IMS V10 Product Enhancements

(Course code CMA0)

Student Notebook

ERC 1.0

Authorized

Training

Trademarks

IBM® is a registered trademark of International Business Machines Corporation.

The following are trademarks of International Business Machines Corporation in the United States, or other countries, or both:

- AIX®
- CICS®
- DB2®
- DFS™
- DFSMS™
- DFSMSdfp™
- DFSMSdss™
- DS8000™
- Enterprise Storage Server®
- FlashCopy®
- IMS™
- MVS™
- Parallel Sysplex®
- ProductPac®
- RACF®
- RAMAC®
- Rational®
- REXX™
- RMF™
- SP™
- SystemPac®
- System z9®
- TotalStorage®
- Virtualization Engine™
- VTAM®
- WebSphere®
- z9™
- z/OS®
- zSeries®

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

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

July 2008 edition

The information contained in this document has not been submitted to any formal IBM test and is distributed on an “as is” basis without any warranty either express or implied. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer’s ability to evaluate and integrate them into the customer’s operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will result elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

© Copyright International Business Machines Corporation 2007, 2008. All rights reserved.
This document may not be reproduced in whole or in part without the prior written permission of IBM.
Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Contents

Trademarks xix

Course description xxi

Agenda xxiii

Unit 1. IMS Version 10 Enhancements 1-1
- [No title] 1-2
- Class Agenda 1-3
- Class Agenda 1-4
- Class Agenda 1-5
- Class Agenda 1-6
- Class Agenda 1-7
- Class Agenda 1-8
- Software Prerequisites 1-9
- Supported Migrations and Coexistence 1-11

Unit 2. System Enhancements 2-1
- System Enhancements 2-2
- [No title] 2-3
- Transaction Scheduling 2-4
- FPCTRL System Definition Macro Eliminated 2-6
- DFSDFxxx PROCLIB Member 2-8
- DFSDFxxx PROCLIB Member 2-9
- DFSDFxxx PROCLIB Member 2-11
- DFSDFxxx PROCLIB Member 2-13
- [No title] 2-14
- Enhanced Display of System Parameters 2-15
- Enhanced Display of System Parameters 2-17
- [No title] 2-18
- VSCR - Savings 2-19
- DIRCA 2-20
- Benefits 2-21
- Sysplex Serial Program Management(SSPM) 2-22
- Sysplex Serial Program Management (SSPM) 2-23
- Implementation via IMS Resource Manager (RM) 2-25
- Implementation via IMS Resource Manager (RM) 2-27
- Sysplex Serial Program Management Summary 2-28
- [No title] 2-29
- Enhanced Log Record Statistics 2-30
- Enhanced Log Record Statistics 2-31
- Enhanced Log Record Statistics 2-32
- Enhanced Log Record Statistics 2-34
- Enhanced Log Record Statistics 2-36

[No title] 2-37
Large Sequential Data Sets 2-38
Large Sequential Data Sets 2-39
[No title] 2-41
Log Transaction Analysis Utility (DFSILTA0) 2-42
Statistical Analysis Utility (DFSISTS0) 2-44
System Utilities Enhancements 2-46
System Utilities Enhancements 2-48
[No title] 2-49
BPE Highlights 2-50
Implementation 2-51
Setup - BPE Configuration 2-52
Setup - BPE Configuration 2-54
Setup - BPE Configuration 2-56
Commands 2-57
DISPLAY Command 2-58
UPDATE Command 2-59
Performance Impacts 2-60
Benefits 2-61
BPE External Trace Appendix 2-62
Setup - GDG Base 2-63
Setup - GDG Base 2-64
Setup - GDG Base 2-65
Setup - GDG Model 2-66
GDG Data Set Allocation/Deallocation Flow 2-67
Data Formatting and Printing 2-68
Data Formatting and Printing - IPCS 2-69
Data Formatting and Printing - IPCS 2-70
Data Formatting and Printing - IPCS 2-71
Data Formatting and Printing - IPCS 2-72
Data Formatting and Printing - IPCS 2-73
Data Formatting and Printing - IPCS 2-74
Data Formatting and Printing - IPCS 2-75
Data Formatting and Printing - IPCS 2-76
Data Formatting and Printing - IPCS 2-77
Data Formatting and Printing - Batch mode 2-78
Data Formatting and Printing - Batch mode 2-79
Formatted Data - Example 2-80
New Messages 2-81
[No title] 2-83
Highlights 2-84
Functionality 2-85
Functionality 2-87
Example of E-mail Notification - Graphics 2-89
Example of E-mail Notification - Text Only 2-90
Setup 2-91
On Demand Interface 2-92
On Demand Interface 2-93

Prerequisites 2-95
Benefits 2-96
System Enhancements 2-97

Unit 3. Dynamic Resource Definition (DRD) 3-1
Dynamic Resource Definition (DRD) Overview 3-2
Environment for DRD 3-4
[No title] 3-6
Resources 3-7
Commands Used in DRD 3-8
Resource Attributes and Statuses 3-9
CREATE Command 3-10
Descriptors 3-11
CREATE Command with LIKE Parameter 3-12
IMS-defined Descriptor 3-13
User-defined Descriptor 3-14
QUERY Command to Show Attributes 3-15
[No title] 3-16
Using DRD 3-17
PROCLIB Members 3-19
PROCLIB Members 3-20
A Procedure to Enable DRD 3-21
Processing Flow with DRD Enabled 3-23
Processing Flow with DRD Enabled 3-25
[No title] 3-26
RDDS Extraction Utility 3-27
Security Considerations 3-28
[No title] 3-29
IMS System Generation Considerations 3-30
Elimination of MODBLKS 3-31
Summary 3-32
Summary 3-33

Unit 4. System Management 4-1
Systems Management Enhancements 4-2
[No title] 4-3
New MSC Commands 4-4
UPDATE MSPLINK 4-5
UPDATE MSPLINK (cont.) 4-6
UPDATE MSLINK 4-7
UPDATE MSLINK (cont.) 4-8
UPDATE MSLINK (cont.) 4-9
UPDATE MSNAME 4-10
UPDATE MSNAME Attributes 4-11
QUERY MSPLINK 4-12
Example – QRY MSPLINK SHOW(ALL) 4-14
Example - QUERY MSPLINK 4-15
QUERY MSLINK 4-16

Example – QRY MSLINK 4-18
QUERY MSNAME 4-19
[No title] 4-21
Example of Using UPDATE Commands for MSC 4-22
[No title] 4-23
[No title] 4-24
[No title] 4-25
MSC Commands Summary 4-26
[No title] 4-27
QUERY (QRY) Command Enhancements 4-28
QRY PGM and QRY RTC Commands 4-29
QRY Resource or Descriptor – Definition Attributes 4-30
QRY Resource – Definition Type and Model 4-31
QRY Resource – Timestamps 4-32
QRY Resource Example 4-34
[No title] 4-35
QUEUE (QUE) TRAN - ENQ 4-36
QUEUE TRAN - ENQ 4-37
QUEUE TRAN - ENQ 4-39
QUEUE (QUE) TRAN - DEQ 4-40
QUEUE LTERM - ENQ 4-41
QUE LTERM - DEQ 4-42
QUEUE (QUE) TRAN Command 4-43
[No title] 4-44
Global Status 4-45
Global Status and Global PLEXPARM 4-46
What Is Global Status? 4-47
Global Status vs Local Status 4-48
What Is the Global PLEXPARM? 4-49
QUERY Command Support 4-51
Batch Jobs and Utilities 4-53
IMS Initialization 4-54
IMS Restart 4-55
Global PLEXPARM – Structure Failure 4-57
Global Status Summary 4-58
Systems Management Enhancements 4-59

Unit 5. Managing Resources. 5-1
Manage Resources Highlights 5-2
Manage Resources Highlights 5-3
Hierarchical Panel Structure 5-4
[No title] 5-5
[No title] 5-6
[No title] 5-7
[No title] 5-8
Create Resources 5-9
[No title] 5-10
[No title] 5-11

[No title] . . . 5-12
[No title] . . . 5-13
Create Resource . . . 5-14
[No title] . . . 5-15
[No title] . . . 5-16
[No title] . . . 5-17
[No title] . . . 5-18
Querying Resources . . . 5-19
[No title] . . . 5-20
Manage Resources Summary . . . 5-21

Unit 6. Operations. . . . 6-1
Operations Enhancements . . . 6-2
[No title] . . . 6-3
OM Terminology Review . . . 6-4
Operations Manager Enhancements . . . 6-5
OM Audit Trail . . . 6-6
OM Audit Trail . . . 6-8
Printing the new OM Audit Trail . . . 6-9
Sample Listing of OM Audit Trail using CSLULALE . . . 6-11
Sample Listing of OM Audit Trail using CSLOERA3 (command input/output) . . . 6-12
Unsolicited Message Support . . . 6-13
Unsolicited Message Support . . . 6-14
Unsolicited Message Support - Tailoring . . . 6-15
OM Audit Trail Migration Considerations . . . 6-17
[No title] . . . 6-18
TSO SPOC Support for OM Enhancements . . . 6-19
REXX Support for Audit Trail . . . 6-20
REXX Functions . . . 6-21
Sample REXX Code . . . 6-22
[No title] . . . 6-24
Batch SPOC Utility (CSLUSPOC) . . . 6-25
CSLUSPOC Parameters . . . 6-26
Sample Batch SPOC Utility JCL . . . 6-27
//SYSIN File . . . 6-28
//SYSPRINT Output . . . 6-29
//SYSPRINT Output . . . 6-30
[No title] . . . 6-31
Prior to V10 . . . 6-32
Sample XML Output . . . 6-33
Using REXX Parser Function . . . 6-34
Sample XML Output with REXX Stem Variables . . . 6-35
Using REXX Parser Function . . . 6-36
Summary . . . 6-37
[No title] . . . 6-38
Action Bar Enhancements . . . 6-39
Action Bar Enhancements . . . 6-40
[No title] . . . 6-41

Improved Help 6-42
Support for OM Audit Trail 6-43
[No title] 6-44
[No title] 6-45

Unit 7. Online Change 7-1
Online Change Enhancements in IMS V10 7-2
[No title] 7-3
ACBLIB Member Online Change 7-4
ACBLIB Member Online Change Overview 7-5
ACBLIB Member Online Change Overview 7-6
Step 1: Build updated member(s) 7-7
Step 2: (Optional) Create Backup of ACTIVE ACBLIB 7-9
Step 3: OLC INIT commands 7-10
Setup for ACBLIB Member Online Change 7-12
Setup for ACBLIB Member Online Change (cont.) 7-14
IMSplex Consideration 7-16
ACBLIB Member Online Change Phases 7-18
Prepare Phase 7-19
Commit Phase 1 7-20
Commit Phase 2 7-21
Example Commands 7-23
INIT OLC PHASE(PREPARE) Command Example 7-24
QUERY OLC SHOW(RSCLIST) Command Example 7-25
QUERY MEMBER TYPE(IMS) Command Example .7-26
INIT OLC PHASE(COMMIT) Command 7-27
TERMINATE OLC Command 7-28
ACBLIB Member Online Change Summary 7-29
[No title] 7-30
Online Change COMMIT Processing Enhancement .7-31
Online Change COMMIT Processing Enhancement .7-33
Online Change COMMIT Processing Enhancement Summary 7-34

Unit 8. Security. 8-1
Highlights 8-2
Support for Mixed-case Passwords 8-3
Support for Mixed-case Passwords 8-5
Security Auditing Improvements 8-6
Security Auditing Improvements 8-8
Security Auditing Improvements 8-9
Security Auditing Improvements 8-10
Enhanced Security for IMS Conversations 8-11
Enhanced Security for IMS Conversations 8-12
Enhanced Security for IMS Conversations 8-13
SMU Support Removed 8-15
SMU Support Removed - Impact 8-16
SMU Support Removed - Impact 8-17
SMU Support Removed – Impact 8-19

SMU Support Removed – Impact … 8-21
SMU Support Removed – Impact … 8-22
SMU Support Removed – Impact … 8-23
SMU to SAF/RACF Migration 8-24

Unit 9. Transaction Manager 9-1
Transaction Manager 9-2
MSC 9-3
Highlights 9-4
Old MSC Routing Exits Removed 9-6
Increased Bandwidth - Background 9-8
Increased Bandwidth… 9-9
Increased Bandwidth… 9-10
Increased Bandwidth… 9-11
Increased Bandwidth 9-13
Increased Bandwidth - Implementation 9-15
Increased Bandwidth - Commands 9-16
Increased Bandwidth - Commands … 9-17
Statistics 9-18
Statistics … 9-20
Statistics … 9-22
Statistics … 9-23
Statistics - Report example 9-24
. . . . 9-25
Example - Bandwidth Mode Setup 9-27
Example - Bandwidth Mode Setup … 9-28
Performance results 9-29
Migration Considerations 9-30
Migration Considerations … 9-31
Benefits 9-32
MSC VGR Support 9-33
MSC VGR Support … 9-34
MSC VGR Support … 9-35
MSC VGR Support … 9-36
MSC VGR Support … 9-37
[No title] 9-38
Highlights 9-39
Enhanced Timeout Granularity 9-40
Enhanced Timeout Granularity - Implementation 9-42
Support for /LOCK and /UNLOCK 9-44
Support for /LOCK and /UNLOCK … 9-45
Support for /LOCK and /UNLOCK … 9-46
Local LU 9-47
Local LU … 9-49
Local LU … 9-50
Local LU … 9-51
Local LU … 9-52
Local LU … 9-53

Local LU 9-54
Local LU Security .9-55
Benefits .9-56
Removal of BTAM Support .9-57
Highlights .9-58
An Example .9-60
Migration Considerations .9-61

Unit 10. Database .10-1
Database Enhancements .10-2
[No title] .10-3
Change Accum and Prefix Resolution Sort .10-4
ACBGEN Exploitation of Storage Above the 16M Line .10-5
HALDB ILDS Rebuild Enhancement .10-6
[No title] .10-8
Image Copy 2 Fast Replication .10-9
Image Copy 2 Fast Replication .10-10
Image Copy 2 Fast Replication Invocation .10-12
Image Copy 2 Fast Replication Invocation .10-14
IC2 Fast Replication Output Data Set Name .10-15
DBRC with Fast Replication .10-16
Fast Replication with Database Recovery .10-18
Fast Replication .10-20
[No title] .10-21
Fuzzy User Image Copy Support .10-22
Fuzzy User Image Copy Support .10-24
Procedures for Clean and Fuzzy User ICs .10-25
Fuzzy User Image Copies .10-27

Unit 11. Fast Path .11-1
Fast Path Enhancements .11-2
[No title] .11-3
Starting All Areas with UPDATE DB command .11-4
Keeping Randomizer Resident when Stopping DB .11-5
Keeping Randomizer Resident when Stopping DB .11-6
[No title] .11-8
Shared VSO Private Buffer Pool Lookaside .11-9
Shared VSO in XRF Tracking and FDBR Systems .11-10
/START AREA GLOBAL Enhancement .11-11
[No title] .11-12
Maximum Number of FP Buffers Increased .11-13
Maximum Number of FP Output Threads Increased .11-14
FDBR FPBUFF= Parameter Default .11-16
[No title] .11-17
DFSVSMxx Specification for QUITCI .11-18
[No title] .11-19
EMH enhancement in IMS V10 .11-20
Fast Path Enhancements Summary .11-22

Unit 12. Connectivity 12-1
Connectivity 12-2
[No title] 12-3
Highlights 12-4
Routing Enhancements 12-5
Routing Enhancements 12-6
Routing Enhancements 12-8
Routing Enhancements 12-10
Resume TPIPE Security 12-12
Resume TPIPE Security 12-14
Resume TPIPE Security 12-15
Message Flood Detection and Control 12-16
Message Flood Detection and Control 12-17
Message Flood Detection and Control 12-19
CM1 (Send-then-Commit) Time-out Control 12-21
CM1 (Send-then-Commit) Time-out Control 12-22
CM1 (Send-then-Commit) Time-out Control 12-24
CM1 (Send-then-Commit) Time-out Control 12-26
TPIPE Storage Clean-up 12-28
TPIPE Storage Clean-up 12-29
Member Level Security 12-30
Member Level Security… 12-31
Asynchronous Message Enhancements 12-32
Asynchronous Message Enhancements 12-33
Asynchronous Message Enhancements 12-34
Asynchronous Message Enhancements 12-35
Asynchronous Message Enhancements 12-36
Asynchronous Message Enhancements 12-37
Asynchronous Message Enhancements 12-38
Asynchronous Message Enhancements 12-40
Asynchronous Message Enhancements 12-41
OTMA /DISPLAY Command Enhancements 12-42
OTMA Processing during Restart 12-44
Migration Considerations 12-45
OTMA Enhancements - Benefits 12-46
OTMA Enhancements - Benefits… 12-47
OTMA Enhancements - Benefits… 12-48
[No title] 12-49
Highlights 12-50
ACEE Aging Value Support 12-51
Client Password Change Request 12-53
Client Password Change Request 12-55
RACF Mixed Case Password Support 12-56
CM1 (Send-then-Commit) Time-out Control 12-58
CM1 (Send-then-Commit) Time-out Control 12-60
Message Flood Control 12-61
Message Flood Control 12-63
Asynchronous Message (CM0) Enhancements 12-64

Resume TPIPE Enhancements - Port Affinity 12-65
Resume TPIPE Enhancements - Port Affinity 12-66
Resume TPIPE Enhancements - Port Affinity 12-68
Resume TPIPE Enhancements - Alternate Clientid 12-69
Resume TPIPE Enhancements - Alternate Clientid 12-70
XML Adapter Support 12-71
XML Adapter Support - Overview 12-72
XML Adapter Support - Implementation 12-73
XML Adapter Support - Implementation 12-75
XML Adapter Support - Implementation 12-77
XML Adapter Support - Implementation 12-79
XML Adapter Support - Implementation 12-81
XML Adapter Support - Flow 12-82
XML Adapter Support - Considerations 12-83
XML Adapter Support - Considerations 12-84
IMS SOA Composite Business Application Support 12-85
IMS SOA Composite Business Application Support 12-86
IMS SOA Composite Business Application Support 12-88
IMS SOA Composite Business Application Support 12-89
IMS SOA Composite Business Application Support 12-90
IMS Connect Enhancements Benefits 12-91

Unit 13. IMS Integration Suite Enhancements. 13-1
IMS Integration Suite 13-2
[No title] 13-3
[No title] 13-4
IMS TM Resource Adapter 13-6
PL/I source message structures 13-7
[No title] 13-8
Commit Mode 1, Sync Level CONFIRM 13-10
[No title] 13-11
Retrieve output using alternate clientID 13-13
IMS TM Resource Adapter 13-14
IMS TM Resource Adapter 13-15
Asynchronous Callout to EJB 13-16
Asynchronous Callout to MDB 13-18
Asynchronous Callout 13-20
IMS TM Resource Adapter 13-21
IMS TM Resource Adapter 13-22
IMS TM Resource Adapter 13-23
IMS TM Resource Adapter 13-25
IMS SOA Composite Business Application support 13-26
[No title] 13-27
[No title] 13-28
IMS TM Resource Adapter 13-30
[No title] 13-31
[No title] 13-32
IMS SOAP Gateway Deployment Utility 13-33

IMS SOAP Gateway V10 Enhancements 13-34
Support on zOS with IMS 10 13-35
Asynchronous Callout to Web Service with XML Adapter 13-36
Asynchronous Callout to Web Service with XML Adapterwith a response 13-37
Asynchronous Callout to Web Service without XML Adapter 13-38
Asynchronous Callout 13-39
[No title] 13-40
IMS MFS Web SupportMFS Web Services and MFS Web Enablement 13-41
MFS Web Enablement 13-42
MFS Web Services – SOA support 13-43
[No title] 13-44
IMS DB Resource Adapter 13-45
IMS DB Resource Adapter 13-46
DB2 Java Stored Procedure return DB2 Result Sets 13-47
IMS DB Resource Adapter 13-49
IMS DB Resource Adapter 13-51
IBM SDK V5 for z/OS support 13-53
IBM SDK V5 for z/OS support 13-54
IBM SDK V5 for z/OS support 13-55
IBM SDK V5 for z/OS support - IMS Java Application sample API 13-56
IBM SDK V5 for z/OS support 13-57
[No title] 13-58
DLIModel Utility 13-59
DLIModel Utility 13-60
DLIModel Utility GUI Download 13-62
DLIModel Utility GUI 13-63
[No title] 13-64
XML Database - Review 13-65
[No title] 13-66
IMS DB Resource Adapter XQuery API 13-67
IMS DB Resource Adapter XQuery 13-69

Unit 14. DBRC 14-1
DBRC Enhancements and Migration 14-2
[No title] 14-3
DBRC Timestamp Precision 14-4
DBRC Timestamp Formats 14-6
DBRC Commands 14-8
Change Accumulation and Database Recovery 14-10
DBRC Timestamp Precision 14-12
[No title] 14-13
SAF (RACF) Support for RECONs 14-14
READONLY Support for RECONs 14-15
READONLY Support for RECONs 14-17
[No title] 14-19
IMSPLEX SCI Registration 14-20
DBRC Sharing Group ID 14-21
[No title] 14-23

DBRC API Enhancements 14-24
Alternate RECON and IMS DD Names 14-25
DBRC API RECON Update Capability 14-26
DBRC API Subsystem Registration 14-27
DBRC API Database Authorization 14-29
DBRC API FUNC=QUERY Enhancements 14-30
DBRC API Security Authorization 14-32
RECON Open 14-34
VERSION parameter in DSPAPI macros 14-35
DBRC API Enhancements 14-37
[No title] 14-38
Parallel RECON Access 14-39
Parallel RECON Access 14-40
Parallel RECON Access 14-41
Transactional VSAM (DFSMStvs) Overview 14-42
DBRC RECONs - DFSMStvs Environment 14-43
Transactional VSAM (DFSMStvs) Overview 14-44
Transactional VSAM (DFSMStvs) Overview 14-45
Parallel RECON Access Definition and Set Up 14-47
IMS Definitions for Parallel RECON Access 14-48
RECON Definitions for Parallel RECON Access .14-49
RECON Definitions for Parallel RECON Access .14-51
IFAPRDxx Member of SYS1.PARMLIB 14-52
SHCDS Data Sets 14-53
SYS1.PARMLIB Members 14-54
IGDSMSxx Member of SYS1.PARMLIB 14-55
Log Streams 14-57
Log Streams 14-59
Log Stream Definitions 14-60
SMSVSAM Lock Structure 14-62
SMSVSAM Cache Structures 14-64
Logger Structures 14-66
Security (RACF) Considerations 14-67
Parallel RECON Access Implementation 14-68
LIST.xxx Command Options 14-69
LIST.RECON output 14-71
Locking of VSAM Records with PRA 14-72
Deadlocks 14-73
Lock Timeouts 14-75
Retries After Lock Timeouts and Deadlocks 14-77
Monitoring of Locks 14-79
SMSVSAM Messages About Excessive Locks 14-80
RMF Information 14-81
RMF III - VSAM RLS Activity report 14-82
Recovery 14-83
Recovery 14-84
Peer Recovery Example 14-86
Changing Execution Parameters 14-87

RECON Header/Header Extension Processing 14-88
RECON Header/Header Extension Processing 14-90
Serial vs. Parallel RECON Access Comparison 14-92
PRA Migration and Coexistence 14-94
PRA Migration and Coexistence 14-95
PRA Summary and Benefits 14-96
Additional Information on DFSMStvs 14-97
DBRC Enhancements 14-99
[No title] 14-100
Supported Migrations and Coexistence 14-101
RECON Upgrade 14-102
RECON Listings 14-103
RECON Upgrade 14-104
MINVERS 14-105
IMSplex Name and DBRC Group ID 14-106
IMSplex and DBRC Sharing Group ID Specification 14-107
RECON Listings 14-109
DBRC Migration Steps 14-110

Unit 15. Installation and Migration 15-1
Installation and Migration 15-2
[No title] 15-3
Packaging 15-4
Software Prerequisites 15-5
Supported Connections 15-7
Supported Migrations and Coexistence - DBRC 15-8
Coexistence - IMS Utilities 15-10
Resource Consistency Checking Eliminated 15-11
Remote Site Recovery (RSR) Migration/Coexistence 15-13
Log Records 15-14
IMS Tools Migration/Coexistence 15-15
[No title] 15-16
IMS Library 15-17
IMS Library Reorganization 15-18
IMS Library Reorganization 15-19
IMS Library Reorganization 15-20
IMS Library Reorganization 15-21
IMS Library Reorganization 15-22
[No title] 15-23
Highlights 15-24
RACF Implementation Support 15-25
IMS Java Sample 15-27
IMS Java Sample 15-29
IMS Connect Sample Application 15-31
IMS Connect Sample Application 15-33
Operations Manager Sample 15-34
Manage Resources 15-35
DRD Example 15-36

DRD Example 15-37
DRD Example 15-38
Miscellaneous 15-39
Miscellaneous 15-40
Benefits 15-41
[No title] 15-42
Highlights 15-43
Invocation 15-44
Support for Members of PROCLIB 15-45
Support for Members of PROCLIB 15-47
Support for Comments 15-48
Support for Members of PROCLIB 15-49
New DISPLAY Formats 15-50
New “Sel” Field Options 15-51
New “Pull Down” Options 15-52
New Member Selection Panel 15-53
New Member Selection Panel 15-54
Benefits 15-55
[No title] 15-56
Knowledge-Base Log Analysis Highlights 15-57
Concurrent Levels of IMS with KBLA 15-58
Concurrent Levels of IMS with KBLA 15-59
Migration to V10 with KBLA 15-60
Migration to V10 with KBLA 15-61
An Example - Migration 15-62
Migration and Coexistence with KBLA 15-63
[No title] 15-64
Installation and Migration Tasks 15-65
Installation and Migration Tasks 15-66
Installation and Migration Tasks 15-68
[No title] 15-69
Migration Considerations 15-70
BTAM Support Removal 15-71
SMU Support Removed 15-73
SMU to SAF/RACF Migration 15-74
SMU Support Removed – Migration Considerations 15-75
Transaction Scheduling 15-76
FPCTRL System Definition Macro Eliminated 15-78
DSP1929I System Startup Message 15-80
IMS Application Menu 15-81
PSB Sizes Increase 15-82
ACBGEN Default Change for BUILD DBD= 15-83
Sysplex Serial Program Management 15-85
MSC Migration Considerations 15-86
MSC Migration Considerations 15-87
OTMA Migration Considerations 15-88
Log Transaction Analysis, Log Merge, and Statistical Analysis utilities 15-89
Change Accum and Prefix Resolution Sort 15-91

DBRC Timestamp Precision 15-92
Database Change Accumulation Utility 15-94
Database Recovery Utility 15-96
RECON Listings 15-98
DBRC API Security Authorization 15-99
VERSION parameter in DSPAPI macros 15-100
IMSplex Name and DBRC Group ID 15-102
IMSplex and DBRC Sharing Group ID Specification 15-103

Bibliography. X-1

Trademarks

The reader should recognize that the following terms, which appear in the content of this training document, are official trademarks of IBM or other companies:

IBM® is a registered trademark of International Business Machines Corporation.

The following are trademarks of International Business Machines Corporation in the United States, or other countries, or both:

AIX®
CICS®
DB2®
DFS™
DFSMS™
DFSMSdfp™
DFSMSdss™
DS8000™
Enterprise Storage Server®
FlashCopy®
IMS™
MVS™
Parallel Sysplex®
ProductPac®
RACF®
RAMAC®
Rational®
REXX™
RMF™
SP™
SystemPac®
System z9®
TotalStorage®
Virtualization Engine™
VTAM®
WebSphere®
z9™
z/OS®
zSeries®

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

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

Course description

IMS V10 Product Enhancements

Duration: 3 days

Purpose

This course describes the enhancements to IMS Version 10.

Audience

The audience for this class includes IMS Systems Programmers, Database Adminstrators, and IMS Applications Programmers.

Prerequisites

A knowledge of IMS is assumed throughout the class.

Agenda

Day 1

Welcome
Unit 1 - System Enhancements
Unit 2 - Dynamic Resource Definition (DRD)

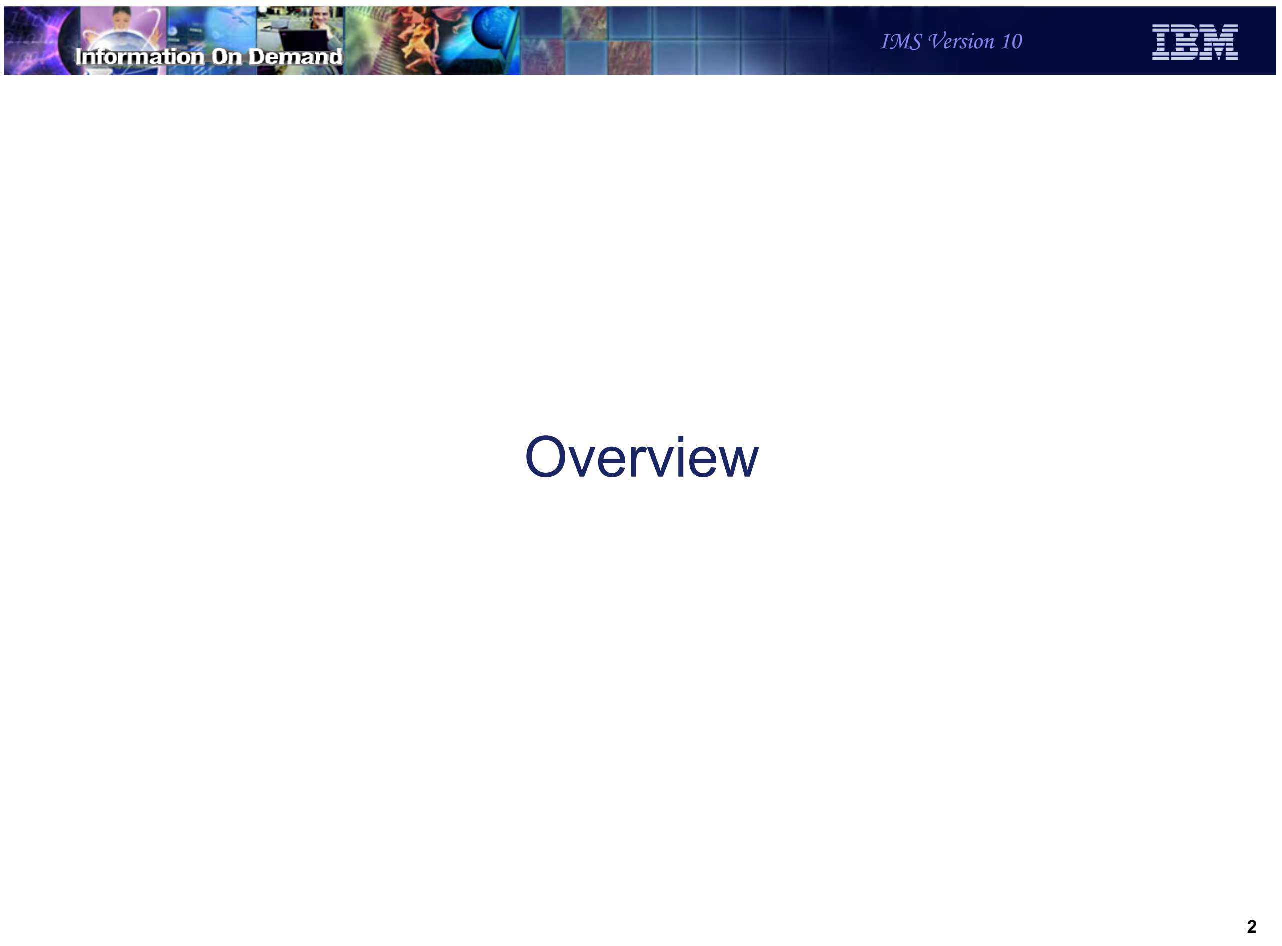
Day 2

Unit 3 - System Management Enhancements
Unit 4 - Managing Resources
Unit 5 - Operations Enhancements
Unit 6 - Online Change Enhancements
Unit 7 - Security
Unit 8 - Transaction Manager Enhancements
Unit 9 - Database Enhancements
Unit 10 - Fast Path Enhancements
Unit 11 - Connectivity Enhancements

Day 3

Unit 12 - IMS Integration Suite Enhancements
Unit 13 - DBRC Enhancements
Unit 14 - Installation and Migration

Unit 1. IMS Version 10 Enhancements



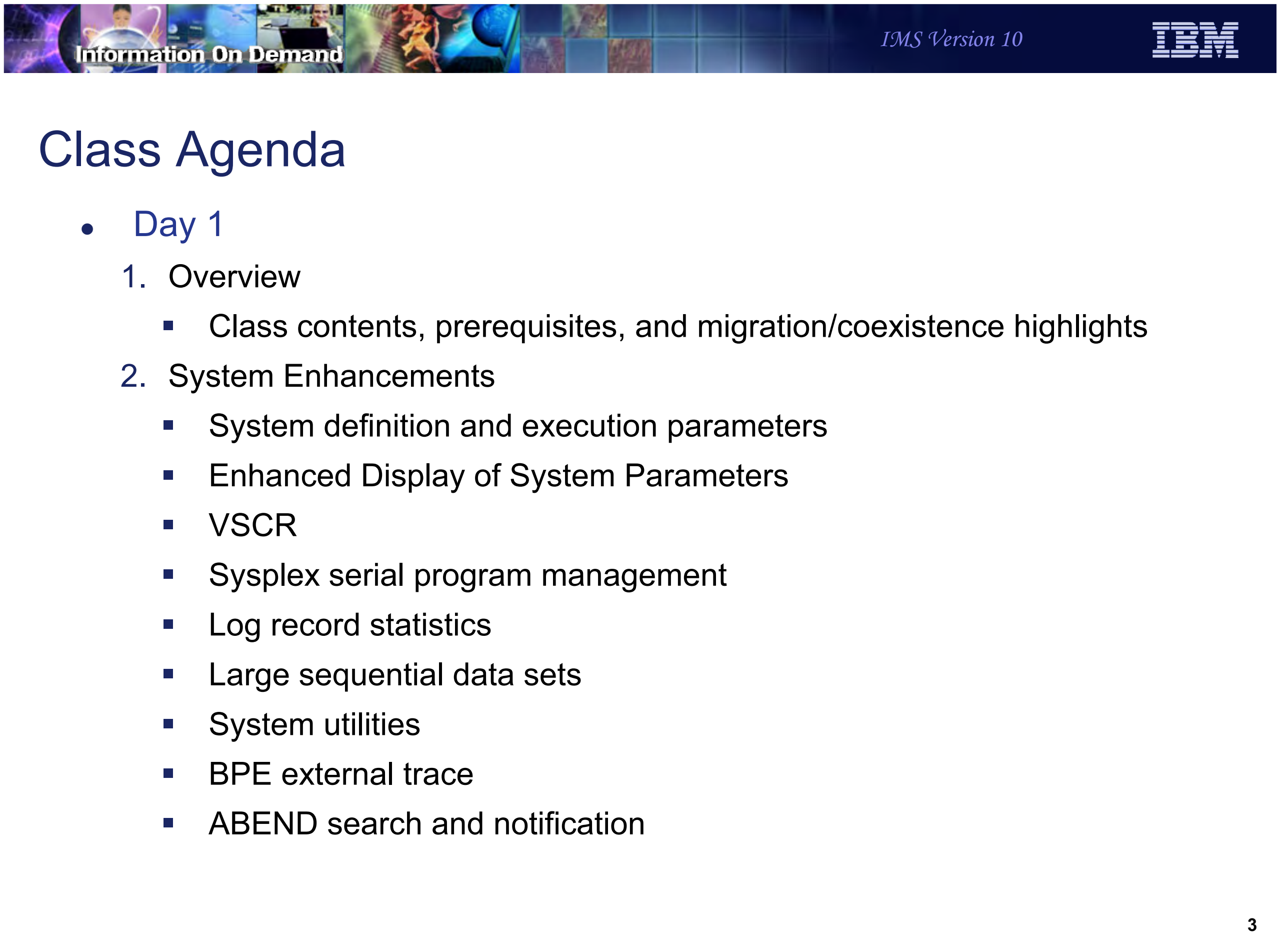
Information On Demand

IMS Version 10

Overview

Figure 1-1. [No title] CMA01.0

Notes:



