, which is 15.1 in IMS 15 for all OLDS record
  types.
- DSPSSRC:
  - The SSRELLVL flag within the SSFIXED field is changed to indicate the
    subsystem release level, which is 15.1 (X'E1') in IMS 15.
  - The SSCOEXLV flag within the SSFIXED field is changed to indicate the
    subsystem coexistence level, which is 15.1 (X'E1') in IMS 15.

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.

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 13 or IMS 14 format to an IMS 15 format. You are not required to change the MINVERS value to when you migrate to IMS 15. 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 .

Attention: Do not issue the IMS 15 CHANGE.RECON UPGRADE command to upgrade the RECON data sets until all IMS Version 13, IMS 14, and IMS 15 systems that access the RECON data sets have the correct supporting products in place and have been tested for IMS 15 support.

Because Remote Site Recovery (RSR) is no longer supported after IMS 14, you must remove all Global Service Group (GSG) information before migrating to IMS 15. Issue the LIST.GSG command to identify any GSG and issue the DELETE.GSG command to remove it before upgrading the RECON data set.

Migration to IMS V15 from pre-IMS V13 may be achieved by migrating to the IMS V13 or the IMS V14 RECON data set format first and then using IMS V15 to issue the CHANGE.RECON UPGRADE command.

To upgrade an IMS Version 13 or IMS 14 RECON data set:

1. Apply the IMS 15 coexistence Small Programming Enhancements (SPEs) to all IMS Version 13 and IMS 14 systems before you upgrade the RECON data set. The coexistence SPEs (APARs/PTFs) are:

   - PI62555 for IMS Version 13
   - PI62558 for IMS 14

   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 15 DBRC Recovery Control utility (DSPURX00) or the IMS 15 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 '13.1' if the value was less than '13.1'. 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 15 Commands, Volume 3: IMS Component and z/OS Commands*. For details about the Query request, see *IMS Version 15 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 15 DBRC Recovery Control utility (DSPURX00) or the IMS 15 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 15 level, you can update the MINVERS value. Before you issue the CHANGE.RECON MINVERS('15.1') command, read "Minimum version value in the RECON data set" on page 67 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 13 or IMS 14, the process for enabling dynamic resource definition (DRD) in an IMS 15 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 15, 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 15 is cold started. If dynamic definition of MSC resources is also enabled, MSC resources are also automatically exported to the repository after IMS 15 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 DFSCG*xxx* member, CSLG=*xxx*. These parameters identify which DFSDF*xxx* and DFSCG*xxx* 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:
    - RDDSDSN=(*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 DFSCG*xxx* member of the IMS.PROCLIB data set
   - The <DYNAMIC_RESOURCES> section of the DFSDF*xxx* member of the IMS.PROCLIB data set

4. Allocate the system RDDS data sets that are specified on the RDDSDSN= parameter in the DFSDFxxx member.

5. Specify the IMS EXEC parameters DFSDF=*xxx* and, if MODBLKS=DYN was specified in the DFSCG*xxx* member, CSLG=*xxx*. These parameters identify which DFSDF*xxx* and DFSCG*xxx* 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 DFSDF*xxx* PROCLIB member to specify AUTOEXPORT=REPO, AUTOEXPORT=AUTO, or AUTOEXPORT=NO.
2. Modify the DFSDF*xxx* 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 DFSCG*xxx* member, CSLG=*xxx*. These parameters identify which DFSDF*xxx* and DFSCG*xxx* 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 MODID 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:

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 15.

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 15

IMS 15 and later systems obtain the user exit header block (UEHB) and any additional buffers that are passed to the DFSAOUE0 exit routine from 31-bit private storage.

The following user exit routines are removed from the IMS nucleus and are loaded as stand-alone load modules during IMS initialization:

- DFSCMTU0
- DFSCNTE0
- DFSCSMB0 and user-defined from TRANSACT EDIT=
- DFSCTTO0 and user-defined from TYPE EDIT= or LINEGRP EDIT=
- DFSFEBJ0
- DFSME000-DFSME127
- DFSPIXT0 and user-defined from TYPE EDIT= or LINEGRP EDIT=
- DFSSIML0

For more information about the migration considerations that are related to the removal of the preceding user exits from the IMS nucleus, see "Migration considerations for removing user exit routine specification from system definition" on page 63.

Migration considerations introduced in IMS 14

In IMS 14 you can dynamically refresh and query type-2 Automated Operator user exit routines (DFSAOE00 and other AOIE type exit routines).

To enable dynamic refresh and query support for the existing type-2 Automated Operator user exit routine (DFSAOE00), specify TYPE=AOIE,EXITS=(DFSAOE00) on the EXITDEF parameter in the USER_EXITS section of the DFSDF*xxx* 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.

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 76.
- 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 CSLDC*xxx* 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 13 (V13) CQS that is using a resource structure.
- One RM and one OM running on the LPAR.
- Two V13 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 15 (V15):
1. Prepare the IMSA libraries and data sets for IMS 15 (for example, perform a system definition, set up JCL, and so on).
2. Stop the V13 SCI.
3. Start the V15 SCI.
4. Start the V15 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 V13 OM.
6. Shut down IMSA.
7. Shut down IMSB.
8. Stop the V13 RM.
9. Stop the V13 CQS.
10. Start the V15 CQS.
11. Start the V15 RM.
12. Cold start IMSA.
13. Restart IMSB.

14. Test IMSA.
15. After IMSA (V15) 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 13 and repeat this process after any problems are fixed.
16. Stop the V13 OM.
17. Take IMSB offline and migrate it to IMS 15.

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 V13 CQS
   - One Common Service Layer (CSL), consisting of an OM, RM, and SCI
   - One V13 IMS system named IMSC

   LPAR2

   - One V13 CQS
   - One Common Service Layer (CSL), consisting of an OM, RM, and SCI
   - Two V13 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 15 (V15):
1. Migrate the subsystems on LPAR1 first by performing the following actions:
   a. Prepare the IMSC libraries and data sets for IMS 15 (for example, perform a system definition, set up JCL, and so on).
   b. Stop the V13 SCI.
   c. Start the V15 SCI.
   d. Stop the V13 OM (assuming the same OM name is used before and after the migration).
   e. Start the V15 OM.
   f. Shut down IMSC.
   g. Stop the V13 RM.
   h. Shut down the V13 CQS.
   i. Start the V15 CQS.
   j. Start the V15 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 13 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 15 (for example, perform a system definition, set up JCL, and so on).

b. Stop the V13 SCI.

c. Start the V15 SCI.

d. Stop the V13 OM.

e. Start the V15 OM.

f. Shut down IMSD.

g. Shut down IMSE.

h. Stop the V13 RM.

i. Stop the V13 CQS.

j. Start the V15 CQS.

k. Start the V15 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 13 and repeat this process after any problems are fixed.

o. After IMSD (V15) has tested successfully, proceed to the next step.

p. Stop the V13 OM.

q. Take IMSE offline and migrate it to IMS 15.

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 V13 CQS
- One Common Service Layer (CSL), consisting of an OM, RM, and SCI
- One V13 IMS system named IMSF
- One IMS Connect that is communicating with IMSF, IMSG, and IMSH

LPAR2
- One V13 CQS
- One Common Service Layer (CSL), consisting of an OM, RM, and SCI
- Two V13 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 15 (V15):
1. Migrate the subsystems on LPAR1 first by performing the following actions:
   a. Prepare the IMSF libraries and data sets for IMS 15 (for example, perform a system definition, set up JCL, and so on).

b. Stop the V13 SCI.

c. Start the V15 SCI.

d. Stop the V13 OM (assuming the same OM name is used before and after the migration).

e. Start the V15 OM.

f. Shut down IMS Connect.

g. Shut down IMSF.

h. Stop the V13 RM.

i. Shut down the V13 CQS.

j. Start the V15 CQS.

k. Start the V15 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 13 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 15 (for example, perform a system definition, set up JCL, and so on).

b. Stop the V13 SCI.

c. Start the V15 SCI.

d. Start the V15 OM.

e. Stop the V13 OM.

f. Shut down IMSG.

g. Shut down IMSH.

h. Stop the V13 RM.

i. Stop the V13 CQS.

j. Start the V15 CQS.

k. Start the V15 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 13 and repeat this process after any problems are fixed.

o. After IMSG (V15) has tested successfully, proceed to the next step.

p. Stop the V13 OM.

q. Take IMSH offline and migrate it to IMS 15.

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 13 or IMS 14 to IMS 15.

Consider the following items when planning to migrate IMS Connect support to IMS 15:

- In IMS 15, the CREATE IMSCON TYPE(PORT) command (Commands) command returns a new completion code when the number of ports that are defined to IMS Connect reaches 200, the new maximum number for defined ports.
- In IMS 15, 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)
- In IMS 15, IMS Connect requires at least 3 MB more storage than in previous releases. Adjust the IMS Connect region size accordingly.
- If both of the following situations occur, you might need to modify code that includes the HWSOMPFX macro:
  - The Network Session ID (NETSID) section or the Network User ID (NETUID) section, or both, is included in the security section of the OTMA message header.
  - Either the **DSECT=**ALL or the **DSECT=**NO option is specified with the HWSOMPFX macro.

  The size of the NETUID and NETSID sections can vary, causing the locations of the fields that are below the security section to change. However, if the **DSECT=**ALL or the **DSECT=**NO option is specified, a contiguous DSECT, the HWSOMPFX DSECT, is generated that does not account for sections that vary in size. Therefore, the fields in the OTMA message header that are below the security section might become inaccessible.

  For the fields of the OTMA message header that are below the security section to be accessed, you need to map the HWSOMUSR, HWSOMAPP, or HWSOMAPX DSECTS of the HWSOMPFX macro to the changed locations of the fields.

  For more information about the fields of the OTMA message header, see OTMA header fields used by IMS Connect (Communications and Connections).
- In IMS 15, processing by the HWSJAVA0 user message exit routine of the user data section that is in the OTMA message header is updated. If the OTMA message header contains network security information and the HWSOMPFX macro is used, the HWSJAVA0 exit routine specifies both the **DSECT=** and the **NETSEC_OPT=**YES options for the HWSOMPFX macro. The **DSECT=** and the **NETSEC_OPT=**YES options cause the following behaviors:
  - An individual DSECT is generated for each section of the OTMA message header.
  - The HWSECDNDS DSECT, or the HWSECARDS DSECT, or both, is generated to map network security information.
  - The HWSOMPFX DSECT is not generated.
- 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

Migrating TMRA connections from Local Option to TCP/IP:

IMS 14 is the last version to support the Local Option for connections between IMS Connect and IMS TM Resource Adapter. In IMS Version 13, Local Option is supported for 31-bit only. Migrating TMRA connections from Local Option to TCP/IP requires changes to the IMS Connect TCP/IP configuration member.

Local Option uses the z/OS Program Call function for communications between IMS Connect and IMS TM Resource Adapter that runs on WebSphere Application Server for z/OS

When you migrate from Local Option to TCP/IP connections, consider changing the following configurations:
- In the IMS Connect HWSCFGxx configuration member, change the LOCAL value that the PORT and PORTID parameters specify to a unique TCP/IP port number. Valid values for the TCP/IP port numbers are decimal numbers from 1 to 65535. For example, change from PORT=(ID=LOCAL) or PORTID=(LOCAL) to PORT=(ID=nnnn) or PORTID=(nnnn).
- In the IMS TM Resource Adapter connection factory Custom Properties pages, complete the following actions:
  - Disable Local Option connections by clearing the **IMSConnectName** field.
  - Enable TCP/IP connections by specifying the **HostName** and **PortName** fields.

Related reference:

➡ TCPIP statement (System Definition)

Related information:

➡ Configuring IMS connection factories (TM Resource Adapter)

➡ HWSX0916W (Messages and Codes)

Migration considerations for network security credential propagation enhancement

There are several migration considerations related to the network security credential propagation enhancement.
- If the security-data section of the OTMA message prefix contains network security credentials, the size of the OTMA message can increase by up to 504 bytes. Therefore, consider increasing the size of the SHMSG and LGMSG message queue data sets and the size of the message queue pool.
- If both of the following situations occur, you might need to modify code that includes the HWSOMPFX macro:
  - The Network Session ID (NETSID) section or the Network User ID (NETUID) section, or both, is included in the security section of the OTMA message header.
  - Either the **DSECT=**ALL or the **DSECT=**NO option is specified with the HWSOMPFX macro.

The size of the NETUID and NETSID sections can vary, causing the locations of the fields that are below the security section to change. However, if the **DSECT=**ALL or the **DSECT=**NO option is specified, a contiguous DSECT, the HWSOMPFX DSECT, is generated that does not account for sections that vary in size. Therefore, the fields in the OTMA message header that are below the security section might become inaccessible.

For the fields of the OTMA message header that are below the security section to be accessed, you need to map the HWSOMUSR, HWSOMAPP, or HWSOMAPX DSECTS of the HWSOMPFX macro to the changed locations of the fields.

For more information about the fields of the OTMA message header, see OTMA header fields used by IMS Connect (Communications and Connections).

- In IMS 15, processing by the HWSJAVA0 user message exit routine of the user data section that is in the OTMA message header is updated. If the OTMA message header contains network security information and the HWSOMPFX macro is used, the HWSJAVA0 exit routine specifies both the **DSECT=** and the **NETSEC_OPT=**YES options for the HWSOMPFX macro. The **DSECT=** and the **NETSEC_OPT=**YES options cause the following behaviors:
  - An individual DSECT is generated for each section of the OTMA message header.
  - The HWSECDNDS DSECT, or the HWSECARDS DSECT, or both, is generated to map network security information.
  - The HWSOMPFX DSECT is not generated.

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 15, 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 15 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 15 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 15 DFSURCL0 utility with IMS 15 log data sets. If you run with mixed versions, the results can be unpredictable.

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.

WADS migration considerations

IMS 15 introduced new changes to the write-ahead data set (WADS).