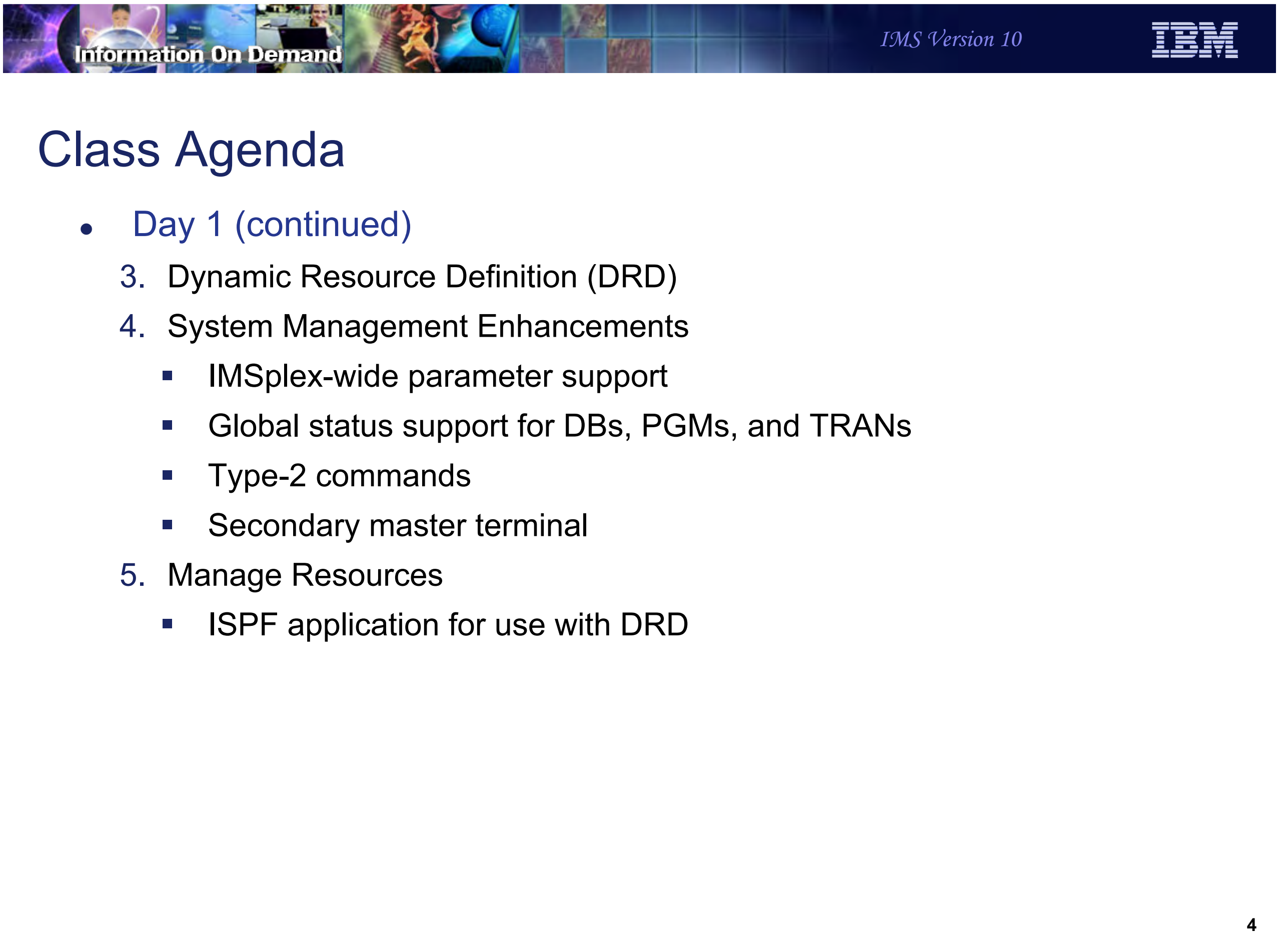
Information On Demand
IMS Version 10
IBM

Class Agenda

- Day 1
  1. Overview
     - Class contents, prerequisites, and migration/coexistence highlights
  2. System Enhancements
     - System definition and execution parameters
     - Enhanced Display of System Parameters
     - VSCR
     - Sysplex serial program management
     - Log record statistics
     - Large sequential data sets
     - System utilities
     - BPE external trace
     - ABEND search and notification

Figure 1-2. Class Agenda CMA01.0

Notes:



Information On Demand

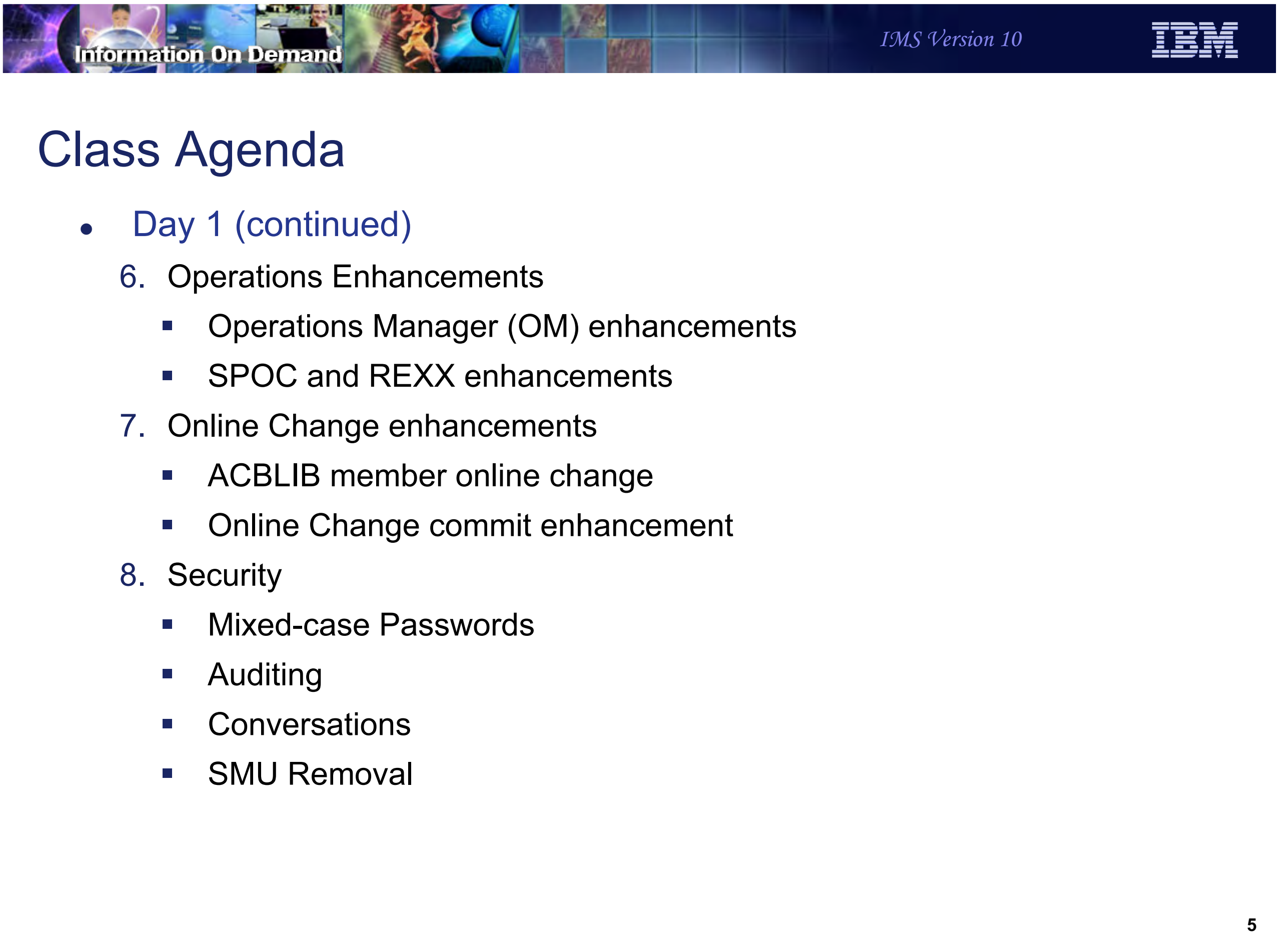
IMS Version 10

Class Agenda

- Day 1 (continued)
  3. Dynamic Resource Definition (DRD)
  4. System Management Enhancements
     - IMSplex-wide parameter support
     - Global status support for DBs, PGMs, and TRANs
     - Type-2 commands
     - Secondary master terminal
  5. Manage Resources
     - ISPF application for use with DRD

Figure 1-3. Class Agenda CMA01.0

Notes:

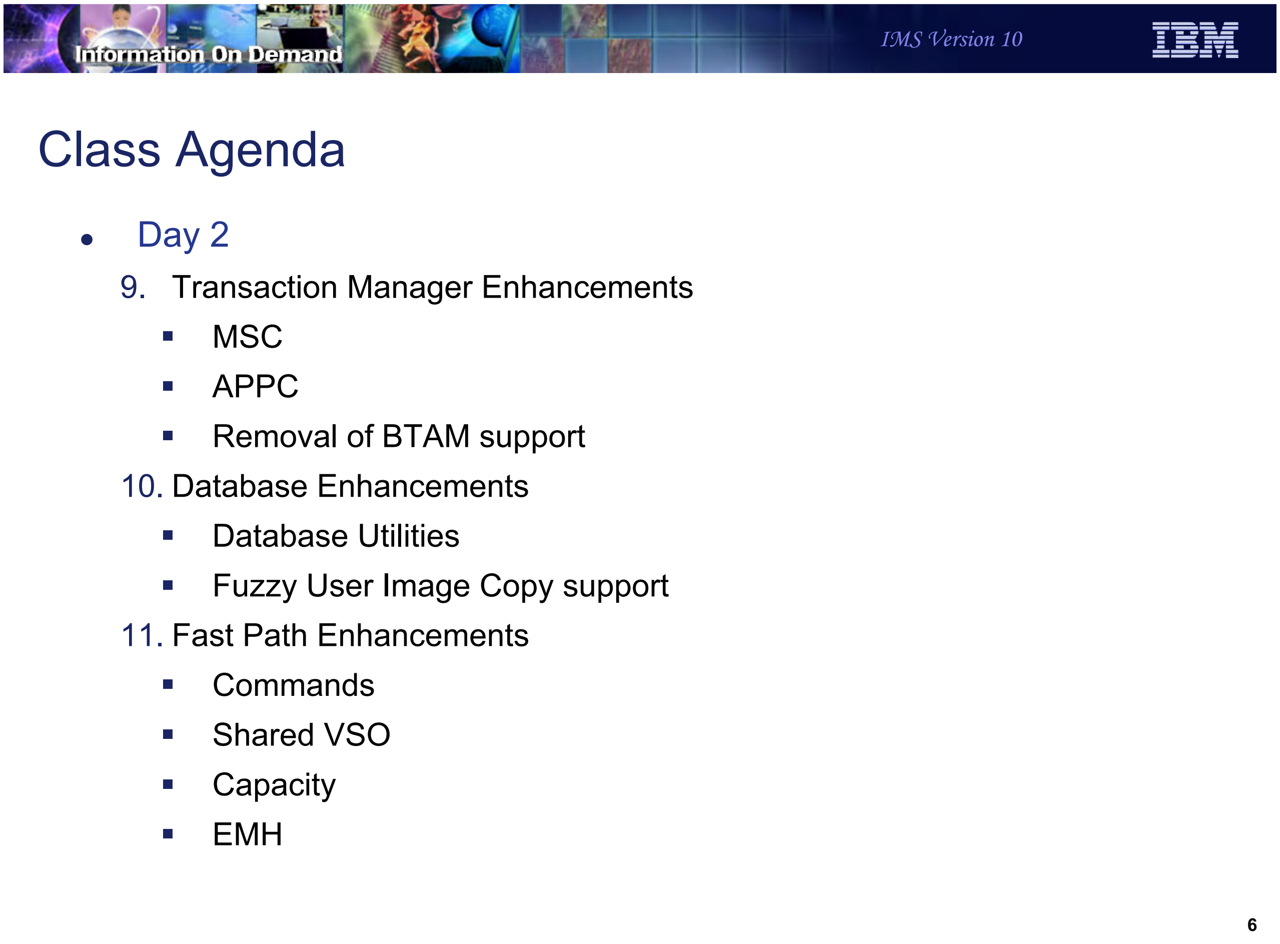

Information On Demand
IMS Version 10
IBM

Class Agenda

- Day 1 (continued)
  6. Operations Enhancements
     - Operations Manager (OM) enhancements
     - SPOC and REXX enhancements
  7. Online Change enhancements
     - ACBLIB member online change
     - Online Change commit enhancement
  8. Security
     - Mixed-case Passwords
     - Auditing
     - Conversations
     - SMU Removal

Figure 1-4. Class Agenda CMA01.0

Notes:


Information On Demand
IMS Version 10
IBM

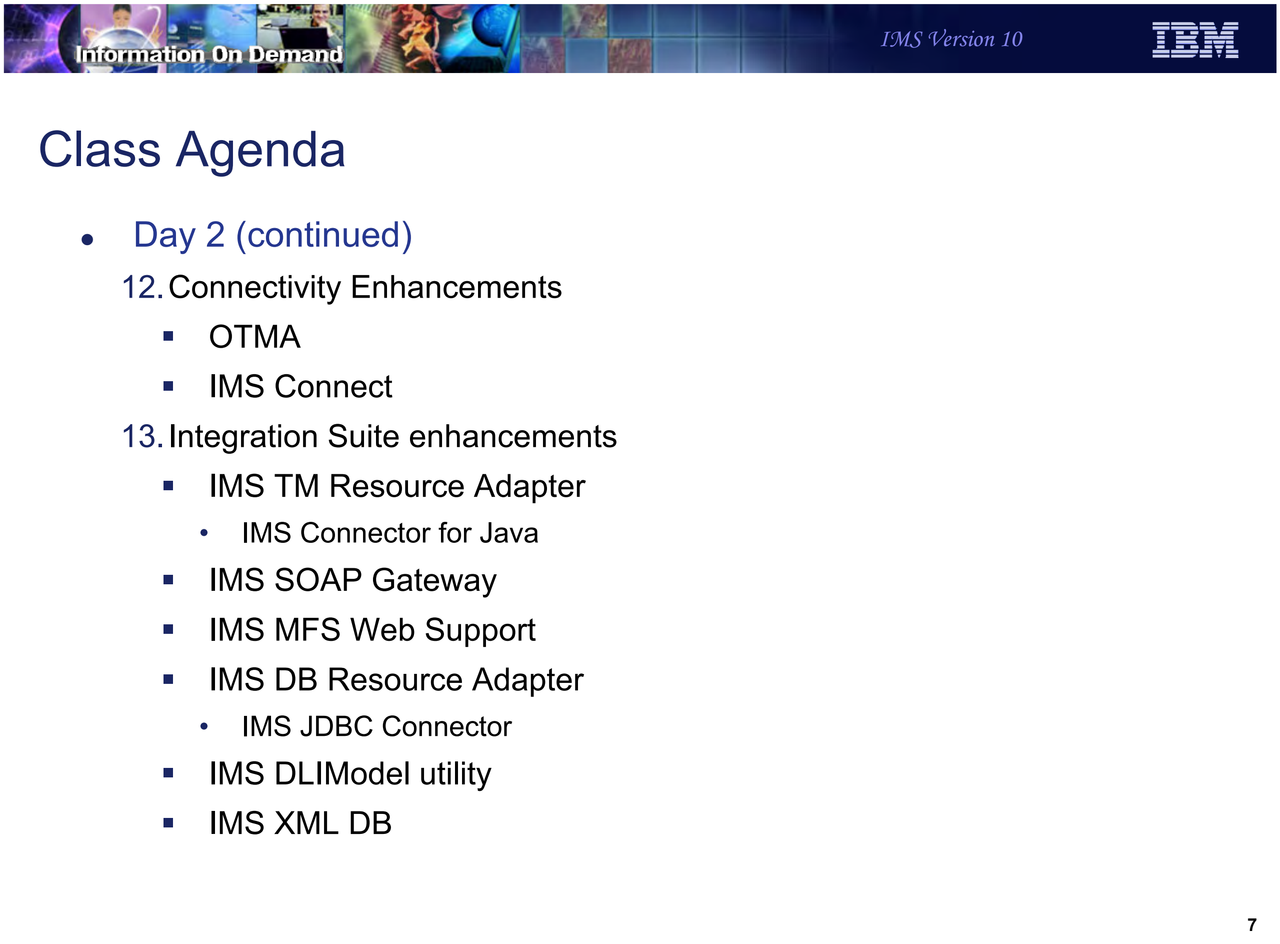
Class Agenda

- Day 2
  9. Transaction Manager Enhancements
     - MSC
     - APPC
     - Removal of BTAM support
  10. Database Enhancements
     - Database Utilities
     - Fuzzy User Image Copy support
  11. Fast Path Enhancements
     - Commands
     - Shared VSO
     - Capacity
     - EMH

6

Figure 1-5. Class Agenda CMA01.0

Notes:

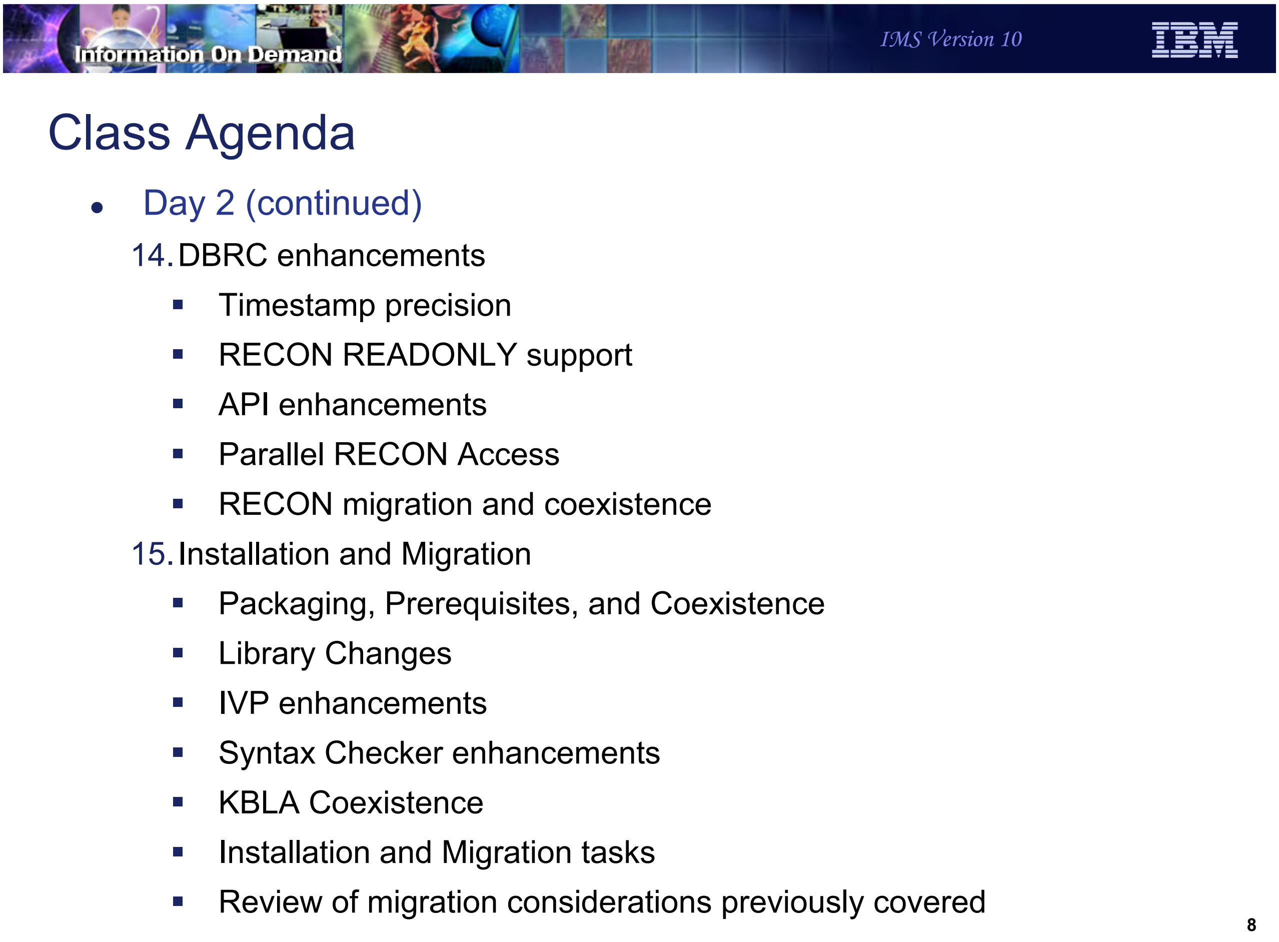

Information On Demand
IMS Version 10
IBM

Class Agenda

- Day 2 (continued)
  12. Connectivity Enhancements
      - OTMA
      - IMS Connect
  13. Integration Suite enhancements
      - IMS TM Resource Adapter
        - IMS Connector for Java
      - IMS SOAP Gateway
      - IMS MFS Web Support
      - IMS DB Resource Adapter
        - IMS JDBC Connector
      - IMS DLIModel utility
      - IMS XML DB

Figure 1-6. Class Agenda CMA01.0

Notes:


Information On Demand
IMS Version 10

Class Agenda

- Day 2 (continued)

14. DBRC enhancements
    - Timestamp precision
    - RECON READONLY support
    - API enhancements
    - Parallel RECON Access
    - RECON migration and coexistence
15. Installation and Migration
    - Packaging, Prerequisites, and Coexistence
    - Library Changes
    - IVP enhancements
    - Syntax Checker enhancements
    - KBLA Coexistence
    - Installation and Migration tasks
    - Review of migration considerations previously covered

Figure 1-7. Class Agenda CMA01.0

Notes:

Software Prerequisites

- Minimum software level prerequisites
  - z/OS V1R7 (5694-A01)
    - RACF, or equivalent, if security is used
    - High Level Assembler Toolkit (5696-234)
  - IRLM 2.2, if IRLM is used

- Minimum software levels for optional functions:
  - IMS V10 Image Copy 2 and Database Recovery fast replication
    - z/OS V1R8
  - DBRC Parallel RECON Access
    - z/OS DFSMStvs, a separately orderable feature of z/OS

Figure 1-8. Software Prerequisites CMA01.0

Notes:

The minimum level of z/OS for IMS V10 is z/OS V1R7. In addition to z/OS the user must install RACF, or an equivalent security product, in order to use security with IMS V10. As with previous IMS releases, the High Level Assembler Toolkit is required to provide assembler macros that IMS uses. If the IRLM is used, IRLM 2.2 is required. Program Isolation (PI) is also supported with IMS V10. IRLM is required for block level data sharing.

z/OS V1.7 runs on the following IBM System z9 and zSeries servers or equivalents:

- z9-109
- z900
- z990
- z800
- z890

z/OS V1R8 is required if the fast replication function of Image Copy 2 and the Database Recovery utility are used.

DBRC Parallel RECON Access requires z/OS DFSMStvs which is a separately orderable feature of z/OS V1R7 or z/OS V1R8.

Information On Demand
IMS Version 10
IBM

Supported Migrations and Coexistence

- IMS V9
  - Upgrade RECONs from IMS V9 to V10
  - Databases are compatible
  - Application programs are compatible

- IMS V8
  - Upgrade RECONs from IMS V8 to V10
  - Databases are compatible
  - Application programs are compatible

Figure 1-9. Supported Migrations and Coexistence CMA01.0

Notes:

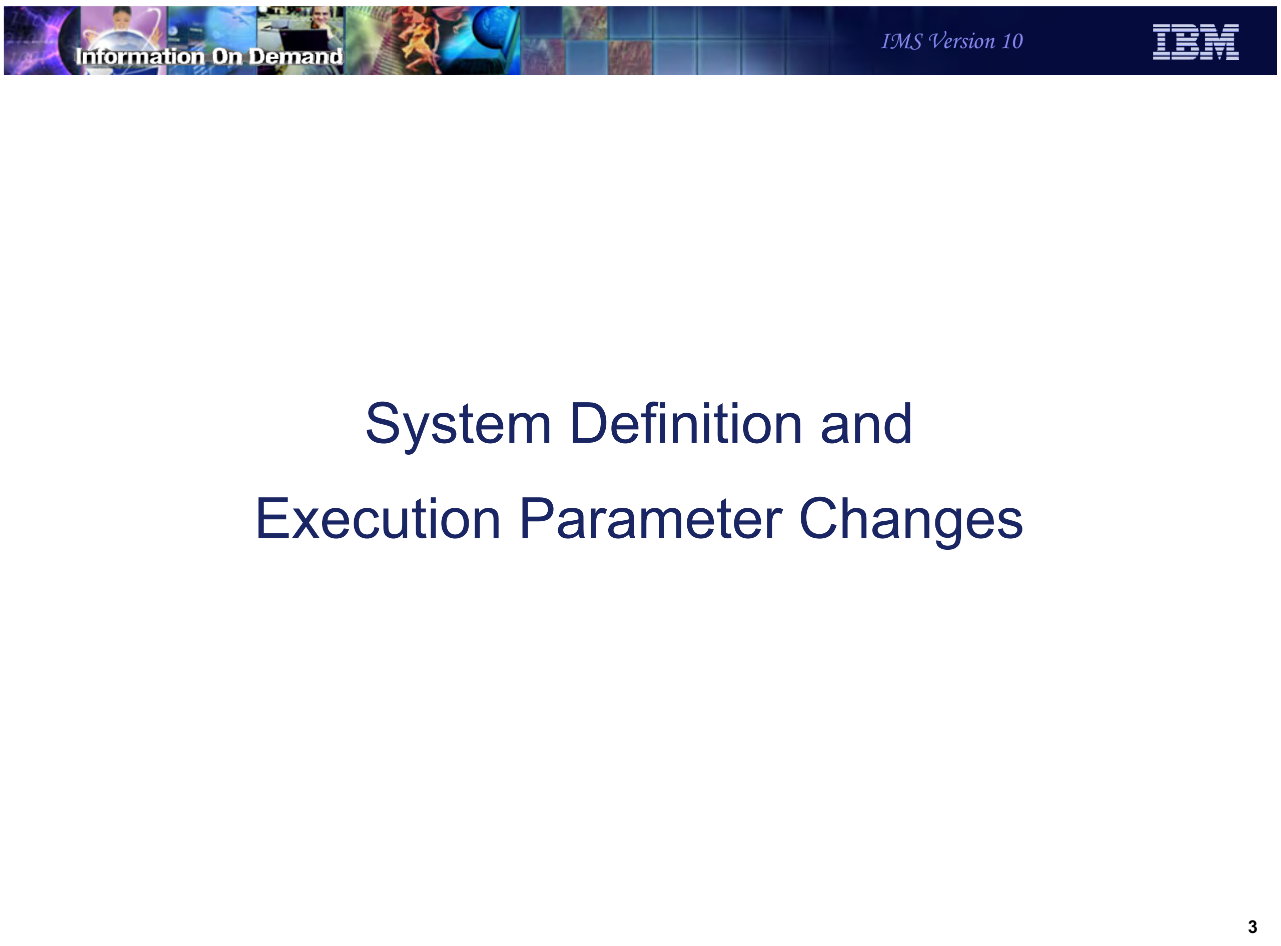
Unit 2. System Enhancements

System Enhancements

- System Definition and Execution Parameter Changes
- Enhanced Display of System Parameters
- Virtual Storage Constraint Relief
- Sysplex Serial Program Management
- Enhanced Log Record Statistics
- Large Sequential Data Set Support
- System Utilities Enhancements
- BPE External Trace
- Abend Search and Notification

Figure 2-1. System Enhancements CMA01.0

Notes:



Information On Demand

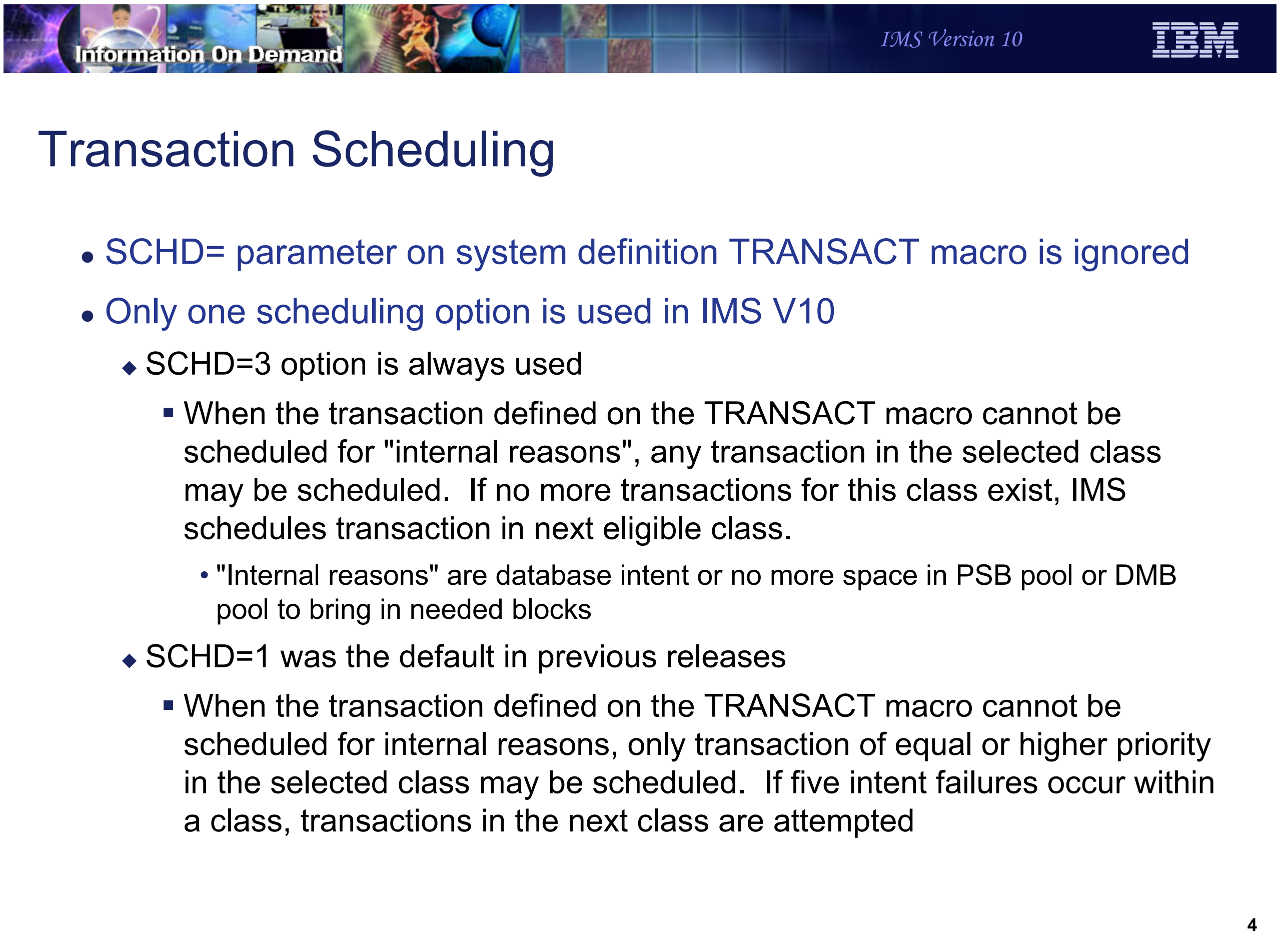
IMS Version 10

System Definition and Execution Parameter Changes

Figure 2-2. [No title] CMA01.0

Notes:

This section covers changes to IMS system definition and execution parameters that affect functions available in previous releases. Definition and execution changes for new functions are discussed where the new functions are explained.



Transaction Scheduling

- SCHD= parameter on system definition TRANSACT macro is ignored
- Only one scheduling option is used in IMS V10
  - SCHD=3 option is always used
    - When the transaction defined on the TRANSACT macro cannot be scheduled for "internal reasons", any transaction in the selected class may be scheduled. If no more transactions for this class exist, IMS schedules transaction in next eligible class.
      - "Internal reasons" are database intent or no more space in PSB pool or DMB pool to bring in needed blocks
  - SCHD=1 was the default in previous releases
    - When the transaction defined on the TRANSACT macro cannot be scheduled for internal reasons, only transaction of equal or higher priority in the selected class may be scheduled. If five intent failures occur within a class, transactions in the next class are attempted

Figure 2-3. Transaction Scheduling CMA01.0

Notes:

In previous release of IMS, the SCHD= parameter on the system definition TRANSACT macro specified the scheduling option used when the transaction defined on the TRANSACT macro could not be scheduled for internal reasons (database intent or no more space in PSB pool or DMB pool to bring in needed blocks). IMS V10 always uses the SCHD=3 option. The default in previous releases was SCHD=1.

SCHD=1 specified that only transactions of equal or higher priority in the selected class would be scheduled. Five consecutive intent conflicts are allowed within a class before IMS starts scheduling the next eligible class.

SCHD=3 specified that any transaction in the selected class could be scheduled. IMS starts scheduling the next eligible class after attempting to schedule all the transactions in the current class

Two other options were available in previous releases.

SCHD=2 specified that only higher-priority transactions in the selected class could be scheduled.

SCHD=4 specified that IMS should skip to the next class and attempt to schedule the highest-priority transaction in that class.

Scheduling failures for the "internal reasons" should be rare. Intent conflicts only occur when PROCOPTs with the E (exclusive) option are used. The PSB and DMB pools should always be large enough to hold the currently required PSBs and DMBs.

FPCTRL System Definition Macro Eliminated

- The system definition FPCTRL macro is ignored
    - Previous releases:
        - Macro was required to enable Fast Path capability
        - Set defaults for Fast Path parameters
- Execution parameters for IMS and DBC procedures
    - FP=Y|N used to enable Fast Path
        - Default is FP=N
    - DBBF=, DBFX=, BSIZ=, and OTHR= used to specify Fast Path parameters
- Benefit
    - Simplifies implementation of Fast Path
- Migration
    - Fast Path users must specify FP=Y and values for DBBF=, DBFX=, BSIZ=, and OTHR= for execution

Figure 2-4. FPCTRL System Definition Macro Eliminated CMA01.0

Notes:

IMS V10 has simplified the implementation of Fast Path. It does not have to specified at system definition time. The FPCTRL system definition macro is ignored. In previous releases it was required to enable Fast Path capabilities. It also was used to specify the default values for some Fast Path parameters. Fast Path capabilities are always generated for DB/DC and DBCTL systems. They are enabled by a parameter at execution time.

You must specify FP=Y at execution time for an online system to enable Fast Path capabilities in IMS V10. The default for the parameter is FP=N. Since defaults for Fast Path parameters cannot be specified at system definition time, they should be specified at execution time. These parameters are DBBF=, DBFX=, BSIZ=, and OTHR=. The default values for these parameters are the same as the default values for the FPCTRL macro in previous releases. These are:

DBBF=10, DBFX=4, BSIZ=2048, and OTHR=2

These values are highly unlikely to be appropriate for most Fast Path users.

When an FPCTRL macro is included in the system definition input, the following message is issued:

```
G1010 THE FPCTRL MACRO IS NO LONGER SUPPORTED.
      THE KEYWORD PARAMETERS WILL BE IGNORED.
      PLEASE USE THE EQUIVALENT EXECUTION
      PARAMETER TO SPECIFY VALUES THAT WERE
      CODED IN THIS MACRO.
```

DFSDFxxx PROCLIB Member

- DFSDFxxx is new IMS PROCLIB member
  - Defines IMS parameters for
    - Dynamic resource definition
    - Shared queues
    - CSL
    - Diagnostics and statistics
  - Eliminates need for DFSSQxxx and DFSCGxxx members
    - They are now optional
    - Values specified in DFSSQxxx and DFSCGxxx override DFSDFxxx values
- IMS execution parameter DFSDF=xxx specifies DFSDFxxx member

Figure 2-5. DFSDFxxx PROCLIB Member CMA01.0

Notes:

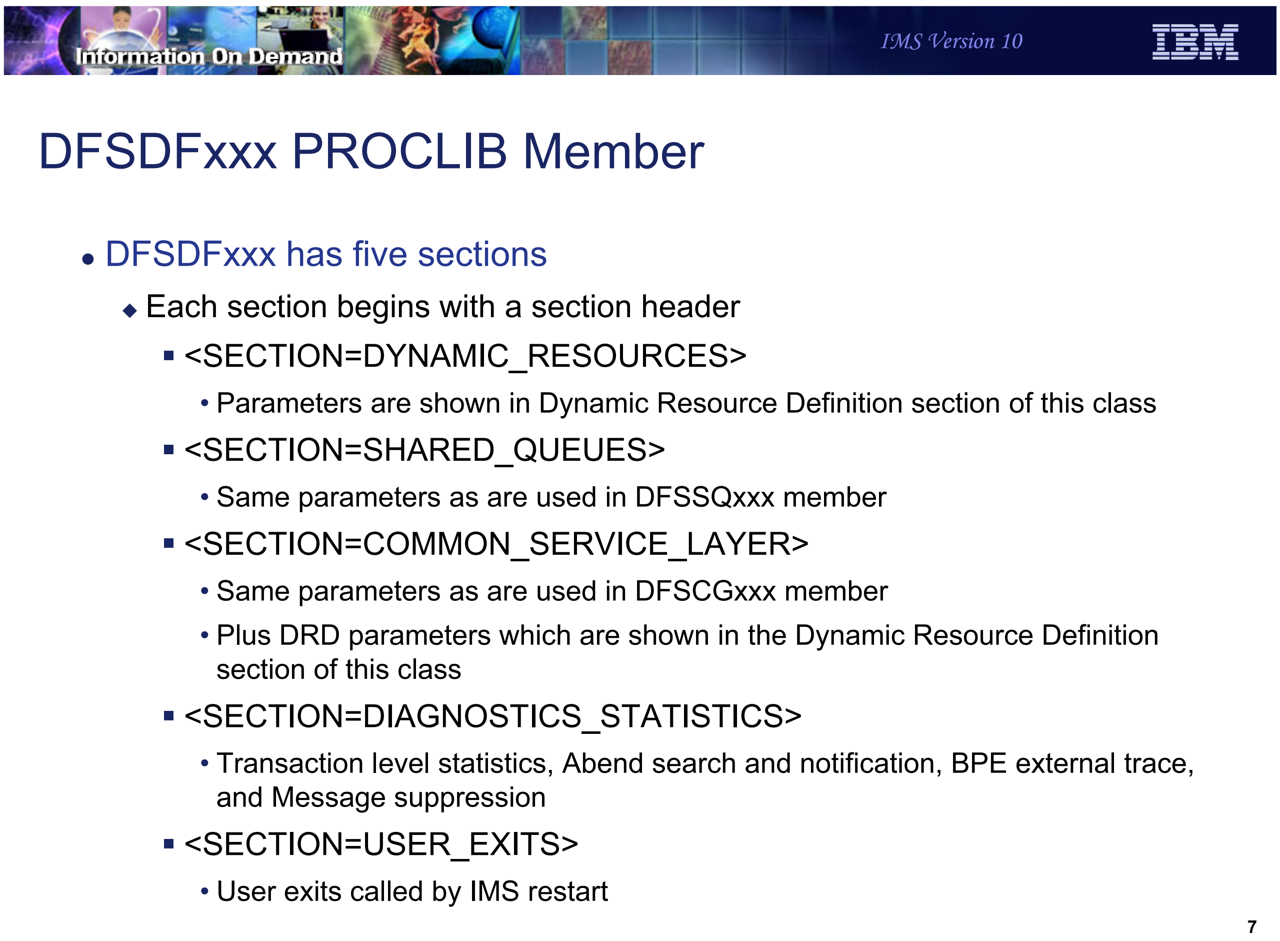
IMS V10 has a new PROCLIB member which is used to consolidate execution definitions which otherwise might be in several PROCLIB members. The DFSDFxxx member may be used to define parameters for several components of IMS.

DFSDFxxx eliminates the need to use a DFSSQxxx member for shared queues parameters or a DFSCGxxx member for CSL parameters. Instead, these parameters may be specified in the DFSDFxxx member. If a parameter is specified in both members, the values specified in DFSSQxxx and DFSCGxxx override those specified in DFSDFxxx.

DFSDFxxx is required to specify some IMS parameters. These are explained later.

The DFSDFxxx member used by IMS is specified in the DFSDF= execution parameter.

.


Information On Demand
IMS Version 10
IBM

DFSDFxxx PROCLIB Member

- DFSDFxxx has five sections
  - Each section begins with a section header
    - <SECTION=DYNAMIC_RESOURCES>
      - Parameters are shown in Dynamic Resource Definition section of this class
    - <SECTION=SHARED_QUEUES>
      - Same parameters as are used in DFSSQxxx member
    - <SECTION=COMMON_SERVICE_LAYER>
      - Same parameters as are used in DFSCGxxx member
      - Plus DRD parameters which are shown in the Dynamic Resource Definition section of this class
    - <SECTION=DIAGNOSTICS_STATISTICS>
      - Transaction level statistics, Abend search and notification, BPE external trace, and Message suppression
    - <SECTION=USER_EXITS>
      - User exits called by IMS restart

7

Figure 2-6. DFSDFxxx PROCLIB Member CMA01.0

Notes:

The DFSDFxxx member has up to five sections of definitions. Each section begins with a section identification statement. These statements are shown on the slide.

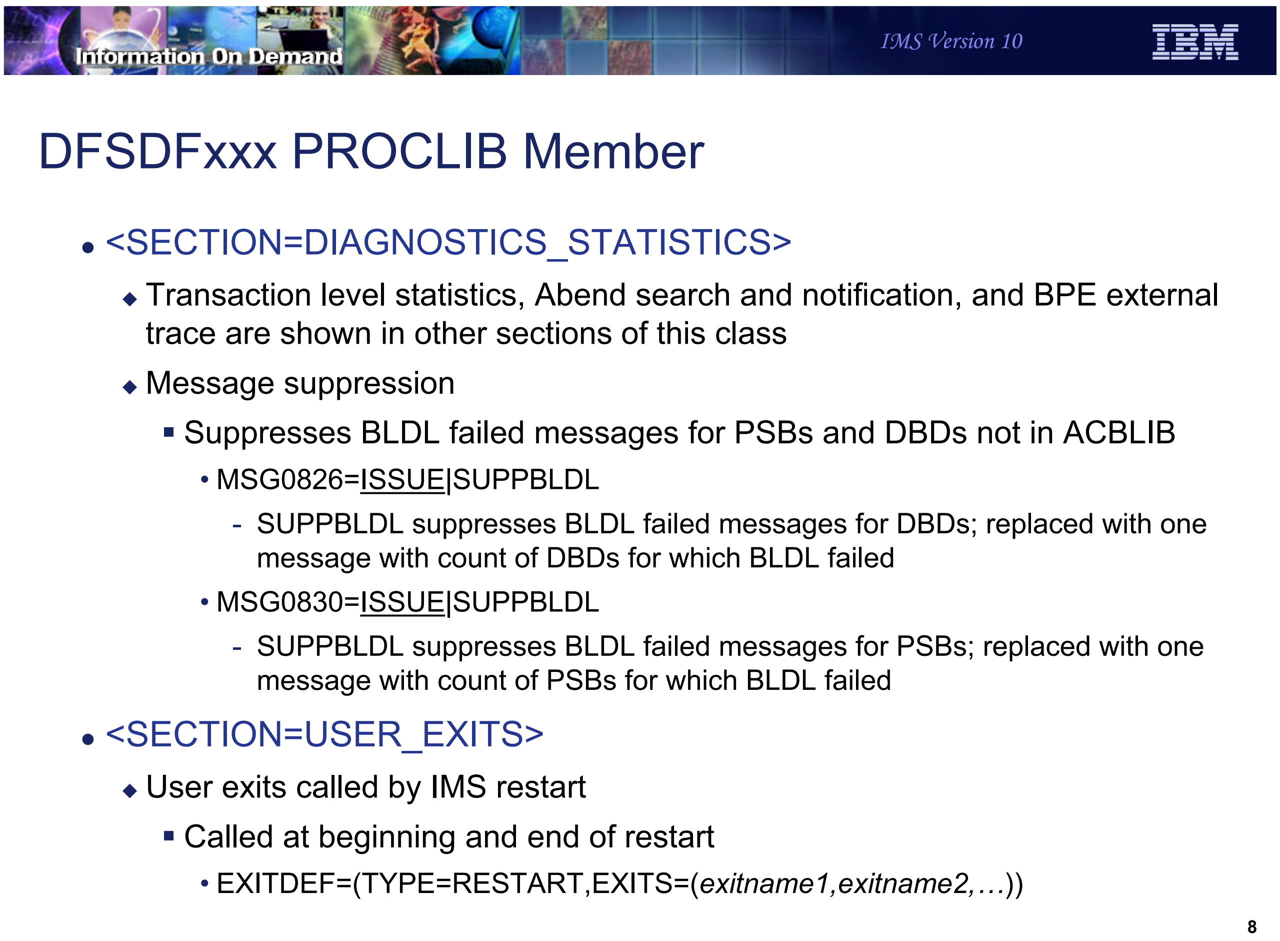
Dynamic resource definition parameters are defined only in DFSDFxxx. They are in the DYNAMIC_RESOURCES and COMMON-SERVICE_LAYER sections.

Shared queues parameters may be defined in the SHARED_QUEUES section.

CSL parameters may be defined in the COMMON_SERVICE_LAYER section.

Some diagnostic and statistics parameters are defined in DFSDFxxx. These are defined in the DIAGNOSTICS_STATISTICS section.

IMS Version 10 includes an IMS restart user exit capability. These exits are specified in the USER_EXITS section. They are called at the beginning and end of IMS restarts. Restart exits are called during all types of IMS restart. An exit routine is passed a function code and a code that indicates the type of restart that is being done. Multiple routines can be defined. The routines are called in the order that they are listed in the EXITDEF parameter. If an exit is defined multiple times, it is called multiple times. The form of the specification is: EXITDEF=(TYPE=RESTART,EXITS=(*exitname*))



Information On Demand
IMS Version 10
IBM

DFSDFxxx PROCLIB Member

- <SECTION=DIAGNOSTICS_STATISTICS>
  - Transaction level statistics, Abend search and notification, and BPE external trace are shown in other sections of this class
  - Message suppression
    - Suppresses BLDL failed messages for PSBs and DBDs not in ACBLIB
      - MSG0826=ISSUE|SUPPBLDL
        - SUPPBLDL suppresses BLDL failed messages for DBDs; replaced with one message with count of DBDs for which BLDL failed
      - MSG0830=ISSUE|SUPPBLDL
        - SUPPBLDL suppresses BLDL failed messages for PSBs; replaced with one message with count of PSBs for which BLDL failed
- <SECTION=USER_EXITS>
  - User exits called by IMS restart
    - Called at beginning and end of restart
      - EXITDEF=(TYPE=RESTART,EXITS=(*exitname1,exitname2,…*))

Figure 2-7. DFSDFxxx PROCLIB Member CMA01.0

Notes:

The DIAGNOSTICS_STATISTICS section includes a capability to suppress some DFS826I and DFS830I messages. There messages are issued by IMS restart to indicate that DBDs and PSBs which are defined in the system do not have members in ACBLIB. These messages are suppressed with the following statements:

MSG0826=SUPPBLDL and MSG0830=SUPPBLDL.

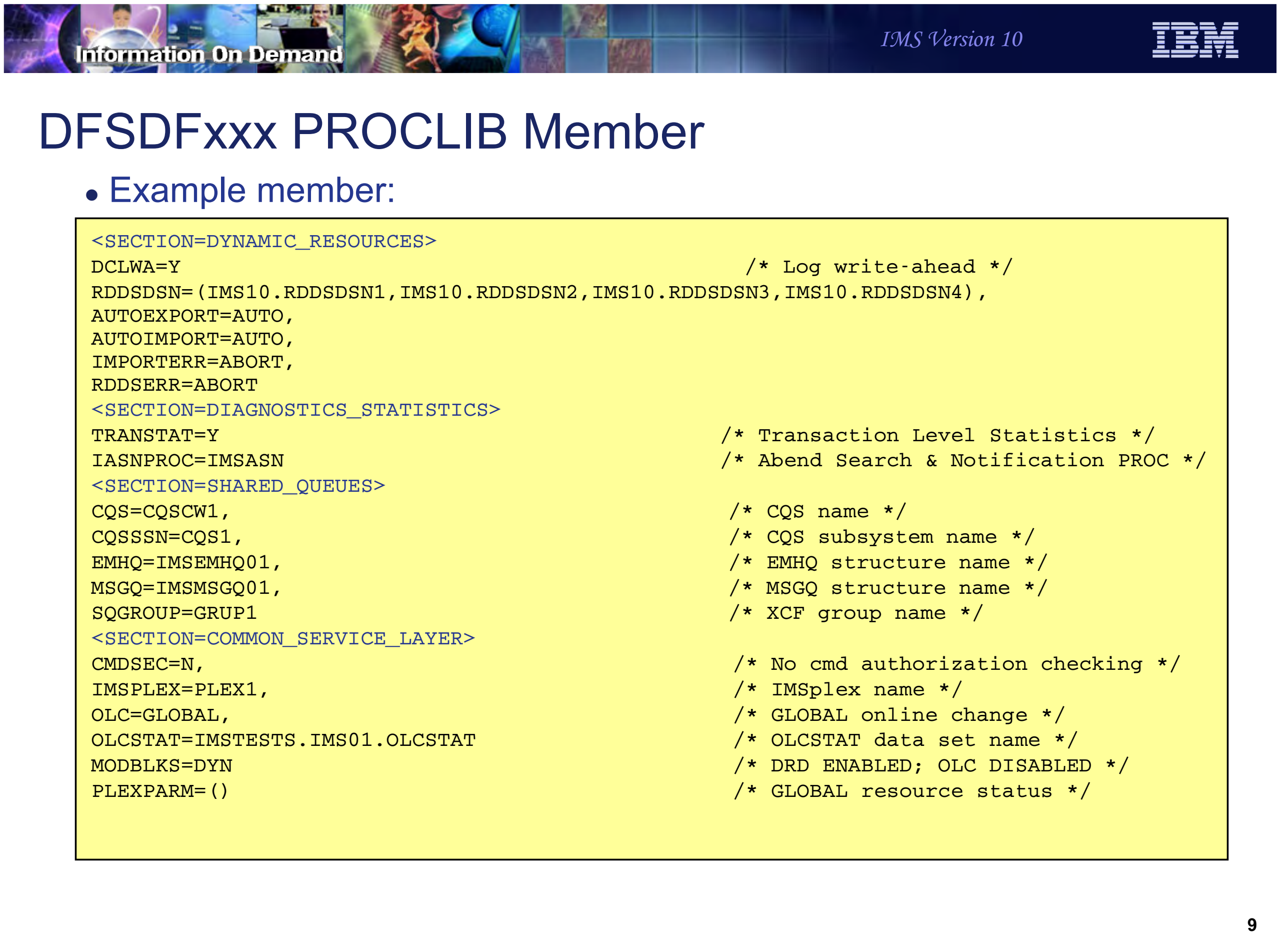
If the DBD messages are not suppressed, there is one message for each DBD for which the BLDL failed. Its form is: DFS826I BLDL FAILED FOR FOLLOWING DBD This is followed by a message which includes a count of the number of DBD BLDL failures. Its form is: DFS826I xxx DBD ERRORS SENT TO JOB LOG

If the DBD messages are suppressed, the following message is issued: DFS826I *xxx* DBD ERRORS SUPPRESSED This includes the count of the messages suppressed.

If the PSB messages are not suppressed, there is one message for each PSB for which the BLDL failed. Its form is: DFS830I BLDL FAILED FOR FOLLOWING PSB This is followed by a message which includes a count of the number of DBD BLDL failures. Its form is: DFS830I xxx PSB ERRORS SENT TO JOB LOG

If the PSB messages are suppressed, the following message is issued: DFS830I *xxx* PSB ERRORS SUPPRESSED This includes the count of the messages suppressed.

IMS Version 10 includes an IMS restart user exit capability. These exits are specified in the USER_EXITS section. They are called at the beginning and end of IMS restarts. Restart exits are called during all types of IMS restart. An exit routine is passed a function code and a code that indicates the type of restart that is being done. Multiple routines can be defined. The routines are called in the order that they are listed in the EXITDEF parameter. If an exit is defined multiple times, it is called multiple times. The form of the specification is: EXITDEF=(TYPE=RESTART,EXITS=(*exitname1.exitname2,…*))



DFSDFxxx PROCLIB Member

- Example member:

```
<SECTION=DYNAMIC_RESOURCES>
DCLWA=Y                                                   /* Log write-ahead */
RDDSDSN=(IMS10.RDDSDSN1,IMS10.RDDSDSN2,IMS10.RDDSDSN3,IMS10.RDDSDSN4),
AUTOEXPORT=AUTO,
AUTOIMPORT=AUTO,
IMPORTERR=ABORT,
RDDSERR=ABORT
<SECTION=DIAGNOSTICS_STATISTICS>
TRANSTAT=Y                                              /* Transaction Level Statistics */
IASNPROC=IMSASN                                         /* Abend Search & Notification PROC */
<SECTION=SHARED_QUEUES>
CQS=CQSCW1,                                              /* CQS name */
CQSSSN=CQS1,                                             /* CQS subsystem name */
EMHQ=IMSEMHQ01,                                          /* EMHQ structure name */
MSGQ=IMSMSGQ01,                                          /* MSGQ structure name */
SQGROUP=GRUP1                                            /* XCF group name */
<SECTION=COMMON_SERVICE_LAYER>
CMDSEC=N,                                                 /* No cmd authorization checking */
IMSPLEX=PLEX1,                                            /* IMSplex name */
OLC=GLOBAL,                                               /* GLOBAL online change */
OLCSTAT=IMSTESTS.IMS01.OLCSTAT                            /* OLCSTAT data set name */
MODBLKS=DYN                                               /* DRD ENABLED; OLC DISABLED */
PLEXPARM=()                                               /* GLOBAL resource status */
```

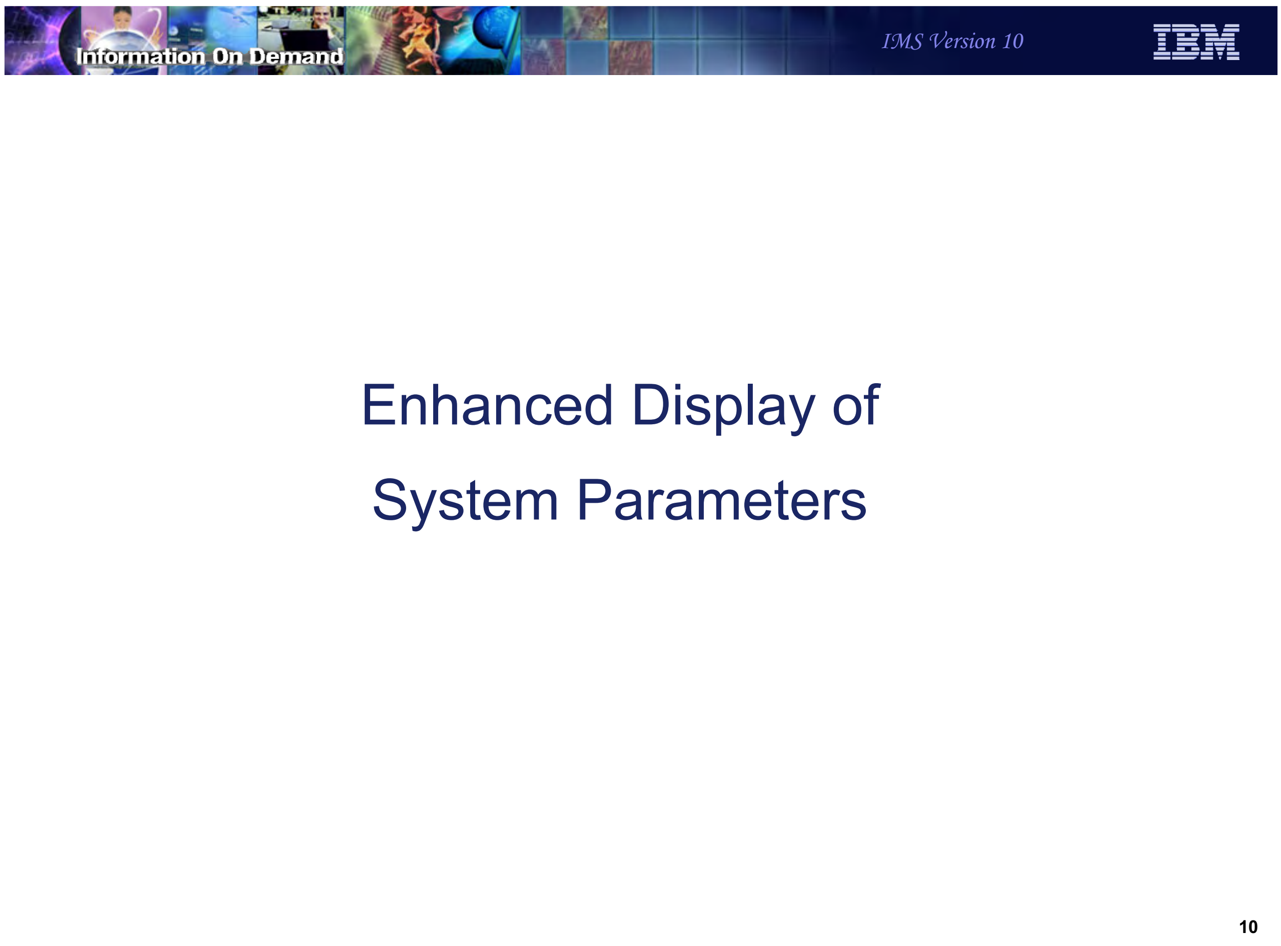
9

Figure 2-8. DFSDFxxx PROCLIB Member CMA01.0

Notes:

This is an example of a DFSDFxxx member. It includes parameters for DRD, traces, transaction level statistics, abend search and notification, shared queues, and CSL.

The meanings of the parameters other that those for shared queues and CSL are explained elsewhere in this class. Shared queues and CSL parameters have the same meaning as they had in the DFSSQxxx and DFSCGxxx members for previous releases.


Information On Demand
IMS Version 10

Enhanced Display of System Parameters

Figure 2-9. [No title] CMA01.0

Notes:

Enhanced Display of System Parameters

- DFS1929I message is written after reading the log
  - Displays message twice
    - First message is compatible with prior releases
      - Displays each system parameter specified in system definition, PROCLIB member, or overridden at execution time
        - These are values after initialization, but before restart
    - Second message is written after restart has read the log
      - Displays actual system parameters that are in effect
  - Message header includes IMS version number and control region type
    - Body of message is compatible with previous releases

```
DFS1929I * IMS SYSTEM PARAMETERS ACTIVE FOR THIS V10.1 DBDC EXECUTION:
DFS1929I *    ALOT            = 1440                        *SYS3
DFS1929I *    AOIP            = 2147483647                  *SYS3
DFS1929I *    AOIS            = N                           *SYS3
…
```

Figure 2-10. Enhanced Display of System Parameters CMA01.0

Notes:

IMS issues message DFS1929I to list the IMS system parameters. Since the message is issued early in the initialization process, the values displayed reflect those provided during system generation or overridden through startup parameters. Every parameter is shown. If a value has not been provided by the user then the default, if one exists, is displayed. The values shown, however, might not reflect the actual values that are used during online execution. The discrepancy between the actual and requested values may be minor, but in some situations it can be large. IMS V10 continues to issue the DFS1929I message early in initialization, for compatibility, but issues it once again after restart is complete and the log has been read so that the values displayed reflect the actual values in effect for the IMS execution.

The DFS1929I message has been enhanced to include the IMS Control Region type along with the IMS Version number. The remainder of the message is compatible with the format of previous releases. The first form of the message that is printed during IMS initialization contains all the IMS parameters with user-specified or default values.

The second DFS1929I message which is produced after the log has been read has been enhanced to list:

- Only the parameters that are applicable to the control region type.
- Only the parameters associated with active components. For example, Fast Path parameters will be shown only if activated. Prior releases always displayed the parameters regardless of whether or not Fast Path was enabled. The new IMS V10 parameter FP=Y | N will always be displayed.
- Actual values that will be used during execution. For example, the CPLOG value displayed in the first DFS1929I message during initialization might not be valid after a warm or emergency restart because its value was changed by a command. As a result, the value in the first DFS1929I message does not help in diagnosing problems. The problem encompasses other parameters such as APPC, APPCSE, OTMA, OTMASE, etc., which can also be changed by command and are recoverable during an IMS restart. The actual values, therefore, are provided in the second DFS1929I message after restart is complete.

Enhanced Display of System Parameters

- Migration
  - DFS1929I message is issued twice
    - Previously issued only once
    - New second message has more accurate information
  - Header has changed to include version and control region type

- Benefits
  - Assists in determining the actual parameters values for optimal configurations and optimal user response times
  - Provides an accurate audit list of execution parameters

Figure 2-11. Enhanced Display of System Parameters CMA01.0

Notes:

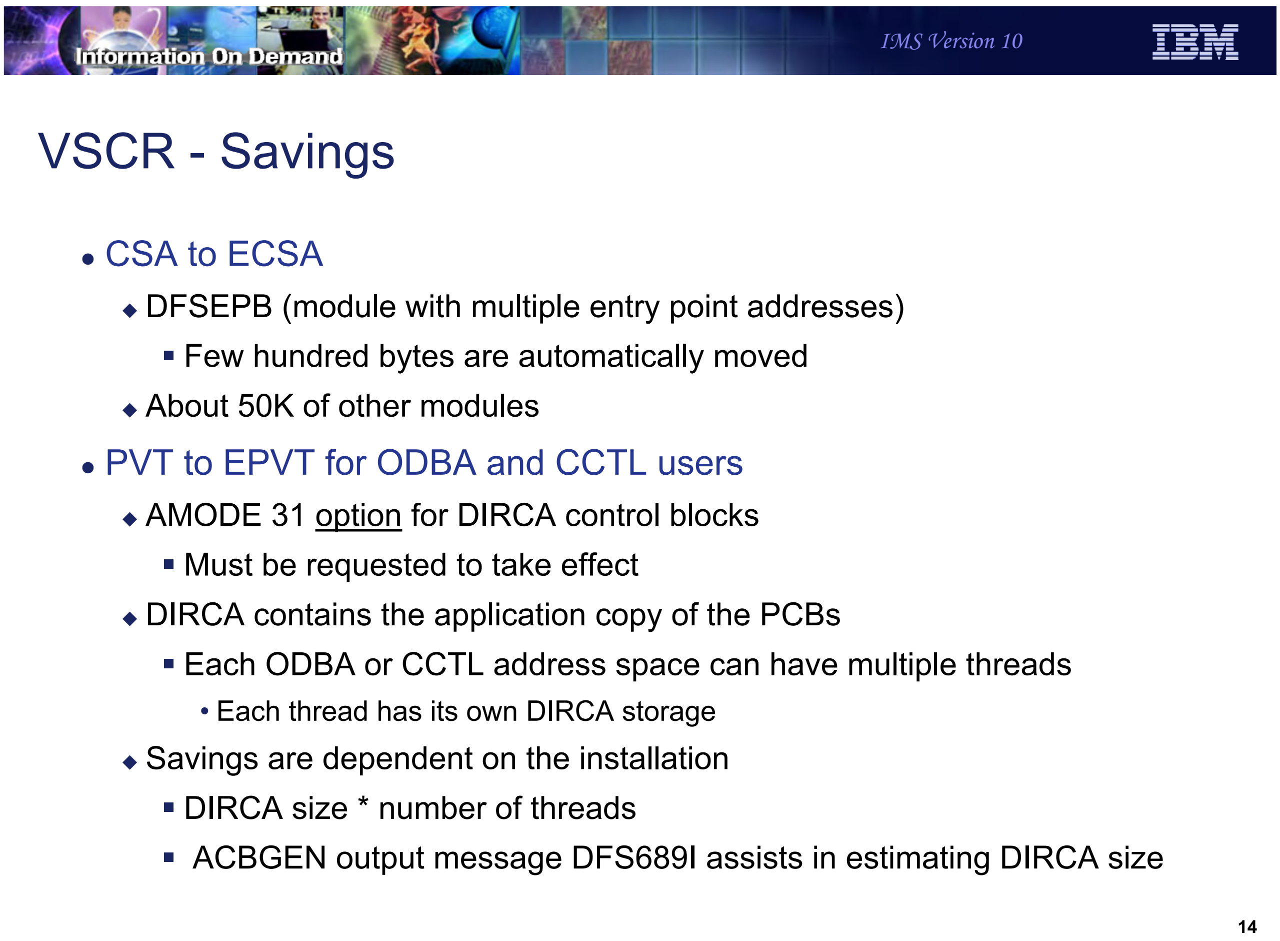
When migrating to IMS V10, user-written routines or procedures that look for the DFS1929I message should recognize the header format change to include the IMS version and Control Region type. Additionally, these routines should be modified to recognize that there are two versions of the message.

Information On Demand
IMS Version 10
IBM

Virtual Storage Constraint Relief

Figure 2-12. [No title] CMA01.0

Notes:


Information On Demand
IMS Version 10
IBM

VSCR - Savings

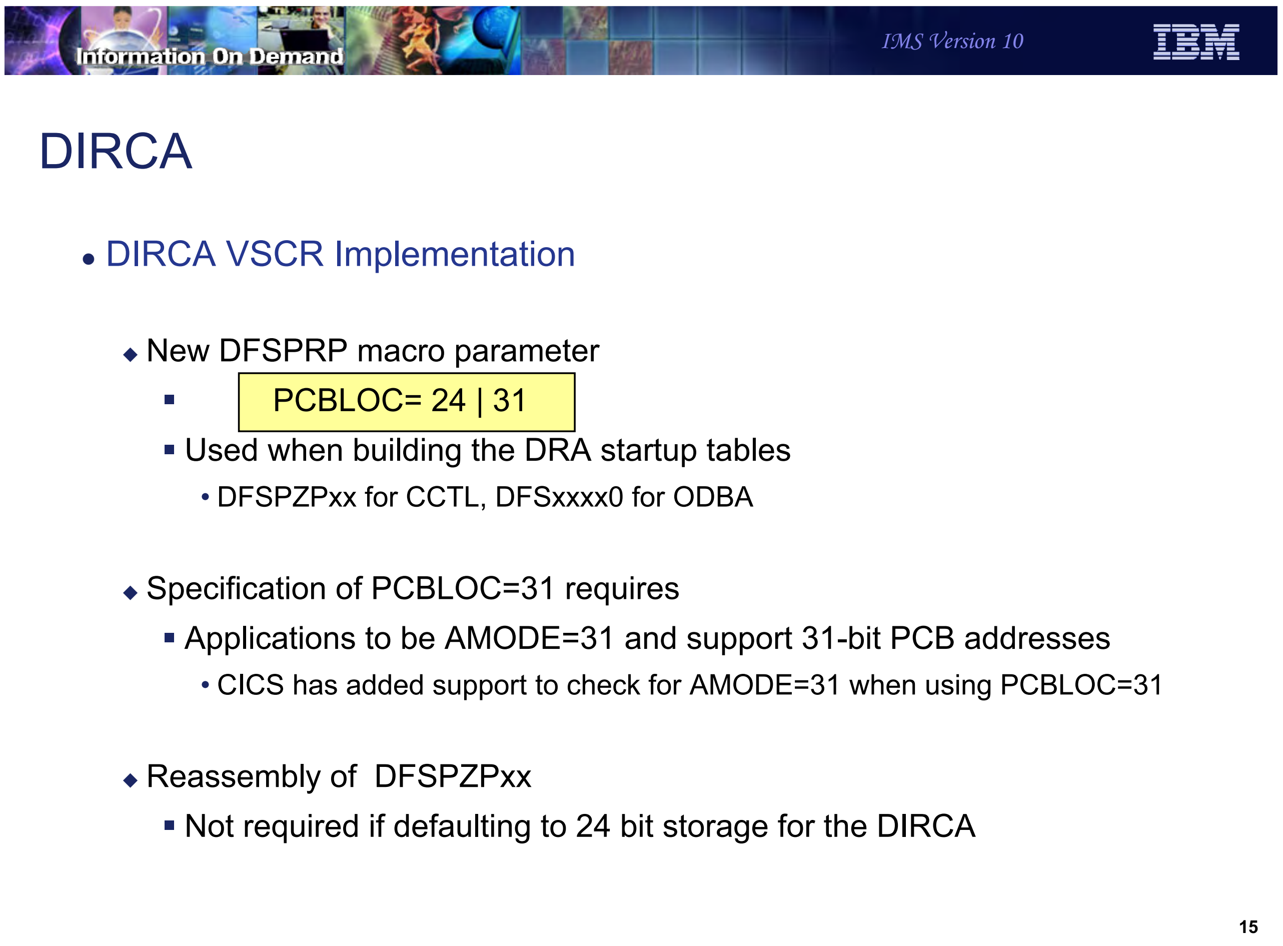
- CSA to ECSA
  - DFSEPB (module with multiple entry point addresses)
    - Few hundred bytes are automatically moved
  - About 50K of other modules
- PVT to EPVT for ODBA and CCTL users
  - AMODE 31 option for DIRCA control blocks
    - Must be requested to take effect
  - DIRCA contains the application copy of the PCBs
    - Each ODBA or CCTL address space can have multiple threads
      - Each thread has its own DIRCA storage
  - Savings are dependent on the installation
    - DIRCA size * number of threads
    - ACBGEN output message DFS689I assists in estimating DIRCA size

Figure 2-13. VSCR - Savings CMA01.0

Notes:

The DFSEPB (Entry Point Block) was introduced in IMS V9 to keep track of entry points of CSECTs of large composite modules that needed to be divided into smaller modules. The movement of this storage from CSA to ECSA saves a few hundred bytes in IMS V10. In addition, a few other modules have been moved from CSA to ECSA. These modules total about 50K bytes.

A more substantial impact is that of potential DIRCA (Dependent Region Interregion Communication Area) storage movement above the line for ODBA and CCTL address spaces. The savings are installation dependent and must be requested for the environment to take advantage of the capability. Each thread from the ODBA/CCTL address space has its own DIRCA and each DIRCA contains the application's copy of the PCBs. This enhancement provides a benefit for those environments, such as CICS, that need to use larger PSBs and more concurrent threads without impacting storage usage below the 16MB line.


Information On Demand
IMS Version 10
IBM

DIRCA

- DIRCA VSCR Implementation
  - New DFSPRP macro parameter
    - PCBLOC= 24 | 31
    - Used when building the DRA startup tables
      - DFSPZPxx for CCTL, DFSxxxx0 for ODBA
  - Specification of PCBLOC=31 requires
    - Applications to be AMODE=31 and support 31-bit PCB addresses
      - CICS has added support to check for AMODE=31 when using PCBLOC=31
  - Reassembly of DFSPZPxx
    - Not required if defaulting to 24 bit storage for the DIRCA

15

Figure 2-14. DIRCA CMA01.0

Notes:

The DFSPRP macro provides the definition for the DRA parameters when building the DRA startup table. A new parameter PCBLOC provides the option of defining whether the DIRCA is to be built using 24-bit or 31-bit storage. Note that a specification of 31 also requires that the applications actually support AMODE 31 and 31-bit PCB addresses.

CICS has added support so that it checks for the use of AMODE=31 by applications when PCBLOC=31 is specified. The application will abend with a new CICS abend code ADCF during schedule of the PSB if it is AMODE=24 and PCBLOC=31 is in effect. This support is added by APARS PK54099 for CICS TS 2.2 and CICS TS 2.3, and PK54100 for CICS TS 3.1 and CICS TS 3.2.

When a CCTL connector starts up, the DRA INIT call can be used to override the PCBLOC specification in the DRA startup table. A new parameter, PPLLPSO, allows specification of 31 or 24. Other values are ignored and the startup defaults to the value specified in the startup table.

Information On Demand

IMS Version 10

IBM

Benefits

- DFSEPB
  - Moves storage from CSA without user action
- DIRCA
  - Supports requirement for larger PSBs and greater number of concurrent threads from CCTL/ODBA applications
  - Opportunity to move the line between CSA and Private for the LPAR
    - Applications (instances of CICS, ODBA, etc.) that are causing the constraint will need to request the 31-bit DIRCA

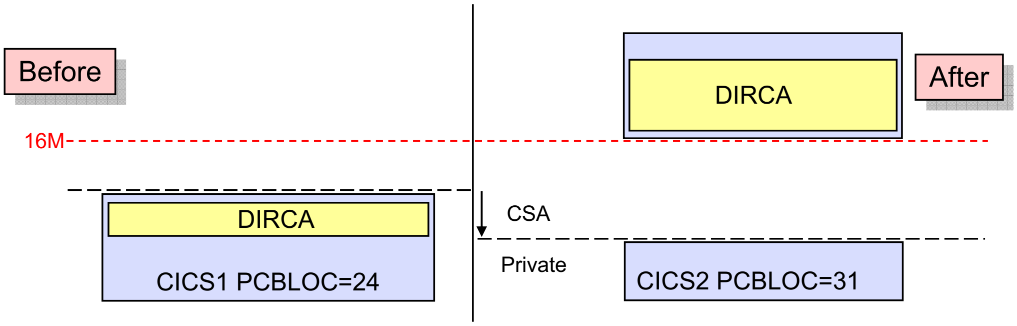
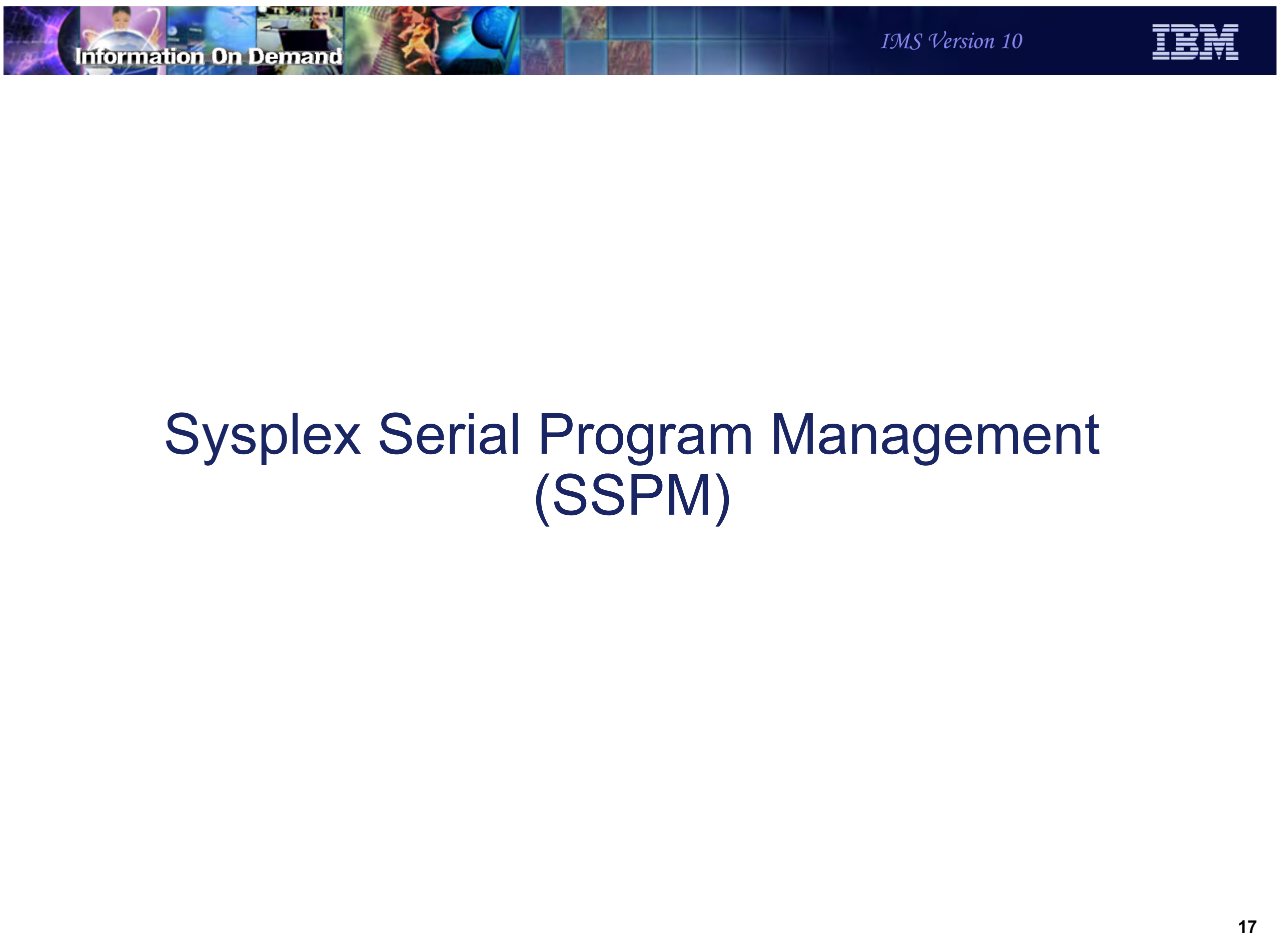


Figure 2-15. Benefits CMA01.0

Notes:

The DFSEPB module is automatically moved above the line from CSA to ECSA.