The WADS must be defined as a linear VSAM data set with a control interval (CI) size of 4 KB (4096-bytes), secondary space allocation of 0, and with the SHAREOPTIONS(3 3) parameter. The access method services (AMS) utility IDCAMS can be used to define the data set.

If the WADS for different IMS systems on different logical partitions (LPARs) share the same data set name, you must specify SHAREOPTIONS(3 3) for the WADS, even though the physical data sets are not shared between the systems.

Recommendation: Use different data set name for the IMS 15 WADS than the name used for the WADS in current IMS version. This allows you to predefine the IMS 15 WADS before shutting down the current IMS system. Otherwise, the WADS must be deleted and redefined after the current IMS system goes down and before the system is started as an IMS 15 system.

You can use the **ZHYPERWRITE=** parameter in the LOGGER section of the DFSDF*xxx* PROCLIB member to enable or disable the use of zHyperWrite to write to the WADS.

Example

The following JCL is an example of allocating the WADS:

```
//AMS EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *

DEFINE CLUSTER -
     (NAME(IMSA.WADS.VSAM) -
     VOLUME(VOL001) -
     CONTROLINTERVALSIZE(4096) -
```

```
        SHAREOPTIONS(3 3) -
        CYLINDERS (20 0) -
        LINEAR)
/*
```

Migration considerations for moving logger parameters to DFSDF*xxx*

IMS 15 includes migration considerations related to the consolidation of the IMS logger parameters into the mandatory LOGGER section of the DFSDF*xxx* member of the IMS PROCLIB data set.

The IMS logger parameters that were previously specified in the DFSVSM*xx* PROCLIB member are moved to the new LOGGER section of the DFSDF*xxx* PROBLIB member. Any logger parameters that are still specified in the DFSVSM*xx* member are ignored by the IMS system and not reported in the error log.

The **ARC=** and **WADS=** parameters, which were previously specified in the DFSPB*xxx* PROCLIB member or on the EXEC parameters of the control region JCL, are also moved to the DFSDF*xxx* PROCLIB member. If you specify **ARC=** or **WADS=** in DFSPB*xxx* or in the control region JCL, the specifications are ignored by the IMS system.

Migration considerations for sample procedures removal enhancement

IMS includes migration considerations for the SDFSPROC data set for sample procedures removal enhancement.

Migration considerations introduced in IMS 15

In IMS 14 and earlier, the **PROCLIB=** parameter of the IMSGEN macro determines whether SYSGEN generates the IMS sample procedures in the IMS.PROCLIB dataset.

In IMS 15, the **PROCLIB=** parameter of the IMSGEN macro is obsolete. Only **PROCLIB=NO** is allowed. The default value for the parameter is changed to **PROCLIB=NO**. If **PROCLIB=YES** or **PROCLIB=IMS** is specified, the IMS system generates an MNOTE return code 2 with message G918 issued.

Delete the **PROCLIB=** keyword, or change **PROCLIB=YES** or **PROCLIB=IMS** to **PROCLIB=NO** to avoid a return code of 2 during stage 1 system definition.

If sample procedures are used, copy them from IMS.SDFSPROC to IMS.PROCLIB data set. Rename the data sets if necessary to match the original data sets generated by SYSGEN.

For sample procedures that are not delivered during the SMP/E processing in the IMS SDFSPROC library, use the DFSPROCB JCL to create the sample procedures and rename the sample procedures whose names do not match what they would have been if they are generated by system definition.

For more information about using the DFSPROCB JCL to create and rename certain sample procedures, see Running the DFSPROCB job to complete sample procedures (System Definition).

In IMS 14 or earlier, most sample procedures generated by SYSGEN includes variable data that the IMS system sets based on various specifications in the stage 1 system definition macros. In IMS 15, because sample procedures are now

provided as copy source during SMP processing, you can update these sample procedures so that the variable data matches what they would have been if generated by SYSGEN when it is necessary.

In IMS 14 or earlier, some sample procedures omit entire sections of the sample procedure, or have different versions of the sample procedure, depending on the type of system or other specifications in stage 1 system definition macros.

In IMS 15, the new sample procedures, in general, assume a 'standard' DB/DC system, because the actual system environment is not known. For each procedure that has conditionally-generated statements for 'non-standard' environments, the prolog in the sample procedure includes detailed instructions on how the sample procedure should be modified.

Migration considerations for removing user exit routine specification from system definition

IMS 15 includes migration considerations for removing user exit routine specification from system definition.

Migration considerations introduced in IMS 15

The following user exit routines are removed from the IMS nucleus and are loaded as stand-alone load modules during IMS initialization:

- DFSCMTU0
- DFSCNTE0
- DFSCSMB0 and user-defined from TRANSACT EDIT=
- DFSCTTO0 and user-defined from TYPE EDIT= or LINEGRP EDIT=
- DFSFEBJ0
- DFSME000-DFSME127
- DFSPIXT0 and user-defined from TYPE EDIT= or LINEGRP EDIT=
- DFSSIML0

In order for the IMS system to properly load the user exits during IMS initialization, either move the user exit routines to an authorized library in the JOBLIB, STEPLIB, or LINKLIST library concatenated in front of the IMS.SDFSRESL, or ensure that the library that is currently specified by the USERLIB= keyword of the IMSGEN macro is added to the JOBLIB, STEPLIB, or LINKLIST concatenation.

The USERLIB= parameter on the IMSGEN macro is no longer used because no user exits are linked during Sysgen. G121 MNOTE0 is issued if USERLIB= is specified.

With the removal of user exit routines from the nucleus, some user-defined options in the system definition COMM macro are no longer valid and are replaced by IMS startup keyword parameter definitions in the DFSPBxxx proclib member or in the JCL override. The following table lists the COMM keywords affected, their replacement keywords in the DFSPBxxx, and the user exit routines that are affected by this change. If the COMM macro keywords are specified, the stage 1 system generation assembly issues an MNOTE 2.

Table 9. COMM macro changes

COMM keyword	DFSPBxxx keyword	User exit	Notes
FESEXIT = (<u>NO</u>|YES, timeout)	FESEXIT = <u>N</u>|Y FESTIM = timeout	DFSFEBJ0	Front End Switch routine. G118 MNOTE 2 is issued if FESEXIT=YES is specified in the COMM macro. The FESTIM= keyword in DFSPBxxx already exists before IMS 15.
MFSEXIT=(field, segment)	MFSEXITF=field MFSEXITS=segment	DFSME000-DFSME127	MFS input message field and segment edit routines. G119 MNOTE 2 is issued if MFSEXIT is specified in the COMM macro. Field specifies the range of field edit routines (000-field). Segment specifies the range of segment edit routines (segment-127).
OPTIONS=USERMSGS	USERMSGS=<u>N</u>|Y	DFSCMTU0	User message table. G120 MNOTE 2 is issued if OPTIONS=USERMSGS is specified in COMM macro.
SIMEXIT=<u>NO</u>|YES	SIMEXIT=<u>N</u>|Y	DFSSIML0	Shared printer message router routine. G117 MNOTE 2 is issued if SIMEXIT=YES is specified in the COMM macro.

If the user exit routines that are removed from the nucleus use callable services, the customer must link DFSCSI00 with each user exit.

Existing user exit routines must be examined to determine if any external references (VCONs) to the nucleus exist. Any such external references must be removed or link edit processing will fail due to unresolved external references.

The actual interfaces to the exits are not being changed. Unless the exits have VCONs to the nucleus, users don't need to update the code in the exits.

IMSCTF macro removal migration considerations

In IMS 15, the IMSCTF macro is no longer supported and is ignored during system generation. If the IMSCTF macro is specified, IMS System generation stage 1 assembly will issue message G122 in an MNOTE 2, which results in a return code of 2 for the assembly.

If you specified the following parameters in the IMSCTF macro, you must specify the parameter values by using the DFSPB*xxx* member of the IMS PROCLIB data set, JCL, or the DFSIDEF0 module for IMS 15.

The `CPLOG=` parameter

If you specified the **`CPLOG=`** parameter in the IMSCTF macro, use one of the following methods to define the checkpoint log frequency for IMS 15. If you do not use one of the following methods to specify the checkpoint log frequency for IMS 15, a default value of 500,000 is used.

- Specify the **`CPLOG=`** parameter in the DFSPBxxx member of the IMS PROCLIB data set.
- Specify the **`CPLOG=`** parameter in JCL.
- Issue the /CHANGE CPLOG command.

The `LOG=` parameter

In IMS 15, the IEFRDER DD statement for the system log is included in the DBBBATCH, DLIBATCH, IMSCOBGO, and IMSPLIGO procedures. The IEFRDER2 DD statement is included as a comment. To use the IEFRDER2 DD statement, remove the asterisks (*).

The IMSMON DD statement is included as a comment in the DBC, DCC, and IMS procedures for IMS 15. To use the IMSMON DD statement, remove the asterisks (*).

The PRDR= parameter

If you specified the **PRDR=** parameter in the IMSCTF macro, use one of the following methods to specify the name for the IMSRDR procedure for IMS 15. If you do not specify a name for the IMSRDR procedure for IMS 15, IMS uses a default name of IMSRDR as the name of the IMSRDR procedure that is started by the /START REGION command.

- Specify the **PRDR=** parameter in the DFSPBxxx member of the IMS PROCLIB data set.
- Specify the **PRDR=** parameter in JCL.

In addition, a sample IMSRDR procedure is supplied in the ADFSPROC and SDFSPROC library data sets. If you specify the **PRDR=** parameter in the DFSPBxxx member or in JCL, the name of the sample IMSRDR procedure does not change.

The RDS= parameter

In IMS 15, use one of the following methods to specify the buffer size for the restart data set (IMS.RDS):

- Specify the **RDS=** parameter in the DFSPB*xxx* member.
- Specify the **RDS=** parameter in JCL.

In IMS 15, you do not need to specify the device type on which the IMS.RDS data set resides.

The SVCNO= parameter

If you defined a type 2 SVC number by using the **SVCNO=** parameter on the IMSCTF macro, perform the following steps to use the type 2 SVC in IMS 15.

Important: If you specify the **SVC2=** parameter in the DFSIDEF0 module, ensure that the DFSIDEF0 module resides in an APF-authorized library that is included in the IMS Control Region JOBLIB or STEPLIB concatenation, or in the z/OS LINKLIST concatenation.

1. Define a type 2 SVC number by specifying the SVC2= keyword in one of the following locations. If you do not use the SVC2= keyword to define a type 2 SVC number, IMS uses a default value of 254.

   Important: If you use the Open Database Access (ODBA) interface or the database resource adapter (DRA) and you do not want to use the default value of 254 for the type 2 SVC number, you can define the number only by using the DFSIDEF macro of the DFSIDEF0 module.

   - In the DFSPBxxx member of the IMS PROCLIB data set. The value that you specify in the DFSPBxxx member overrides the value that is specified on the DFSIDEF macro.
   - In JCL. The value that you specify in JCL overrides the SVC2= keyword value that is specified in the DFSPBxxx member and the value that is specified on the DFSIDEF macro.
   - On the DFSIDEF macro of the DFSIDEF0 module by specifying the following code:

```
DFSIDEF TYPE=PARM,SVC2=
```

> The value that is specified on the DFSIDEF macro is overidden by the value that is specified in the DFSPBxxx member or in JCL.

2. If you used the SVC2= parameter on the DFSIDEF macro to define a type 2 SVC number, assemble and link the DFSIDEF0 module into IMS.SDFSRESL.
3. Run JCL to re-link the type 2 SVC routine, DFSVC200, as IGC*xxx*, where *xxx* is the type 2 SVC number. To generate sample JCL for re-linking the routine, specify `TYPE=GEN,SVC2=`*xxx* in the DFSIDEF macro.
4. Bind the type 2 SVC routine into the z/OS nucleus.

If you defined a type 4 SVC number by using the **SVCNO=** parameter on the IMSCTF macro, perform the following steps to use the type 4 SVC in IMS 15.

Important: If you specify the **SVC4=** parameter in the DFSIDEF0 module, ensure that the DFSIDEF0 module resides in an APF-authorized library that is included in the IMS Control Region JOBLIB or STEPLIB concatenation, or in the z/OS LINKLIST concatenation.

1. To define a type 4 SVC number, specify the following code on the DFSIDEF macro of the DFSIDEF0 module. If you do not use the SVC4= keyword to specify a type 4 SVC number, IMS uses a default value of 255.
   ```
   DFSIDEF TYPE=PARM,SVC4=
   ```
2. Assemble and link the DFSIDEF0 module into IMS.SDFSRESL.
3. Run JCL to re-link the type 4 SVC routine, DSP00MVS, as IGC00*yyy*, where *yyy* is the EBCDIC representation of the zoned-decimal type 4 value. You can use the sample JCL, which is generated when you specify `TYPE=GEN,SVC4=`*yyy* on the DFSIDEF macro, to re-link the routine.
4. Bind the type 4 SVC routine in an LPA or MLPA library.

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 you IMS system uses ACB libraries, the ACB Maintenance utility, ACBGEN, is required.
- If your IMS system uses IMS-managed ACBs...
- For each IMS that you are migrating to a pre-15 level, complete the following steps:
  1. Ensure that the status of all databases and PSBs updated by IMS 15 are correct.
  2. Resolve DBRC issues. See "DBRC fallback considerations" on page 67.
  3. Shut down IMS 15.
  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 15 messages to IMS Version 13 or IMS 14 message queues.

If you fall back from IMS 15 to IMS 14, ensure that the IMS 14 DBRC Coexistence APAR PI62558 (PTF UI44433) is applied to the IMS 14 system.

If you fall back from IMS 15 to IMS Version 13, ensure that the IMS Version 13 DBRC Coexistence APAR PI62555 (PTF UI44432) is applied to the IMS Version 13 system.

DBRC fallback considerations

Certain steps must be taken to revert the level of DBRC from IMS 15 to IMS Version 13 or IMS 14.