The DIRCA is moved above the 16MB line from private to extended private. This provides two benefits. First, PSBs may be larger. Second, the CICS or ODBA address space private area does not have to be as large, so the line between private and CSA may be lower. This provides more room in CSA.

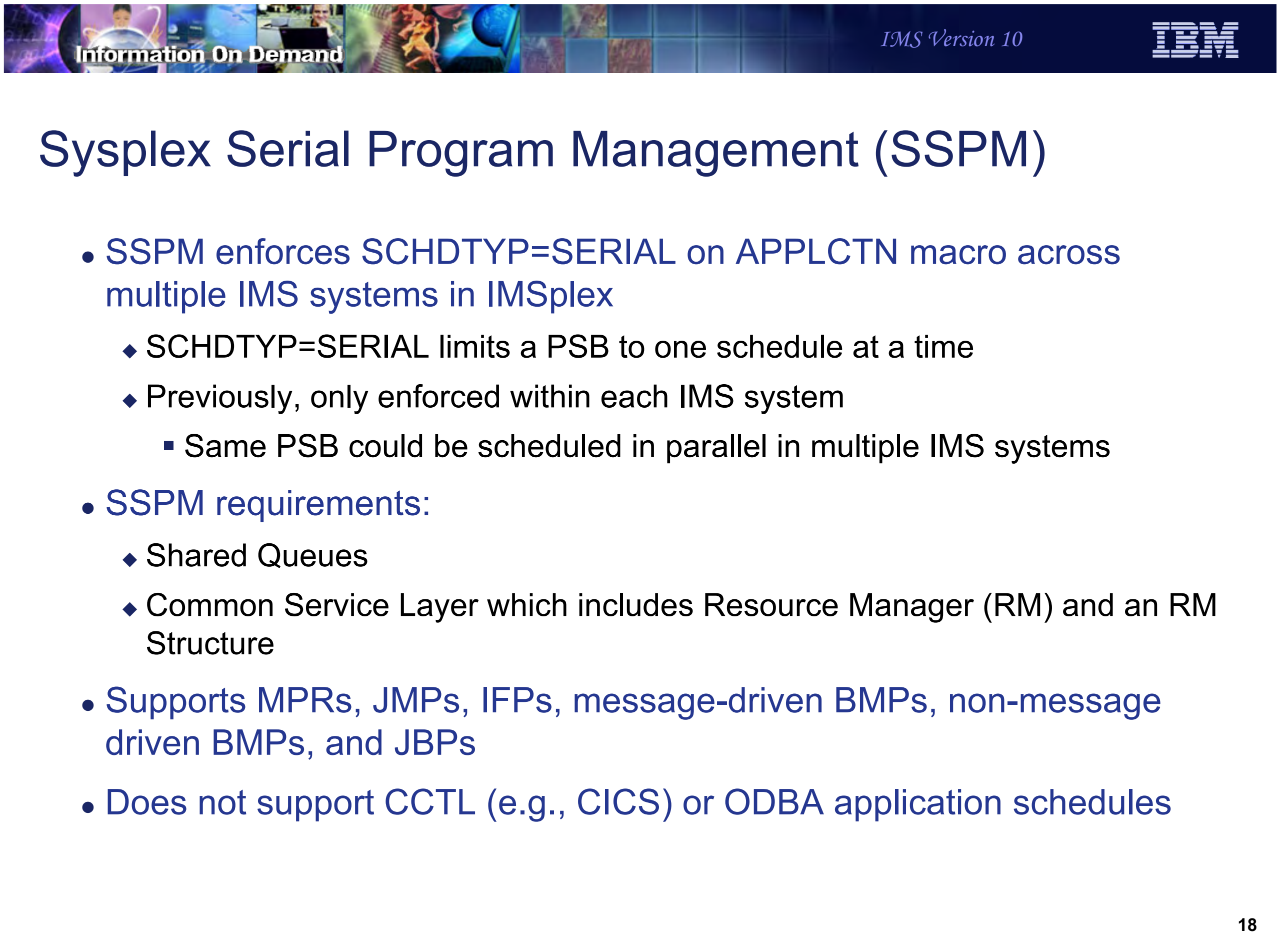


Information On Demand
IMS Version 10

Sysplex Serial Program Management (SSPM)

Figure 2-16. Sysplex Serial Program Management(SSPM) CMA01.0

Notes:


Information On Demand
IMS Version 10
IBM

Sysplex Serial Program Management (SSPM)

- SSPM enforces SCHDTYP=SERIAL on APPLCTN macro across multiple IMS systems in IMSplex
  - SCHDTYP=SERIAL limits a PSB to one schedule at a time
  - Previously, only enforced within each IMS system
    - Same PSB could be scheduled in parallel in multiple IMS systems
- SSPM requirements:
  - Shared Queues
  - Common Service Layer which includes Resource Manager (RM) and an RM Structure
- Supports MPRs, JMPs, IFPs, message-driven BMPs, non-message driven BMPs, and JBPs
- Does not support CCTL (e.g., CICS) or ODBA application schedules

Figure 2-17. Sysplex Serial Program Management (SSPM) CMA01.0

Notes:

IMS Version 10 has enhanced the support for serialization of programs that are defined as SCHDTYP=SERIAL to be across all the members of an IMSplex, rather than just within a single IMS when using Shared Queues. This serialization is only done in systems using shared queues and a Resource Manager with a Resource Manager structure.

IMS V10 will ensure that SCHDTYP=SERIAL programs will be scheduled in only one IMS dependent region (MPR, JMP, IFP, message-driven BMP, non-message driven BMP, JBP) across an entire IMSplex with Shared Queues at a time.

CICS and ODBA are not supported for SSPM. SCHDTYP=SERIAL may be specified for an APPLCTN used by CICS or ODBA. As in previous releases, this will serialize the scheduling of the PSB by CICS, ODBA, and IMS TM within an IMS system. On the other hand, the serialization across multiple IMS systems will only be enforced for IMS TM regions including MPPs, IFPs, JMPs, BMPs, and JBPs.

SCHDTYP=SERIAL is the default if not specified on the APPLCTN macro.

IMS will use the IMS Resource Manager (RM) component of the IMS Common Services Layer (CSL) to provide this serialization.

SCHDTYP=SERIAL is specified on the APPLCTN macro or on the CREATE PGM SCHDTYPE(SERIAL) command when using DRD. This should not be confused with the SERIAL=YES parameter on the TRANSACT macro. SCHDTYP=SERIAL for APPLCTN limits the scheduling of a PSB. SERIAL=YES for TRANSACT limits the scheduling of a transaction. The meaning and enforcement of SERIAL=YES on the TRANSACT macro is not changed in IMS Version 10.

Information On Demand IMS Version 10 IBM

Implementation via IMS Resource Manager (RM)

- When a program defined as SCHDTYP=SERIAL is chosen for scheduling
  - IMS scheduler attempts to create a PROGRAM record in the RM structure
    - If PROGRAM does not exist in structure
      - PROGRAM record created
      - Scheduling continues
      - Program record deleted from RM structure when program/PSB is unscheduled
    - If PROGRAM exists in structure
      - Scheduling fails
      - MPRs and JMPs available to schedule the next eligible program
        - Message remains on shared queues, picked up by currently scheduled program
      - BMPs, IFPs, and JBPs abend with U457 as in previous versions

19

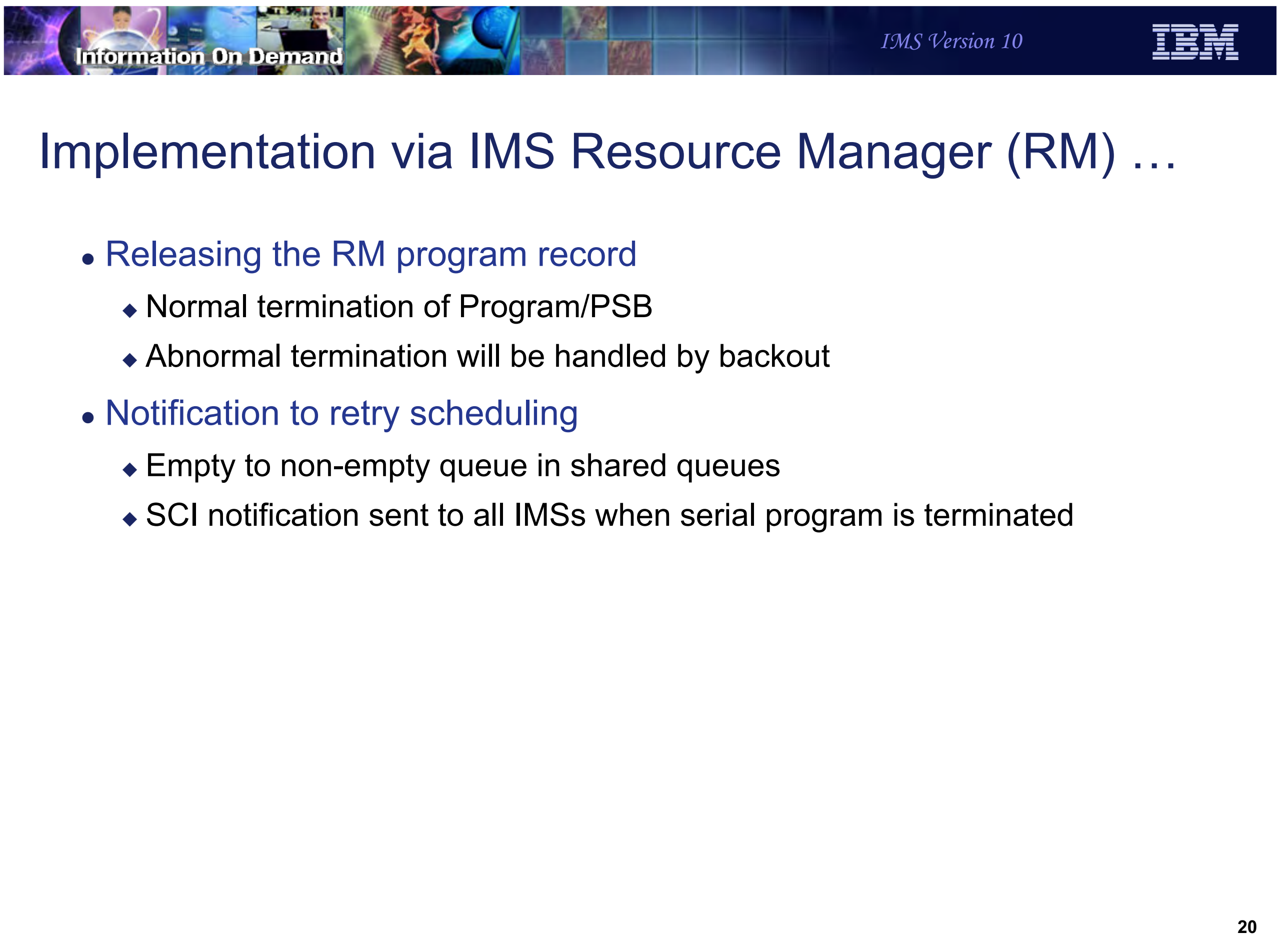
Figure 2-18. Implementation via IMS Resource Manager (RM) CMA01.0

Notes:

IMS Version 10 will determine if a SCHDTYP=SERIAL program is already scheduled by interacting with the IMS Resource Manager and its resource structure. IMS uses RM to attempt to create the program resource in the resource structure.

If the program resource does not exist in the resource structure, that means no other copy of this program is scheduled at this time. Therefore, this scheduling can continue. A program record for this program will be created in the resource structure. For MPRs, JMPs, and IFPs, the program resource remains active in the resource structure until the PSB is unscheduled. For BMPs and JBPs, the program resource will be deleted when the batch message processing program terminates.

If the serial program is already scheduled on another IMS, the program resource already exists in RM and the attempt to create the program resource in RM fails. Therefore, this attempt to schedule what would be the second copy of the serial program will fail. For MPRs and JMPs, the dependent region will be available to schedule the next eligible program. The message that got the scheduling failure will remain on the shared queues (was never taken off) and will hopefully be picked up by the currently scheduled program. BMPs, JMPs, and IFPs will get a U457 abend when scheduling finds the program is already scheduled as they would have before in previous versions within a single IMS versus in IMS V10 this abend would be on another IMS in the IMSplex.



Information On Demand
IMS Version 10
IBM

Implementation via IMS Resource Manager (RM) …

- Releasing the RM program record
  - Normal termination of Program/PSB
  - Abnormal termination will be handled by backout
- Notification to retry scheduling
  - Empty to non-empty queue in shared queues
  - SCI notification sent to all IMSs when serial program is terminated

Figure 2-19. Implementation via IMS Resource Manager (RM) … CMA01.0

Notes:

The program record in the RM structure is deleted when the program terminates. It the program ABENDs, backout will delete the program record.

After a scheduling attempt fails due to this serialization, the IMS system will not attempt to schedule the PSB again until it receives a notification. This can occur in two ways. It may occur due to the queue going from an empty to a non-empty state. It may occur due to an SCI notification which is sent when the serial program is terminated.

Sysplex Serial Program Management Summary

- Benefits
  - Guarantees that only one copy of an IMS TM program will be scheduled at any point in time across the entire IMSplex with shared queues
    - Enforced for V10 only
  - Removes requirement for specialized customer procedures to handle serialized programs in an IMSplex with Shared Queues
    - Still needed for V8 and V9
  - No program changes/definition changes

Figure 2-20. Sysplex Serial Program Management Summary CMA01.0

Notes:

Information On Demand
IMS Version 10
IBM

Enhanced Log Record Statistics

Figure 2-21. [No title] CMA01.0

Notes:

Information On Demand
IMS Version 10
IBM

Enhanced Log Record Statistics

- New and enhanced log record statistics
  - Transaction level statistics in optional 56FA log record
    - Written at each commit
      - Written for each input message with MODE=MULT
    - Includes all statistics from 07 log record of previous releases
    - Also includes additional statistics
  - Additional information in existing 07, 08 and 0A07 log records

Figure 2-22. Enhanced Log Record Statistics CMA01.0

Notes:

Enhanced statistics are written in several IMS log records. This includes a new 56FA log record which records statistics for individual transactions, not just for executions of programs or schedulings of PSBs. When chosen it is written for each commit for transactions defined as MODE=SNGL and for non-message driven BMPs and JBPs. For transactions defined as MODE=MULT the log record is written for each message. This record includes all of the statistics which were written to the 07 log records in previous releases. This includes CPU time and call counts. In addition, there are also new statistics which are shown on the next page.

Additional information has been added to the application start accounting log record (08), the application termination accounting log record (07), and the CPI-C driven transaction termination log record (0A07).

Information On Demand
IMS Version 10
IBM

Enhanced Log Record Statistics

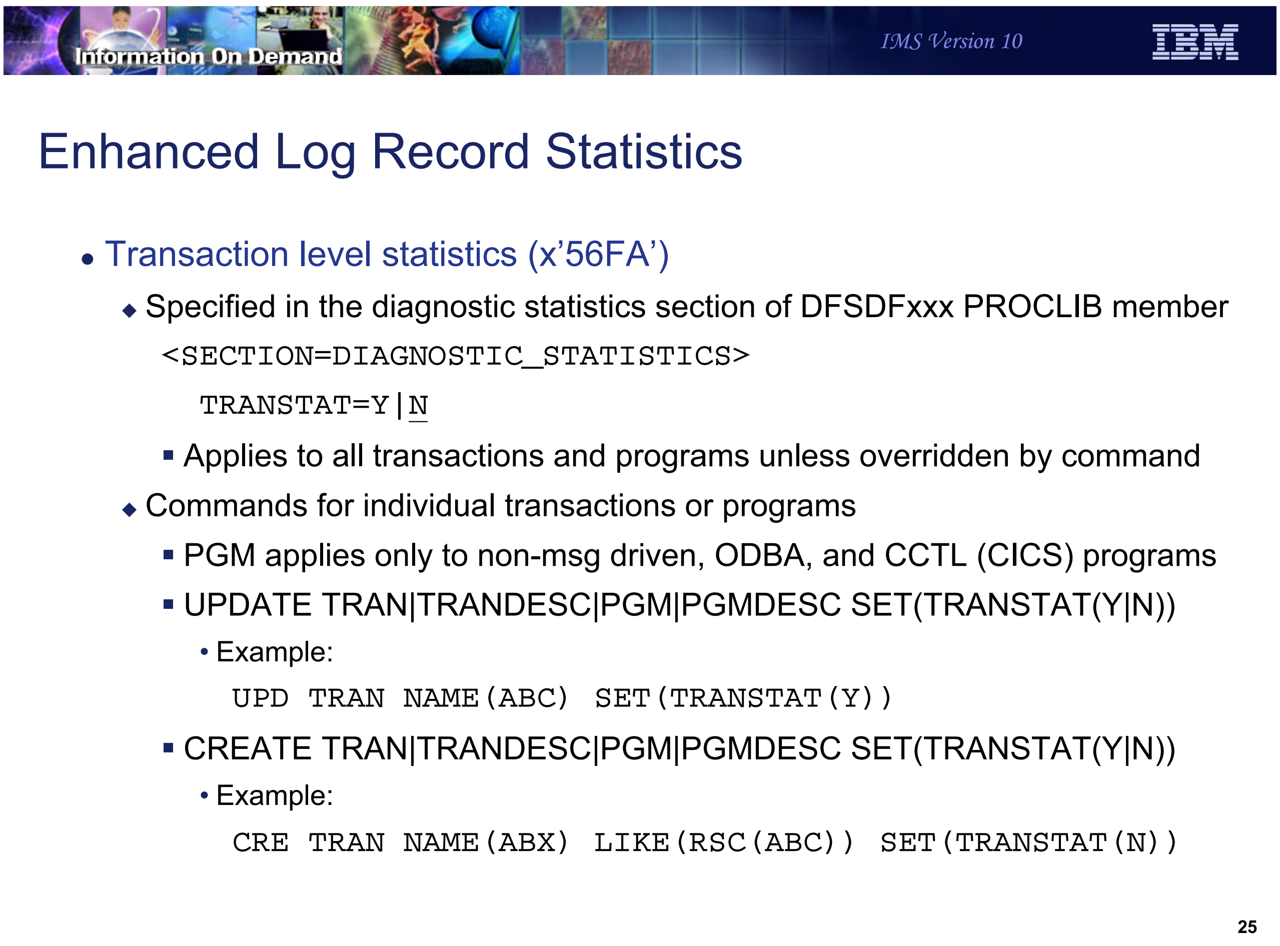
- Transaction level statistics (x’56FA’)
  - Fields in addition to those in the x’07’ log record
    - VSAM I/O reads and writes
    - OSAM I/O reads and writes
    - Total ESAF (DB2) calls
    - CPU time in TOD clock format
    - Elapsed time of database I/Os
    - Elapsed wait time for database locks
    - More…
      - See DFSETPCP macro for mapping of fields
  - Record is 528 bytes
    - You can easily estimate the logging increase
      - 528 bytes times the transaction rate

Figure 2-23. Enhanced Log Record Statistics CMA01.0

Notes:

The additional fields in the x’56FA’ log record include read counts and write counts for both VSAM and OSAM database data sets. There are counts of External Subsystem Attach Facility calls. There are typically calls to DB2, but they also may be for MQ or another subsystem. The CPU time for the transaction is written in time of day (TOD) format. The statistics include the total elapsed time for database I/Os and the total elapsed time for waits for locks. These two elapsed times were previously available in the x’07’ log records for DBCTL threads. It was not available for IMS TM except in IMS Monitor trace records. The x’56FA’ log record is mapped by the DFSETPCP macro.

The x’56FA’ log record is approximately 528 bytes. It is written for every transaction and for every commit for non-message driven programs for those transactions and programs with the option. If you choose to use this option, you can easily estimate its effect on logging volumes. The increase in logging may be noticeable, but most IMS systems will easily handle this increase.



Information On Demand
IMS Version 10
IBM

Enhanced Log Record Statistics

- Transaction level statistics (x'56FA')
  - Specified in the diagnostic statistics section of DFSDFxxx PROCLIB member

```
<SECTION=DIAGNOSTIC_STATISTICS>
  TRANSTAT=Y|N
```

    - Applies to all transactions and programs unless overridden by command
  - Commands for individual transactions or programs
    - PGM applies only to non-msg driven, ODBA, and CCTL (CICS) programs
    - UPDATE TRAN|TRANDESC|PGM|PGMDESC SET(TRANSTAT(Y|N))
      - Example:

```
UPD TRAN NAME(ABC) SET(TRANSTAT(Y))
```

    - CREATE TRAN|TRANDESC|PGM|PGMDESC SET(TRANSTAT(Y|N))
      - Example:

```
CRE TRAN NAME(ABX) LIKE(RSC(ABC)) SET(TRANSTAT(N))
```

25

Figure 2-24. Enhanced Log Record Statistics CMA01.0

Notes:

Transaction level statistics log records are optional. The TRANSTAT specification in the diagnostic statistics section of the DFSDFxxx PROCLIB member determines the default setting for all transactions and non-message driven programs. TRANSTAT=N is the default. TRANSTAT=Y causes the log records to be written. The setting in this member may be overridden for individual transactions and non-message driven programs by using commands.

There are two commands that may be used to set transaction level statistics for transactions and non-message driven programs.

The UPDATE command can be used to change the current setting of transaction level statistics for any of the entities.

The CREATE command includes the SET(TRANSTAT(Y|N)) parameter for setting transaction level statistics on or off when creating a new transaction, transaction descriptor, program, or program descriptor. It defaults to the setting in the descriptor or the transaction or program specified in the LIKE parameter.

There is also a QUERY command that may be used to show the current setting for a transaction (TRAN), transaction descriptor (TRANDESC), program (PGM), or program descriptor (PGMDESC).

Enhanced Log Record Statistics

- New information in application termination accounting log record (x'07')
  - Shared queue "false schedule" indicator
  - Application execution CPU time in TOD clock format
  - Database I/O time and total lock wait time
  - VSAM I/O counts, OSAM I/O counts, ESAF calls, FP FLD calls, FP POS calls, and Abend reason code
  - DLI calls used for Fast Path DBs are included in DLI call accounting area
- New information in application start accounting log record (x'08')
  - Wait times for intent conflicts and pool space
  - Elapsed time of scheduling process
  - Program name and PSB name
- New information in CPI-C driven tran. termination log record (x'0A07')
  - Total CPU time in TOD clock format

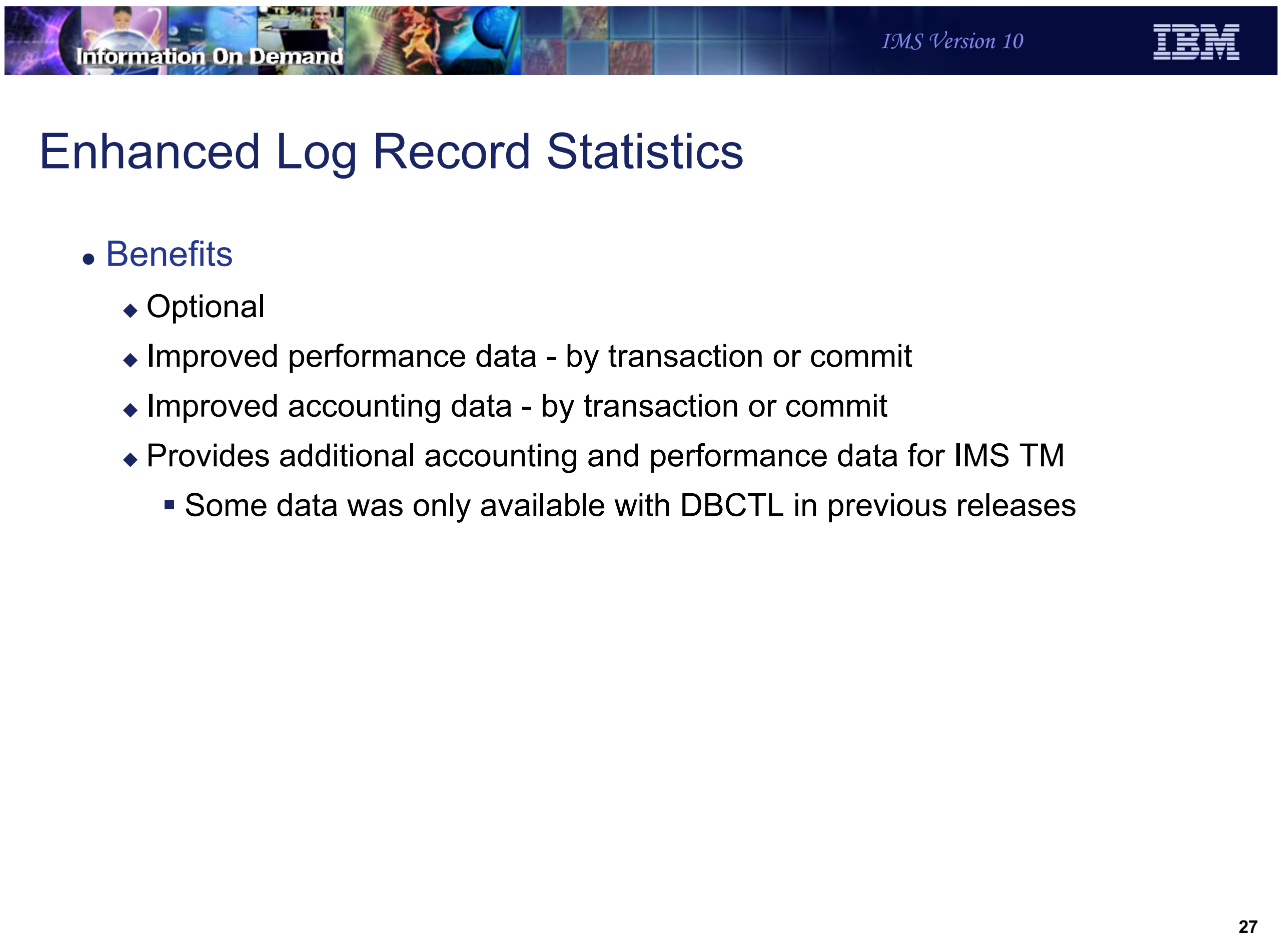
Figure 2-25. Enhanced Log Record Statistics CMA01.0

Notes:

The application termination accounting log record (x'07') has been enhanced. The DLRFALSE bit in the DLRFLAG2 byte indicates that the scheduling of the transaction was false in a shared queues environment. The DLREXTIM field contains the CPU time for the program in time of day (TOD) format. The DLRTMEIO and DLRTMEPL fields are now populated in all environments. Previously, only DBCTL threads caused these fields to be populated. DLRTMEIO contains the total I/O time for database calls. DLRTMEPL contains the total wait time for database locks. Additional fields have been added. These include counts for VSAM and OSAM database data set reads and writes, ESAF calls, such as those to DB2, and additional Fast Path call counts for FLD calls and POS calls. When an abend occurs the abend reason code is included when available. The DLI call counts for GU, GN. ISRT, etc. now include both full function and Fast Path database calls. The log record is mapped by the DFSLOG07 macro. The offsets to some fields have changed from previous releases.

The application start accounting log record (x'08') is written for each scheduling of a program. The statistics in the log record have been enhanced. Statistics about the scheduling of the program were written for DBCTL threads in previous releases. IMS V10 writes these statistics for all programs, not just DBCTL threads. These statistics show the time that scheduling waited due to intent conflicts as could occur when a PROCOPT includes E, the time that scheduling waited due to a lack of pool space, such as in the PSB pool, and the elapsed time of the scheduling process. Two new fields have been added. They are LINTPGM which contains the program name and LINTPSB which contains the PSB name. The program name and PSB name were previously available in the LINTSY1 and LINTSY2 fields, however, there were complicated rules governing the contents of these two fields. LINTPGM and LINTPSB greatly simplify the discovery of the program name and PSB name. LINTSY1 and LINTSY2 remain in IMS V10 and contain the same contents as in previous releases. The log record is mapped by the DFSLOG08 macro.

The x’0A07’ log record is written when a CPI-C driven transaction terminates. It includes accounting data similar to that in the x’07’ log records. IMS V10 has added the L0AEXT field to the log record. It contains the CPU time in time of day (TOD) format. Previously, this time was only available in timer units in the L0AETIM field in the log record. The log record is mapped by the DFSLOG0A macro.


Information On Demand
IMS Version 10
IBM

Enhanced Log Record Statistics

- Benefits
  - Optional
  - Improved performance data - by transaction or commit
  - Improved accounting data - by transaction or commit
  - Provides additional accounting and performance data for IMS TM
    - Some data was only available with DBCTL in previous releases

27

Figure 2-26. Enhanced Log Record Statistics CMA01.0

Notes:

IMS V10 provides enhanced performance and accounting information in its log records. This can provide for granular information for accounting and charge out purposes. It also provides more detailed information for understanding the performance of individual transactions and programs. Some of the information in the log records was previously available only when the IMS Monitor trace was on for IMS TM environments. It was then processed by the IMS Monitor report program or the IMS Performance Analysis tool. IMS V10 provides new options for obtaining this data.

Large Sequential Data Sets

Figure 2-27. [No title] CMA01.0

Notes:

Information On Demand
IMS Version 10
IBM

Large Sequential Data Sets

- z/OS 1.7 DFSMS adds support for large sequential data sets on DASD
  - More than 65,535 tracks on one volume for a data set
    - More than 4369 cylinders
  - Theoretical limit is 16,777,215 tracks per volume

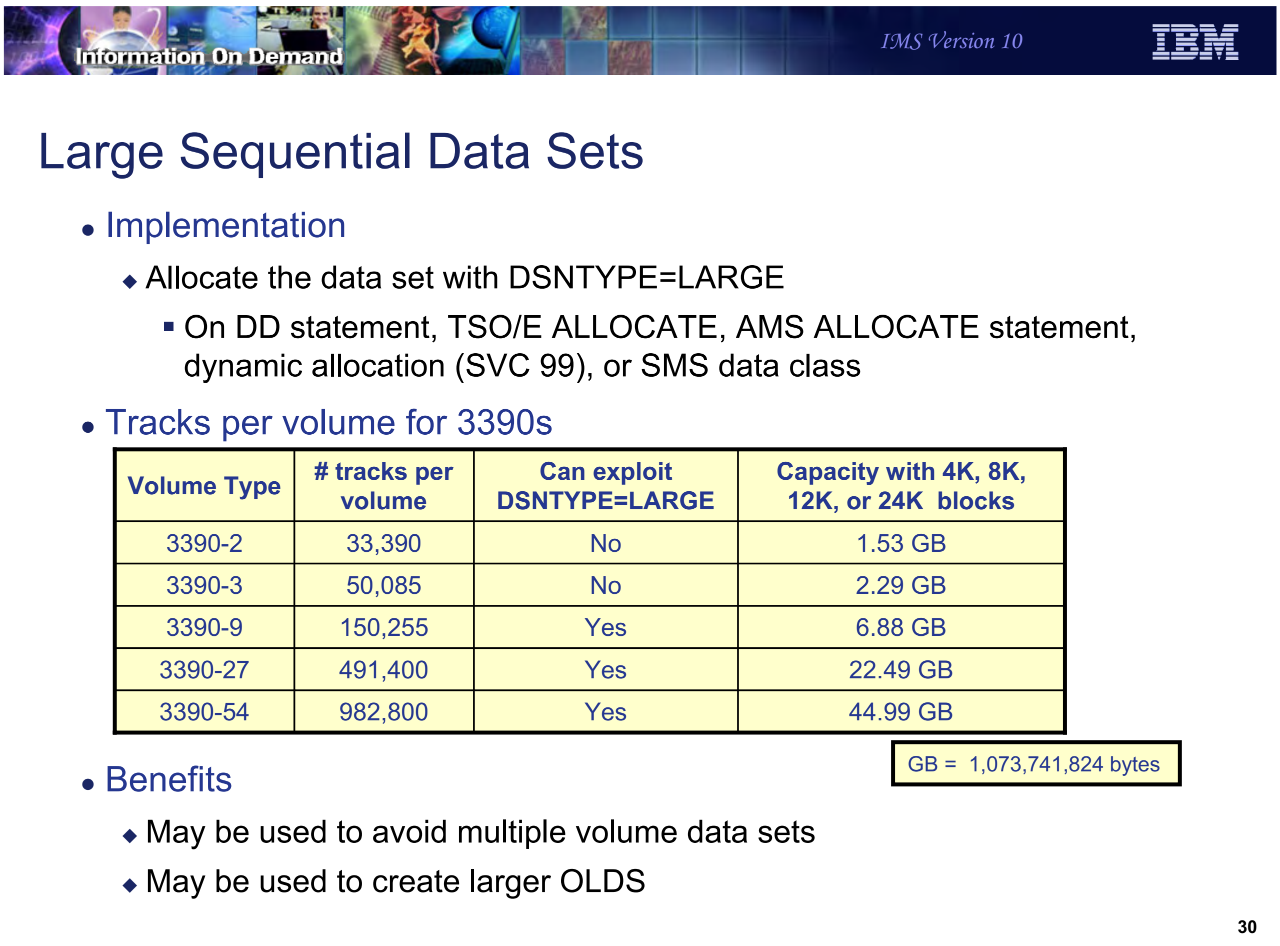
- IMS V10 adds large sequential data set support
  - GSAM/BSAM and OSAM data sets
    - Includes
      - Logs
      - Trace data sets
      - Message queue data sets
      - GSAM/BSAM files
      - OSAM database data sets

Figure 2-28. Large Sequential Data Sets CMA01.0

Notes:

z/OS 1.7 added support for large sequential data sets on DASD. These data sets may occupy more than 65,535 tracks on a single volume. Without this support, physical sequential data sets (DSORG=PS) were restricted to 65,535 tracks on any volume. Multi-volume data sets could have up to 65,535 tracks on each volume. The architectural (theoretical) limit with large data set support is 16,777,215 tracks per volume. This is far beyond the maximum number of tracks per volume that storage subsystems support.

IMS Version 10 adds support for large sequential data sets. This applies to GSAM/BSAM and OSAM data sets. These include logs (OLDS and SLDS), trace data sets, message queue data sets, GSAM files, and OSAM database data sets.



Information On Demand
IMS Version 10
IBM

Large Sequential Data Sets

- Implementation
  - Allocate the data set with DSNTYPE=LARGE
    - On DD statement, TSO/E ALLOCATE, AMS ALLOCATE statement, dynamic allocation (SVC 99), or SMS data class
- Tracks per volume for 3390s

Volume Type	# tracks per volume	Can exploit DSNTYPE=LARGE	Capacity with 4K, 8K, 12K, or 24K blocks
3390-2	33,390	No	1.53 GB
3390-3	50,085	No	2.29 GB
3390-9	150,255	Yes	6.88 GB
3390-27	491,400	Yes	22.49 GB
3390-54	982,800	Yes	44.99 GB

GB = 1,073,741,824 bytes

- Benefits
  - May be used to avoid multiple volume data sets
  - May be used to create larger OLDS

Figure 2-29. Large Sequential Data Sets CMA01.0

Notes:

A data set may use large data set support only if if is allocated with DSNTYPE=LARGE specified. That is, large data set support only applies to data sets that are created with this specification. DSNTYPE=LARGE may be specified on the DD statement, a TSO/E ALLOCATE statement, or an AMS ALLOCATE statement. For dynamic allocation (SVC 99) a DALDSNT text unit may be used to specify LARGE.

The following is an example of the use of a DD statement with DSNTYPE=LARGE specified.

```
//ABCDEF   DD   DSN=IMS.ABC.XYZ,DSNTYPE=LARGE,
//              UNIT=SYSDA,SPACE=(CYL,(4500,100)),
//              DISP=(NEW,CATLG),VOL=SER=LRGVOL1
```

The table is provided to show which volume types may contain large format data sets. 3390-1, 3390-2, and 3390-3 have less than 65,535 tracks per volume. Large format data sets may be defined on them, however, they cannot take advantage of the larger number of tracks per volume. The table also shows the approximate capacity of a volume for certain block sizes. This applies to 4K, 8K, 12K and 24K. Other block sizes may yield more or less capacity due to the number of blocks which will fit on a track. For example, a 3390-9 will hold a 7.45 GB data set with a block size of 26K.

Large format data sets may be used for any log, trace data set, message queue data set, GSAM/BSAM data set or OSAM database data set. They may be used to avoid the requirement for multiple volumes for very large data sets. Since OLDS data sets must reside on only one volume, large format data sets allow OLDS to be larger than previously possible.

OSAM database data sets are limited to a maximum of 8 gigabytes. The large format data set support does not change this limit.

Information On Demand
IMS Version 10

System Utilities Enhancements

Figure 2-30. [No title] CMA01.0

Notes:

Log Transaction Analysis Utility (DFSILTA0)

- Rewritten to improve performance
- Accepts logs from multiple IMS systems
  - Log Merge utility not required for MSC users
  - Support for shared queues
- Support for non-recoverable transactions has been added

Figure 2-31. Log Transaction Analysis Utility (DFSILTA0) CMA01.0

Notes:

The Log Transaction Analysis utility (DFSILTA0) has several improvements.

In previous releases the utility could produce reports for systems connected by MSC. This required that the user invoke the Log Merge utility to merge the logs from multiple systems and use its output as input to the Log Transaction Analysis utility. The requirement to run Log Merge has been eliminated. Logs from multiple IMS systems may be input to a single execution of the utility. In previous releases the Log Transaction Analysis utility did not support shared queues. This restriction has been eliminated. One execution of the utility can process logs from multiple systems using shared queues.

When logs from multiple systems are input to the utility multiple LOGINxxx DD statements are used. Any valid JCL characters may be used to replace “xxx” in the DD names. The last character in “xxx” is used as the System ID in reports. When multiple log data sets from one IMS system are inputs, they are concatenated under a single DD name.

In previous releases the utility did not report non-recoverable transactions, including APPC messages. Support for reporting non-recoverable transactions has been added in IMS V10.

The Log Transaction Analysis utility has an option to write the output report to an output log data set, to a printer DD, or to both. In previous releases this was controlled by the OUT= parm on the EXEC statement. Both outputs were produced unless the OUT= parameter eliminated them. Possible values were OUT=NOREPORT, OUT=NOLOG, and OUT=(NOREPORT,NOLOG). IMS V10 has eliminated this parm. If you do not want either output just do not include its DD statement. PRINTER is used for the report and LOGOUT is used for the output log data set. The need for the output log data set has been reduced in IMS V10. In previous releases it was used to create a log for input to the Statistical Analysis utility. This is available in IMS V10 but not required.

Information On Demand
IMS Version 10
IBM

Statistical Analysis Utility (DFSISTS0)

- Rewritten to improve performance
  - Six steps replaced by single step job
    - Internal sorts done in single step
- Accepts logs from multiple IMS systems
  - Log Merge utility not required for MSC users
  - Support for shared queues
- Support for non-recoverable transactions has been added
- New message select output order statement
  - Controls the order of messages in the report

Figure 2-32. Statistical Analysis Utility (DFSISTS0) CMA01.0

Notes:

The Statistical Analysis utility (DFSISTS0) has several improvements.

In previous releases the Statistical Analysis utility reports were created by a six step job which included two sorts. The reports are now produced by a single step. This steps invokes internal sorts.

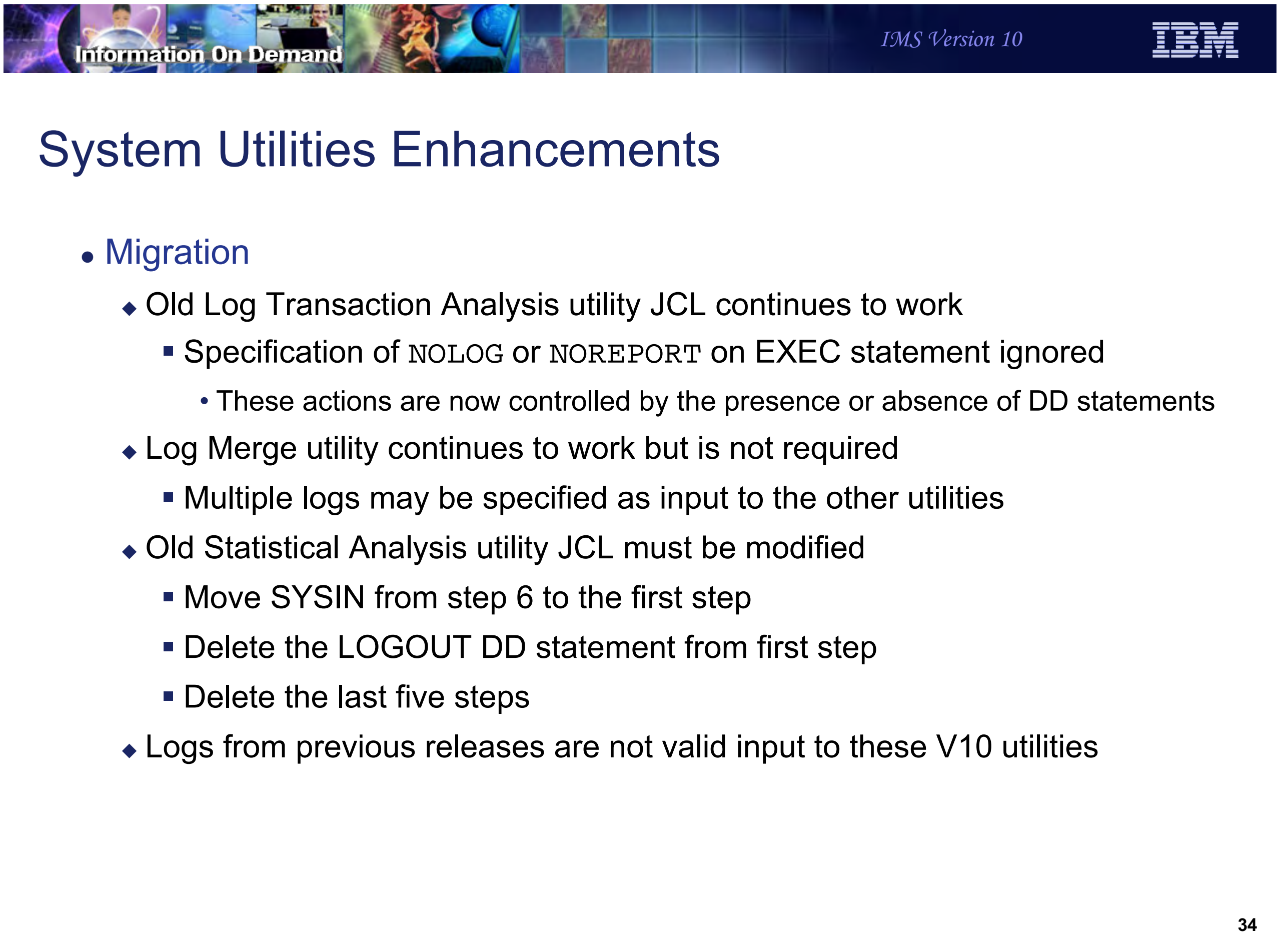
Support for shared queues and MSC is comparable to that in the Log Transaction Analysis utility for IMS V10. This includes the support for multiple LOGINxxx DD statements. When logs from multiple systems are input to the utility multiple LOGINxxx DD statements are used. Any valid JCL characters may be used to replace “xxx” in the DD names. When multiple log data sets from one IMS system are inputs, they are concatenated under a single DD name. The capability to specify multiple input logs eliminates the requirement to use an output log from the Log Transaction Analysis utility when you want a report from multiple IMS systems. In previous releases this was the way that log records from multiple IMS systems were given to the Statistical Analysis utility.

In previous releases the utility did not report non-recoverable transactions, including APPC messages. Support for reporting non-recoverable transactions has been added in IMS V10.

The message select output order statement is a new utility control statement specified in the SYSIN DD data set. The format of the message select output order statement is:

```
ORDER=CREATE|DEST|SOURCE
```

This statement determines the order in which the message select function lists or copies messages. If CREATE is specified, the messages are ordered by the time of the first input message. CREATE is the default. If SOURCE is specified, the messages are ordered by the originating LTERM names. If DEST is specified, the messages are ordered by the original destination. This is the original transaction unless it is a message switch.


Information On Demand
IMS Version 10
IBM

System Utilities Enhancements

- Migration
  - Old Log Transaction Analysis utility JCL continues to work
    - Specification of `NOLOG` or `NOREPORT` on EXEC statement ignored
      - These actions are now controlled by the presence or absence of DD statements
  - Log Merge utility continues to work but is not required
    - Multiple logs may be specified as input to the other utilities
  - Old Statistical Analysis utility JCL must be modified
    - Move SYSIN from step 6 to the first step
    - Delete the LOGOUT DD statement from first step
    - Delete the last five steps
  - Logs from previous releases are not valid input to these V10 utilities

Figure 2-33. System Utilities Enhancements CMA01.0

Notes:

The JCL used in previous releases for the Log Transaction Utility can be used with IMS V10. There is one slight incompatibility. Previously you could specify an OUT= parameter on the EXEC statement. OUT=NOLOG specified that an output report should not be written to the LOGOUT DD data set. OUT=NOREPORT specified that an output report should not be written to the PRINTER DD data set. In IMS V10 if you do not want either of these data sets do not include the DD statement for it. IMS V10 ignores the OUT= parameter on the EXEC statement.

The Log Merge utility was used in previous releases to merge logs from different systems. The output of Log Merge was used as input to the Log Transaction Analysis or Statistical Analysis utilities. The Log Merge utility can be used with IMS V10, but it is not required. Both analysis utilities can merge multiple logs as part of their processing.

The JCL used for the Statistical Analysis utilities in previous releases must be changed for IMS V10. The actions needed to change the JCL are listed on the slide. The control statements were specified in the SYSIN data set of the sixth step. The SYSIN data set is now part of the first (and only) step. The LOGOUT DD in the old first step was used to pass data to the following steps. Since only one step is used in IMS V10, the LOGOUT DD is no longer used. The last five steps are no longer used, so their JCL should be eliminated.

Logs produced by IMS V8 and IMS V9 are not valid input to the IMS V10 Log Transaction Analysis utility and the IMS V10 Statistical Analysis utility.

System Utilities Enhancements

- Benefits
  - Support for shared queues
  - Support for non-recoverable transactions
  - Simplified execution
  - Improved reporting by Statistical Analysis utility
  - Improved performance

Figure 2-34. System Utilities Enhancements CMA01.0

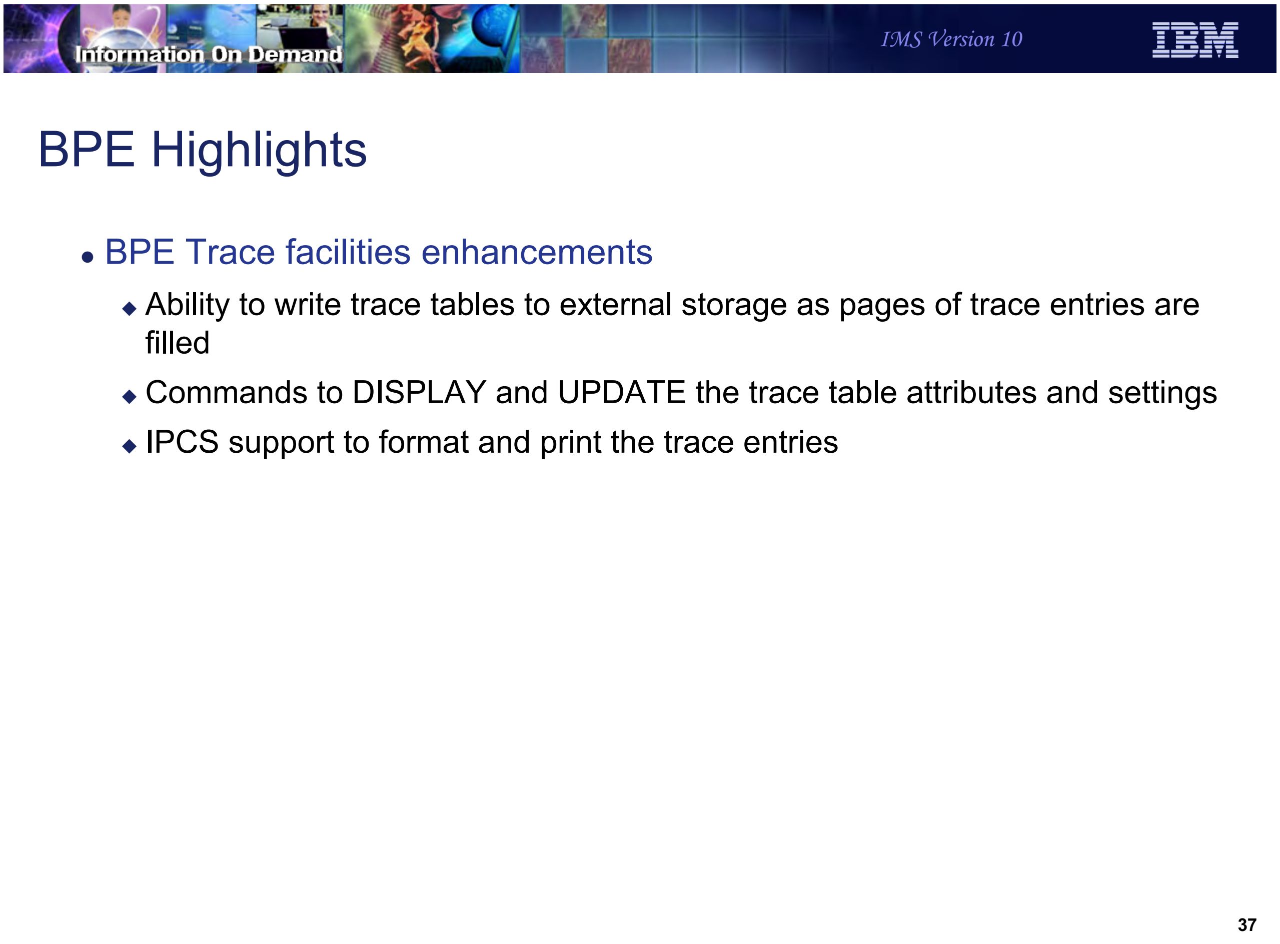
Notes:

The Log Transaction Analysis utility (DFSILTA0) and the Statistical Analysis utility (DFSISTS0) enhancements provide several benefits. They now provide information for shared queues systems and for non-recoverable transactions in any systems. Their execution has been simplified in two ways. You can now feed logs from multiple systems into both utilities without first merging the logs with the Log Merge utility. The Statistical Analysis utility is now just a single step execution instead of multiple programs and sorts executed in multiple steps. The Statistical Analysis utility has new reporting options that provide you a choice in how the report lines are ordered. The simplification of execution also provides improved performance since entire data sets are not written in each step and passed to following steps.

BPE External Trace

Figure 2-35. [No title] CMA01.0

Notes:


Information On Demand
IMS Version 10
IBM

BPE Highlights

- BPE Trace facilities enhancements
  - Ability to write trace tables to external storage as pages of trace entries are filled
  - Commands to DISPLAY and UPDATE the trace table attributes and settings
  - IPCS support to format and print the trace entries

Figure 2-36. BPE Highlights CMA01.0

Notes:

The Base Primitive Environment (BPE) is a critical component of several IMS-related components such as IMS Connect, the Common Queue Server, and several of the address spaces introduced with the Common Service Layer. In prior releases, problem determination using the BPE traces often required an increase in the number of trace pages and a corresponding requirement to recreate the problem and capture an MVS dump at the correct moment. This process increased the chance of multiple repeats, thereby decreasing productivity and increasing the time to find problem resolution.

IMS V10 enhances the BPE Trace facilities to provide a means to write the trace tables to external storage, Tape or DASD, as pages of trace entries are filled thereby increasing the capacity associated with tracing. Additionally, commands are provided to display the trace table attributes and update the trace level settings. To view data, trace formatting is provided through IPCS exits and formatting routines.

Information On Demand
IMS Version 10
IBM

Implementation

- External data sets use Generation Data Groups (GDGs)
  - GDG name and other trace parameters are specified in the BPE configuration PROCLIB member (e.g., BPECFGxx)
    - New parameter EXTTRACE, *and*
    - Existing TRCLEV parameter
      - New EXTERNAL sub-parameter
    - Note: as in previous releases, each component can create a separate BPE configuration member or share one
  - Creating Generation Data Group (GDG) definitions requires:
    - Defining GDG base
    - Defining GDG model

Figure 2-37. Implementation CMA01.0

Notes:

The BPE external trace data sets are defined as GDGs and must be specified in the BPE configuration member in PROCLIB. The primary tasks include the specification of new values in BPECFGxx as well as creating the GDG definitions.

Both TRCLEV with EXTERNAL=YES and the EXTTRACE statement have to be defined for external trace to start recording data. The existing TRCLEV statement is associated with a particular trace table and has been enhanced with the EXTERNAL=YES/NO parameter which specifies whether that trace table is to be externalized or not. The EXTTRACE statement is for the data set definition options and is needed for the function to be activated. If a user specifies, EXTERNAL=YES on a table and does not specify an EXTTRACE statement, an error message is produced.

Setup - BPE Configuration

- BPECFGxx
  - New EXTTRACE parameter
    - Provides information about the new external data sets

```
►►─EXTTRACE(─GDGDEF(─DSN(data_set_name)───────────────────────────►
                                       ├UNIT(unit) VOLSER(volser)─┤
                                       └STORCLAS(storclas)────────┘

 ►─ SPACE(space)─────────────────────────────────────────────────────────────►
                ├SPACEUNIT(  BLK  )─┤└BLKSIZE(blksize)─┘└DATACLAS(dataclas)─┘
                |           ├ TRK ┤  |
                |           └ CYL ┘  |
                └AVGREC(  U  )───────┘
                        ├ K ┤
                        └ M ┘

►────────────────────────────────────────────────────────────────────)─►◄
   └COPYBUFS(copybufs)─┘  └IOBUFS(iobufs)─┘  └COMP(ims_component)─┘
```

Figure 2-38. Setup - BPE Configuration CMA01.0

Notes:

The EXTTRACE keyword definition is an enhancement to the BPECFGxx configuration definition set. One EXTTRACE keyword can be specified for each IMS component running with BPE. Alternatively, one EXTTRACE definition can be shared among multiple IMS components. If the COMP specification is omitted, the associated keyword applies to all components.

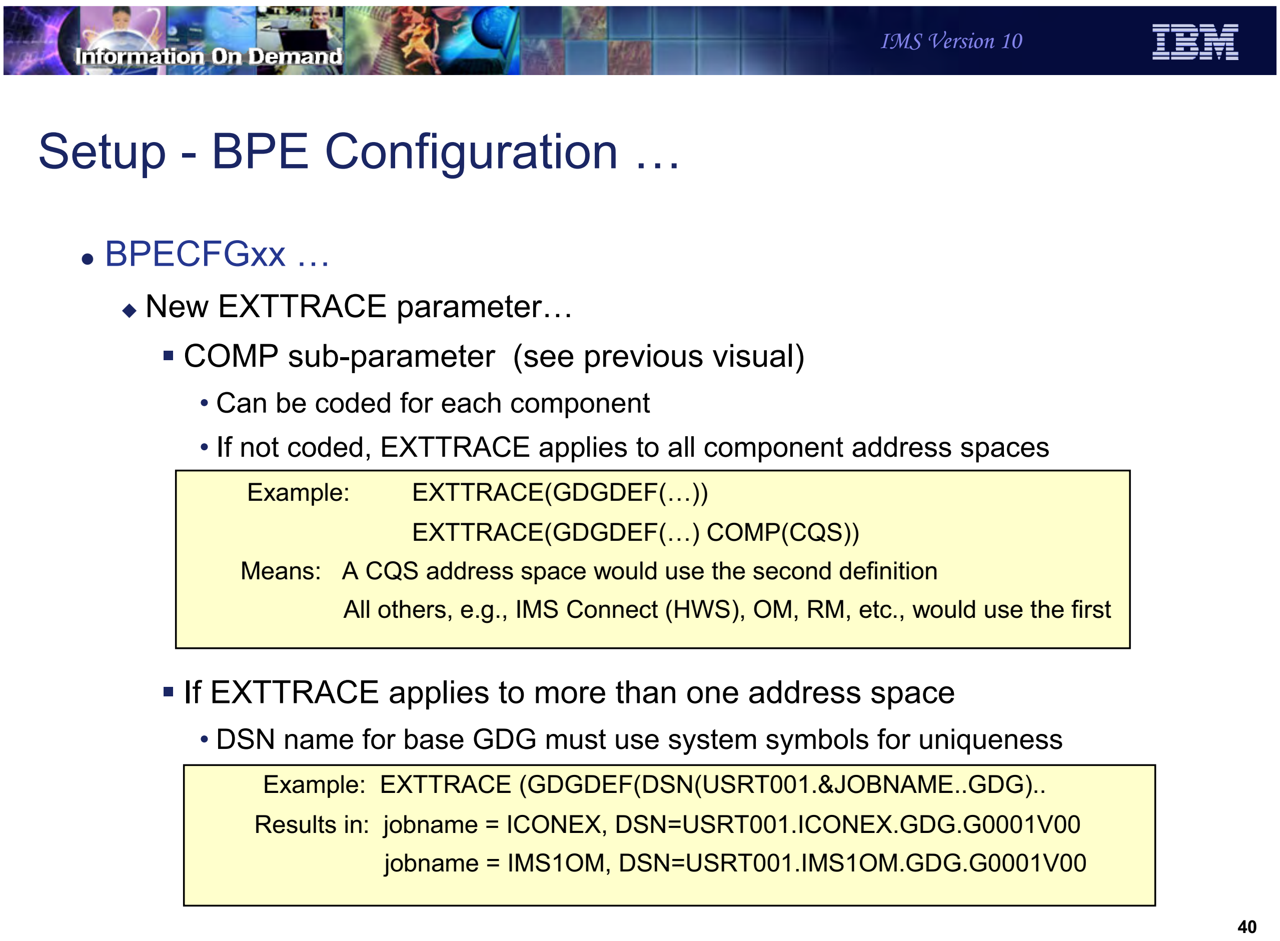
- GDGDEF(DSN(data_set_name),....) This is the name of the GDG data set. It does not include the relative generation number in parentheses. It does not include the absolute generation number. (For definition of absolute and relative generation number, refer z/OS JCL Reference).
- Location: Choices include STORCLAS(class_name), or, (UNIT(unit_name) and VOLSER(serial_number)

SPACE(space): specify number of units of space to be allocated for a data set

SPACEUNIT(BLK,TRK,CYL): quantity value of the SPACE operand as the number of blocks, tracks or cylinders.

AVGREC(U | K | M), in SMS, determines the size of the dataset allocation. The following are the valid values for AVGREC: (U - Use the primary space quantity specified on the SPACE operand; K - Multiply the primary space quantity specified on the SPACE operand by 1024 (1 K); M - Multiply the primary space quantity specified on the SPACE operand by 1,048,576 (1 M)).

- BLKSIZE(blksize): specifies the block size for the data set. The maximum allowable decimal value for block size is 32760.
- DATACLAS(dataclas): specifies the data class for the data set .
- COPYBUFS - are the buffers that will be used to copy data (trace table entries) from BPE tables. Data from these buffers are then copied into IOBUFS. The default value for number of buffers is 15 with max value of 64,000.
- IOBUFS - are the buffers that will copy data into external trace data sets. Number of IOBUFS defined by default is 2 and max value is 99.
- COMP(ims_component) - Specifies the IMS component to which the EXTTRACE statement applies. Possible values are: (CQS, HWS, OM, RM, SCI).


Information On Demand
IMS Version 10
IBM

Setup - BPE Configuration …

- BPECFGxx …
  - New EXTTRACE parameter…
    - COMP sub-parameter (see previous visual)
      - Can be coded for each component
      - If not coded, EXTTRACE applies to all component address spaces

Example: EXTTRACE(GDGDEF(…))
EXTTRACE(GDGDEF(…) COMP(CQS))
Means: A CQS address space would use the second definition
All others, e.g., IMS Connect (HWS), OM, RM, etc., would use the first

- If EXTTRACE applies to more than one address space
  - DSN name for base GDG must use system symbols for uniqueness

Example: EXTTRACE (GDGDEF(DSN(USRT001.&JOBNAME..GDG)..
Results in: jobname = ICONEX, DSN=USRT001.ICONEX.GDG.G0001V00
jobname = IMS1OM, DSN=USRT001.IMS1OM.GDG.G0001V00

Figure 2-39. Setup - BPE Configuration … CMA01.0

Notes:

If COMP is not coded, then the EXTTRACE statement applies to all component address space types that are using the BPE proclib member, and that do not have an EXTTRACE statement with a specific COMP specified for their component type. For example, for a BPE configuration PROCLIB member with the following statements:

EXTTRACE(GDGDEF(...))

EXTTRACE(GDGDEF(...) COMP(CQS))

If a CQS address space were started, it would use the external trace definitions from the second EXTTRACE statement. If any other type of address space were started, it would use the external trace definition from the first EXTTRACE statement. Note that if you have the EXTTRACE apply to more than one address space, you must use z/OS system symbols in the data set name (DSN) for the GDG base name, to ensure that each data set has a unique name. For example:

DSN(USRT001.&JOBNAME..GDG) and jobname=ICONEX then
DSN=USRT001.ICONEX.GDG.G0001V00

DSN(USRT001.&JOBNAME..GDG) and jobname=IMS1OM then
DSN=USRT001.IMS1OM.GDG.G0001V00

…

Setup - BPE Configuration …

- BPECFGxx …
  - New EXTERNAL sub-parameter in the TRCLEV keyword definition
    - Indicates whether trace entries created in the specified table should be written to an external trace data set

```
                                                            ┌─,EXTERNAL=NO──┐
▶▶──TRCLEV=(type, level, ims_component──────────────────────┼───────────────┼─)────▶◀
                                      └─,PAGES=num_pages─┘  └─,EXTERNAL=YES─┘
```

- EXTERNAL=YES/NO
  - NO (the default) indicates that trace entries in the table are not to be externalized.
  - YES indicates that trace entries are to be externalized

- Trace entries will only be written if external trace data sets have been defined (EXTTRACE= parameter)

Figure 2-40. Setup - BPE Configuration … CMA01.0

Notes:

The new EXTERNAL sub-parameter in the TRCLEV keyword definition provides the specification of whether or not the trace entries are to be externalized.

Information On Demand IMS Version 10 IBM

Commands

- BPE commands are invoked through the z/OS MODIFY command
  - TRACETABLE resource type - internal BPE trace tables managed by
    - BPE
    - IMS component using BPE
      - IMS Connect, CQS, OM, RM, or SCI
- Two command verbs operate on the TRACETABLE resource type:
  - DISPLAY - shows the trace level and number of trace table pages of specified trace tables
  - UPDATE - modifies the trace level attribute of specified trace tables

Figure 2-41. Commands CMA01.0

Notes:

BPE commands are invoked through the z/OS MODIFY command.

The TRACETABLE resource type refers to the internal BPE-managed trace tables defined either by BPE (for example: DISP, CBS, STG, LATC), or by the IMS component using BPE (for example: IMS Connect, CQS, OM, RM, SCI).

Two command verbs operate on the TRACETABLE resource type: DISPLAY and UPDATE.

DISPLAY Command

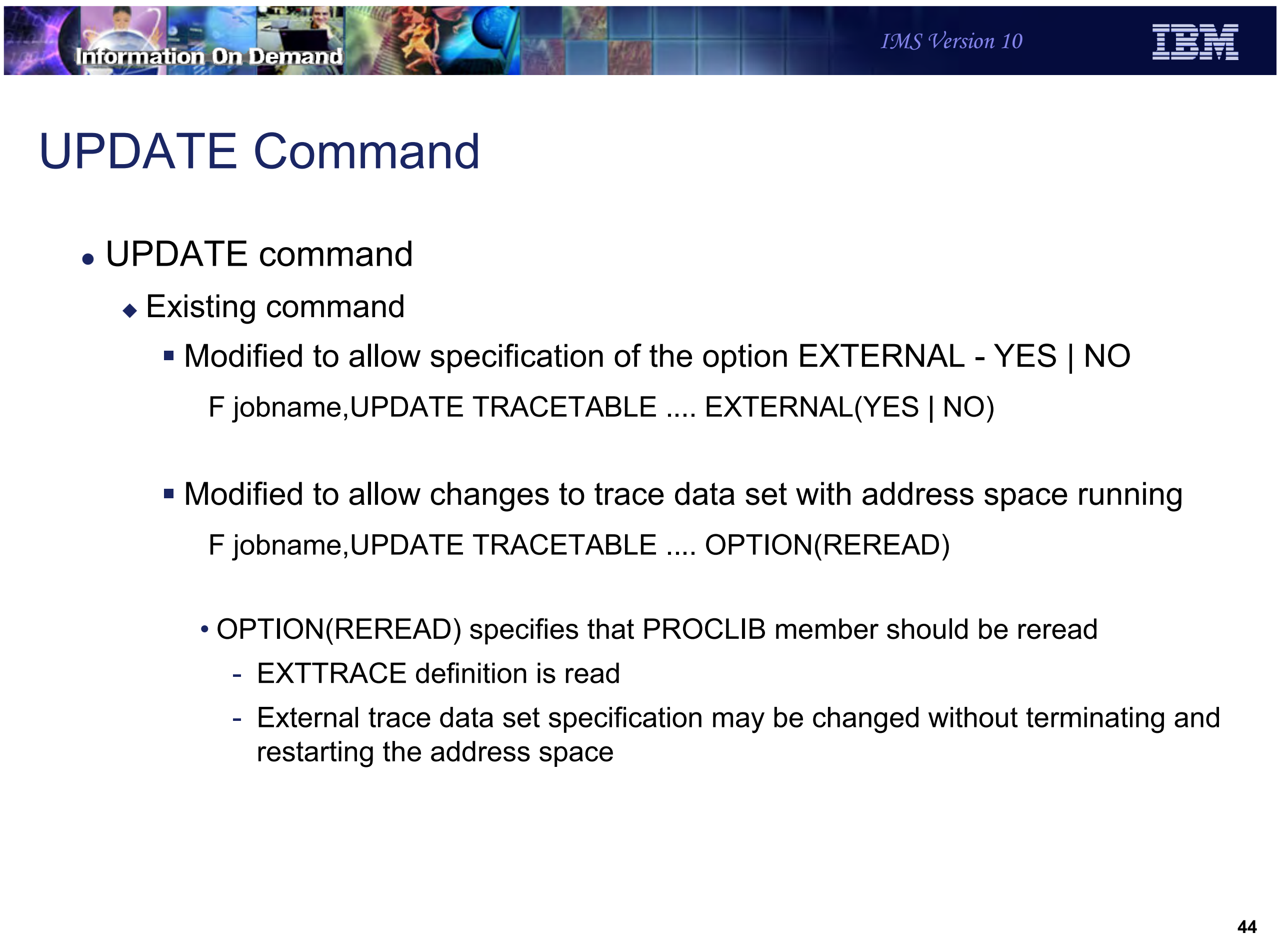
- DISPLAY command
  - Existing command
    - Output is modified to show a new EXT column
      - Indicates YES or NO to show the external trace is set on or off for a particular trace table
  - Example

```
F  CQS1,DIS TRTAB NAME(*) OWNER(CQS)

BPE0030I TABLE    OWNER     LEVEL   #PAGES       EXT
BPE0000I CQS      CQS       MEDIUM     4         YES
BPE0000I ERR      CQS       HIGH       4         NO
BPE0000I INTF     CQS       LOW        8         NO
BPE0000I STR      CQS       HIGH       8         YES
BPE0032I DISPLAY TRACETABLE COMMAND COMPLETED
```

Figure 2-42. DISPLAY Command CMA01.0

Notes:

The BPE DISPLAY TRACETABLE command is an existing BPE command that is used to display the current attribute settings for the requested trace tables. IMS V10 modifies the command output by adding a new column to indicate, YES or NO, whether the external trace is set on or off for a particular trace table.


Information On Demand
IMS Version 10
IBM

UPDATE Command

- UPDATE command
  - Existing command
    - Modified to allow specification of the option EXTERNAL - YES | NO

```
F jobname,UPDATE TRACETABLE .... EXTERNAL(YES | NO)
```

    - Modified to allow changes to trace data set with address space running

```
F jobname,UPDATE TRACETABLE .... OPTION(REREAD)
```

      - OPTION(REREAD) specifies that PROCLIB member should be reread
        - EXTTRACE definition is read
        - External trace data set specification may be changed without terminating and restarting the address space

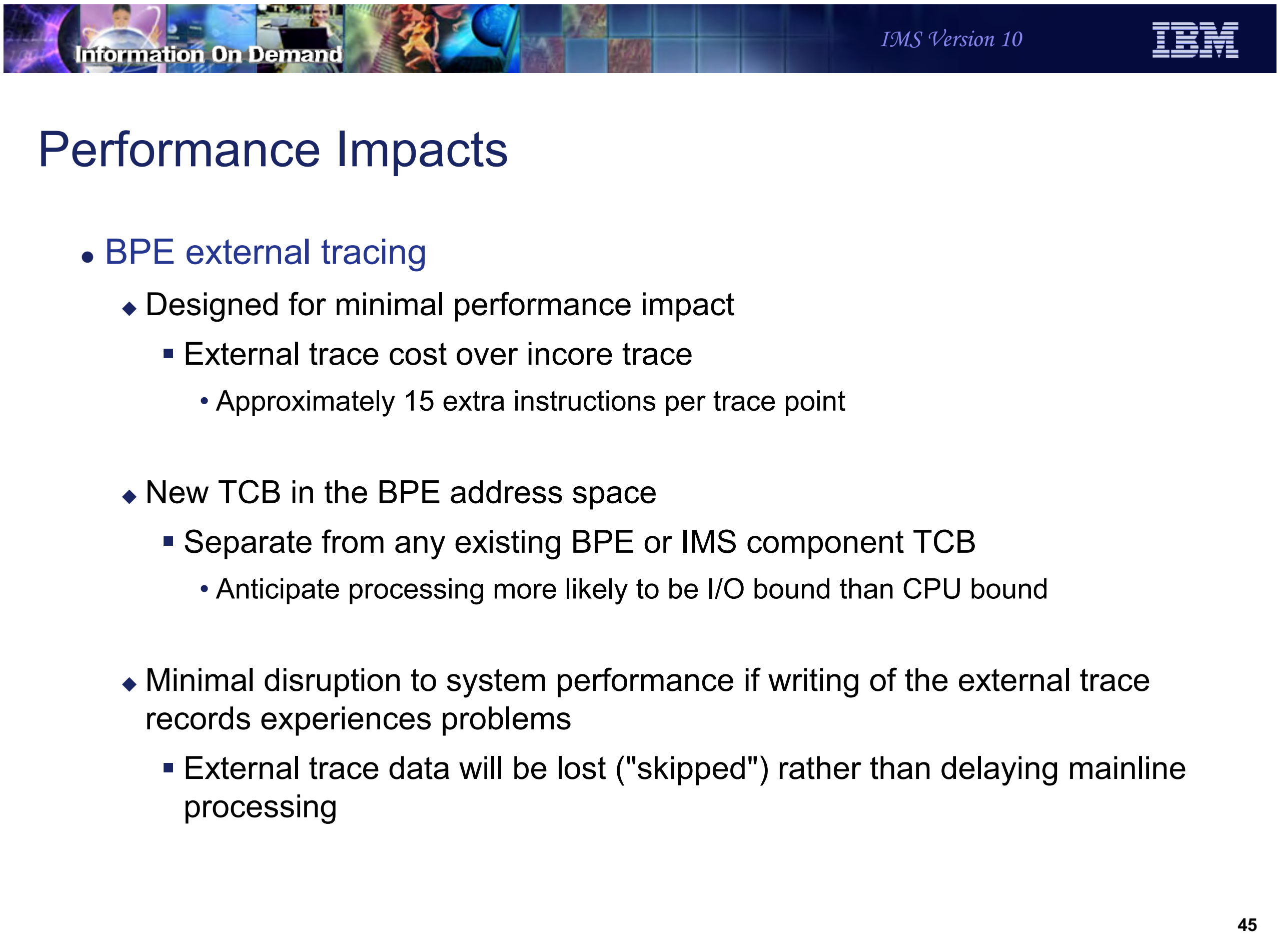
Figure 2-43. UPDATE Command CMA01.0

Notes:

The BPE UPDATE TRACETABLE command is used to change the trace level setting for the requested trace tables. The command syntax for UPDATE TRACETABLE has been extended to allow specification of the option EXTERNAL from YES to NO or vice versa.

EXTERNAL is an optional parameter that specifies whether or not the trace entries for the trace tables affected by the command should be written to the external trace data set. If the keyword is not specified on an UPD TRTAB command, then there is no change made to the external trace setting for the affected tables. Note that specification of EXTERNAL(YES) requires that the BPE configuration PROCLIB member contain the parameters to define a trace data set.

The command syntax for UPDATE TRACETABLE also has been extended to allow specification of the option OPTION(REREAD). REREAD is the only valid value for OPTION. REREAD causes the PROCLIB member to be reread for the information in the specified EXTTRACE statement. This allows the external trace data set specification to be changed or added without terminating the address space and restarting it.