Minimum version value in the RECON data set

If the MINVERS value is set to '15.1'and you need to fall back from IMS 15 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 '15.1'.

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

1. Shut down all IMS 15 subsystems.
2. Ensure that all IMS 15 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 15 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 15.

To change the MINVERS value to '14.1', complete the following steps:

1. Shut down all IMS 15 subsystems.
2. Ensure that all IMS 15 subsystem records are removed from the RECON data set.
   - To view the subsystem records in the RECON data set, issue the LIST.SUBSYS command.
   - If IMS 15 subsystem records exist, issue the DELETE.SUBSYS command to delete them.
3. Issue the CHANGE.RECON MINVERS('14.1') command to reset the MINVERS value to fall back to IMS 14 from IMS 15.

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 15 system that uses the IMS repository function to an IMS 14 or IMS Version 13 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 15.

2. Cold start the IMS 14 or IMS Version 13 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. Reinstitute 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 15 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.
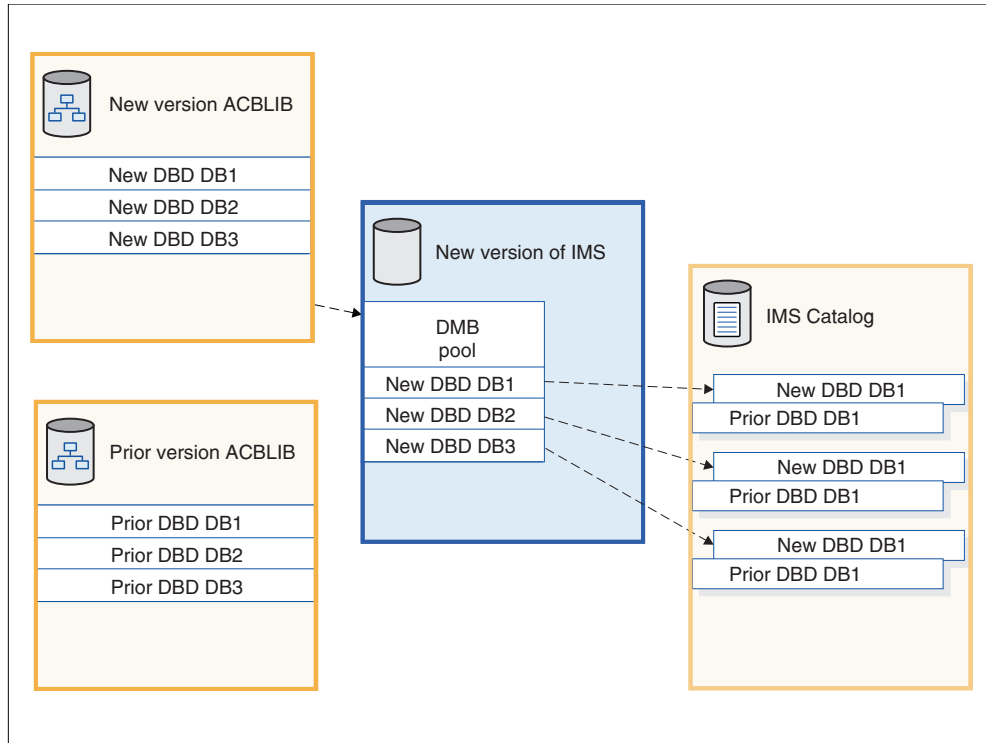
Figure 1. DBDs that IMS uses before fallback

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.
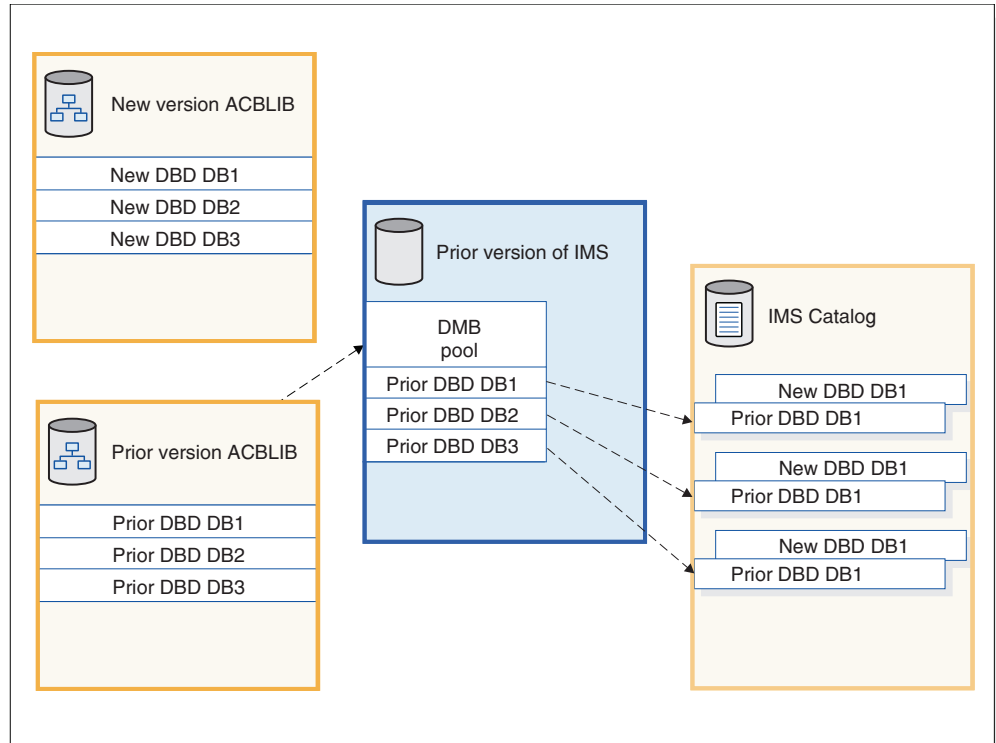
Figure 2. DBDs that IMS uses after fallback

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.

WADS fallback considerations

When you fall back from IMS 15 to a previous version of IMS, ensure that the WADS is allocated consistently with the prior version.

You must use a non-VSAM WADS data set in an IMS system of previous version.

If you defined WADS data set names for IMS 15 during migration that were different from the names used by the previous IMS version, use the previous names for fallback. The fallback process replaces the reference to the WADS data sets systematically. Otherwise, the WADS must be deleted and redefined after the IMS 15 system is shut down, and before the system is started at the previous IMS version.

At the end of fallback, cold start IMS with the FORMAT ALL or FORMAT WA keyword.

CQS automatic structure checkpoint fallback considerations

You can fall back from a common queue server (CQS) that can perform automatic structure checkpoint to an earlier version of CQS.

To fall back from an IMS 15 CQS that can perform automatic structure checkpoint to an IMS 14 or IMS Version 13 CQS, you must remove or comment out the **STRCHKPT** parameter in the CQSSLxxx member of the IMS PROCLIB data set.

Chapter 6. Coexistence with IMS 15

Restrictions and compatibility considerations apply for coexistence of IMS 15 with earlier versions of IMS.

If an IMS 15 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 13 and IMS 14 must have certain APARs installed to coexist with IMS 15.

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 10. APARs and PTFs needed for IMS 15 coexistence with IMS Version 13 and IMS 14

IMS 15 function	IMS Version 13 coexistence APAR/PTF	IMS 14 coexistence APAR/PTF	Additional information
DBRC RECON data sets	PI62555/UI44432	PI62558/UI44433	For more information, see "DBRC coexistence considerations" on page 77.
IMSRSC support for MSC	PI49334/UI32565	PI49208/UI32429	
OTMA TPIPE Parallelism Coexistence	PM93878/UK98632	No APAR needed	
IMS repository enhancements - Automatic export	PI27285/UI23504	No APAR needed	
IMS Managed ACB Coexistence	None	• PI79314/UI51473 • PI82231/UI51271	

Determining which coexistence service needs to be installed

You can identify the outstanding service that needs to be installed on your IMS Version 13 and IMS 14 systems to enable them to coexist with IMS 15 by using SMP/E and the IMS 15 FIXCAT category.

The FIXCAT category for IMS 15 is `IBM.Coexistence.IMS.V15` with the keyword `IMSV15COEX/K`.

The following is an example of the steps to determine what IMS service needs to be installed on IMS Version 13 or IMS 14 to coexist with IMS 15:

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 13 or IMS 14 zone (for example):

```
SET BOUNDARY (GLOBAL) .
    REPORT MISSINGFIX ZONES (targetzone)
    FIXCAT(IBM.Coexistence.IMS.V15)
```

Related information:
Enhanced HOLDDATA for z/OS
IBM Fix Category Values and Descriptions

General coexistence considerations

Although IMS 15 can coexist with earlier versions of IMS, general coexistence considerations apply.

IMS 15 can coexist with earlier versions, so existing applications and data can be used without change. Migration and coexistence support is provided for IMS Version 13 and IMS 14.

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 15 SVCs are downward compatible with IMS Version 13 and IMS 14. Only IMS 15 requires the IMS 15 SVCs.
- 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 15, see Chapter 9, "Log record changes in IMS 15," on page 99.
- **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 15 have specific coexistence considerations.

The following topics describe specific coexistence considerations for IMS 15.

Common Queue Server coexistence considerations

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

IMS 15 introduces the CQS automatic structure checkpoint function. An IMS 15 or later CQS that uses automatic structure checkpoint can coexist with IMS Version 13 and IMS 14 CQSs that are connected to the same shared queue structure. However, you can specify **STRCHKPT=** in the CQSSLxxx member of the IMS PROCLIB data set only in IMS 15 CQSs. Only IMS 15 or later CQSs can trigger automatic structure checkpoints, but all versions of CQS participate in the checkpoints.

General CQS coexistence rules

The following general coexistence rules apply to CQS in IMS 15:
- An IMS Version 13 or an IMS 14 can register with an IMS 15 CQS.
- An IMS 15 cannot register with a pre-IMS 15 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.
- User or vendor-written CQS clients that want to register with an IMS 15 CQS must be compiled with IMS Version 13 or later CQS macros.
- User or vendor-written CQS clients that are compiled with the IMS 15 CQS macros cannot register with a pre-IMS 15 CQS.
- 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.
- An IMS Version 13 CQS, an IMS 14 CQS, and an IMS 15 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 15 CQS supports clients that are assembled with the IMS Version 13, IMS 14, or IMS 15 CQS macros.
- A client that is assembled with the IMS 15 CQS macros can register only with an IMS 15 CQS. The IMS 15 client cannot register with an IMS Version 13 or an IMS 14 CQS. Similarly, a client that is assembled with the IMS 14 CQS macros cannot register with an IMS Version 13 CQS.

DBRC coexistence considerations

An IMS 15 Database Recovery Control (DBRC) instance can coexist with an IMS Version 13 DBRC or an IMS 14 DBRC if you install the DBRC coexistence small programming enhancements (SPEs) to the IMS Version 13 or IMS 14 systems, and upgrade your RECON data set to the IMS 15 format by issuing a CHANGE.RECON UPGRADE command. Additional coexistence considerations apply to other DBRC functions.

SPEs are available for IMS Version 13 and IMS 14 that enable the coexistence of the earlier version DBRC with DBRC for IMS 15. With the appropriate SPE installed, IMS 15 and earlier version DBRCs can share the upgraded RECON data set. If the RECON data set has not been upgraded to IMS 15, 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 15.

The following coexistence SPEs must be installed to the IMS Version 13 or IMS 14 DBRC for coexistence purposes:
- IMS Version 13: PI62555
- IMS Version 14: PI62558

Restriction: After a RECON data set is upgraded to the IMS 15 level, it is not accessible to any pre-IMS 15 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 15 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 15 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 14 Batch Backout utility cannot back out logs that were created on IMS 15.

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 14 Database Recover utility can use logs that were created on IMS Version 13 or IMS 14, but not on IMS 15.

DEDB Alter enhancement coexistence considerations

The DEDB Alter utility has requirements for coexistence in IMS 15.

You can run the DEDB Alter utility in a coexistence environment; however, if you are invoking the DEDB Alter utility for an IMS 15 function such as ALTERDB, the DEDB Alter utility detects the version of the IMS system and terminates if the IMS version does not support the particular change.

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 15 DBRC uses time-stamp precision to the microsecond.

Recommendation: Use DBRC with all IMS 15 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 15 utilities must be used whenever the input data for a DBDS contains log, image copy, or change accumulation records created by the IMS 15 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 14 utilities can process either IMS Version 13 or IMS 14 log records, but cannot process IMS 15 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 15 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 15 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 82

Exit routine coexistence considerations

Exit routines that run in IMS Version 13 or IMS 14 will work without modification in IMS 15. 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.

Related concepts:

"Exit routine migration considerations" on page 52

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 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 or later with the CDSLID value set to 2.

IMS 64-bit storage manager coexistence considerations

If the QUERY POOL command is routed to an IMS 15 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 14 and IMS 15 systems using IMS managed ACBs

When migrating multiple IMS systems in a data sharing environment, it might not be possible to shut down all IMS systems at once. Consequently, IMS 14 and IMS 15 systems must be able to support non-native directory ACBs for coexistence.

During migration, IMS 14 and IMS 15 systems use separate directory data sets. To allow IMS 14 and IMS 15 systems to support sharing the same directory data set, install the following APARs:

- APAR PI82231/UI51271 for IMS 14.
- APAR PI79314/UI51473 for IMS 15.

APAR PI79314 for IMS 15 allows an IMS 15 system to support directory ACBs generated from an IMS 14 system. APAR PI82231 for IMS 14 allows an IMS 14 system to support directory ACBs generated from an IMS 15 system.

Related concepts:

➡ Overview of coexistence APARs (Release Planning)

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 15 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 13, using DRD with an RDDS.
- IMSB is running IMS 14, using DRD with an RDDS.
- IMSC is running IMS 15 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 15, IMSRSC repository is enabled if the DFSDFxxx member has AUTOEXPORT=AUTO explicitly defined and autoexport export to the IMSRSC repository is enabled after IMS 15 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 13, using DRD with an RDDS, and using an IMS 15 CSL.
- IMSB is running IMS 14, using DRD with an RDDS, and using an IMS 15 CSL.

- IMSC and IMSD are running IMS 15 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(RDDS) 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(RDDS) 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 15 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 15 system.

IMS Connect coexistence considerations

The IMS 15 IMS Connect function supports IMS 14 and IMS Version 13, although new functions might not be available when connecting with the older versions.

IMS Connect instances from IMS 14 or IMS Version 13 might also be able to connect to an IMS 15 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
- IMS 15 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 15 Transaction Manager subsystem to other supported versions of IMS by using Multiple Systems Coupling (MSC).

The following IMS versions can coexist with an IMS 15 system in an MSC network:
- IMS 14 (5635-A05)
- IMS Version 13 (5635-A04)

For an IMS that was generated with MSC resources to coexist with IMS 15, 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 15 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 15 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 15, 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

Use one of the following methods to install the coexistence APARs that are listed in the preceding list:
- First, install 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 13 APAR/PTF PI49334/UI32565
  - IMS 14 preconditioning APAR/PTF PI49208/UI32429
  - IMS 14 preconditioning APAR/PTF PI45186

Network security credential propagation enhancement coexistence considerations

The network security credential propagation enhancement introduces some specific considerations for coexistence with IMS 15.

If network security credentials are sent from a client application of IMS TM resource adapter, the credentials can be passed to IMS and audited in IMS log records only if IMS, IMS Connect, and IMS TM resource adapter are all Version 15 or later. If network security credentials are included in messages from IMS TM resource adapter Version 15 and IMS Connect is Version 14 or earlier, the messages are sent to IMS Connect without the network security credentials, even if the IMS system is Version 15 or later. If network security credentials are sent from IMS TM resource adapter Version 15 and the IMS system is V14 or earlier system, you must modify the HWSJAVA0 user message exit to remove the network security credentials from the security-data section of the OTMA message prefix.

If network security credentials are sent from a user-written client application that uses the HWSSMPL0 or the HWSSMPL1 user message exit, the credentials can be passed to IMS and audited in IMS log records only if IMS and IMS Connect are both Version 15 or later. If network security credentials are sent from an application that uses HWSSMPL0 or the HWSSMPL1 and IMS Connect is Version 14 or earlier, the IRM extensions that contain the network security credentials are ignored, even if the IMS system is Version 15 or later. If network security credentials are sent from an application that uses HWSSMPL0 or the HWSSMPL1 and the IMS system is Version 14 or earlier, the user message exit removes the security credentials from the OTMA message prefix.

If network security credentials are included in IMS callout messages to IMS TM resource adapter Version 14 or earlier, the credentials are removed from the callout messages.

Open Database coexistence considerations

Coexistence restrictions apply to the use of Open Database.

The IMS Universal drivers that come with IMS 15 can be used by Java applications that access IMS 14 or IMS Version 13 databases as long as all the IMS systems are part of a mixed-version IMSplex that includes an IMS 15 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 CSLDC*xxx* member of the IMS PROCLIB data set.

OTMA coexistence considerations

When an IMS 15 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

Shared message queue coexistence considerations

Instances of IMS Version 13, IMS 14, and IMS 15 can share an IMS message queue
in an IMSplex.

Syntax Checker coexistence considerations

The IMS 15 Syntax Checker supports IMS Version 13, IMS 14, and IMS 15.

Be sure that the version shown is correct when you use the Syntax Checker to
check the parameters of earlier versions.

Restrictions for IMS 15 during coexistence

There are no general restrictions for IMS 15 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.

Chapter 7. Command changes in IMS 15

In IMS 15, a number of commands have been enhanced to support new functionality and some commands have been removed.

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.

If a command appears in more than one of the following subsections, the command was enhanced to support multiple new or changed functions in IMS 15.

Subsections:
- "Program Creation user exit routine (PGMCREAT) enhancement"
- "IMS Connect command enhancements"
- "RSR commands removed in IMS 15"
- "DBRC migration and coexistence" on page 90
- "Function level activation control enhancement" on page 90

Program Creation user exit routine (PGMCREAT) enhancement

Program Creation user exit routine (PGMCREAT) enhances the following commands:
- QUERY DB command (Commands) and QUERY PGM command (Commands) commands can now display a new definition type of PGMCREAT.
- QUERY USEREXIT command (Commands) and REFRESH USEREXIT command (Commands) commands can show PGMCREAT as new Type. For example,

```
TYPE(exit_type)
PGMCREAT - Program Creation user exit
```

IMS Connect command enhancements

In IMS 15, the CREATE IMSCON TYPE(PORT) command (Commands) command returns a new completion code when the number of ports that are defined to IMS Connect reaches 200, the new maximum number for defined ports.

RSR commands removed in IMS 15

Commands, command keywords, and command output that are specific to the discontinued Remote Site Recovery function have been removed or are otherwise no longer effective.

Support for RSR keywords on commands that are not specific to RSR has been removed. Some RSR-specific keywords return a message or code indicating the RSR is no longer supported, while other keywords are ignored.

The following commands that are specific to RSR have been removed from IMS 15:
- CHANGE.PRILOG TSLDS
- CHANGE.SECLOG TSLDS

- CHANGE.SG
- DELETE.GSG
- DELETE.SG
- /DISPLAY TRACKING STATUS
- GENJCL.RECEIVE
- INIT.SG
- INIT.GSG
- LIST.GSG
- RESET.GSG
- /START ISOLOG
- /START SERVGRP
- /START XRCTRACK
- /STOP SERVGRP
- /STOP XRCTRACK

DBRC migration and coexistence

DBRC migration and coexistence changes the default value of the `MINVERS` parameter to "13.1" in the following commands:
- /RMxxxxxx commands (Commands)
- CHANGE.RECON command (Commands)
- INIT.RECON command (Commands)

Function level activation control enhancement

The following commands are added for the function level activation control enhancement:
- Use the QUERY IMSFUNC command (Commands) command to see all of the IMS functions that are defined in the IMS function table.
- Use the UPDATE IMSFUNC command (Commands) command to dynamically enable or disable IMS functions that are not enabled by default.

Chapter 8. Message and code changes in IMS 15

IMS 15 includes new and changed messages and codes. Also, many messages were deleted from IMS 15.

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 15

IMS 15 includes new messages, as shown in the lists that follow.

New CQS messages

The following messages are new:
   CQS0224E

New DFS messages

The following messages are new:
   DFS0745E
   DFS3253W
   DFS3458
   DFS4690I
   DFS4691I
   DFS4692E
   DFS4861W
   DFS4862W
   DFS4863W
   DFS4864W
   DFS4878I
   DFS4879I
   DFS4881I

New G messages

The following messages are new:
   G116
   G117
   G118
   G119
   G120
   G121
   G122

New HWS messages

The following messages are new:
  HWSX0915W
  HWSX0916W

New abend codes

The following abend codes are new:
  0421
  0423
  0424
  0425
  0426

New component codes

The following component codes are new:
  Return and reason codes for IMS Connect exits

Changed messages and codes for IMS 15

This release includes changed DSP messages, as shown in the lists that follow.

Changed DFS messages

The following messages are changed:
  DFS0413I
  DFS1934E
  DFS2205I
  DFS2385E
  DFS2679A
  DFS3254I
  DFS3377
  DFS3398E
  DFS4610E
  DFS4615E
  DFS554A
  DFS3438
  DFS3446

Changed DSP messages

The following messages are changed:
  DSP0012I

Changed abend codes

The following abend codes are changed:
  U0711
  U0070
  U0071
  U0073
  U1002
  1143

Changed component codes

The following component codes are changed:
OTMA C/I return codes

Deleted messages and codes for IMS 15

The messages that are listed in the following sections were removed from IMS 15.

The following messages were deleted in this release:
DFS2211I
G040
G043
G904
G042
G1205
G1212

LGEN messages deleted

The following messages were removed with the removal of support for large system definitions (LGEN).

DFS3512I	G008
DFS3514I	G1211
DFS3516I	
DFS3518I	
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	

RSR messages deleted

The following messages and abends were removed with the removal of support for Remote Site Recovery (RSR). These messages include messages issued by the Transport Manager Subsystem and other RSR services, such as online forward recovery, log router, DBRC service groups.

	DSP messages (DBRC)
DFS4072A	DFS4130I
DFS4073A	DFS4131A
DFS4074W	DFS4132A
DFS4075A	DFS4133A
DFS4076A	DFS4134I
DFS4077I	DFS4135I
DFS4078I	DFS4136I
DFS4079A	DFS4150I
DFS4085W	DFS4152A
DFS4086W	DFS4153W
DFS4087A	DSPM058I
DFS4088A	DSPM059I
DFS4089A	DSP1001I
DFS4090A	DSP1002I
DFS4091I	DSP1030I
DFS4092A	DSP1004I
DFS4094A	DSP1005I
DFS4095A	DSP1006I
DFS4096A	DSP1007I
DFS4097I	DSP1009I
DFS4098A	DSP1010I
DFS4099A	DSP1011I
DFS4100I	DSP1012I
DFS4101A	DSP1013I
DFS4102A	DSP1014I
DFS4103I	DSP1016I
DFS4104A	DSP1017I
DFS4105A	DSP1021I
DFS4106I	DSP1022I
DFS4107A	DSP1025I
DFS4108A	DSP1027I
DFS4109	DSP1028I
DFS4111I	DSP1029I
DFS4112A	DSP1030I
DFS4113I	DSP1032I
DFS4114A	DSP1035I
DFS4115I	DSP1036I
DFS4121I	DSP1037I
DFS4122A	DSP1038I
DFS4123I	DSP1042I
DFS4124I	DSP1043I
DFS4125I	DSP1044I
DFS4126I	
DFS4127A	

ELX messages (TMS)	ELX0201W	G messages (SYSGEN)
ELX0001A	ELX0202I	G1205
ELX0002A	ELX0203W	G1212
ELX0003A	ELX0204I	
ELX0004A	ELX0205I	
ELX0005A	ELX0209I	
ELX0100A	ELX0210A	
ELX0101W	ELX0211I	
ELX0102W	ELX0212A	
ELX0103I	ELX0213A	
ELX0104A	ELX600A	
ELX0105I	ELX1001E	
ELX0106A	ELX1002I	
ELX0107I	ELX1003W	
ELX0108I	ELX1004W	
ELX0109I	ELX1005W	
ELX0111A	ELX1010I	
ELX0112I	ELXC106I	
ELX0113A	ELXC109I	
ELX0114I	ELXC112A	
ELX0115I	ELXC160I	
ELX0116A	ELXC161I	
ELX0118A	ELXC162A	
ELX0119I	ELXC181I	
ELX0120I	ELXC315I	
ELX0122I		
ELX0128I		
ELX0129I		
ELX0130I		
ELX0150I		
ELX0170I		
ELX0171A		
ELX0181A		
ELX0183A		
ELX0184I		
ELX0185I		
ELX0187I		
ELX0188I		
ELX0189I		
ELX0200I		

Chapter 9. Log record changes in IMS 15

The following table lists the log records that are new or changed in IMS 15.

Table 11. New or changed log records for IMS 15

Log record	Description of enhancement
X'01'	Enhanced to record distributed network security credentials if the credentials are passed to IMS in the OTMA message prefix.
X'03'	Enhanced to record distributed network security credentials if the credentials are passed to IMS in the OTMA message prefix.
X'07'	A new field DLREXTOF is added. DLREXTOF contains an offset to an extension of the call statistics section. The offset value can be used to determine whether additional information exists in the log record extension. Zero indicates no extension is added to the log record. A non-zero offset value can also be used to generate the mapping address to the new extension.
X'22'	Enhanced to record runtime resources creation status after the PGMCREAT user exit returns to IMS: • An X'22' log record, with the map byte field set to X'20', is written when a runtime database resource is created by IMS after the PGMCREAT user exit returns to IMS and indicates that a database resource is to be created. • An X'22' log record, with the map byte field set to X'21', is written when a runtime program resource is created by IMS after the PGMCREAT user exit returns to IMS and indicates that a program resource is to be created.
X'42'	Enhanced to include the current IMS function level at offset X'104'.
X'4500'	Enhanced to include the current IMS function level at offset X'C0' plus some reserved space.
X'4050'	Added to log the following information: • The IMS functions that are enabled so that the function enablement value can be retrieved at IMS restart. • The current IMS function level.
X'4507'	The following fields are added or removed for the IMS Logger enhancements of encryption and use of Media Manager to write to WADS: • ST4507_WADSCIS (new). • ST4507_FLG2 (new). • ST4507_ENC_WADS_MTX (new) • ST4507_WADSTRACKS (removed). • ST4507_WADSBLKSTRK (removed).

Table 11. New or changed log records for IMS 15 (continued)

Log record	Description of enhancement
X'45FF'	Three new feature flags are added to indicate the following status: • Enablement of Fast Path. • Use of Fast Path Shared EMHQ. • Use of Fast Path 64-bit DEDB buffers. This log record is also enhanced to include the following information for IMS functions: • At offset X'50', the current IMS function level. • At offset X'54', the offset to the IMS functions enabled bitmap from the start of the X'45FF' log record. • At offset X'30', some reserved space.
X'56FA'	A new field TPCEXTOF is added. TPCEXTOF contains an offset to an extension of the accounting statistics section. The offset value can be used to determine whether additional information exists in the log record extension. Zero indicates no extension is added to the log record. A non-zero offset value can also be used to generate the mapping address to the new extension.
X'5901'	Enhanced to record distributed network security credentials in the following situations: • If a Fast Path message that contains the credentials is processed by the Fast Path expedited message handler (EMH) on the local IMS system. • If a Fast Path message that contains the credentials is processed by using the EMH queue (EMHQ) in a shared-queues environment, the credentials are included in the log record in the processing, back-end IMS system.
X'5911'	Enhanced to record distributed network security credentials in a front-end IMS system if a Fast Path message that contains network security credentials is processed by using the EMH queue (EMHQ) in a shared-queues environment.