Information On Demand
IMS Version 10
IBM

Performance Impacts

- BPE external tracing
  - Designed for minimal performance impact
    - External trace cost over incore trace
      - Approximately 15 extra instructions per trace point
  - New TCB in the BPE address space
    - Separate from any existing BPE or IMS component TCB
      - Anticipate processing more likely to be I/O bound than CPU bound
  - Minimal disruption to system performance if writing of the external trace records experiences problems
    - External trace data will be lost ("skipped") rather than delaying mainline processing

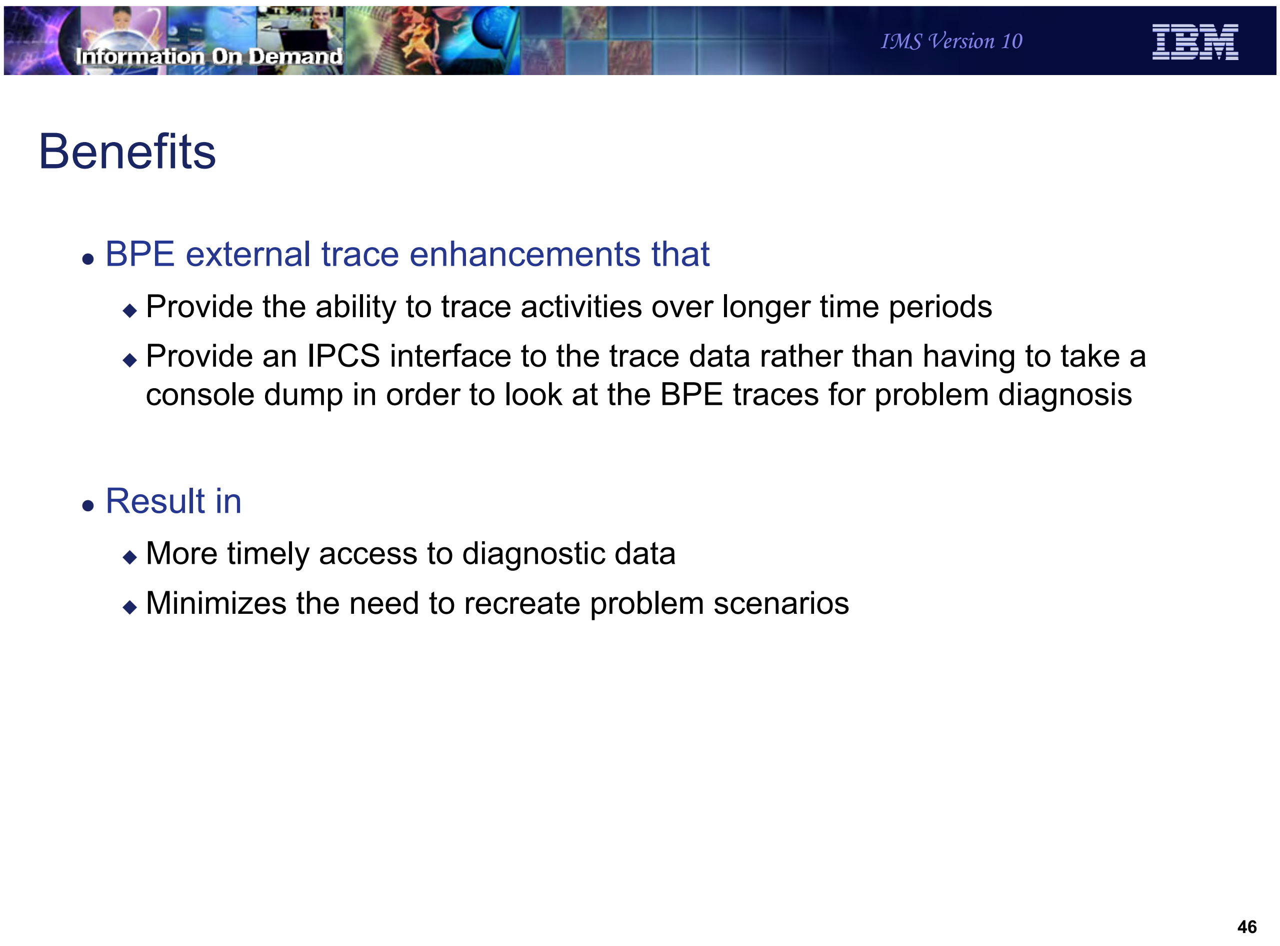
Figure 2-44. Performance Impacts CMA01.0

Notes:

Performance is always a consideration especially when tracing is activated. The BPE external trace function is designed to minimize the performance overhead incurred at trace points in the mainline code. Running with external tracing enabled results in an approximate cost of 15 extra instructions per trace point compared to running with incore tracing.

The majority of the processing required to externalize BPE trace data occurs under a new TCB in the BPE address space. This TCB is separate from any existing BPE or IMS component TCB. Work under existing TCBs, therefore, is not directly impacted by work performed by the new external trace TCB. The processing under the external trace TCB is more likely to be I/O bound than CPU bound.

If the writing of the external trace records cannot keep up with the rate of the generation of the records, external trace data will be lost ("skipped"), rather than delaying mainline processing. The goal is to make running with BPE external trace enabled minimally disruptive to system performance.


Information On Demand
IMS Version 10
IBM

Benefits

- BPE external trace enhancements that
  - Provide the ability to trace activities over longer time periods
  - Provide an IPCS interface to the trace data rather than having to take a console dump in order to look at the BPE traces for problem diagnosis
- Result in
  - More timely access to diagnostic data
  - Minimizes the need to recreate problem scenarios

Figure 2-45. Benefits CMA01.0

Notes:

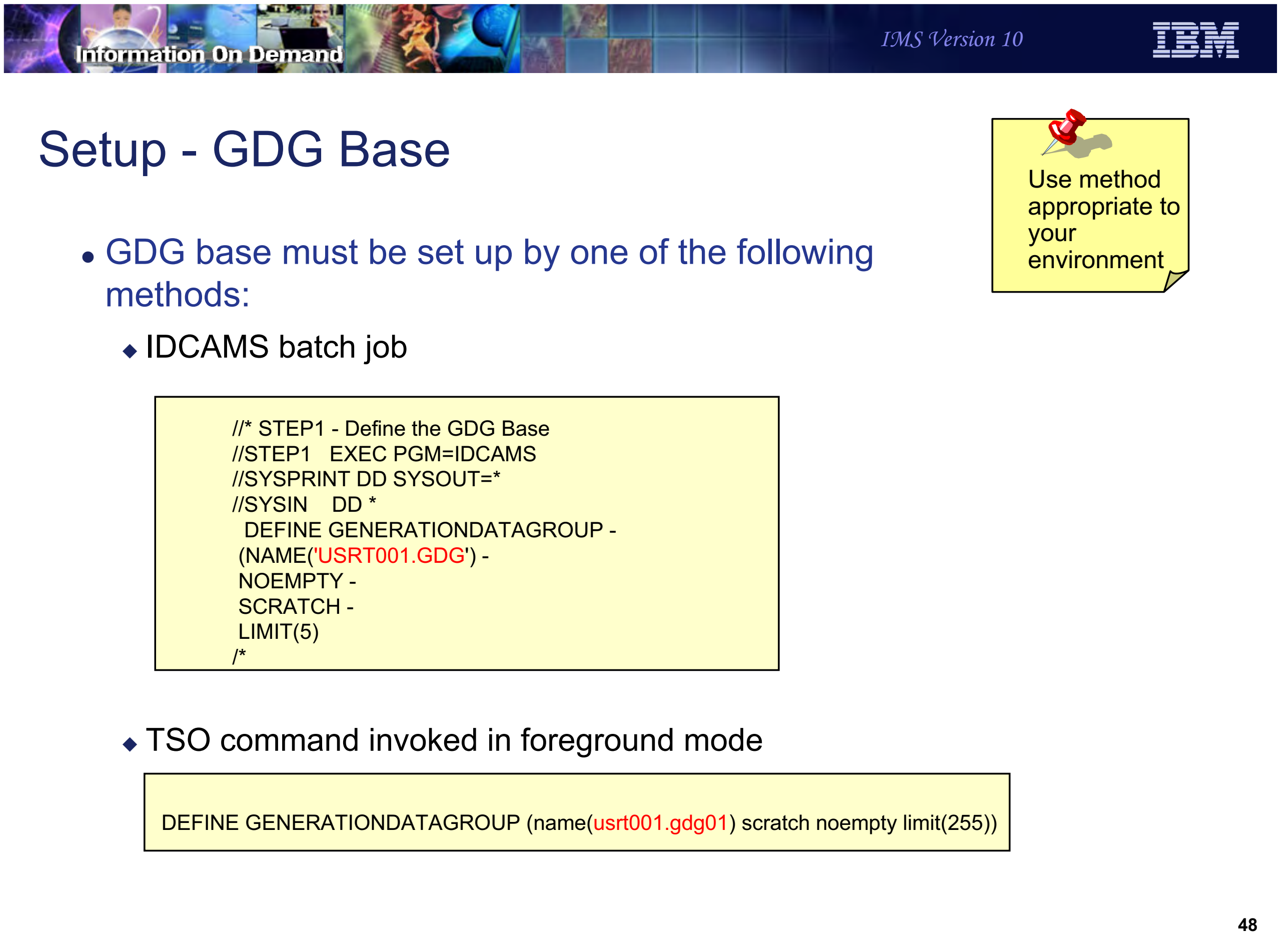
Information On Demand
IMS Version 10

BPE External Trace Appendix

- This appendix for BPE External Trace includes:
  - GDG set up instructions
  - External trace formatting and printing documentation
    - IPCS examples
  - New messages

Figure 2-46. BPE External Trace Appendix CMA01.0

Notes:



Information On Demand

IMS Version 10

IBM

Setup - GDG Base

Use method appropriate to your environment

- GDG base must be set up by one of the following methods:
  - IDCAMS batch job

```
//* STEP1 - Define the GDG Base
//STEP1   EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  DEFINE GENERATIONDATAGROUP -
 (NAME('USRT001.GDG') -
 NOEMPTY -
 SCRATCH -
 LIMIT(5)
/*
```

  - TSO command invoked in foreground mode

```
DEFINE GENERATIONDATAGROUP (name(usrt001.gdg01) scratch noempty limit(255))
```

Figure 2-47. Setup - GDG Base CMA01.0

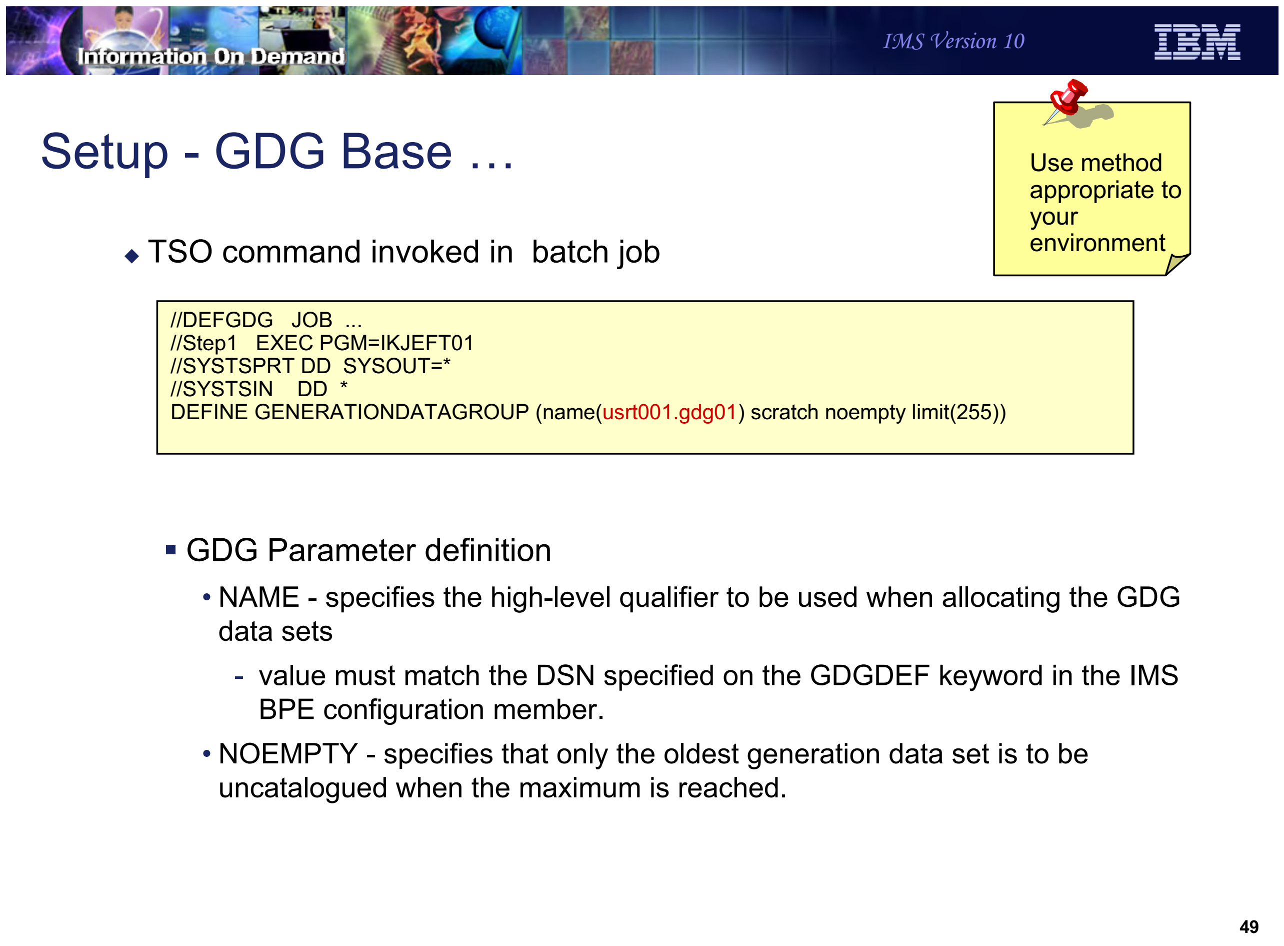
Notes:

Users must define the GDG base prior to using BPE external trace functionality. This definition can be performed in any of the following ways:

1) IDCAMS batch job

2) TSO command invoked in the foreground

3) TSO command invoked in a batch job



Setup - GDG Base …

Use method appropriate to your environment

- TSO command invoked in batch job

```
//DEFGDG   JOB  ...
//Step1   EXEC PGM=IKJEFT01
//SYSTSPRT DD  SYSOUT=*
//SYSTSIN    DD  *
DEFINE GENERATIONDATAGROUP (name(usrt001.gdg01) scratch noempty limit(255))
```

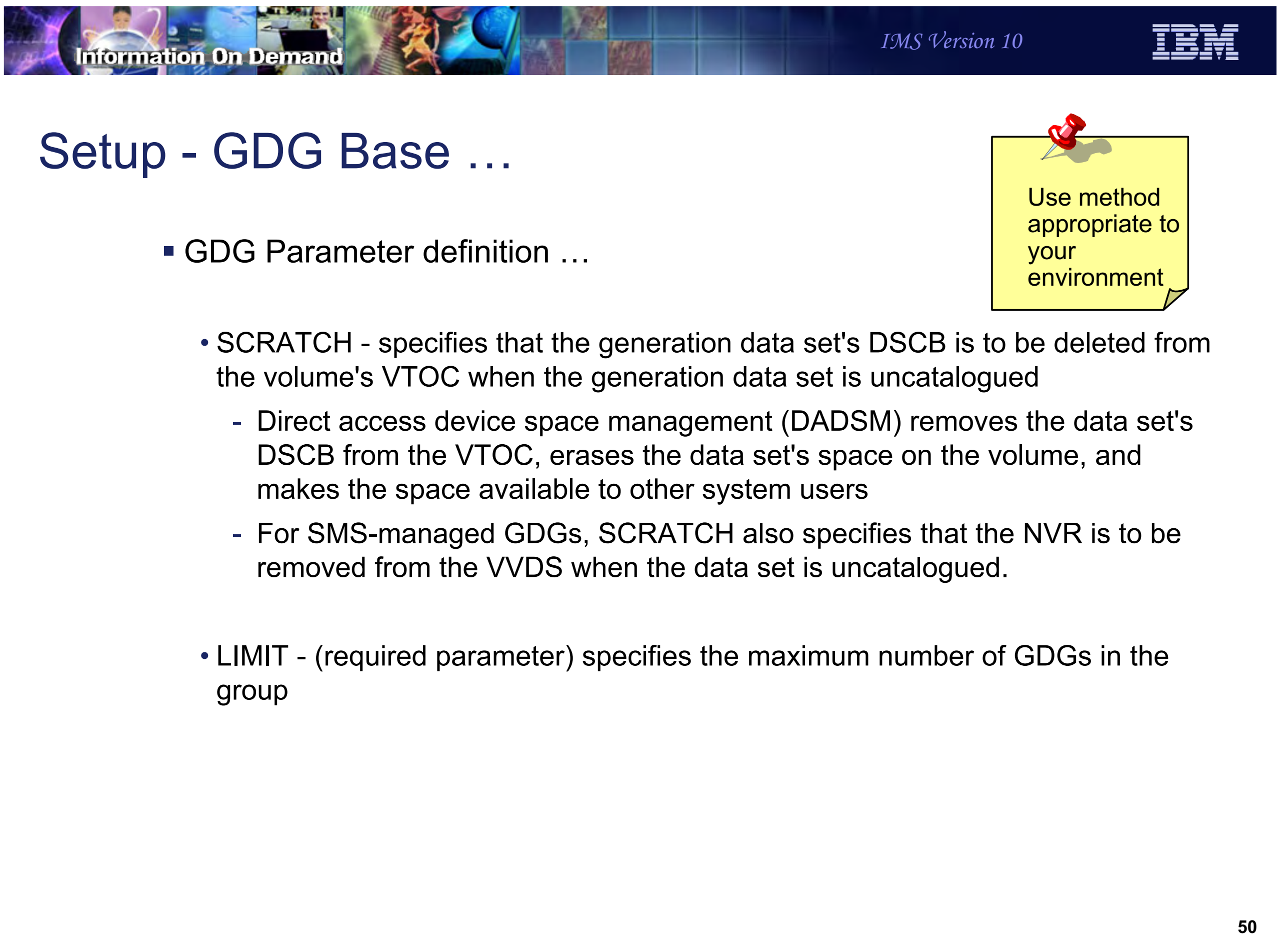
- GDG Parameter definition
  - NAME - specifies the high-level qualifier to be used when allocating the GDG data sets
    - value must match the DSN specified on the GDGDEF keyword in the IMS BPE configuration member.
  - NOEMPTY - specifies that only the oldest generation data set is to be uncatalogued when the maximum is reached.

Figure 2-48. Setup - GDG Base … CMA01.0

Notes:

Regardless of which method is chosen, the GDG parameter definition must specify a value for NAME which is the high level qualifier that will be used when allocating the DGD data sets. This value must match the value in the DSN parameter of the EXTTRACE keyword definition in the BPE configuration member.

The specification of NOEMPTY provides for the uncataloguing of the oldest GDG data set when the maximum number is reached.


Information On Demand
IMS Version 10
IBM

Setup - GDG Base …

Use method appropriate to your environment

- GDG Parameter definition …
  - SCRATCH - specifies that the generation data set's DSCB is to be deleted from the volume's VTOC when the generation data set is uncatalogued
    - Direct access device space management (DADSM) removes the data set's DSCB from the VTOC, erases the data set's space on the volume, and makes the space available to other system users
    - For SMS-managed GDGs, SCRATCH also specifies that the NVR is to be removed from the VVDS when the data set is uncatalogued.
  - LIMIT - (required parameter) specifies the maximum number of GDGs in the group

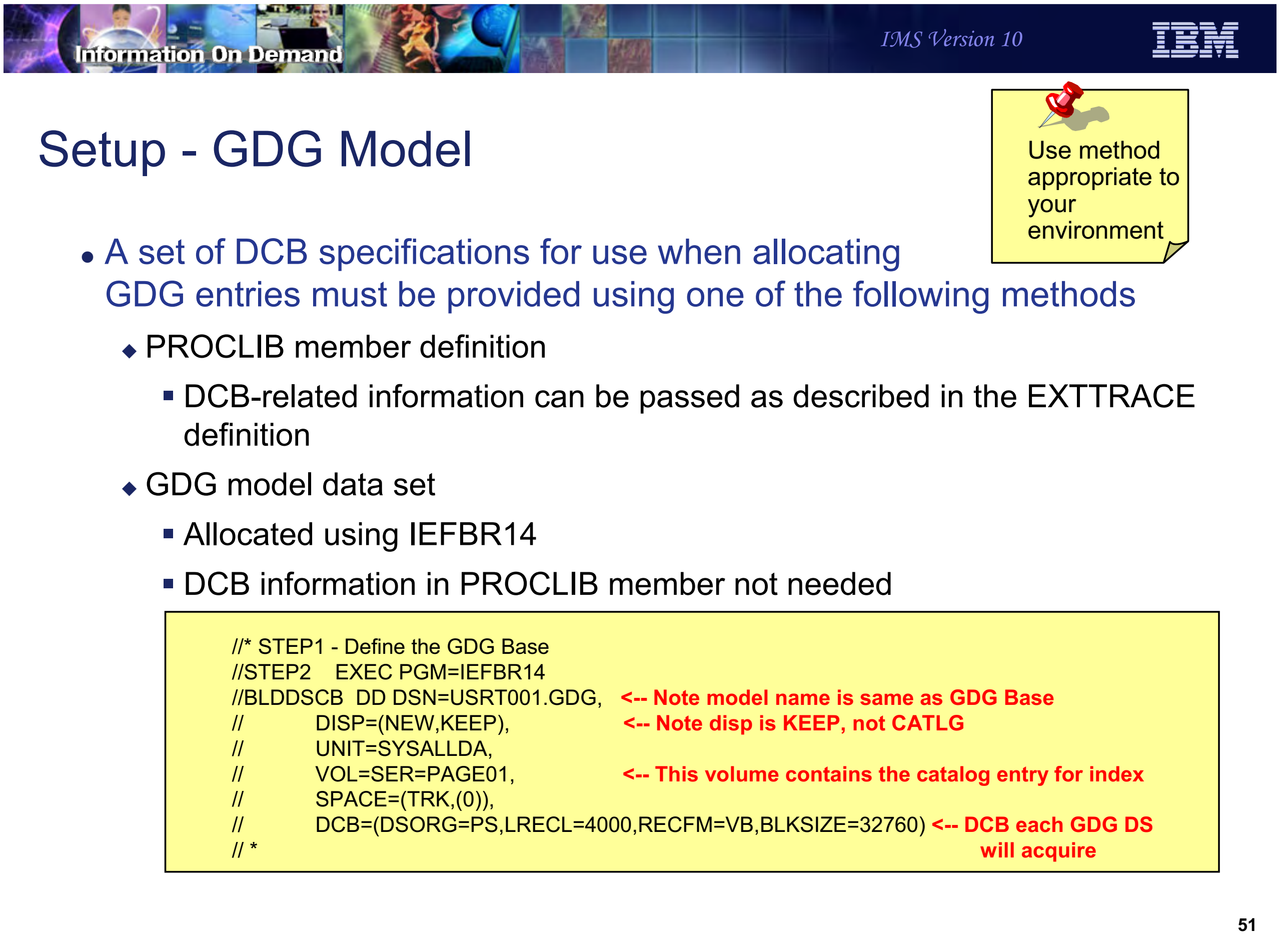
50

Figure 2-49. Setup - GDG Base … CMA01.0

Notes:

SCRATCH specifies that the generation data set's DSCB is to be deleted from the volume's VTOC when the generation data set is uncatalogued. Direct access device space management (DADSM) removes the data set's DSCB from the VTOC, erases the data set's space on the volume, and makes the space available to other system users. The generation data set ceases to exist. For SMS-managed GDSs, SCRATCH also specifies that the NVR is to be removed from the VVDS when the data set is uncatalogued.

LIMIT is a required parameter and specifies the maximum number of GDGs in the group. In the example on the previous visuals, this value is 255.



Information On Demand

IMS Version 10

Setup - GDG Model

Use method appropriate to your environment

- A set of DCB specifications for use when allocating GDG entries must be provided using one of the following methods
  - PROCLIB member definition
    - DCB-related information can be passed as described in the EXTTRACE definition
  - GDG model data set
    - Allocated using IEFBR14
    - DCB information in PROCLIB member not needed

```
//* STEP1 - Define the GDG Base
//STEP2    EXEC PGM=IEFBR14
//BLDDSCB  DD DSN=USRT001.GDG,   <-- Note model name is same as GDG Base
//            DISP=(NEW,KEEP),   <-- Note disp is KEEP, not CATLG
//            UNIT=SYSALLDA,
//            VOL=SER=PAGE01,    <-- This volume contains the catalog entry for index
//            SPACE=(TRK,(0)),
//            DCB=(DSORG=PS,LRECL=4000,RECFM=VB,BLKSIZE=32760) <-- DCB each GDG DS
// *                                                                 will acquire
```

51

Figure 2-50. Setup - GDG Model CMA01.0

Notes:

As with all GDGs, DCB information must be provided for use when the GDG entries are created. The DCB can be provided via one of the following methods:

- PROCLIB member definition - the DCB-related information can be passed in the EXTTRACE definition
- GDG model data set - a GDG model data set can be allocated using IEFBR14 and put on the same volume as the user catalog and GDG data sets. With a GDG model, providing the DCB information in EXTTRACE is not required. The visual gives an example of the IEFBR14 JCL.

GDG Data Set Allocation/Deallocation Flow

- BPE external trace data sets are allocated and opened:
  - When the first trace table is specified with external option
    - Either at BPE initialization
      - EXTERNAL=YES is coded on the TRCLEV statement
    - Or, when the first table is updated with EXTERNAL(YES) on the UPD TRTAB command

- Data sets are closed and deallocated:
  - When the last trace table is stopped
    - E.g., UPD TRTAB EXTERNAL(NO) command.
  - Or, when they are filled and the next data set in the generation is subsequently opened for write access

Figure 2-51. GDG Data Set Allocation/Deallocation Flow CMA01.0

Notes:

BPE external trace data sets are allocated and opened in one of two situations: during BPE initialization if the TRCLEV statement specifies EXTERNAL=YES, or as a result of the UPD TRTAB command and a specification of EXTERNAL(YES).

The trace data sets are closed and deallocated when the last trace table is stopped by a specification of EXTERNAL(NO) in the UPD TRTAB EXTERNAL command, or when the data sets are filled and the next data set in the generation is subsequently opened for write access.

Data Formatting and Printing

- BPE external trace data formatting options
  - Offline, through IPCS exits and formatting routines
  - Batch processing

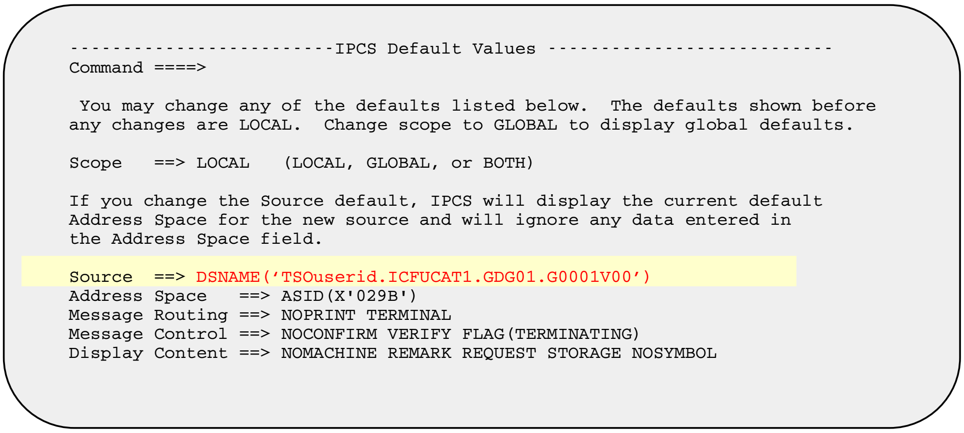
Figure 2-52. Data Formatting and Printing CMA01.0

Notes:

There are two modes available to format and print the external trace data. The first is through IPCS routines. The second is through batch processing.

Data Formatting and Printing - IPCS

- IPCS
  - From the IPCS main menu, select option 0 to specify the GDG used
  - The selection brings up the IPCS Default Values panel, hit enter



```
------------------------IPCS Default Values --------------------------
Command ====>

 You may change any of the defaults listed below.  The defaults shown before
any changes are LOCAL.  Change scope to GLOBAL to display global defaults.

Scope    ==> LOCAL    (LOCAL, GLOBAL, or BOTH)

If you change the Source default, IPCS will display the current default
Address Space for the new source and will ignore any data entered in
the Address Space field.

Source   ==> DSNAME('TSOuserid.ICFUCAT1.GDG01.G0001V00')
Address Space    ==> ASID(X'029B')
Message Routing ==> NOPRINT TERMINAL
Message Control ==> NOCONFIRM VERIFY FLAG(TERMINATING)
Display Content ==> NOMACHINE REMARK REQUEST STORAGE NOSYMBOL
```

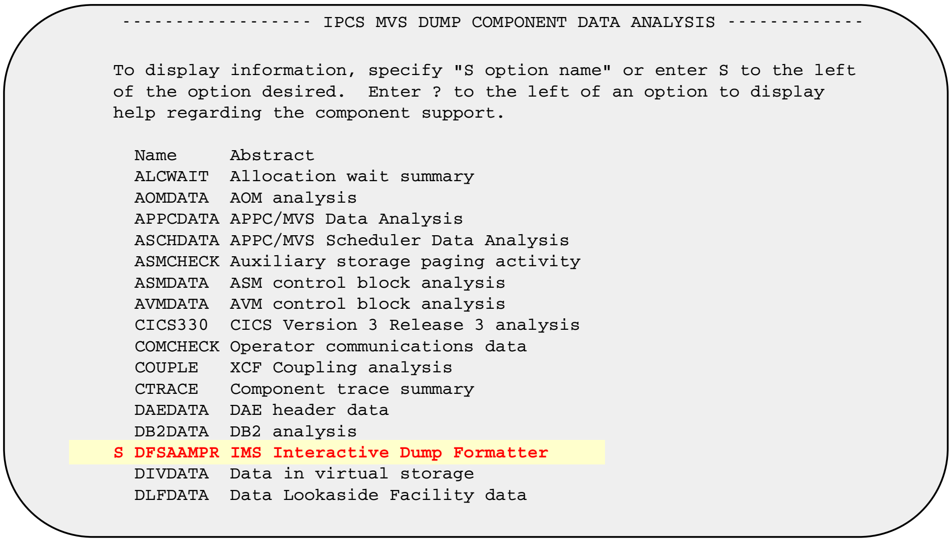
Figure 2-53. Data Formatting and Printing - IPCS CMA01.0

Notes:

This and the next several visuals give an example of using IPCS to format and print the external data set trace records.

Data Formatting and Printing - IPCS …

- The IPCS MVS Dump Component Data Analysis panel shows the available options for analysis
- Select DFSAAMPR - IMS Interactive Dump Formatter



```
 ------------------ IPCS MVS DUMP COMPONENT DATA ANALYSIS ------------

To display information, specify "S option name" or enter S to the left
of the option desired.  Enter ? to the left of an option to display
help regarding the component support.

  Name     Abstract
  ALCWAIT  Allocation wait summary
  AOMDATA  AOM analysis
  APPCDATA APPC/MVS Data Analysis
  ASCHDATA APPC/MVS Scheduler Data Analysis
  ASMCHECK Auxiliary storage paging activity
  ASMDATA  ASM control block analysis
  AVMDATA  AVM control block analysis
  CICS330  CICS Version 3 Release 3 analysis
  COMCHECK Operator communications data
  COUPLE   XCF Coupling analysis
  CTRACE   Component trace summary
  DAEDATA  DAE header data
  DB2DATA  DB2 analysis
S DFSAAMPR IMS Interactive Dump Formatter
  DIVDATA  Data in virtual storage
  DLFDATA  Data Lookaside Facility data
```

Figure 2-54. Data Formatting and Printing - IPCS … CMA01.0

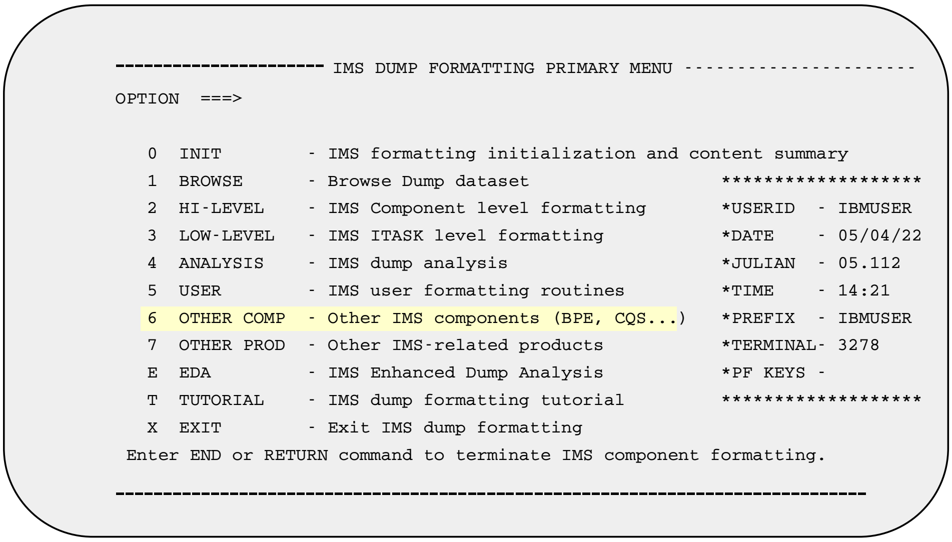
Notes:

The IMS Interactive Dump Formatter is named DFSAAMPR.

Information On Demand IMS Version 10 IBM

Data Formatting and Printing - IPCS …

- Select Option 6



```
---------------------- IMS DUMP FORMATTING PRIMARY MENU ---------------------
OPTION  ===>

   0  INIT        - IMS formatting initialization and content summary
   1  BROWSE      - Browse Dump dataset                 *******************
   2  HI-LEVEL    - IMS Component level formatting      *USERID   - IBMUSER
   3  LOW-LEVEL   - IMS ITASK level formatting          *DATE     - 05/04/22
   4  ANALYSIS    - IMS dump analysis                   *JULIAN   - 05.112
   5  USER        - IMS user formatting routines        *TIME     - 14:21
   6  OTHER COMP  - Other IMS components (BPE, CQS...)  *PREFIX   - IBMUSER
   7  OTHER PROD  - Other IMS-related products          *TERMINAL- 3278
   E  EDA         - IMS Enhanced Dump Analysis          *PF KEYS -
   T  TUTORIAL    - IMS dump formatting tutorial        *******************
   X  EXIT        - Exit IMS dump formatting
 Enter END or RETURN command to terminate IMS component formatting.

-----------------------------------------------------------------------
```

56

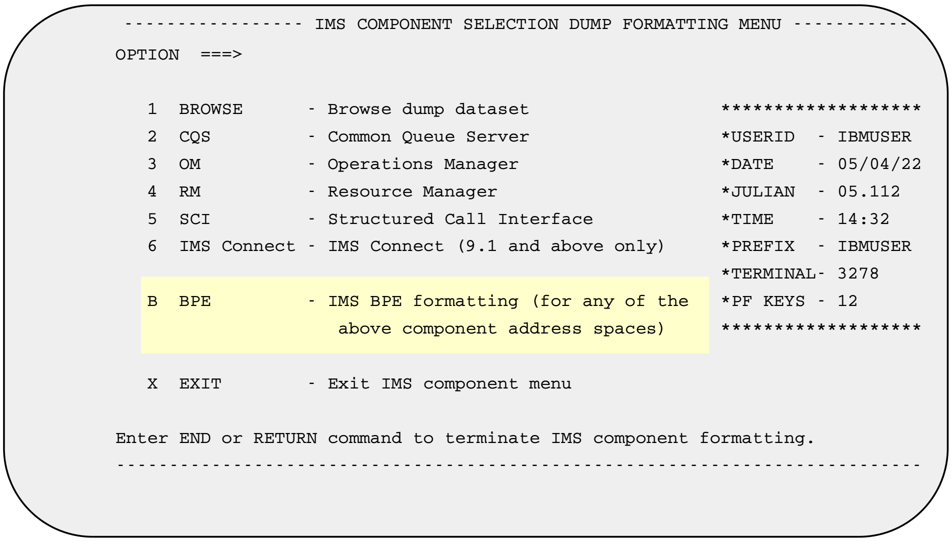
Figure 2-55. Data Formatting and Printing - IPCS … CMA01.0

Notes:

The IMS Dump Formatting Primary Menu shows the available IMS dump formatting options. Select Option 6 - ‘OTHER COMP’.

Data Formatting and Printing - IPCS …

- Select Option B (BPE)



```
 ---------------- IMS COMPONENT SELECTION DUMP FORMATTING MENU ----------
OPTION  ===>

   1  BROWSE      - Browse dump dataset                 *******************
   2  CQS         - Common Queue Server                 *USERID  - IBMUSER
   3  OM          - Operations Manager                  *DATE    - 05/04/22
   4  RM          - Resource Manager                    *JULIAN  - 05.112
   5  SCI         - Structured Call Interface           *TIME    - 14:32
   6  IMS Connect - IMS Connect (9.1 and above only)    *PREFIX  - IBMUSER
                                                        *TERMINAL- 3278
   B  BPE         - IMS BPE formatting (for any of the  *PF KEYS - 12
                     above component address spaces)    *******************

   X  EXIT        - Exit IMS component menu

Enter END or RETURN command to terminate IMS component formatting.
----------------------------------------------------------------------------
```

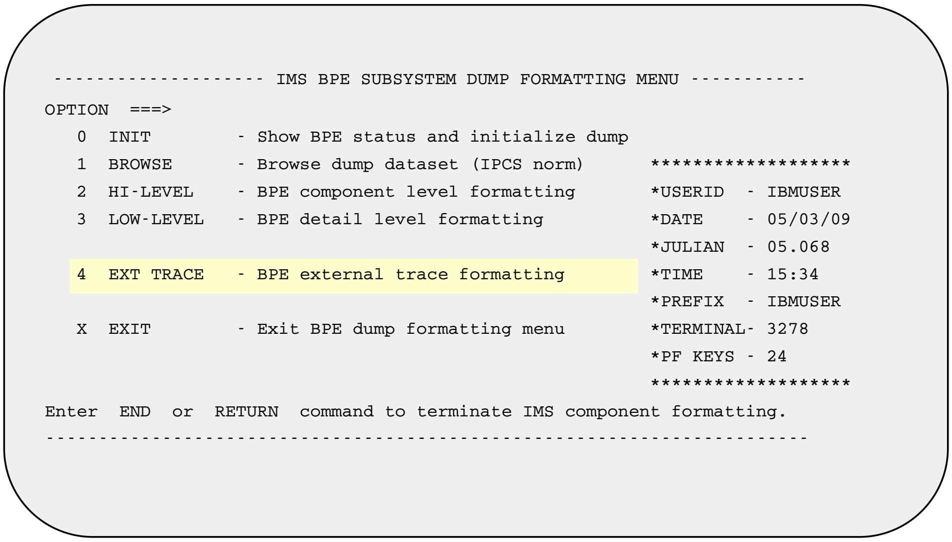
Figure 2-56. Data Formatting and Printing - IPCS … CMA01.0

Notes:

The next panel identifies IMS BPE Formatting. Select B to access this formatter. Alternatively, each of the components can be chosen separately.