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 12. 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.

Related concepts:

"Network security credential propagation enhancement" on page 113

Part 2. IMS 15 enhancements

The enhancements to introduced in IMS 15 span the following areas of IMS: system definition, administration, troubleshooting, and reference.

The following topics provide planning information for the enhancements to IMS 15.

Chapter 10. 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 one or more new IMS functions or support for new technologies or products, are released as PTFs.

The following topics provide an overview of the IMS continuous delivery model and a list of the newest IMS functions.

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. You can enable disabled enhancements dynamically by using the UPDATE IMSFUNC command. Or, you can enable the function statically at IMS cold start by defining the parameter for the function in the DFSDF*xxx* member of the IMS PROCLIB data set. 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 default after you install the PTF.

IMS continuous delivery functions

IMS functions are delivered using PTFs as part of the IMS continuous delivery enhancement process and allow you to use new enhancements faster and more frequently.

Some of the PTFs that provide IMS continuous delivery enhancements are identified using IMS function levels. For more information about IMS function levels, see IMS function levels overview (System Administration).

The following table describes the IMS functions that are provided as part of the IMS continuous delivery model.

Table 13. IMS continuous delivery functions

IMS function level	APAR/PTF	Function name	Can be enabled	Can be disabled	Enabled by default	Must be applied globally	DBCTL	DCCTL	Batch	Activation bits	Link to further information
00000001	PI83839/ UI52153	FUNCTIONLEVEL	N	N	Y	N	Y	Y	Y	Not applicable because function is enabled by default	"Function level activation control enhancement" on page 124

Related concepts:

→ IMS function levels overview (System Administration)

Related reference:

→ QUERY IMSFUNC command (Commands)

→ UPDATE IMSFUNC command (Commands)

Related information:

Chapter 11. IMS Database Manager enhancements

The IMS 15 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 13, "IMS system enhancements," on page 119 in IMS 15 might also impact IMS DB.

DBRC migration and coexistence

DBRC enhancements include changes to the RECON data set format for IMS 15 and support that allows IMS Version 13 and IMS 14 systems to coexist until all of the systems are upgraded to IMS 15.

The changes to the RECON data set changes for IMS 15 include:
- The header version indicator is V15R1.
- The default value for MINVERS is set to "13.1" (X'D1').
- The RECON level is set to "15.1" (X'E1').
- The log release level in the PRILOG, SECLOG, PRISLD, SECSLD, PRIOLD, and SECOLD (and interim) records is set to X'3C', but is listed as "15.1".
- The version and coexistence level in the subsystem record is set to "15.1" (X'E1').

Because Remote Site Recovery (RSR) is no longer supported after IMS 14, you must remove all Global Service Group (GSG) information before migrating to IMS 15. Issue the LIST.GSG command to identify any GSG and issue the DELETE.GSG command to remove it before upgrading the RECON data set.

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:
- 7.0 for the IMS 15 API.
- 6.0 for the IMS 14 API.
- 5.0 for the IMS Version 13 API.

Coexistence considerations

The INIT.GSG command is not supported in IMS 15. If you try to use this command for IMS 15, the DSP1035I message is issued.

You can use 8GB OSAM HALDB without OLR Support for IMS 15 by configuring the following settings. This support was introduced in IMS 14. With this support, you can define an HALDB database to use up to eight gigabytes of data in its OSAM PHDAM or PHIDAM OSAM data sets.
- Set the MINVERS value to 14.1.
- Set the MINVERS value to 13.1 if the cross DBRC service level ID (CDSLID) value is set to 2. The CDSLID set to 2 indicates that the IMS Version 13 8GB

OSAM Support for HALDBS without OLR SPE has been applied to all IMS Version 13 resident libraries (RESLIBS) accessing the RECON data set.

Performance considerations

Before you upgrade a RECON with parallel access active, close the RECON data sets and perform a check for retained locks. The upgrade fails if retain locks exist. The RECON data sets are reopened in LSR mode when parallel access is active. This improves performance by eliminating the need for locking.

Excessive CA/CI splits might delay RECON upgrade.

While the RECON is being upgraded, other jobs wait, which impacts the performance of any DBRC requests.

Before you issue the CHANGE.RECON UPGRADE CHECKUP command against the production RECON data set, issue the command against a copy of the production RECON data set to understand any possible performance issues.

Log record changes

The following log records are new or changed by this enhancement in IMS 15:
- DSPRCNRC
- DSPRCR1
- DSPLOGRC
- DSPOLDRC
- DSPSSRC

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 15, see Chapter 8, "Message and code changes in IMS 15," on page 91.

Changes to commands

The CHANGE.RECON UPGRADE command is changed to upgrade an IMS Version 13 or IMS 14 RECON data set.

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 utilities

Batch Backout, Log Recovery and Log Archive utilities can only run on the release which created the log(s). That is, an IMS Version 13 or an IMS 14 Batch Backout utility cannot be used to back out an IMS 15 log.

The Database Recovery utility and the Change Accumulation utility must be run with highest level of logs being used. For example, the IMS 14 utilities may use IMS 14 logs but not IMS 15 logs.

DEDB Alter utility enhancements

In IMS 15, the DEDB Alter utility is enhanced to enable you to add a new field to existing free space in a segment while the DEDB area remains online.

This enhancement increases the availability of your DEDB database. Previously, a DEDB database had to be taken offline to add a new field to a segment.

Requirements:
- A two-stage randomizer must be used, which enables areas to be processed individually.
- All IMS data sharing systems must be at IMS 15 or later.
- 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.
- 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

If you are invoking the DEDB Alter utility for an IMS 15 function, such as ALTERDB, 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

You can run the DEDB Alter utility in a coexistence environment; however, if you are invoking the DEDB Alter utility for an IMS 15 function such as ALTERDB, the DEDB Alter utility detects the version of the IMS system and terminates if the IMS version does not support the particular change.

Changes 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.

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 15, see Chapter 8, "Message and code changes in IMS 15," on page 91.

Changes 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.

Chapter 12. IMS Transaction Manager enhancements

The enhancements to IMS Transaction Manager (IMS TM) in IMS 15 include enhancements to Multiple Systems Coupling (MSC), Open Transaction Manager Access (OTMA), IMS Connect, synchronous callout, and more.

The Chapter 13, "IMS system enhancements," on page 119 in IMS 15 might also impact IMS TM.

CQS automatic structure checkpoint

In IMS 15, you can configure IMS to take checkpoints of a common queue server (CQS) structure automatically after a user-specified number of log records have been written by a CQS.

Coexistence considerations

An IMS 15 CQS that performs automatic structure checkpoints can coexist with IMS Version 13 and IMS 14 CQSs that are connected to the same shared queue structure. However, you can use the **STRCHKPT=** parameter in the CQSSLxxx member of the IMS PROCLIB data set to trigger the automatic structure checkpoint only for IMS 15 CQSs. Only the IMS 15 CQSs can become the master CQS that performs automatic structure checkpoints.

Fallback considerations

Before reverting an IMS 15 CQS that performs automatic structure checkpoint to a CQS of an earlier version of IMS, remove or comment out the **STRCHKPT=** parameter in the CQSSLxxx member of the IMS PROCLIB data set.

Network security credential propagation enhancement

In IMS 15, you can associate the security credentials that are entered by a user in a distributed environment with the end-to-end processing of a transaction in IMS. The distributed security credentials can even be propagated to synchronous callout requests that are initiated by the ICAL call of the IMS DL/I interface.

With the network security credential propagation enhancement, when network security credentials are entered by a user in a distributed environment and are passed to IMS, the security credentials are audited in IMS log records. Previously, when a user initiated an IMS transaction in a distributed environment and entered security credentials, the credentials were not propagated to IMS and were therefore excluded from IMS log records.

The distributed network security credentials can include a network user ID and a network session ID.

Network user ID
> The distributed identity of the user. The maximum length of a network user ID is 246 bytes. For users of the IMS TM Resource Adapter, the network user ID is a Distinguish Name (DN) in the X.500 series of standards.

Network session ID
> The session identity of the distributed user. The maximum length of a network session ID is 254 bytes. For users of the IMS TM Resource Adapter, the network session ID is a domain name, realm, or registry name.

Network security credentials can be propagated from user-written IMS Connect client applications that use the HWSSMPL0 or the HWSSMPL1 user message exit routines. Two new IRM extension specifications are added with this enhancement to enable applications that use the HWSSMPL0 or the HWSSMPL1 user message exit routines to pass network security credentials to IMS. An IRM extension with an ID of *NETUID* can be used to pass a network user ID to IMS and an IRM extension with an ID of *NETSID* can be used to pass a network session ID to IMS. If network security credentials are included in an IMS callout request, the RESUME TPIPE call of the IMS Connect client application can be defined to support the credentials.

IMS TM resource adapter client applications that use the HWSJAVA0 user message exit routine can also propagate network security credentials to IMS. An extendable Java Authentication and Authorization Service (JAAS) login module is provided with IMS TM resource adapter to enable network security credentials to be passed from a Java EE application that uses the HWSJAVA0 user message exit routine to IMS. The activation specification is enhanced with the resumeTpipeNsc property to enable IMS TM resource adapter to support network security credentials in IMS synchronous callout messages. To enable IMS TM resource adapter to support network security credentials in asynchronous callout messages, the IMS interaction specification is enhanced with the setResumeTpipeNSC property.

Migration considerations

- If the security-data section of the OTMA message prefix contains network security credentials, the size of the OTMA message can increase by up to 504 bytes. Therefore, consider increasing the size of the SHMSG and LGMSG message queue data sets and the size of the message queue pool.
- If both of the following situations occur, you might need to modify code that includes the HWSOMPFX macro:
  - The Network Session ID (NETSID) section or the Network User ID (NETUID) section, or both, is included in the security section of the OTMA message header.
  - Either the **DSECT=**ALL or the **DSECT=**NO option is specified with the HWSOMPFX macro.

  The size of the NETUID and NETSID sections can vary, causing the locations of the fields that are below the security section to change. However, if the **DSECT=**ALL or the **DSECT=**NO option is specified, a contiguous DSECT, the HWSOMPFX DSECT, is generated that does not account for sections that vary in size. Therefore, the fields in the OTMA message header that are below the security section might become inaccessible.

  For the fields of the OTMA message header that are below the security section to be accessed, you need to map the HWSOMUSR, HWSOMAPP, or HWSOMAPX DSECTS of the HWSOMPFX macro to the changed locations of the fields.

  For more information about the fields of the OTMA message header, see OTMA header fields used by IMS Connect (Communications and Connections).
- In IMS 15, processing by the HWSJAVA0 user message exit routine of the user data section that is in the OTMA message header is updated. If the OTMA

message header contains network security information and the HWSOMPFX macro is used, the HWSJAVA0 exit routine specifies both the **DSECT=** and the **NETSEC_OPT=**YES options for the HWSOMPFX macro. The **DSECT=** and the **NETSEC_OPT=**YES options cause the following behaviors:

– An individual DSECT is generated for each section of the OTMA message header.
– The HWSECDNDS DSECT, or the HWSECARDS DSECT, or both, is generated to map network security information.
– The HWSOMPFX DSECT is not generated.

Coexistence considerations

Network security credentials can be passed to IMS and audited in IMS log records only if both IMS Connect and IMS are Version 15 or later. If network security credentials are passed between IMS and client applications of IMS TM resource adapter in inbound and outbound, or callout, messages, IMS TM resource adapter must also be V15 or later.

Log record changes

Because distributed network security credentials are passed to IMS in the security-data section of the OTMA message prefix, all IMS log records that contain information about the message prefix, such as log records X'01' and X'03', include the distributed security credentials.

If a Fast Path message contains network security credentials and is processed by the Fast Path expedited message handler (EMH) on the local IMS system, the credentials are logged in the X'5901' log record.

If a Fast Path message that contains network security credentials is processed by using the EMH queue (EMHQ) in a shared-queues environment, in the front-end IMS system, the credentials are included in the X'5911' log record. In the back-end IMS system, which is the processing IMS system, the credentials are included in the X'5901' log record.

Requirements

The following prerequisites must be enabled for IMS TM resource adapter client applications, except for the client applications that use the IMS service provider in IBM® z/OS® Connect Enterprise Edition (z/OS Connect EE), to support network security credentials:

• One of the following application servers:
– WebSphere Application Server Version 8.0 or later
– WebSphere Liberty Version 8.5.5.9 or later
• Container-managed security.
• An external user account registry, such as an LDAP server, that contains authorized users.

Restrictions

Distributed network security credentials from DataPower, IMS Connect API, and SOAP Gateway clients are not supported by IMS Connect and therefore are not audited in IMS log records.

When the IMS Connect Recorder Trace facility is active, IMS Connect takes a snapshot of the first 670 bytes of messages at key points during IMS Connect processing. Because messages that contain network security credentials might be larger than 670 bytes, the information for the network user ID and the network session ID might not be included in an IMS Connect Recorder trace record.

Changes to installing and defining IMS

The LOGSTR= parameter is added to the OTMA client descriptor in the DFSYDTx member of the IMS™ PROCLIB data set. You can use the LOGSTR= parameter to specify whether the first 255 bytes of network security credentials, which includes a network user ID or network session ID, or both, is included in the RACF SMF process records.

Changes to programming for IMS

The IMS OTMA Callable Interface (OTMA C/I) is enhanced with the otma_send_receivey and otma_send_asyncx APIs. You can use these APIs to pass the network user ID and the network session ID to IMS. For each API, up to 100 bytes for the network user ID and up to 100 bytes for the network session ID can be passed to IMS.

The INQY call with the MSGINFO subfunction is enhanced to return the network user ID and the network session ID that are submitted to IMS. IMS application programs can use the INQY MSGINFO call to identify the distributed user who initiated a transaction.