Data Formatting and Printing - IPCS …

- Select Option 4 for the BPE external trace formatting



```
 ------------------- IMS BPE SUBSYSTEM DUMP FORMATTING MENU ----------
OPTION  ===>
   0  INIT        - Show BPE status and initialize dump
   1  BROWSE      - Browse dump dataset (IPCS norm)      ******************
   2  HI-LEVEL    - BPE component level formatting       *USERID  - IBMUSER
   3  LOW-LEVEL   - BPE detail level formatting          *DATE    - 05/03/09
                                                         *JULIAN  - 05.068
   4  EXT TRACE   - BPE external trace formatting        *TIME    - 15:34
                                                         *PREFIX  - IBMUSER
   X  EXIT        - Exit BPE dump formatting menu        *TERMINAL- 3278
                                                         *PF KEYS - 24
                                                         ******************
Enter  END  or  RETURN  command to terminate IMS component formatting.
----------------------------------------------------------------------
```

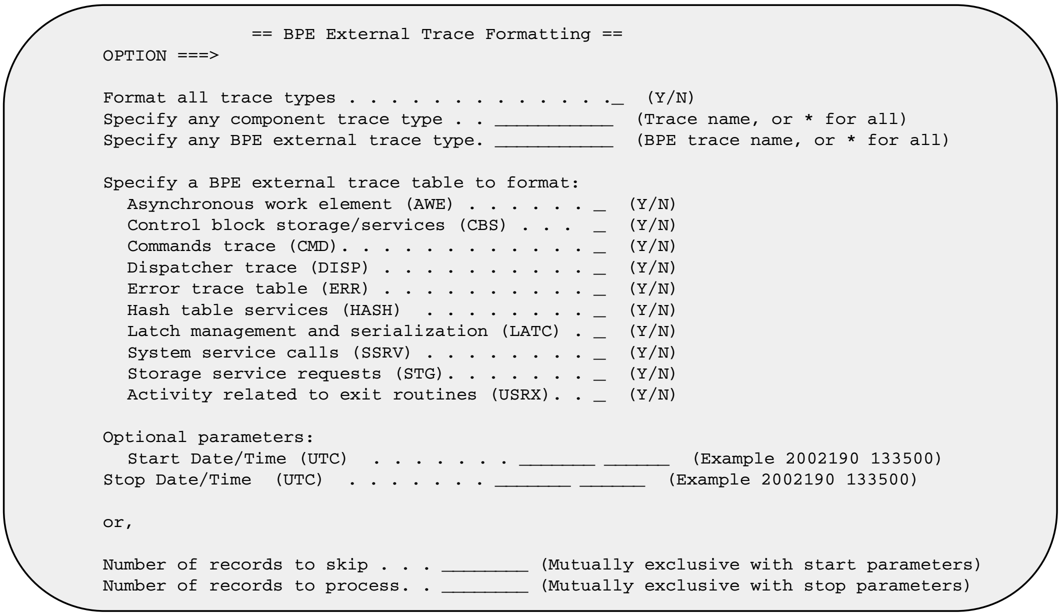
Figure 2-57. Data Formatting and Printing - IPCS … CMA01.0

Notes:

Option 4 provides the access to the new BPE External Trace formatting capability.

Data Formatting and Printing - IPCS …

- Select the traces to view



```
                 == BPE External Trace Formatting ==
OPTION ===>

Format all trace types . . . . . . . . . . . . ._  (Y/N)
Specify any component trace type . . ___________  (Trace name, or * for all)
Specify any BPE external trace type. ___________  (BPE trace name, or * for all)

Specify a BPE external trace table to format:
  Asynchronous work element (AWE) . . . . . . . _  (Y/N)
  Control block storage/services (CBS) . . .  _  (Y/N)
  Commands trace (CMD). . . . . . . . . . . . . _  (Y/N)
  Dispatcher trace (DISP) . . . . . . . . . . . _  (Y/N)
  Error trace table (ERR) . . . . . . . . . . . _  (Y/N)
  Hash table services (HASH)  . . . . . . . . . _  (Y/N)
  Latch management and serialization (LATC) . _  (Y/N)
  System service calls (SSRV) . . . . . . . . . _  (Y/N)
  Storage service requests (STG). . . . . . . . _  (Y/N)
  Activity related to exit routines (USRX). . _  (Y/N)

Optional parameters:
  Start Date/Time (UTC)  . . . . . . . _______ ______  (Example 2002190 133500)
Stop Date/Time  (UTC)  . . . . . . . _______ ______  (Example 2002190 133500)

or,

Number of records to skip . . . ________ (Mutually exclusive with start parameters)
Number of records to process. . ________ (Mutually exclusive with stop parameters)
```

Figure 2-58. Data Formatting and Printing - IPCS …

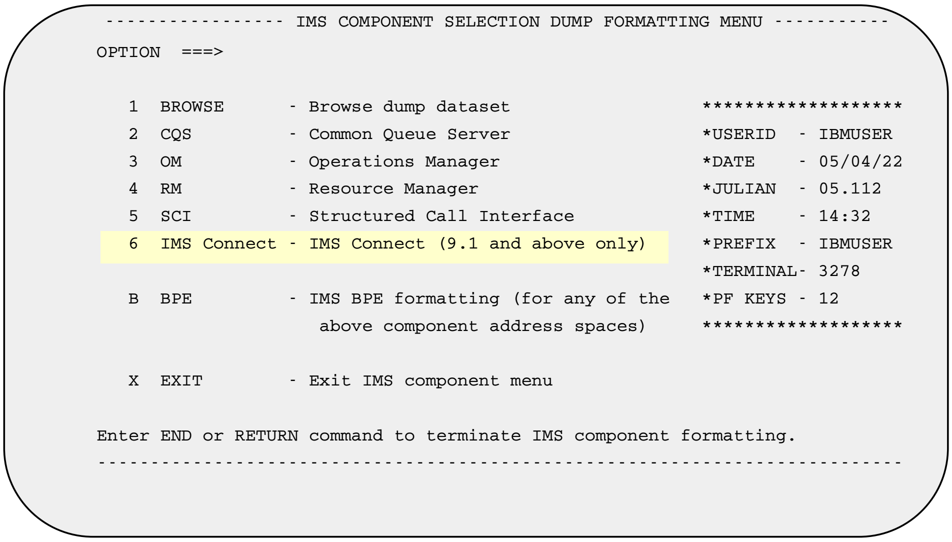
Notes:

The BPE External Trace Formatting panel provides the options to select how the trace is to be produced.

Information On Demand
IMS Version 10
IBM

Data Formatting and Printing - IPCS …

- Each of the components can also be chosen separately
  - For example, Option 6 will request IMS Connect formatting



```
 ----------------- IMS COMPONENT SELECTION DUMP FORMATTING MENU -----------
OPTION  ===>

   1  BROWSE      - Browse dump dataset                *******************
   2  CQS         - Common Queue Server                *USERID  - IBMUSER
   3  OM          - Operations Manager                 *DATE    - 05/04/22
   4  RM          - Resource Manager                   *JULIAN  - 05.112
   5  SCI         - Structured Call Interface          *TIME    - 14:32
   6  IMS Connect - IMS Connect (9.1 and above only)   *PREFIX  - IBMUSER
                                                       *TERMINAL- 3278
   B  BPE         - IMS BPE formatting (for any of the *PF KEYS - 12
                     above component address spaces)   *******************

   X  EXIT        - Exit IMS component menu

Enter END or RETURN command to terminate IMS component formatting.
----------------------------------------------------------------------------
```

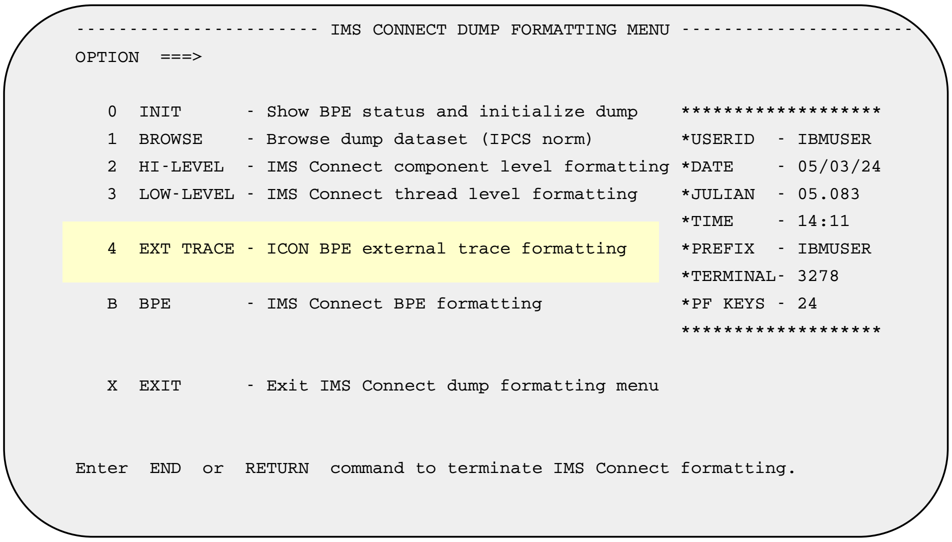
Figure 2-59. Data Formatting and Printing - IPCS … CMA01.0

Notes:

IPCS also supports accessing each component separately. This next example will show the panels associated with IMS Connect.

Data Formatting and Printing - IPCS …

- Select Option 4 for External Trace formatting



```
----------------------- IMS CONNECT DUMP FORMATTING MENU ----------------------
OPTION  ===>

   0  INIT      - Show BPE status and initialize dump    *******************
   1  BROWSE    - Browse dump dataset (IPCS norm)        *USERID   - IBMUSER
   2  HI-LEVEL  - IMS Connect component level formatting *DATE     - 05/03/24
   3  LOW-LEVEL - IMS Connect thread level formatting    *JULIAN   - 05.083
                                                         *TIME     - 14:11
   4  EXT TRACE - ICON BPE external trace formatting     *PREFIX   - IBMUSER
                                                         *TERMINAL- 3278
   B  BPE       - IMS Connect BPE formatting             *PF KEYS  - 24
                                                         *******************

   X  EXIT      - Exit IMS Connect dump formatting menu


Enter  END  or  RETURN  command to terminate IMS Connect formatting.
```

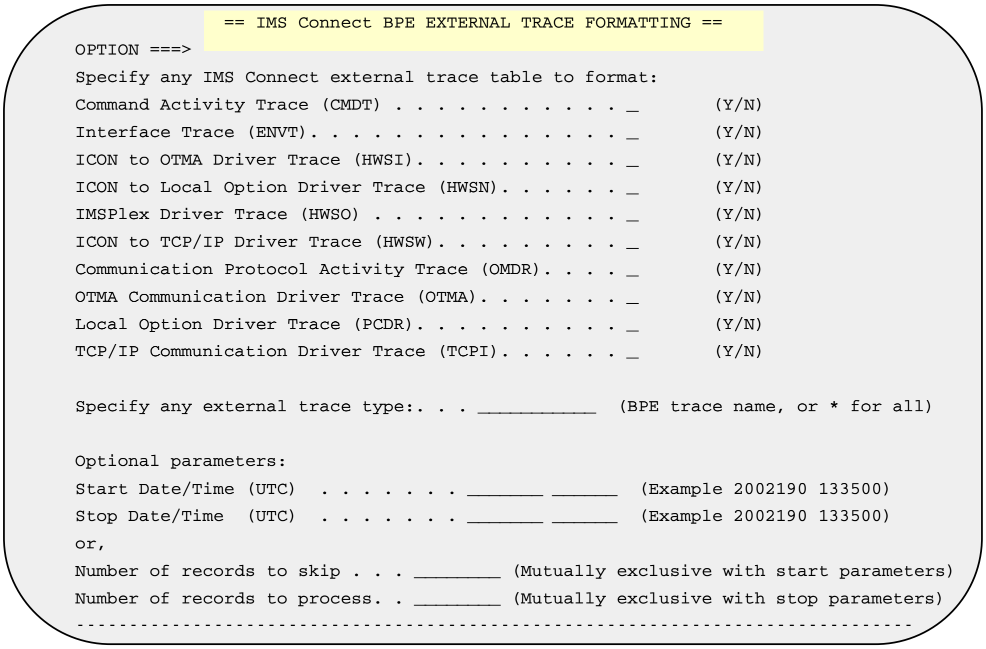
Figure 2-60. Data Formatting and Printing - IPCS … CMA01.0

Notes:

The IMS Connect Dump Formatting panel provides several options. Option 4 is new and shows how to access the BPE external trace formatting for IMS Connect.

Data Formatting and Printing - IPCS …

- New panel



```
              == IMS Connect BPE EXTERNAL TRACE FORMATTING ==
OPTION ===>
Specify any IMS Connect external trace table to format:
Command Activity Trace (CMDT) . . . . . . . . . . . _       (Y/N)
Interface Trace (ENVT). . . . . . . . . . . . . . . _       (Y/N)
ICON to OTMA Driver Trace (HWSI). . . . . . . . . . _       (Y/N)
ICON to Local Option Driver Trace (HWSN). . . . . . _       (Y/N)
IMSPlex Driver Trace (HWSO) . . . . . . . . . . . . _       (Y/N)
ICON to TCP/IP Driver Trace (HWSW). . . . . . . . . _       (Y/N)
Communication Protocol Activity Trace (OMDR). . . . _       (Y/N)
OTMA Communication Driver Trace (OTMA). . . . . . . _       (Y/N)
Local Option Driver Trace (PCDR). . . . . . . . . . _       (Y/N)
TCP/IP Communication Driver Trace (TCPI). . . . . . _       (Y/N)

Specify any external trace type:. . . ___________  (BPE trace name, or * for all)

Optional parameters:
Start Date/Time (UTC)  . . . . . . . _______ ______  (Example 2002190 133500)
Stop Date/Time  (UTC)  . . . . . . . _______ ______  (Example 2002190 133500)
or,
Number of records to skip . . . ________ (Mutually exclusive with start parameters)
Number of records to process. . ________ (Mutually exclusive with stop parameters)
-------------------------------------------------------------------------------
```

Figure 2-61. Data Formatting and Printing - IPCS … CMA01.0

Notes:

The visual shows the new panel for IMS Connect. Note that the other components, e.g., CQS, RM, OM and SCI also have panels that relate specifically to each of those functions.

Data Formatting and Printing - Batch mode

- An alternative to IPCS is Batch mode
  - BPE external trace records can be formatted in batch mode by using JCL similar to the following:

```
 //USER001 JOB (USER001,A421,090,M31),'USER001',
//                             CLASS=B,TIME=(,15),USER=USER001,
//                             MSGLEVEL=(1,1),MSGCLASS=H
//JOBLIB        DD DSNAME=IMSBLD.I10ATS17.CRESLIB,DISP=SHR
//                    DD DSNAME=SYS1.MIGLIB,DISP=SHR
//IPCSDMP  EXEC PGM=IKJEFT01,REGION=4096K
//SYSTSPRT  DD SYSOUT=*
//IPCSPRNT   DD SYSOUT=*
//OUTFILE      DD SYSOUT=*
//INDEX          DD SYSOUT=*
//SYSABEND  DD SYSOUT=*
//SYSTSIN      DD *
IPCSDDIR 'USER001.DUMP.DIR.DD'
ALLOC FILE(IPCSDDIR) +
  DA('USER001.DUMP.DIR.DD') +
  REUSE SHR
ALLOC FILE(INFILE) +
  DA('N900380.BPETRACE.ALL') +
  REUSE SHR
IPCS NOPARM
VERBX BPETRFM0 'COMP(ALL) TRACE(TYPE(ALL))‘
END
```

Figure 2-62. Data Formatting and Printing - Batch mode CMA01.0

Notes:

Batch invocation is an alternative to IPCS. This visual gives an example of how to run a batch job for data formatting and printing.

Information On Demand
IMS Version 10
IBM

Data Formatting and Printing - Batch mode …

- VERBX - keyword definition that defines the formatting request

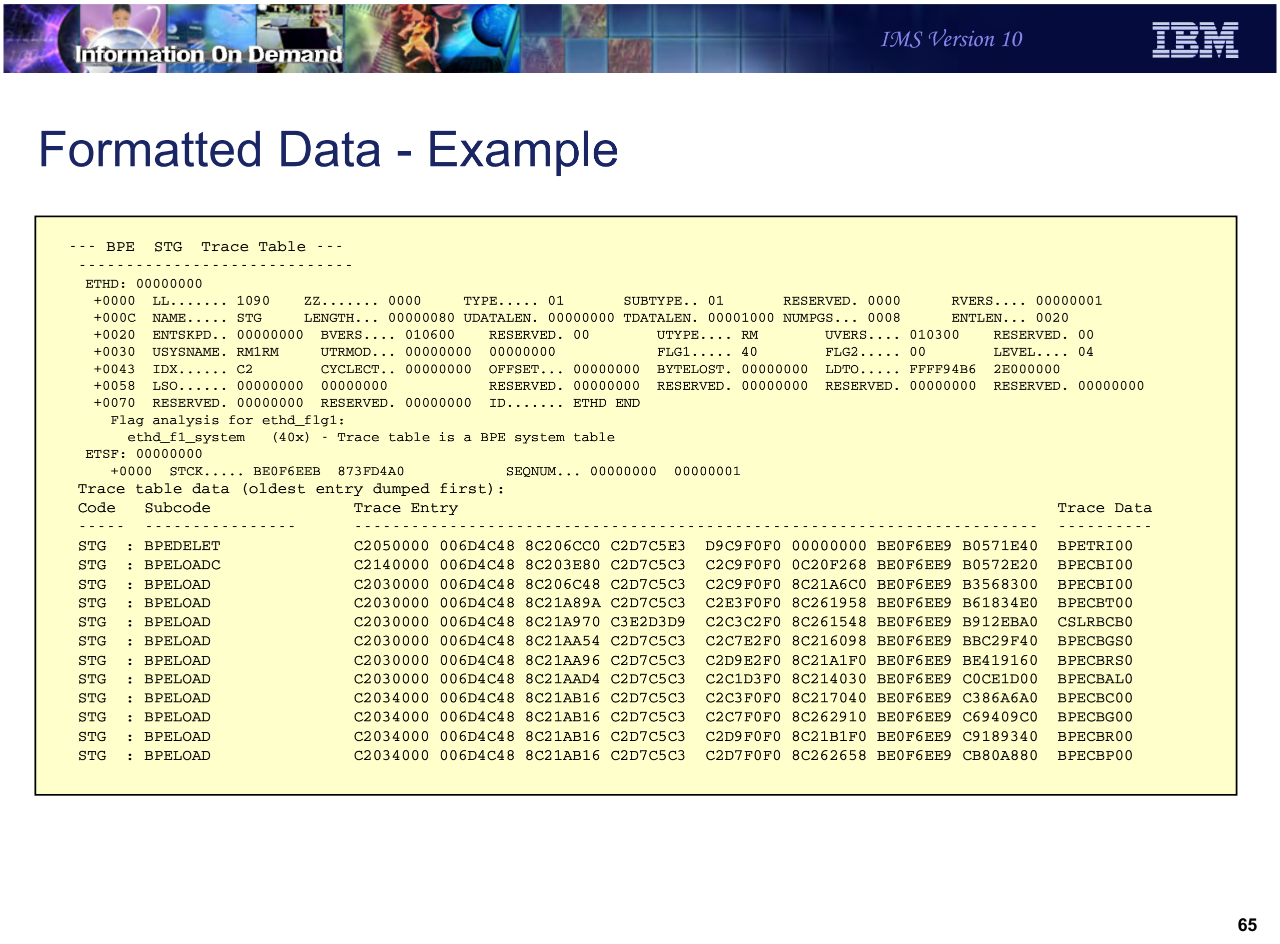
Keyword	Description
COMP()	- (required) specify the component type to format {ALL,HWS,CQS,RM,OM,SCI }
TRACE()	- (required, may repeat) specify the trace type to format. Example: TRACE(TYPE(AWE)) TRACE(TYPE(ERR)) ...
SDATE(), STIME(), EDATE(), ETIME()	- (optional) specify filtering for start and end date, time format: YYYYDDD and HHMMSS
UL()	- whether the filtering time is UTC or Local; value = { U, L }
CSTCK()	- determines whether to print the STCK at the end of each entry in the original HEX or formatted value = { Y, N }

- Example:
  - **VERBX BPETRFM0 'COMP(**ALL**) TRACE(TYPE(**ALL**)) SDATE(**2006001**) STIME(**120000**) EDATE(**2006100**) ETIME(**120000**) UL(**U**) CSTCK(**N**)'**
    - Format trace records for all component and for all trace type between 1st day of 2006, 12pm and 100-th day of 2006, 12pm‘
  - **VERBX BPETRFM0 'COMP(**CQS**) TRACE(TYPE(**ERR**)) TRACE(TYPE(**INTF**))'**
    - Format trace records for only CQS component with trace type ERR or INTF

Figure 2-63. Data Formatting and Printing - Batch mode … CMA01.0

Notes:

The VERBX keyword provides the definitions that request the type of formatting to be done along with filtering requirements.



Formatted Data - Example

```
--- BPE  STG  Trace Table ---
 ----------------------------
  ETHD: 00000000
   +0000  LL....... 1090    ZZ....... 0000     TYPE..... 01       SUBTYPE.. 01       RESERVED. 0000      RVERS.... 00000001
   +000C  NAME..... STG     LENGTH... 00000080 UDATALEN. 00000000 TDATALEN. 00001000 NUMPGS... 0008      ENTLEN... 0020
   +0020  ENTSKPD.. 00000000  BVERS.... 010600    RESERVED. 00        UTYPE.... RM        UVERS.... 010300    RESERVED. 00
   +0030  USYSNAME. RM1RM     UTRMOD... 00000000  00000000            FLG1..... 40        FLG2..... 00        LEVEL.... 04
   +0043  IDX...... C2        CYCLECT.. 00000000  OFFSET... 00000000  BYTELOST. 00000000  LDTO..... FFFF94B6  2E000000
   +0058  LSO...... 00000000  00000000            RESERVED. 00000000  RESERVED. 00000000  RESERVED. 00000000  RESERVED. 00000000
   +0070  RESERVED. 00000000  RESERVED. 00000000  ID....... ETHD END
     Flag analysis for ethd_flg1:
       ethd_f1_system   (40x) - Trace table is a BPE system table
  ETSF: 00000000
     +0000  STCK..... BE0F6EEB  873FD4A0            SEQNUM... 00000000  00000001
 Trace table data (oldest entry dumped first):
 Code   Subcode                 Trace Entry                                                                Trace Data
 -----  ----------------        -------------------------------------------------------------------------  ----------
 STG  : BPEDELET                C2050000 006D4C48 8C206CC0 C2D7C5E3  D9C9F0F0 00000000 BE0F6EE9 B0571E40  BPETRI00
 STG  : BPELOADC                C2140000 006D4C48 8C203E80 C2D7C5C3  C2C9F0F0 0C20F268 BE0F6EE9 B0572E20  BPECBI00
 STG  : BPELOAD                 C2030000 006D4C48 8C206C48 C2D7C5C3  C2C9F0F0 8C21A6C0 BE0F6EE9 B3568300  BPECBI00
 STG  : BPELOAD                 C2030000 006D4C48 8C21A89A C2D7C5C3  C2E3F0F0 8C261958 BE0F6EE9 B61834E0  BPECBT00
 STG  : BPELOAD                 C2030000 006D4C48 8C21A970 C3E2D3D9  C2C3C2F0 8C261548 BE0F6EE9 B912EBA0  CSLRBCB0
 STG  : BPELOAD                 C2030000 006D4C48 8C21AA54 C2D7C5C3  C2C7E2F0 8C216098 BE0F6EE9 BBC29F40  BPECBGS0
 STG  : BPELOAD                 C2030000 006D4C48 8C21AA96 C2D7C5C3  C2D9E2F0 8C21A1F0 BE0F6EE9 BE419160  BPECBRS0
 STG  : BPELOAD                 C2030000 006D4C48 8C21AAD4 C2D7C5C3  C2C1D3F0 8C214030 BE0F6EE9 C0CE1D00  BPECBAL0
 STG  : BPELOAD                 C2034000 006D4C48 8C21AB16 C2D7C5C3  C2C3F0F0 8C217040 BE0F6EE9 C386A6A0  BPECBC00
 STG  : BPELOAD                 C2034000 006D4C48 8C21AB16 C2D7C5C3  C2C7F0F0 8C262910 BE0F6EE9 C69409C0  BPECBG00
 STG  : BPELOAD                 C2034000 006D4C48 8C21AB16 C2D7C5C3  C2D9F0F0 8C21B1F0 BE0F6EE9 C9189340  BPECBR00
 STG  : BPELOAD                 C2034000 006D4C48 8C21AB16 C2D7C5C3  C2D7F0F0 8C262658 BE0F6EE9 CB80A880  BPECBP00
```

Figure 2-64. Formatted Data - Example CMA01.0

Notes:

The formatted data is printed in a format similar to what is shown on this visual.

New Messages

- Several new message are added for BPE processing:
  - BPE0043E ERROR DURING EXTERNAL TRACE PROCESSING IN MODULE module
  - BPE0044I BPE EXTERNAL TRACE FUNCTION NOT ACTIVE
  - BPE0046I EXTERNAL TRACE DATA SET dsn OPENED ON VOL vol
  - BPE0046I EXTERNAL TRACE DATA SET dsn CLOSED ON VOL vol
  - BPE0047I ERROR OCCURRED DURING UPDATE TRACETABLE OPTION PROCESSING

Figure 2-65. New Messages CMA01.0

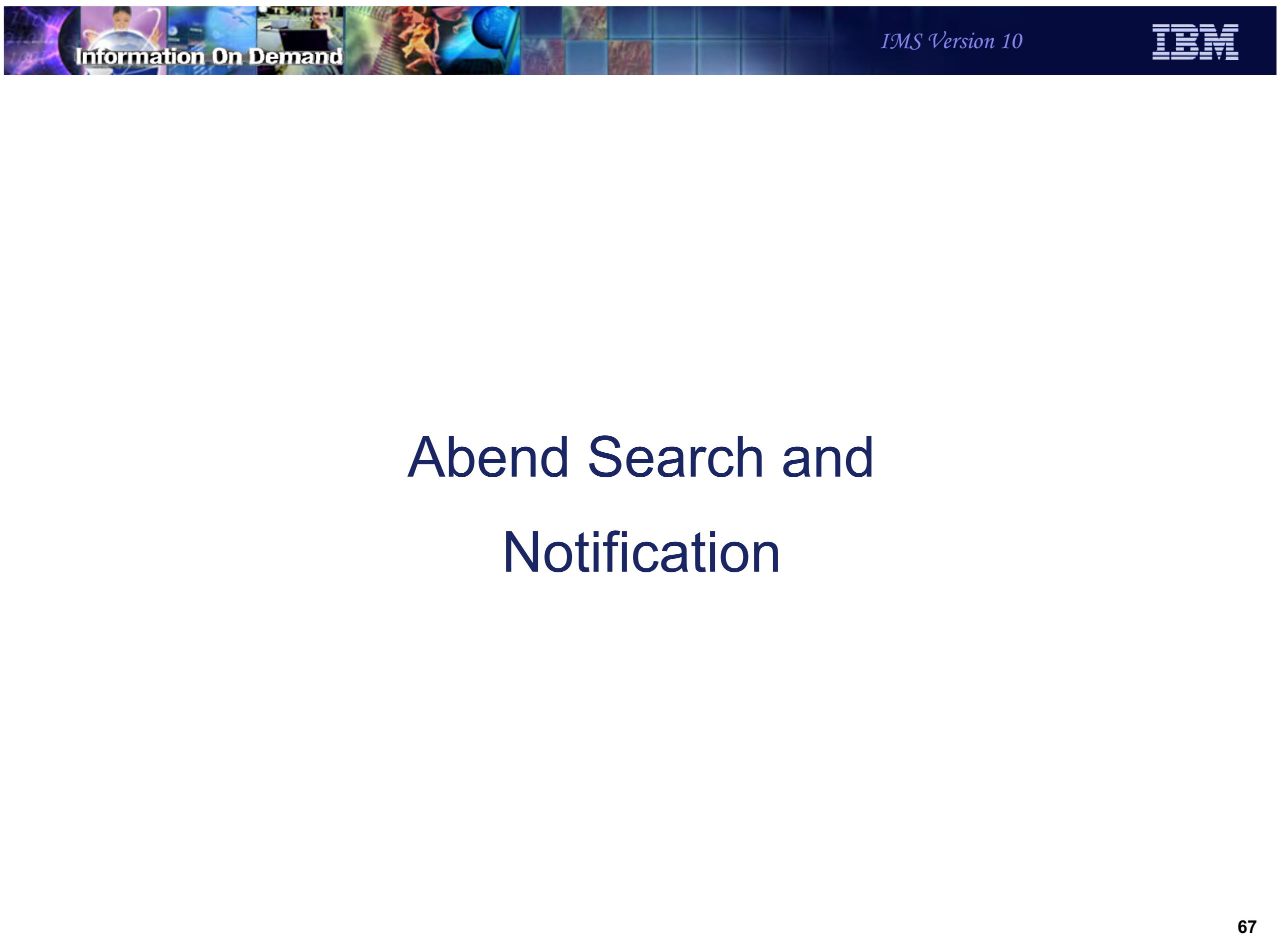
Notes:

If the BPE external trace function is enabled, an installation’s Operations Procedures documentation should be updated to reflect the possibility of several new error messages associated with external tracing. This visual documents the new messages.

If an error occurs during UPDATE TRACETABLE processing, the text of the message produces one of the following:

- UNABLE TO GET NECESSARY STORAGE
- ERROR READING BPE CONFIGURATION PROCLIB MEMBER member address space
- ERROR PARSING BPE CONFIGURATION PROCLIB MEMBER member processing
- UNKNOWN ERROR rc FROM PARSING MODULE BPEPCF10
- LOAD FAILED FOR BPEPCF10, BPELOADC RC=rc
- BPE CONFIGURATION PROCLIB MEMBER NAME WAS NOT CODED ON STARTUP PARMS

- PREVIOUS UPDATE COMMAND REREAD PROCESSING IN PROGRESS
- NO MATCHING EXTTRACE STATEMENT FOUND FOR ADDRESS SPACE



Information On Demand
IMS Version 10
IBM

Abend Search and Notification

Figure 2-66. [No title]

CMA01.0

Notes:

Highlights

- IMS Abend Search and Notification (IMS ASN)
  - Notification mechanism for IMS failures
    - E-mails to user-specified recipients
    - Text messages to user-specified cellular devices, …
- Provides direct and real time access to the ABEND information and description
  - Automatic creation of hyperlinks (URLs) to IBM-supplied Internet resources for analyzing and resolving problems
  - Security
    - E-mails are sent internally from the IMS system task to pre-defined e-mail addresses
    - No sensitive information, such as customer data, is sent

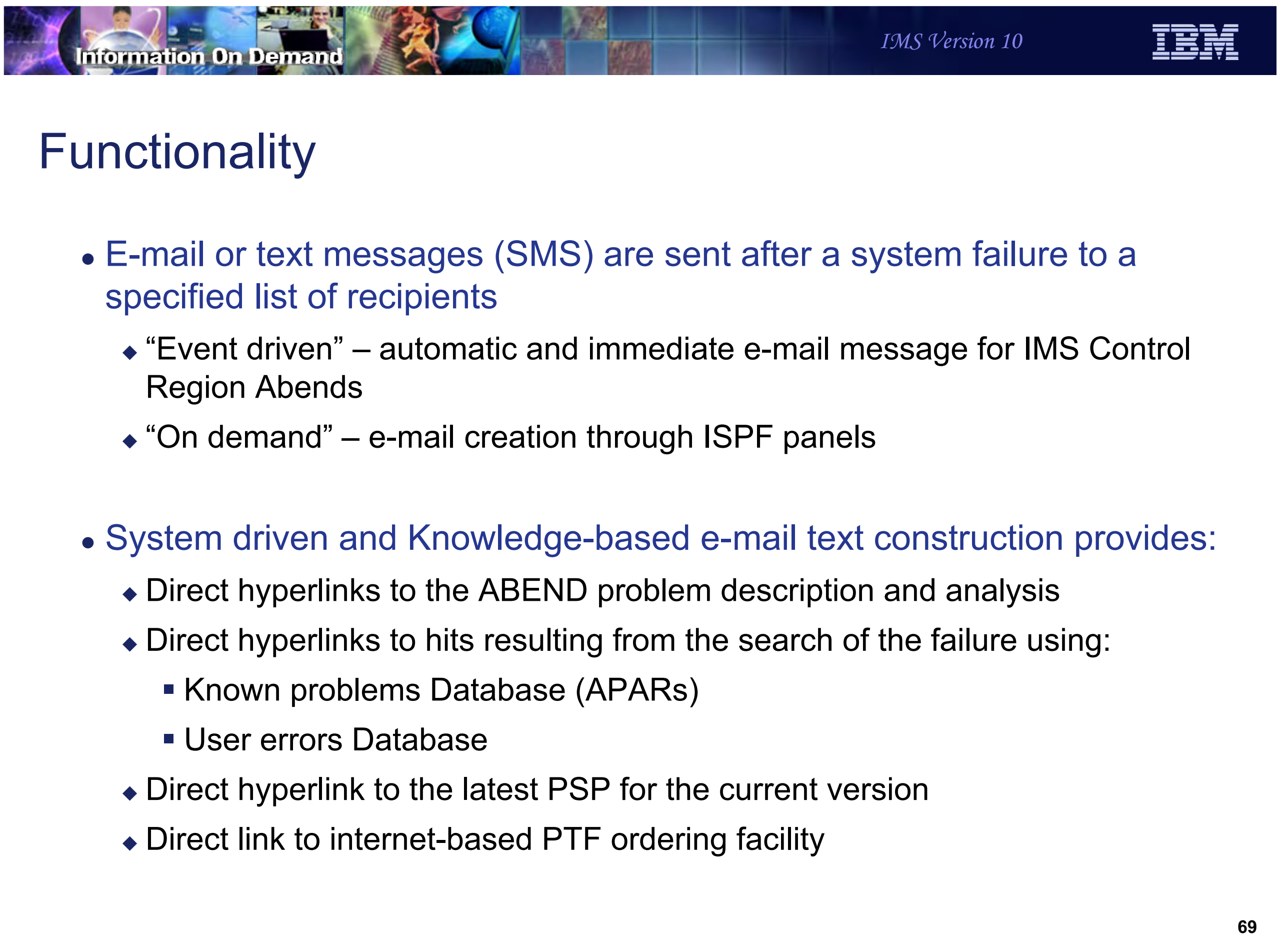
Figure 2-67. Highlights CMA01.0

Notes:

The IMS Abend Search and Notification (IMS ASN) function in IMS V10 provides a mechanism for 'real-time' automatic notification of IMS problems, such as abnormal terminations, to designated personnel who may not have immediate or direct access to the operator consoles.

When the abend occurs (or when the system fails), IMS ASN gathers information about the abend, the return code, the module affected, and, when possible, the APAR level.

Note that security is not an issue because all associated e-mails are sent internally from the IMS system task to the list of pre-defined e-mail addresses. No sensitive information, such as customer data, is ever sent.


Information On Demand
IMS Version 10
IBM

Functionality

- E-mail or text messages (SMS) are sent after a system failure to a specified list of recipients
  - "Event driven" – automatic and immediate e-mail message for IMS Control Region Abends
  - "On demand" – e-mail creation through ISPF panels
- System driven and Knowledge-based e-mail text construction provides:
  - Direct hyperlinks to the ABEND problem description and analysis
  - Direct hyperlinks to hits resulting from the search of the failure using:
    - Known problems Database (APARs)
    - User errors Database
  - Direct hyperlink to the latest PSP for the current version
  - Direct link to internet-based PTF ordering facility

Figure 2-68. Functionality CMA01.0

Notes:

The goal of IMS ASN is to notify one or more pre-defined recipients, via mechanisms such as e-mail or a text message to a cellular device, that an abend just occurred in a given subsystem. There are two ways to initiate an IMS ASN e-mail. The first is "event driven" which occurs based upon an event such as an IMS sytem abend. The second is "on demand" by initiating a request that is sent from the ISPF panels that are a part of the IMS ASN support.