Changes to troubleshooting for IMS

Recommendation: If network security credentials are included in IMS Connect client input messages, enable the BPE External Trace facility for the IMS Connect Recorder Trace facility. If network security credentials are passed to IMS Connect, the size of both input and output messages to and from IMS Connect might be larger than 670 bytes and the BPE External Trace facility would be required to capture the data of the entire message.

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 15, see Chapter 8, "Message and code changes in IMS 15," on page 91.

Changes to exit routines

The following IMS Connect user message exit routines are enhanced to propagate network security credentials between IMS and applications in a distributed environment:
- HWSSMPL0
- HWSSMPL1
- HWSJAVA0

The following transaction manager exit routines are enhanced to include the address of the security-data section of the OTMA message prefix. Because the

OTMA security-data section can include a network user ID and a network session ID, you can use the following exit routines to access the network security credentials if the credentials are passed to IMS.

- DFSYIOE0
- DFSYPRX0
- DFSYDRU0

The DFSCTRN0 exit routine is enhanced to add two OTMA fields, which include the addresses of the network user ID and the network session ID, in the storage area that is mapped the CTRNPARM DSECT.

Chapter 13. IMS system enhancements

The enhancements to the IMS system in IMS 15 include enhancements to the IMS repository, user exits, the total cost of ownership, dependent regions, commands, and more.

The enhancements to the IMS 15 system are not specific to either the IMS Database Manager or the IMS Transaction Manager and might impact both.

Consolidation of IMS logger parameters in the DFSDF*xxx* PROCLIB member

In IMS 15, the parameters that define the IMS logger function are consolidated into a new section, `<SECTION=LOGGER>`, in the DFSDF*xxx* member of the IMS PROCLIB data set. The LOGGER section of the DFSDF*xxx* PROCLIB member is mandatory.

The IMS logger parameters that were previously specified in the DFSVSM*xx* PROCLIB member are moved to the new LOGGER section of the DFSDF*xxx* PROBLIB member. Any logger parameters that are still specified in the DFSVSM*xx* member are ignored by the IMS system and not reported in the error log.

The `ARC=` and `WADS=` parameters, which were previously specified in the DFSPB*xxx* PROCLIB member or on the EXEC parameters of the control region JCL, are also moved to the DFSDF*xxx* PROCLIB member. If you specify `ARC=` or `WADS=` in DFSPB*xxx* or in the control region JCL, the specifications are ignored by the IMS system.

Migration considerations

IMS 15 is the first IMS release where the DFSDF*xxx* member of the PROCLIB data set is mandatory.

Log record changes

The following log records are new or changed by this enhancement in IMS 15:
*
    X'4507' - The following fields are added:
    – ST4507_FLG2 (X'80': ST4507_F2_ZHYPOLDS and X'40': ST4507_F2_ZHYPWADS)
    – ST4507_WADSCIS
*
    X'4507' - The following fields are removed:
    – ST4507_WADSTRACKS
    – ST4507_WADSBLKSTRK

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 15, see Chapter 8, "Message and code changes in IMS 15," on page 91.

Changes to exit routines

The control statement for the log edit user exit (LOGEDIT) is moved from the DFSVSM*xx* member of the IMS PROCLIB data set to the new LOGGER section of the DFSDF*xxx* member.

CQS return code enhancements for z/OS Logger write errors

In IMS 15, if the z/OS system logger encounters write errors in support of the IMS Common Queue Server (CQS), CQS issues new return and completion codes to identify the problem as a z/OS system logger write error.

The CQSPUT request, CQSREAD request, and CQSMOVE request can receive new reason codes.

The CQSUNLCK request and CQSCHKPT request can receive new completion codes.

Message CQS0035E can contain new return codes.

Migration considerations

Any customer- or vendor-written CQS client that processes CQS return, reason, or completion codes should be evaluated to see if it should be updated for the new codes.

Data set support for zHyperWrite

In IMS 15, you can specify whether to use zHyperWrite for writing data to the write-ahead log data sets (WADS) and online log data sets (OLDS).

WADS support for zHyperWrite

In IMS 15, the IMS Logger uses the DFSMS Media Manager to write data to the write-ahead log data set (WADS). As a result, the WADS must be defined as a linear VSAM data set.

The DFSMS Media Manager enables the usage of the hardware features like High Performance FICON® for z Systems™ (zHPF), which increases I/O throughput, and zHyperWrite, which reduces latency time for synchronous replication products. For reading from the WADS, regular VSAM is used.

You can use the **ZHYPERWRITE=** parameter in the new LOGGER section of the DFSDF*xxx* PROCLIB member to enable or disable the use of zHyperWrite for the WADS.

Migration considerations

The WADS must be defined as a linear VSAM data set with a control interval (CI) size of 4 KB (4096-bytes), secondary space allocation of 0, and with the SHAREOPTIONS(3 3) parameter. The access method services (AMS) utility IDCAMS can be used to define the data set.

Recommendation: Use different data set name for the IMS 15 WADS than the name used for the WADS in current IMS version. This allows you to predefine the IMS 15 WADS before shutting down the current IMS system. Otherwise, the WADS must be deleted and redefined after the current IMS system goes down and before the system is started as an IMS 15 system.

Coexistence considerations

For XRF systems, both of the coexisting systems must be of the same IMS version.

Fallback

When you fall back from IMS 15 system to a previous version of IMS, ensure that you use a non-VSAM data set for WADS.

If you defined WADS data set names for IMS 15 during migration that were different from the names used by the previous IMS version, use the previous names for fallback. The fallback process replaces the reference to the WADS data sets systematically. Otherwise, the WADS must be deleted and redefined after the IMS 15 system is shut down, and before the system is started at the previous IMS version.

At the end of fallback, cold start IMS with the FORMAT ALL or FORMAT WA keyword.

Log record changes

The following log records are new or changed by this enhancement in IMS 15:

*

   X'4507' - The following fields are added:
   – ST4507_WADSCIS, which specifies the number of control intervals in the WADS.
   – ST4507_FLG2 (X'40': ST4507_F2_ZHYPWADS)

*

   X'4507' - The following fields are removed:
   – ST4507_WADSTRACKS, which specifies the number of physical tracks in the WADS data set.
   – ST4507_WADSBLKSTRK, which specifies the number of physical blocks per track in the WADS data set.

Trace record changes

The following trace subcodes are added to the IMS logger's trace. They are defined in the ILOG macro.

Table 14. New trace subcodes introduced to the IMS Logger's trace.

Trace subcode	Description
x'31'	Completed a CONNECT request.
x'32'	Completed a DISCONNECT request.
x'33'	Completed a format request.
x'34'	Completed writing instance ID request.
x'38'	The Media Manager termination exit was driven.

Table 14. New trace subcodes introduced to the IMS Logger's trace (continued).

Trace subcode	Description
x'39'	The Media Manager error exit was driven.

Requirements

The support of zHyperWrite requires that both the primary and secondary data set volumes be in a Metro Mirror relationship managed by HyperSwap.

Changes to installing and defining IMS

You must define the WADS as a linear VSAM data set with a control interval (CI) size of 4 KB (4096-bytes), secondary space allocation of 0, and the SHAREOPITON (3 3) parameter.

The following JCL example demonstrates how to define the WADS:

```
//AMS EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *

DEFINE CLUSTER -
    (NAME(IMSA.WADS.VSAM) -
    VOLUME(VOL001) -
    CONTROLINTERVALSIZE(4096) -
    SHAREOPTIONS(3 3) -
    CYLINDERS (20 0) -
    LINEAR)
/*
```

The installation verification program (IVP) is changed to define the WADS as linear VSAM data sets.

Changes to administering IMS

The IMS System Administrator defines all WADS to be used as VSAM linear data sets before starting the IMS 15 system for the first time.

Changes to troubleshooting for IMS

This enhancement introduces new and changed IMS messages related to Logger Media Manager.

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 15, see Chapter 8, "Message and code changes in IMS 15," on page 91.

Changes to utilities

When IMS reads the write-ahead data sets (WADS), the Log Recovery utility might issue the DFS3253W warning message to explain the MVS messages that VSAM issues in the job log.

OLDS support for zHyperWrite

In IMS 15, users can choose whether or not the IMS Logger uses zHyperWrite to write data to the online log data sets (OLDS).

Using zHyperWrite for the OLDS decreases replication latency and makes writes to the OLDS complete quicker in a synchronous replication environment.

Note: This optional enhancement requires that DFSMS APAR OA51385 be applied to your environment and the OLDS be defined as extended format data sets.

Log record changes

No changes to log records. However, as part of the IMS 15 base, the value of the OLDS sub-parameter of the **ZHYPERWRITE** parameter is included in the x'4507' record as flag ST4507_F2_ZHYPOLDS.

Changes to installing and defining IMS

zHyperWrite can be used for writing to the OLDS, which decreases replication latency. Set OLDS=YES in the ZHYPERWRITE= parameter in the LOGGER section of the DFSDFxxx PROCLIB member to enable the IMS Logger's use of zHyperWrite for writes to the OLDS. You must ensure that all the OLDS that will be used by the IMS system are defined as extended format data sets.

To disable this enhancement, remove the **OLDS=YES** parameter or change it to **OLDS=NO**.

Encryption support in IMS

In IMS 15, the following types of data sets can be encrypted beside those already supported in earlier versions of IMS.

Write-ahead data set (WADS) encryption

In IMS 15, you can encrypt the data on the Write-ahead data set (WADS).

APAR PI84947 must be installed to enable this feature.

To encrypt WADS data sets, define WADS as extended format and with a key label.

For online IMS operation, the ID associated with the IMS control region must have access to the key label. Additionally, the IDs associated with any utilities that read the WADS also requires access to the key label.

This enhancement offers additional security benefits. Support personnel can operate on the data sets without the authority to read the data sets.

Log record changes

In the x'4507' record, the following 2-byte field is added: ST4507_ENC_WADS_MTX. This indicates which WADS are encrypted.

Function level activation control enhancement

In IMS 15 with APAR/PTF PI83839/UI52153, you can dynamically enable or disable new IMS functions that are delivered as PTFs under the IMS continuous delivery model without causing an IMS system outage. After you install a new IMS function, you control when to enable and disable the function by using the UPDATE IMSFUNC command.

You can continue to enable or disable IMS functions statically by using IMS PROCLIB data set members as with earlier versions of IMS.

This enhancement also provides the following features to support the continuous delivery of IMS functions:

Function level
> A number that identifies a PTF containing one or more IMS functions.

Function table
> Contains information about significant new or changed IMS functions delivered by IMS henceforth, including the current IMS function level and an entry for each function to provide more information about the function.

IMS functions enabled bitmap
> For IMS functions that are delivered disabled, a bit for every IMS function that becomes enabled. The functions enabled bitmap also contains the current IMS function level.

QUERY IMSFUNC command
> Displays one or more or all of the IMS functions that are defined in the IMS function table and information about the functions. You can also use the QUERY IMSFUNC command to see the current IMS function level.

This enhancement, delivered with APAR/PTF PI83839/UI52153, is the first IMS function, and has a function name of FUNCTIONLEVEL and a function level of 00000001.

Some IMS functions, due to either technical or strategic requirements or because it has no impact until you choose to use it, are enabled by default after you install the PTF. For functions that are enabled by default, you do not need to issue the UPDATE IMSFUNC command to start using the enhancement. This function, delivered with APAR/PTF PI83839/UI52153, is enabled by default.

Log record changes

The following log records are new or changed by this enhancement:

X'42' Enhanced to include the IMS function level at offset X'104'.

X'4500'
> Enhanced to include the IMS function level at offset X'C0'.

X'4050'
> Newly added to log the following information:
> - The IMS functions that are enabled so that the function enablement value can be retrieved at IMS restart.
> - The current IMS function level.

X'45FF'
> Enhanced to include the following information:

- At offset X'50', the current IMS function level.
- At offset X'54', the offset to the IMS functions enabled bitmap from the start of the X'45FF' log record.
- At offset X'30', some reserved space.

Changes to defining and administering IMS

With this enhancement, after you install a PTF that contains a new continuous delivery IMS function, you decide when to enable and disable IMS functions that are not enabled by default. You can change the function enablement value dynamically by using the UPDATE IMSFUNC command. Alternatively, you can enable the new function statically by defining the parameter for the function in the DFSDF*xxx* member of the IMS PROCLIB data set and cold starting IMS.

Tip: If you enable or disable an IMS function by using the UPDATE IMSFUNC command, specify the same enablement value in the parameter that defines the function in the IMS PROCLIB data set member. This ensures that the enablement value for the function is retrieved at IMS cold start.

If you change a function enablement value by using the UPDATE IMSFUNC command and IMS is cold started, one of the following situations occurs:

- For local functions, the enablement value is retrieved from the DFSDF*xxx* PROCLIB member during cold start. In this case, the UPDATE IMSFUNC command might need to be issued following the cold start to return the enablement value to a previous state.
- For global functions, if you use RM, CQS, and a resource structure, the enablement value is retrieved from the resource structure during cold start.
- For catalog functions, the enablement value is retrieved from the catalog during cold start.

Changes that you make by using the UPDATE IMSFUNC command are logged in the x'22' map byte x'31' log record and are recoverable across an IMS restart. For example, if a function is enabled in the DFSDF*xxx* member and later disabled by using the UPDATE IMSFUNC, the disabled value is recovered if IMS restarts.

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 15, see Chapter 8, "Message and code changes in IMS 15," on page 91.

Changes to commands

The UPDATE IMSFUNC command is added to allow you to dynamically enable or disable an IMS function, if the function is not enabled by default, whilst IMS remains online.

The QUERY IMSFUNC command is added to allow you to display one or more or all of the IMS functions that are defined in the IMS function table and information about the functions. You can also use the QUERY IMSFUNC command to see the current IMS function level.

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 sample IMS initialization and termination user exit DFSITRX0 is enhanced to show how to access the current IMS function level from the system contents directory (SCD) field SCDFNCE.

IMS Connect enhancements

In IMS 15, IMS Connect is enhanced in multiple ways to improve reliability, availability, serviceability, and security.

In addition to the enhancements, IMS Connect support for SSL and local option connections is removed.

Maximum number of IMS Connect ports enhancement

In IMS 15, the maximum total number of ports that you can define is increased to 200.

The previous maximum number was 50.

All ports that are defined to an IMS Connect instance count toward the maximum, including CICS and DRDA ports.

Changes to installing and defining IMS

When you are configuring IMS Connect ports, the maximum number of ports that you can specify is now 200. This is a new maximum for the sum total of all defined ports, including CICS and DRDA ports.

Changes to troubleshooting for IMS

At IMS Connect startup, if more than 200 ports are defined in the IMS Connect HWSCFGxx configuration member, IMS Connect issues existing message HWSX0909E and abends with existing abend code U3401.

Changes to commands

The CREATE IMSCON TYPE(PORT) command issues a new completion code when the maximum number of ports is reached.

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.

Program Creation user exit routine (PGMCREAT) enhancement

In IMS 15, you can use the new PGMCREAT type exit to dynamically create the runtime control block (PDIR) for an application program when the application program is scheduled by IMS. The program creation user exit routine (PGMCREAT) can also create the runtime control block for an associated database (DDIR). The Program Creation user exit routine (PGMCREAT) enhancement

simplifies the creation of runtime program resources for application programs that run in BMP and JBP regions. With the Program Creation user exit, an IMS system programmer can schedule a BMP or JBP region without having to define the application program to IMS using the stage-1 APPLCTN macro or the IMS type-2 CREATE PGM command.

Instead of defining the program resource to IMS during IMS SYSGEN or with the CREATE PGM command, the Program Creation user exit can be used to postpone the creation of the program resource until the program is to be scheduled in a BMP or JBP region. When the program is created, the user exit can also optionally create a database resource.

Log record changes

The PGMCREAT user exit can return to IMS to create a new runtime program or database resource. When a program resource is created, an X'22' map byte X'21' log record is written. When a database resource is created, an X'22' map byte X'20' log record is written.

Requirements

Dynamic resource definition must be configured and enabled by specifying MODBLKS=DYN in the DFSDF*xx* or DFSCG*xxx* PROCLIB member.

Restrictions

If a program resource is created, the PSB associated with the program must reside in ACBLIB or the IMS Catalog. If a database resource is created, the DBD associated with the database must also reside in ACBLIB or the IMS Catalog.

Changes to system definition

To work with this exit, the system programmer needs to add the USER_EXITS section in the DFSDFxxx member of the PROCLIB data set. In the USER_EXITS section, specify the PGMCREAT exit type and the name of the user exit routine. For example,

```
<SECTION=USER_EXITS>
          EXITDEF=(TYPE=PGMCREAT,
            EXITS=(exitname))
```

Changes to administering IMS

Define the PGMCREAT user exit routines to IMS specifying EXITDEF=(TYPE=PGMCREAT,EXITS=()) in the DFSDF*xxx* member.

The QUERY USEREXIT command can be used to query the information of the PGMCREAT exit.

The REFRESH USEREXIT command can be used to bring in a new copy of the program create user exit.

Changes to operating and automating

IMS system programmer can schedule a BMP or JBP region without having to define the application program to IMS using the stage-1 APPLCTN macro or the

IMS type-2 CREATE PGM command. When the program is created the user exit can specify to optionally create a database resource.

The REFRESH USEREXIT command can be used to bring in a new copy of the updated program create user exit without having to bring IMS down. For example,

REFRESH USEREXIT TYPE(PGMCREAT) MEMBER(PX0).

PGMCREAT is the user exit type and PX0 is the suffix for DFSDF*xxx* member. For more details, see IMS commands (Commands).

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 15, see Chapter 8, "Message and code changes in IMS 15," on page 91.

Changes to commands

QUERY DB and QUERY PGM commands can now display a new definition type of PGMCREAT.

QUERY USEREXIT and REFRESH USEREXIT commands can show PGMCREAT as new Type. For example,

```
TYPE(exit_type)
PGMCREAT - Program Creation User Exit
```

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 Program Creation user exit routine can request that IMS dynamically create the runtime program resource for a program that is to be scheduled in a BMP or JBP dependent region. Optionally, the exit can also request that IMS create a runtime database resource. The program and optional database resource are created without requiring a restart of IMS or a type-2 CREATE command to be issued.

Reduce need for IMS system definition enhancements

In IMS 15, the IMS system programmers can define and generate an IMS system more easily with the following enhancements.

IMS.OPTIONS data set removal

The IMS.OPTIONS data set removal enhancement simplifies IMS system definition and generation in IMS 15.

The IMS.OPTIONS data set contains only one member, DFSVTAM, which is used to define and set the *&DFSVTAM* and *&DFSBTAM* global variables.

The DFSVTAM member is not used during system generation, so the IMS.OPTIONS data set is removed.

Changes to installing and defining IMS

The IMS.OPTIONS data set is not created during IMS system generation.

The following sample installation jobs no longer references the IMS.OPTIONS data set:
- DFSALOC1, which allocates target and distribution libraries
- DFSDDEF1, which defines SMP/E DDDEFs for the IMS system

The JCL that is generated as the output of stage 1 or during stage 2 no longer references the IMS.OPTIONS data set.

BUFPOOLS macro removal

In IMS 15, the BUFPOOLS macro removal enhancement simplifies IMS system definition and generation.

The BUFPOOLS macro is an IMS System Definition Stage 1 macro that allows the customer to specify default storage buffer pool sizes in DB/DC and DBCTL environments

For versions earlier than IMS 15, customers can define buffer pool sizes by using corresponding keywords in the DFSPBxxx member of the IMS PROCLIB data set or JCL instead of the BUFPOOLS macro.

In IMS 15, the BUFPOOLS macro will be ignored. If the BUFPOOLS macro is included in the IMS Definition Stage 1, IMS will generate an MNOTE return code 2 with message G116 issued.

Because IMS no longer processes the BUFPOOLS macro, the only way to specify buffer pool sizes is to use the corresponding keywords in the DFSPBxxx member of the IMS PROCLIB data set or in the JCL. If no value is provided, IMS will assign default sizes for each buffer pool. The following table shows the BUFPOOLS macro keywords, their corresponding overrides in the DFSPBxxx member of the IMS PROCLIB data set or JCL, and the default value IMS will assign if no value is provided. For detailed description of the keywords in the BUFPOOLS macro, see BUFPOOLS macro (System Definition).

Table 15. Default value for DFSPBxxx or JCL keywords

BUFPOOLS keyword	DFSPBxxx member or JCL override	Default value	Description
DMB=	DMB=	12288 bytes	Size of the DMB control block pool
EPCB=	EPCB=	8192 bytes	Size of the EPCB pool
FORMAT=	FBP=	20480 bytes	Size of the message format buffer pool

Table 15. Default value for DFSPBxxx or JCL keywords (continued)

BUFPOOLS keyword	DFSPBxxx member or JCL override	Default value	Description
FRE=	FRE=	30	Number of fetch request elements (FREs) for loading MFS control blocks into the message format buffer pool
PSB=	PSB=	12288 bytes	Size of the PSB control block pool in subpool 231 storage if the DL/I address space option is not used
PSBW	PSBW=	12288 bytes	Size of the PSB work area pool
SASPSB=(size1,)	CSAPSB=	4096 bytes	Size of the PSB control block pool in the z/OS common storage area (CSA) if the DL/I address space option (LSO=S) is used
SASPSB=(,size2)	DLIPSB=	8192 bytes	Size of the PSB control block pool in the DL/I address space if the DL/I address space option (LSO=S) is used

To override the IMS defaults, provide the keyword values in the DFSPBxxx member of the IMS PROCLIB data set or in the JCL. If you used the BUFPOOLS macro, move the BUFPOOLS macro definitions into the DFSPBxxx member of the IMS PROCLIB data set or the JCL so that IMS uses the correct buffer pool definitions after the next cold start.

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 15, see Chapter 8, "Message and code changes in IMS 15," on page 91.

IMSCTF macro removal

In IMS 15, the IMSCTF macro is removed to simplify IMS system definition and generation.

If the IMSCTF macro is specified, IMS System generation stage 1 assembly issues message G122 in an MNOTE 2, which results in a return code of 2 for the assembly.

Migration considerations

With this enhancement, options that you previously defined in the IMSCTF macro can be defined in IMS 15 by using the DFSPB*xxx* member of the IMS PROCLIB data set, JCL, or the DFSIDEF0 module. If you do not migrate the values that you previously specified on the IMSCTF macro, default values are used by IMS instead.

Table 16. Methods you can use in IMS 15 to specify previously defined IMSCTF macro parameters, and default values for the parameters in IMS 15

IMSCTF macro parameter	How to specify the parameter for IMS 15	Default value in IMS 15
CPLOG=	Use one of the following methods: • The **CPLOG=** parameter in the DFSPB*xxx* member • The **CPLOG=** parameter in JCL • The /CHANGE CPLOG command	500,000
LOG=	If you specified **LOG=(DUAL)** in the IMSCTF macro, use the sample DBBBATCH, DLIBATCH, IMSCOBGO, or IMSPLIGO procedures and remove the asterisk (*) for the IEFRDER2 DD statement. If you specified **LOG=(,MONITOR)** in the IMSCTF macro, use the sample DBC, DCC, or IMS procedures and remove the asterisk (*) for the IMSMON DD statement.	No default values are provided.
PRDR=	Use one of the following methods: • The **PRDR=** parameter in the DFSPB*xxx* member • The **PRDR=** parameter in JCL	IMSRDR
RDS=	Use one of the following methods: • The **RDS=** parameter in the DFSPB*xxx* member • The **RDS=** parameter in JCL	4096

Table 16. Methods you can use in IMS 15 to specify previously defined IMSCTF macro parameters, and default values for the parameters in IMS 15 (continued)

IMSCTF macro parameter	How to specify the parameter for IMS 15	Default value in IMS 15

SVCNO=	Use one of the following methods to specify a type 2 SVC number:
	Important: If you use the Open Database Access (ODBA) interface or the database resource adapter (DRA) and you do not want to use the default value of 254 for the type 2 SVC number, you can define the number only by using the DFSIDEF macro of the DFSIDEF0 module.
	• The **SVC2=** parameter in the DFSPB*xxx* member
	• The **SVC2=** parameter in JCL
	• Specify the following code in the DFSIDEF0 module:
	`DFSIDEF TYPE=PARM SVC2=`
	
	To specify a type 4 SVC number, specify the following code in the DFSIDEF0 module:
	`DFSIDEF TYPE=PARM SVC4=`
		The default value for the type 2 SVC number is 254.
		
		The default value for the type 4 SVC number is 255.

For detailed information on how to migrate the parameters that you specified on the IMSCTF macro, see "IMSCTF macro removal migration considerations" on page 64.

Changes to installing and defining IMS

The Installation and Verification Program (IVP) is modified to assist in re-linking the type 2 and type 4 SVC routines.

The Syntax Checker is modified to assist in specifying the new DFSPBxxx member parameters.

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 15, see Chapter 8, "Message and code changes in IMS 15," on page 91.

SDFSPROC data set for sample procedures removal

The IMS system definition process no longer generates sample procedures. Instead, most of the procedures that were generated by system definition are now built during SMP/E processing and placed in the new ADFSPROC and SDFSPROC library data sets.

The following procedures are removed from SYSGEN.

Table 17. Sample procedures that are removed from SYSGEN

ACBGEN to DFSMPR	DFSMREC to IMS	IMSBATCH to IMSWTxxx (can be multiple)	INITMOD to REFCPY
ACBGEN	DFSMREC	IMSBATCH	INITMOD
CBLTDLI	DFSPBDBC	IMSCOBGO	MFSBACK
DBBBATCH	DFSPBDCC	IMSCOBOL	MFSBTCH1
DBC	DFSPBIMS	IMSDALOC	MFSBTCH2
DBDGEN	DFSUOLC	IMSFP	MFSDCT
DBRC or from IMSCTRL DBRCNM=	DFSVSM00	IMSJBP	MFSREST
DCC	DFSWTxxx (can be multiple)	IMSJMP	MFSRVC
DFSACBCP	DLIBATCH	IMSMSG	MFSTEST
DFSBMP	DLISAS or from IMSCTRL DLINM=	IMSMSV	MFSUTL
DFSIFP	FDR	IMSPLI	OLCUTL
DFSJBP	FMTCPY	IMSPLIGO	PLITDLI
DFSJMP	FPUTIL	IMSRDR or from IMSCTF PRDR=	PSBGEN
DFSMPR	IMS	IMSWTxxx (can be multiple)	REFCPY

The following procedures from the table are created by DFSPROCB JCL and placed in the ADFSPROC and SDFSPROC data sets.

- DFSACBCP
- CBLTDLI
- DFSMREC
- FMTCPY
- PLITDLI
- REFCPY

The following procedures from the table are renamed by DFSPROCB JCL and placed in the ADFSPROC and SDFSPROC data sets.

- DFSDBDGN is renamed as DBDGEN.
- DFSIMSBT is renamed as IMSBATCH.
- DFSPSBGN is renamed as PSBGEN.

The rest of the procedures from the table are now built during SMP/E processing and placed in the ADFSPROC and SDFSPROC data sets

The sample procedures that were generated by SMP/E before IMS 15 are unchanged and are not placed into the ADSFPROC and SDFSPROC libraries.

Before IMS 15, certain variables in the procedures that were specific to your installation were set based on specifications that are provided in your stage-1 system definition macros. In IMS 15, SMP/E sets those variables to the IMS system defaults, so when you tailor the procedures to your environment, be sure to include these variables also.

Migration considerations

Sample procedures are no longer generated during the SYSGEN processing but installed in the IMS.SDFSPROC and IMS.ADFSPROC data sets during SMP processing. These sample procedures contain default values for variables that the IMS system used to set based on the specifications during IMS stage 1 system definition. The variables should be modified to match what they would have been if generated by the SYSGEN processing. Use the DFSPROCB JCL to modify the sample procedures. Then copy the procedures to the IMS.PROCLIB data set.

Before IMS 15, the **PROCLIB** parameter in the IMSGEN macro determined whether the sample procedures were generated during system definition. In IMS 15, the **PROCLIB** parameter is obsolete. Specifications of **PROCLIB=YES** or **PROCLIB=IMS** trigger a warning message. Specifications of **PROCLIB=NO** are ignored.

The installation verification program (IVP) is modified to accommodate the changes to the sample procedures.

Changes to installing and defining IMS

The IVP is modified to use the sample procedures that are now provided during the SMP processing. The IVP copies the sample procedures from the IMS.SDFSPROC to IMS.PROCLIB data set and uses the symbolic variables that are introduced in the sample procedures in the IMS.SDFSPROC data set. Some IVP steps are modified because of this enhancement.

The **PROCLIB** parameter of the IMSGEN macro is ignored. The default value of this parameter is set to **PROCLIB=NO**. Only **PROCLIB=NO** is allowed.

If PROCLIB=YES or PROCLIB=IMS is specified, the IMS system generates an MNOTE return code 2, with message G918 issued.

Changes to administering IMS

The sample procedures are installed by the SMP processing in the IMS.SDFSPROC. Copy the sample procedures to IMS.PROCLIB data set. Rename and modify them so that the variable values match what they would have been if generated by the SYSGEN processing.

The IVP is modified to accommodate the changes to the sample procedures.

Changes to utilities

This enhancement does not change any utilities; however, it does change how the sample procedures that a small number of utilities can use are created.

Type-1 Automated Operator Interface enhancement for 31-bit storage support

For IMS 15, the Type-1 Automated Operator Interface (AOI) exit routine interface module is enhanced to support obtaining 31-bit private storage from the Automated Operator Interface buffer pool (AOIP).

To pass information to the DFSAOUE0 exit routine, the AO exit routine interface module is modified to obtain 31-bit private storage from the AOI buffer pool,

instead of obtaining 24-bit private storage from the communication I/O buffer pool (CIOP). If available space is not sufficient in the AOI buffer pool, the new DFS3458 message is issued.

Migration considerations

IMS 15 and later systems obtain the user exit header block (UEHB) and any additional buffers that are passed to the DFSAOUE0 exit routine from 31-bit private storage.

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 15, see Chapter 8, "Message and code changes in IMS 15," on page 91.

Part 3. IBM IMS Tools support for IMS 15

The IBM IMS Tools products are designed to enhance the performance and operation of IMS, and are upgraded and enhanced to work with IMS 15.

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 15.

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 14. 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, V2.2

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:
- Chapter 15, "IBM IMS Administration Tool for z/OS V1.1," on page 143
- "IBM IMS Database Reorganization Expert for z/OS, V4.1" on page 170
- "IBM IMS High Performance Image Copy for z/OS, V4.2" on page 149
- "IBM IMS High Performance Load for z/OS, V2.1" on page 170
- "IBM IMS High Performance Pointer Checker for z/OS, V3.1" on page 171
- "IBM IMS High Performance Prefix Resolution for z/OS, V3.1" on page 171
- "IBM IMS High Performance Unload for z/OS, V1.2" on page 171
- "IBM IMS Index Builder for z/OS, V3.1" on page 149
- "IBM IMS Library Integrity Utilities for z/OS, V2.2" on page 172

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 , V2.2.

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 149
- "IBM IMS Library Integrity Utilities for z/OS, V2.2" on page 172

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 157
- "IBM IMS Performance Analyzer for z/OS, V4.4" on page 158
- "IMS Problem Investigator for z/OS, V2.4" on page 158

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 149
- "IBM IMS Index Builder for z/OS, V3.1" on page 149

Program number: 5655-ISR

IBM Tools Base for z/OS, V1.6

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.

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 V1.6 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 16, "IBM Management Console for IMS and Db2 for z/OS, V1.1," on page 145

Chapter 15. IBM IMS Administration Tool for z/OS V1.1

IBM IMS Administration Tool centralizes the control of key functions for IMS database administrators, giving you the tools you need to manage IMS DBD and PSB resources.

IMS Administration Tool includes reports on IMS catalog space usage that help you compare catalog contents to your current ACB libraries. You can also issue IMS commands, generate simple, task-based JCL for IMS maintenance, and run SQL statements to access your IMS data. All of this is available in one tool through a graphical web user interface or a traditional ISPF interface.

Related information:

➥ IBM IMS Administration Tool for z/OS

Chapter 16. 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 17. 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 18. 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 149
- "IBM IMS Index Builder for z/OS, V3.1" on page 149

Program number: 5655-ISR

Chapter 19. 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, V2.2

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:
- Chapter 15, "IBM IMS Administration Tool for z/OS V1.1," on page 143
- "IBM IMS Database Reorganization Expert for z/OS, V4.1" on page 170
- "IBM IMS High Performance Image Copy for z/OS, V4.2" on page 149
- "IBM IMS High Performance Load for z/OS, V2.1" on page 170
- "IBM IMS High Performance Pointer Checker for z/OS, V3.1" on page 171
- "IBM IMS High Performance Prefix Resolution for z/OS, V3.1" on page 171
- "IBM IMS High Performance Unload for z/OS, V1.2" on page 171
- "IBM IMS Index Builder for z/OS, V3.1" on page 149
- "IBM IMS Library Integrity Utilities for z/OS, V2.2" on page 172

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 , V2.2.

Program number: 5655-DSP

Chapter 20. 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 increases 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 21. 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 increases 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 22. 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 157
- "IBM IMS Performance Analyzer for z/OS, V4.4" on page 158
- "IMS Problem Investigator for z/OS, V2.4" on page 158

Program number: 5655-S42

Chapter 23. 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 24. 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 25. 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 26. 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 27. 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 to 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, V1.6" on page 140 (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, V2.2

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:
- Chapter 15, "IBM IMS Administration Tool for z/OS V1.1," on page 143
- "IBM IMS Database Reorganization Expert for z/OS, V4.1" on page 170
- "IBM IMS High Performance Image Copy for z/OS, V4.2" on page 149
- "IBM IMS High Performance Load for z/OS, V2.1" on page 170
- "IBM IMS High Performance Pointer Checker for z/OS, V3.1" on page 171
- "IBM IMS High Performance Prefix Resolution for z/OS, V3.1" on page 171
- "IBM IMS High Performance Unload for z/OS, V1.2" on page 171
- "IBM IMS Index Builder for z/OS, V3.1" on page 149
- "IBM IMS Library Integrity Utilities for z/OS, V2.2" on page 172

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 , V2.2.

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 149
- "IBM IMS Library Integrity Utilities for z/OS, V2.2" on page 172

Program number: 5655-W14

Chapter 28. Miscellaneous IBM tools that support IMS

The following IMS tools also support IMS 15.

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, V14.1
> 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 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 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 15

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 29. 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 applictions to call REST APIs (in z/OS Connect EE documentation in IBM Knowledge Center).

IBM z/OS Connect Enterprise Edition V3.0 Program number: 5655-CE3

Chapter 30. Application development with IMS Explorer for Development

IMS Explorer for Development (IMS Explorer) is a component in IMS Enterprise Suite that facilitates application development and extends access to IMS transactions and data through an Eclipse-based user interface.

Version 3.3 of IMS Enterprise Suite is made available for the distributed platform only to provide a new release of IMS Explorer that is based on Eclipse Neon V4.6. For shell-sharing with Eclipse Luna V4.4-based IBM products, use the IMS Explorer in IMS Enterprise Suite Version 3.2 instead.

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

Version 3.3 adds the support for IMS V15.

Related reference:

"IMS Enterprise Suite software requirements" on page 14

➥ IMS Explorer for Development overview

Chapter 31. 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 32. Integrated access with IBM DataPower Gateway

The IBM DataPower Gateway provides integrated support for IMS 15. 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® 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 15 library.

Title	Acronym	Order number
IMS Version 15 Application Programming	APG	SC27-6778
IMS Version 15 Application Programming APIs	APR	SC27-6779
IMS Version 15 Commands, Volume 1: IMS Commands A-M	CR1	SC27-6780
IMS Version 15 Commands, Volume 2: IMS Commands N-V	CR2	SC27-6781
IMS Version 15 Commands, Volume 3: IMS Component and z/OS Commands	CR3	SC27-6782
IMS Version 15 Communications and Connections	CCG	SC27-6783
IMS Version 15 Database Administration	DAG	SC27-6784
IMS Version 15 Database Utilities	DUR	SC27-6785
IMS Version 15 Diagnosis	DGR	GC27-6786
IMS Version 15 Exit Routines	ERR	SC27-6787SC19-4208
IMS Version 15 Installation	INS	SC27-6788
IMS Version 15 Licensed Program Specifications	LPS	GC27-6799
IMS Version 15 Messages and Codes, Volume 1: DFS Messages	MC1	GC27-6789
IMS Version 15 Messages and Codes, Volume 2: Non-DFS Messages	MC2	GC27-6790
IMS Version 15 Messages and Codes, Volume 3: IMS Abend Codes	MC3	GC27-6791
IMS Version 15 Messages and Codes, Volume 4: IMS Component Codes	MC4	GC27-6792
IMS Version 15 Operations and Automation	OAG	SC27-6793
IMS Version 15 Release Planning	RPG	GC27-6794
IMS Version 15 System Administration	SAG	SC27-6795
IMS Version 15 System Definition	SDG	GC27-6796
IMS Version 15 System Programming APIs	SPR	SC27-6797
IMS Version 15 System Utilities	SUR	SC27-6798

Index

A

ACB (application control block)
  IMS management of ACBs 32
    coexistence considerations 81, 82
ACBs (application control blocks)
  fallback
    IMS management of ACBs 71, 72
  falling back from the IMS
    management of ACBs 71, 72
  migration considerations 33
accessibility
  features x
  keyboard shortcuts x
Administration Tool for z/OS, IMS 143
application control block (ACB)
  IMS management of ACBs
    coexistence considerations 81, 82
application control blocks (ACBs)
  fallback
    IMS management of ACBs 71, 72
  falling back from the IMS
    management of ACBs 71, 72
  migration considerations 33
application management tools 147
Application Performance Analyzer for
  z/OS 175
application programs supported 16

B

BUFPOOLS macro removal
  enhancement 129

C

callout function
  software requirements 10
catalog
  fallback considerations for
    IMS-managed ACBs 71, 72
  fallback considerations for the IMS
    catalog 69
catalog, IMS
  migration considerations 30
  migration steps 30, 32
Changed messages and codes for IMS
  15 92
checkpoints
  CQS automatic structure checkpoint
    enhancement 113
CICS subsystems supported 15
codes
  changed in this release 92
  deleted in this release 93
  new in this release 91
  new, changed, and deleted 91
coexistence
  APAR summary 75

coexistence *(continued)*
  DBRC
    small programming
      enhancements 77
  exit routines 80
  Fast Database Recovery (FDBR) 80
  functional areas 76
  IMS Universal drivers 86
  Open Database enhancement 86
  PTF summary 75
  PTFs
    unprocessed coexistence service,
      identifying 75
  restrictions for IMS 15 87
  unprocessed coexistence service,
    identifying 75
coexistence considerations
  ACBs, IMS management of 81, 82
  CQS 76
  database recovery utilities 79
  DEDB Alter enhancement 79
  DRA 79
  dynamic resource definition 80
  general 76
  HALDB support for 8-GB OSAM data
    sets enhancement 80
  IMS 64-bit storage manager 81
  IMS catalog 81
  IMS Connect 84
  IMS management of ACBs 81, 82
  IMS repository function 82
  MSC 85
  Network security credential
    propagation enhancement 86
  OTMA 86
  shared message queues 87
  Syntax Checker 87
  utilities 79
Command Control Facility for z/OS,
  IMS 163
commands
  enhancements 89
common queue server (CQS)
  automatic checkpoint
    fallback considerations 74
Common Queue Server (CQS)
  automatic structure checkpoint
    enhancement 113
  return code enhancements for z/OS
    Logger write errors 120
  z/OS Logger, return code
    enhancements for 120
Communication Controller for Linux on
  System z (CCL)
    withdrawal from support 41
Configuration Manager for z/OS,
  IMS 167
continuous delivery
  explanation 105
  overview 105

coupling facility
  requirements 3
CQS (Common Queue Server)
  automatic structure checkpoint
    enhancement 113
  return code enhancements for z/OS
    Logger write errors 120
  z/OS Logger, return code
    enhancements for 120

D

DASD requirements 4
Data Encryption for Db2 and IMS
  Databases, InfoSphere Guardium 161
Data Explorer
  See Watson Explorer
data sharing
  requirements 3, 10
Database Change Accumulation utility
  (DFSUCUM0)
    fallback considerations 67
Database Control Suite for z/OS,
  IMS 169
Database Recovery Control (DBRC)
  coexistence requirements 77
  fallback considerations 67
  migration to IMS 15 41
  RECON data set
    migrating 43
Database Reorganization Expert for z/OS,
  IMS 170
Database Resource Adapter (DRA)
  coexistence considerations 79
  software requirements 10
Database Solution Pack for z/OS,
  IMS 139, 151, 173
database versioning
  migration considerations 29
databases
  DEDB Alter utility enhancements 111
  modifying
    DEDB Alter utility
      enhancements 111
DataPower Gateway
  IMS support 185
Db2 for z/OS
  subsystem supported 15
DBRC
  migration
    changes to the RECON data set in
      IMS 15 42
  RECON data set
    changes in IMS 15 42
DBRC (Database Recovery Control)
  coexistence requirements 77
  fallback considerations 67
  migration to IMS 15 41
  RECON data set
    migrating 43

IBM®

Spine information:

IMS   Version 15.1.0   Release Planning

IBM