IMS ASN also attempts to construct a message that provides more information on the specific nature of the problem along with potential solutions. The function automatically attempts to create hyperlinks to:

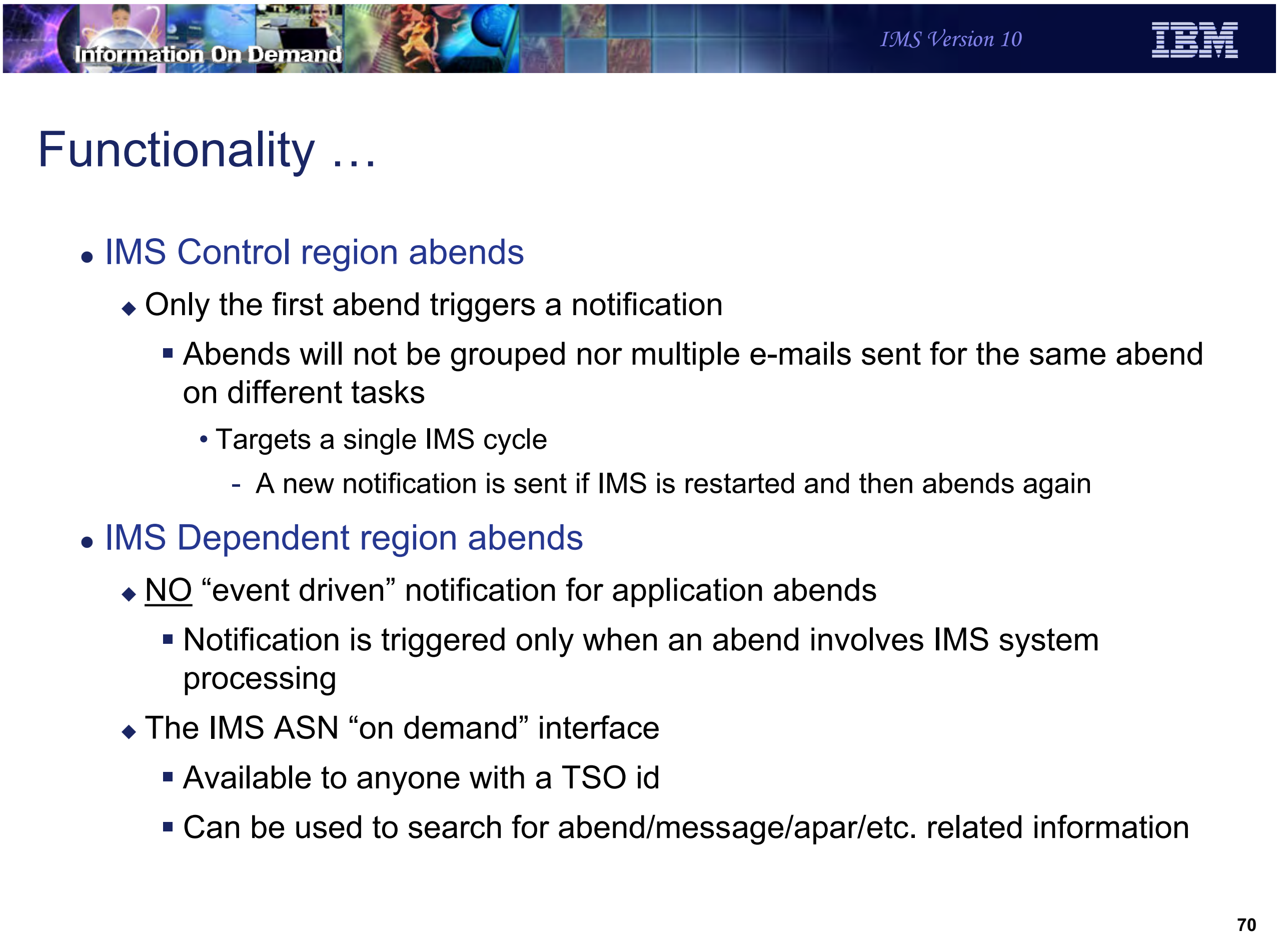
- An abend explanation in the IMS Messages and Codes manual
- Pertinent INTERNET database search engines (in practical terms, the hyperlink will be a URL with a pre-built search argument). The following are examples of links but note that, since they are links, then they are subject to change:

Technotes:
http://www-1.ibm.com/support/search.wss?rs=81&tc=SSEPH2&dc=DB520+D800+D900+
DA900+

DA800&dtm

PSP bucket: https://techsupport.services.ibm.com/server/390.psp390
Messages:
http://publibfp.boulder.ibm.com/cgi-bin/bookmgr/BOOKS/dfsmc1g1/CCONTENTS
PMR: https://www-925.ibm.com/software/support/esr/esr_home.do

List of possible related APARs (fix), or APARs in error (PE APAR).
Note: this hyperlink will be present only if the APAR information is available to IMS ASN components at time of error. The APAR hyperlink is
http://www-1.ibm.com/support/search.wss?rs=81&tc=SSEPH2&dc=DB550+D100&dtm



Information On Demand
IMS Version 10
IBM

Functionality …

- IMS Control region abends
  - Only the first abend triggers a notification
    - Abends will not be grouped nor multiple e-mails sent for the same abend on different tasks
      - Targets a single IMS cycle
        - A new notification is sent if IMS is restarted and then abends again
- IMS Dependent region abends
  - NO "event driven" notification for application abends
    - Notification is triggered only when an abend involves IMS system processing
  - The IMS ASN "on demand" interface
    - Available to anyone with a TSO id
    - Can be used to search for abend/message/apar/etc. related information

Figure 2-69. Functionality …

Notes:

Abend notification is triggered when an abend occurs that causes the IMS Control Region to abnormally terminate. Only the first abend triggers a notification. Additionally, IMS ASN does not group abends, nor does it send multiple e-mails for the same abend that might occur in different TCBs. The notification is intended for a single IMS cycle so if IMS is restarted and then abends again, IMS ASN sends a new notification.

Before creating the notification, IMS ASN logic checks a new dump suppression table in the dump module DFSFDMP0 to determine if an abend requires notification. If it does, then IMS ASN gathers information about the abend, the return code, the module affected, and, when possible, the APAR level. On the other hand, there are abends that may have notification interest, but for which no search is meaningful or necessary. The following list shows abends for which either no notification, or notification without hyperlinks is generated:

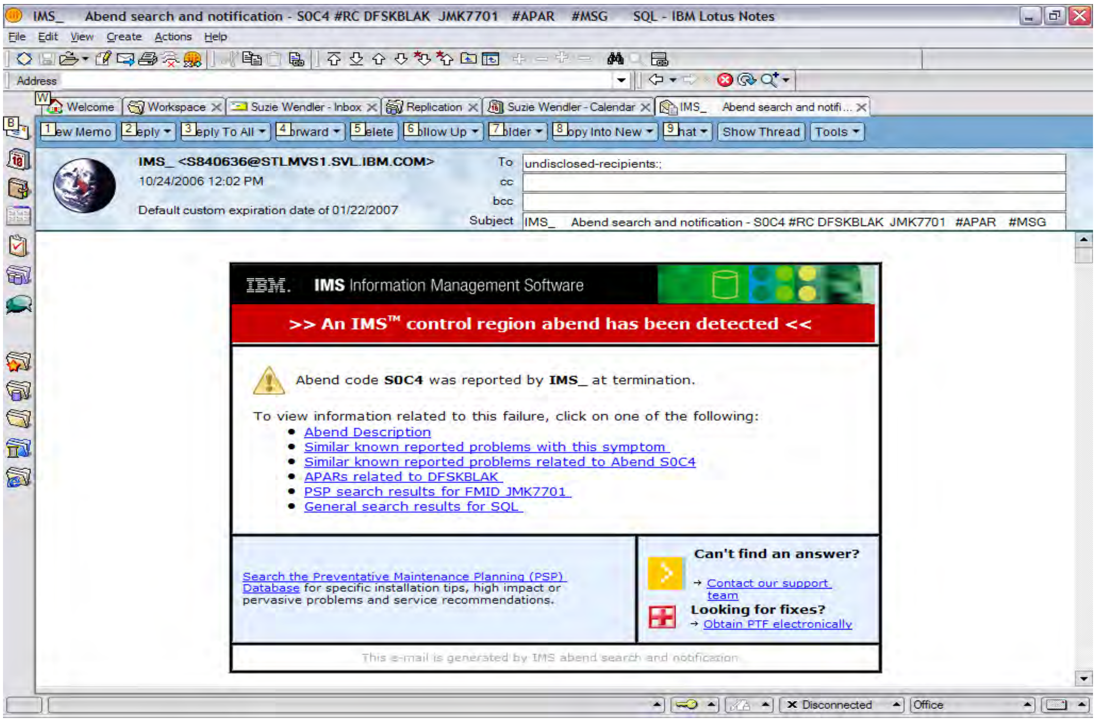
- U0020 - notification only, no hyperlinks with search arguments provided
- U0150 - no notification necessary. CTL region abend already generated a notification

- U0604 - no notification necessary. /SWITCH SYSTEM FORCE/ACTIVE entered (XRF)
- U4095 - no notification necessary. CTL region abend already generated a notification
- SFFF - non-IBM program termination. No notification necessary
- SFFE - non-IBM program termination. No notification necessary

Note that dependent regions abends (MPP or BMP) trigger the notification and search "event driven" function only if the abend involves IMS system processing. Application abends are not recognized by the "event driven" capability.

Dependent Region abend searches, however, can be accomplished using the IMS ASN functionality. The IMS ASN "on demand interface", available to anyone with a TSO id, can be used to search for abend/message/apar/etc. related information.

Example of E-mail Notification - Graphics



IMS_ Abend search and notification - S0C4 #RC DFSKBLAK JMK7701 #APAR #MSG SQL - IBM Lotus Notes
File Edit View Create Actions Help
Address
Welcome
Workspace
Suzie Wendler - Inbox
Replication
Suzie Wendler - Calendar
IMS_ Abend search and notifi...
New Memo
Reply
Reply To All
Forward
Delete
Follow Up
Folder
Copy Into New
Chat
Show Thread
Tools
Disconnected
Office

IMS_ <S840636@STLMVS1.SVL.IBM.COM>
10/24/2006 12:02 PM
Default custom expiration date of 01/22/2007

To: undisclosed-recipients:;
cc:
bcc:
Subject: IMS_ Abend search and notification - S0C4 #RC DFSKBLAK JMK7701 #APAR #MSG

IBM. **IMS** Information Management Software

>> An IMS™ control region abend has been detected <<

Abend code **S0C4** was reported by **IMS_** at termination.

To view information related to this failure, click on one of the following:

- Abend Description
- Similar known reported problems with this symptom
- Similar known reported problems related to Abend S0C4
- APARs related to DFSKBLAK
- PSP search results for FMID JMK7701
- General search results for SQL

Search the Preventative Maintenance Planning (PSP) Database for specific installation tips, high impact or pervasive problems and service recommendations.

Can't find an answer?
→ Contact our support team

Looking for fixes?
→ Obtain PTF electronically

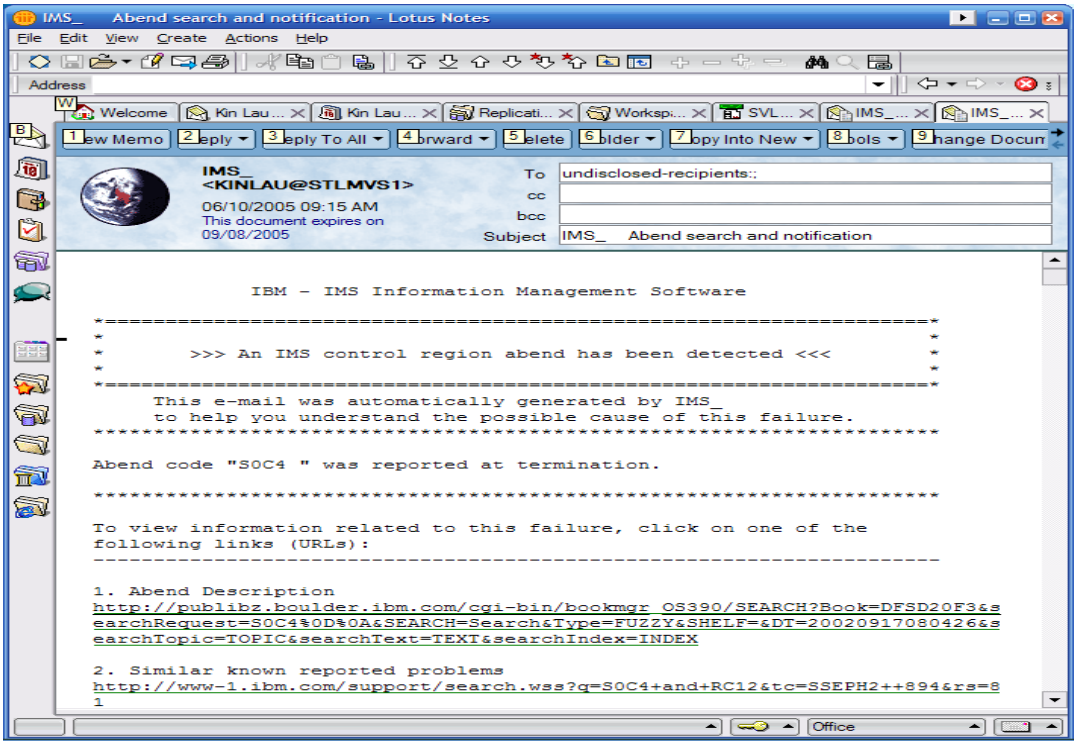
This e-mail is generated by IMS abend search and notification

Figure 2-70. Example of E-mail Notification - Graphics CMA01.0

Notes:

This visual gives an actual example of an e-mail notification in an environment that supports full graphics.

Example of E-mail Notification - Text Only


IMS_ Abend search and notification - Lotus Notes
File Edit View Create Actions Help
Address
Welcome
Kin Lau ...
Kin Lau ...
Replicati...
Workspi...
SVL...
IMS_...
IMS_...
1 New Memo
2 Reply
3 Reply To All
4 Forward
5 Delete
6 Folder
7 Copy Into New
8 Tools
9 Change Docum
Office

IMS_
<KINLAU@STLMVS1>
06/10/2005 09:15 AM
This document expires on
09/08/2005

To: undisclosed-recipients:;
cc:
bcc:
Subject: IMS_ Abend search and notification

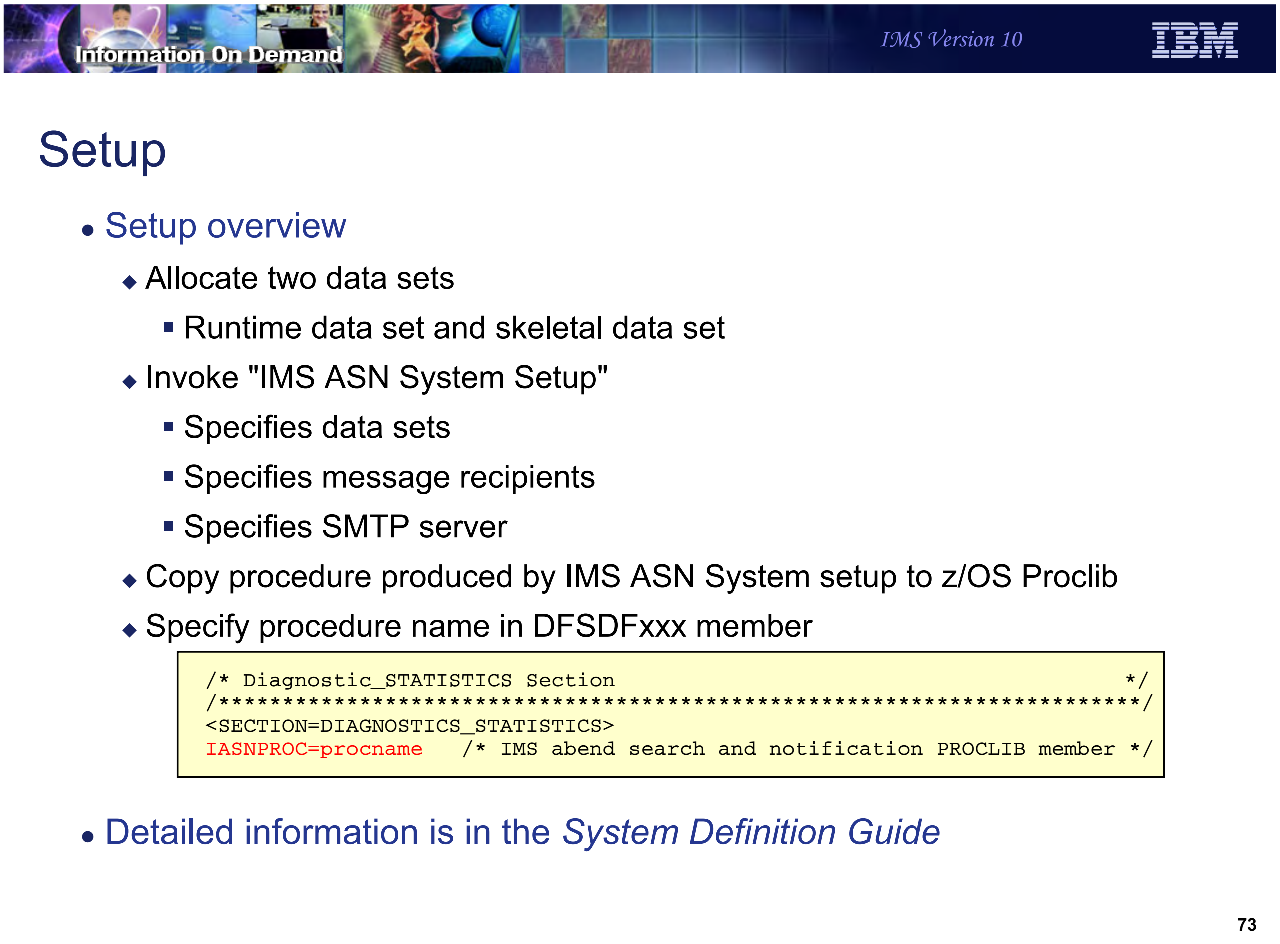
```
                IBM - IMS Information Management Software

*=================================================================*
*                                                                 *
*       >>> An IMS control region abend has been detected <<<     *
*                                                                 *
*=================================================================*
     This e-mail was automatically generated by IMS_
     to help you understand the possible cause of this failure.
*******************************************************************

Abend code "S0C4 " was reported at termination.

*******************************************************************

To view information related to this failure, click on one of the
following links (URLs):
-------------------------------------------------------------------

1. Abend Description
http://publibz.boulder.ibm.com/cgi-bin/bookmgr_OS390/SEARCH?Book=DFSD20F3&s
earchRequest=S0C4%0D%0A&SEARCH=Search&Type=FUZZY&SHELF=&DT=20020917080426&s
earchTopic=TOPIC&searchText=TEXT&searchIndex=INDEX

2. Similar known reported problems
http://www-1.ibm.com/support/search.wss?q=S0C4+and+RC12&tc=SSEPH2++894&rs=8
1
```

72

Figure 2-71. Example of E-mail Notification - Text Only CMA01.0

Notes:

For environments with limited graphics capabilities, text-only notification is available. This visual provides an example of the text-only e-mail format.


Information On Demand
IMS Version 10
IBM

Setup

- Setup overview
  - Allocate two data sets
    - Runtime data set and skeletal data set
  - Invoke "IMS ASN System Setup"
    - Specifies data sets
    - Specifies message recipients
    - Specifies SMTP server
  - Copy procedure produced by IMS ASN System setup to z/OS Proclib
  - Specify procedure name in DFSDFxxx member

```
/* Diagnostic_STATISTICS Section                                      */
/*********************************************************************/
<SECTION=DIAGNOSTICS_STATISTICS>
IASNPROC=procname   /* IMS abend search and notification PROCLIB member */
```

- Detailed information is in the *System Definition Guide*

73

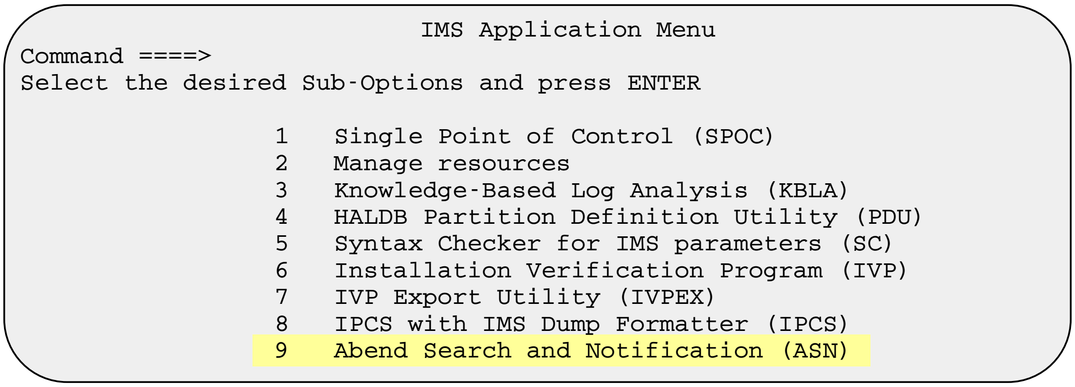
Figure 2-72. Setup CMA01.0

Notes:

IMS ASN has a set of ISPF panels that are used to set up the environment. The application can be invoked either by keying in an exec command in the TSO ISPF Option 6 panel or selecting option 9 in the IMS Application Menu program.

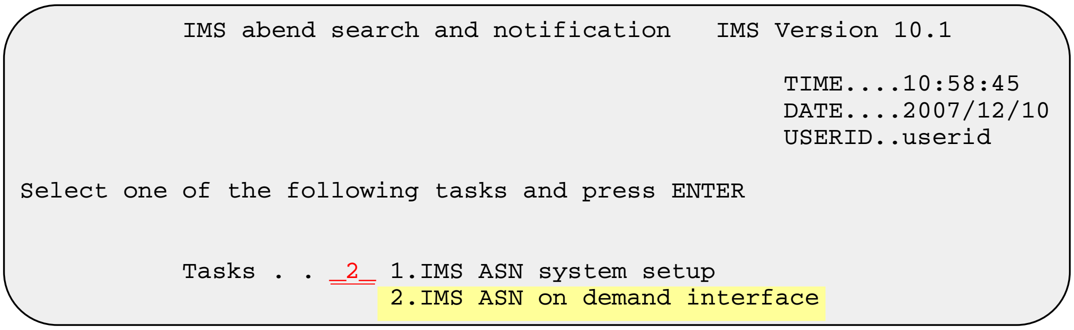
On Demand Interface

1



```
                           IMS Application Menu
Command ====>
Select the desired Sub-Options and press ENTER

                  1   Single Point of Control (SPOC)
                  2   Manage resources
                  3   Knowledge-Based Log Analysis (KBLA)
                  4   HALDB Partition Definition Utility (PDU)
                  5   Syntax Checker for IMS parameters (SC)
                  6   Installation Verification Program (IVP)
                  7   IVP Export Utility (IVPEX)
                  8   IPCS with IMS Dump Formatter (IPCS)
                  9   Abend Search and Notification (ASN)
```

2



```
           IMS abend search and notification   IMS Version 10.1

                                                   TIME....10:58:45
                                                   DATE....2007/12/10
                                                   USERID..userid

Select one of the following tasks and press ENTER


           Tasks . . _2_ 1.IMS ASN system setup
                         2.IMS ASN on demand interface
```

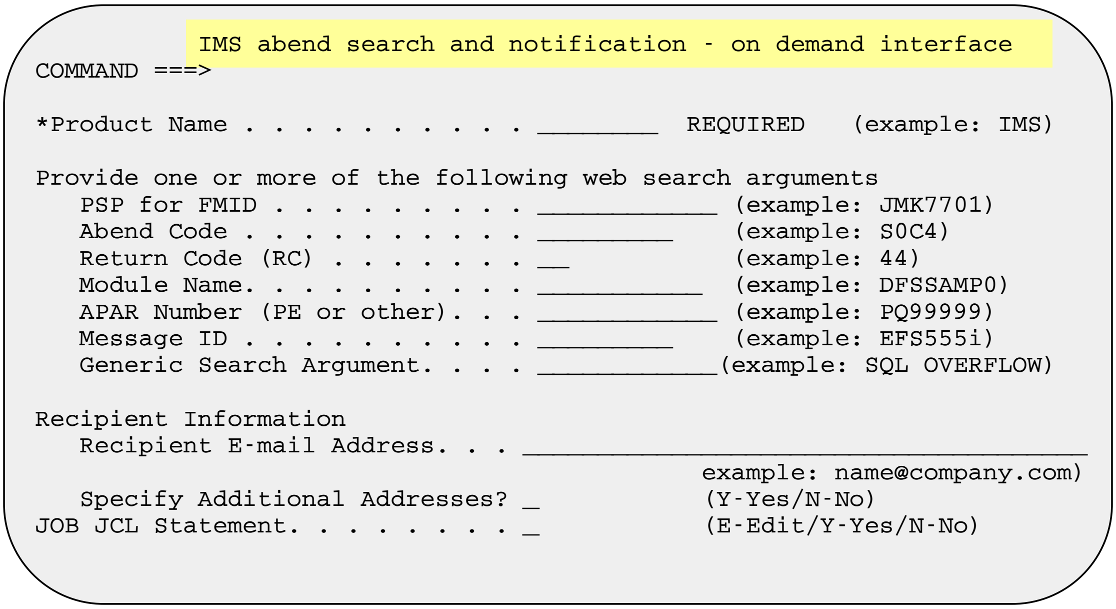
Figure 2-73. On Demand Interface CMA01.0

Notes:

To invoke the "On Demand" interface, you first go to the IMS Application Menu. ASN is option 9 on this screen. This invokes the IMS abend search and notification screen. The "on demand" interface is option 2 on this screen.

On Demand Interface …

- Specifications for “on demand” processing



```
                  IMS abend search and notification - on demand interface
COMMAND ===>

*Product Name . . . . . . . . . . ________  REQUIRED   (example: IMS)

Provide one or more of the following web search arguments
   PSP for FMID . . . . . . . . . ____________  (example: JMK7701)
   Abend Code . . . . . . . . . . _________     (example: S0C4)
   Return Code (RC) . . . . . . . __            (example: 44)
   Module Name. . . . . . . . . . ___________   (example: DFSSAMP0)
   APAR Number (PE or other). . . ____________  (example: PQ99999)
   Message ID . . . . . . . . . . _________     (example: EFS555i)
   Generic Search Argument. . . . ____________(example: SQL OVERFLOW)

Recipient Information
   Recipient E-mail Address. . . ______________________________________
                                            example: name@company.com)
   Specify Additional Addresses? _          (Y-Yes/N-No)
JOB JCL Statement. . . . . . . . _          (E-Edit/Y-Yes/N-No)
```

Figure 2-74. On Demand Interface …

Notes:

The panel for the “on demand” interface is used to activate IMS ASN dynamically from a TSO session. It also provides the ability to specify search criteria, or to replicate the search criteria which was initiated by an abend-driven invocation of IMS ASN.

The program driving this panel parses and assembles the search criteria into a JCL stream. The user can then submit the job generated by IMS ASN.

If ‘Y’ is specified in the ‘Specify Additional Addresses?’ field above, the user will be prompted in edit mode on a member of a PDS where an unlimited number of e-mail recipients can be entered.

If ‘E’ is specified in the “JOB JCL Statement” field, a JOB statement is presented for the user to customize. It is retained for future use. If ‘Y’ is specified, the previously customized job statement is used.

‘Product Name’ is a required field and at least one search argument must be provided on the panel.

Information On Demand
IMS Version 10
IBM

Prerequisites

- IMS ASN e-mail notification uses a set of commands that are part of the Simple Mail Transfer Protocol (SMTP)
  - SMTP is an MVS function provided with TCP/IP
    - MVS TCP V3.1 or later
    - Both local and user-specified external SMTP servers are supported

- Receiving e-mail system
  - For graphic formats, MIME1.0 or higher is needed
    - Lotus Notes clients need patch, MJUK42GR8K, in addition to MIME 1.0 support
  - Otherwise a text-only email is displayed

Figure 2-75. Prerequisites CMA01.0

Notes:

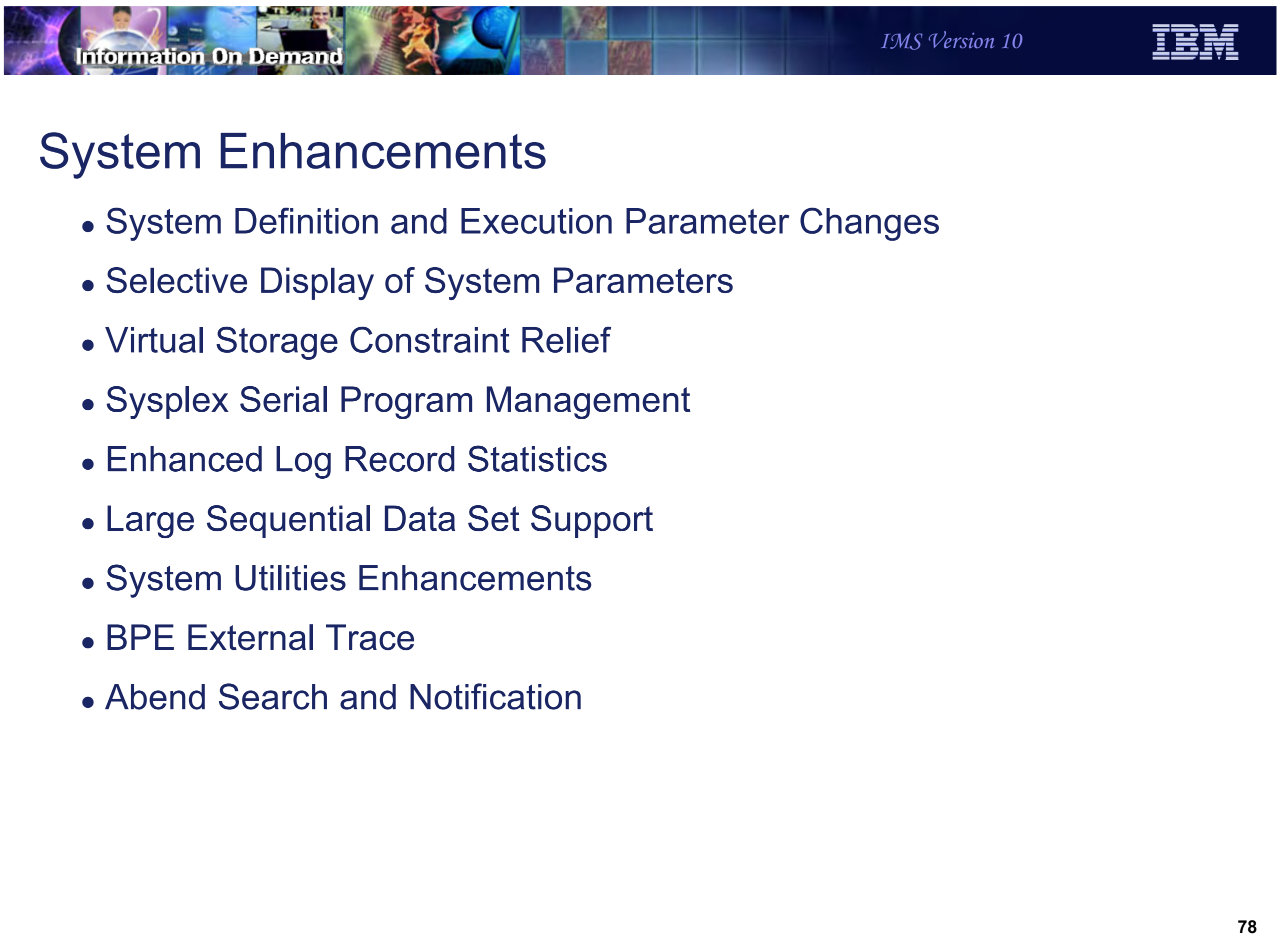
Since the e-mail is sent using SMTP, the IMS system must reside on a z/OS system that has enabled MVS TCP/IP. SMTP is a standard application that is delivered with the TCP/IP stack. For optimal results, the e-mail manager should support graphics and images. With consideration for the small screen size of cellular devices, the sample text message does not include the creation of URLs. The purpose of the text message is to notify users of the occurrence of the abend.

Benefits

- The IMS Abend Search and Notification function
  - Provides
    - Timely and automatic notification of IMS system problems
    - Quick access to explanation of the error and possible causes
    - Quick and direct access to IBM-supplied diagnostics
  - Facilitates problem determination
    - Inclusion of links that are of interest to the problem
  - Reduces problem resolution time

Figure 2-76. Benefits CMA01.0

Notes:


Information On Demand
IMS Version 10

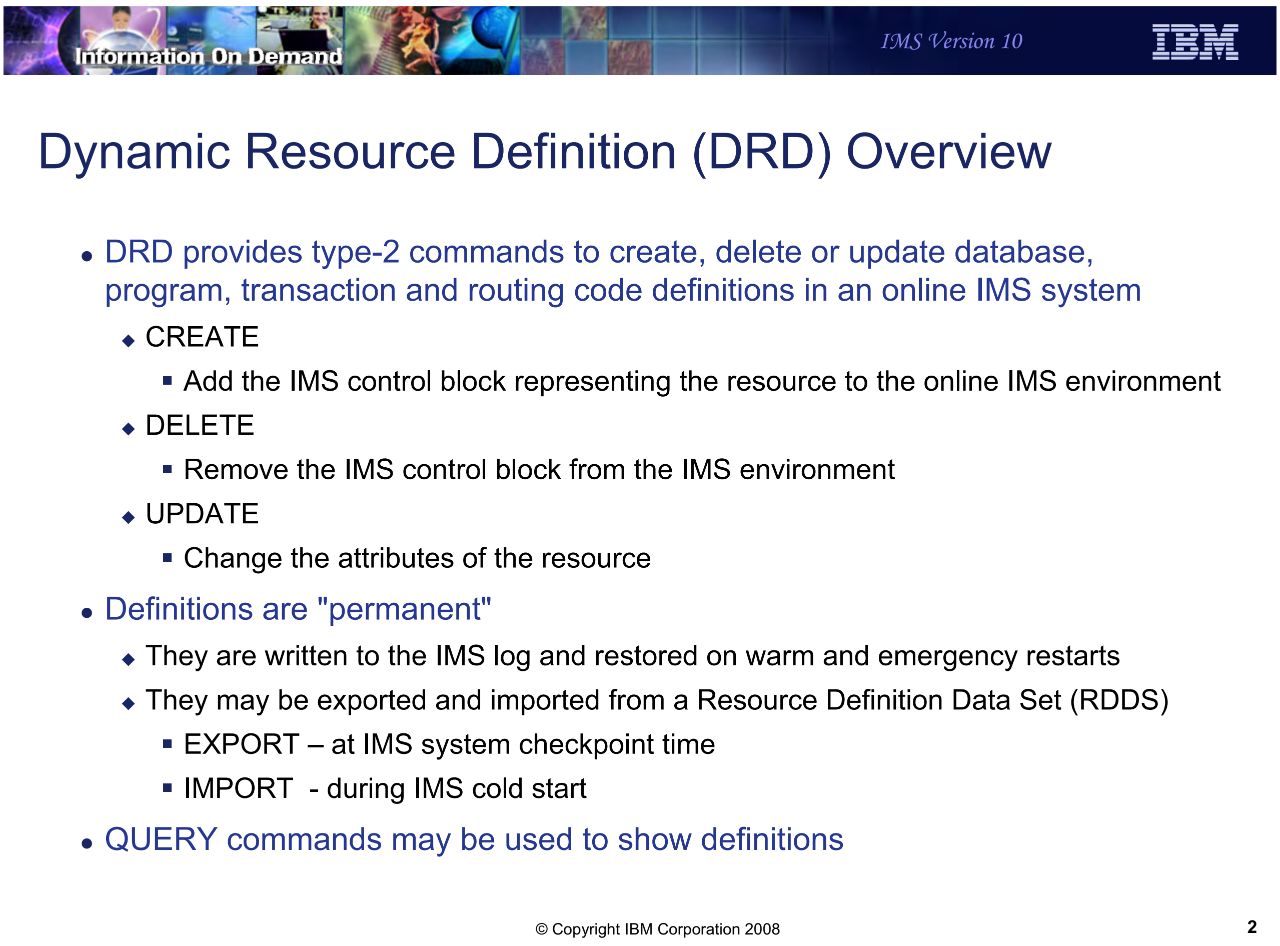
System Enhancements

- System Definition and Execution Parameter Changes
- Selective Display of System Parameters
- Virtual Storage Constraint Relief
- Sysplex Serial Program Management
- Enhanced Log Record Statistics
- Large Sequential Data Set Support
- System Utilities Enhancements
- BPE External Trace
- Abend Search and Notification

Figure 2-77. System Enhancements CMA01.0

Notes:

Unit 3. Dynamic Resource Definition (DRD)


Information On Demand
IMS Version 10
IBM

Dynamic Resource Definition (DRD) Overview

- DRD provides type-2 commands to create, delete or update database, program, transaction and routing code definitions in an online IMS system
  - CREATE
    - Add the IMS control block representing the resource to the online IMS environment
  - DELETE
    - Remove the IMS control block from the IMS environment
  - UPDATE
    - Change the attributes of the resource
- Definitions are "permanent"
  - They are written to the IMS log and restored on warm and emergency restarts
  - They may be exported and imported from a Resource Definition Data Set (RDDS)
    - EXPORT – at IMS system checkpoint time
    - IMPORT - during IMS cold start
- QUERY commands may be used to show definitions

© Copyright IBM Corporation 2008

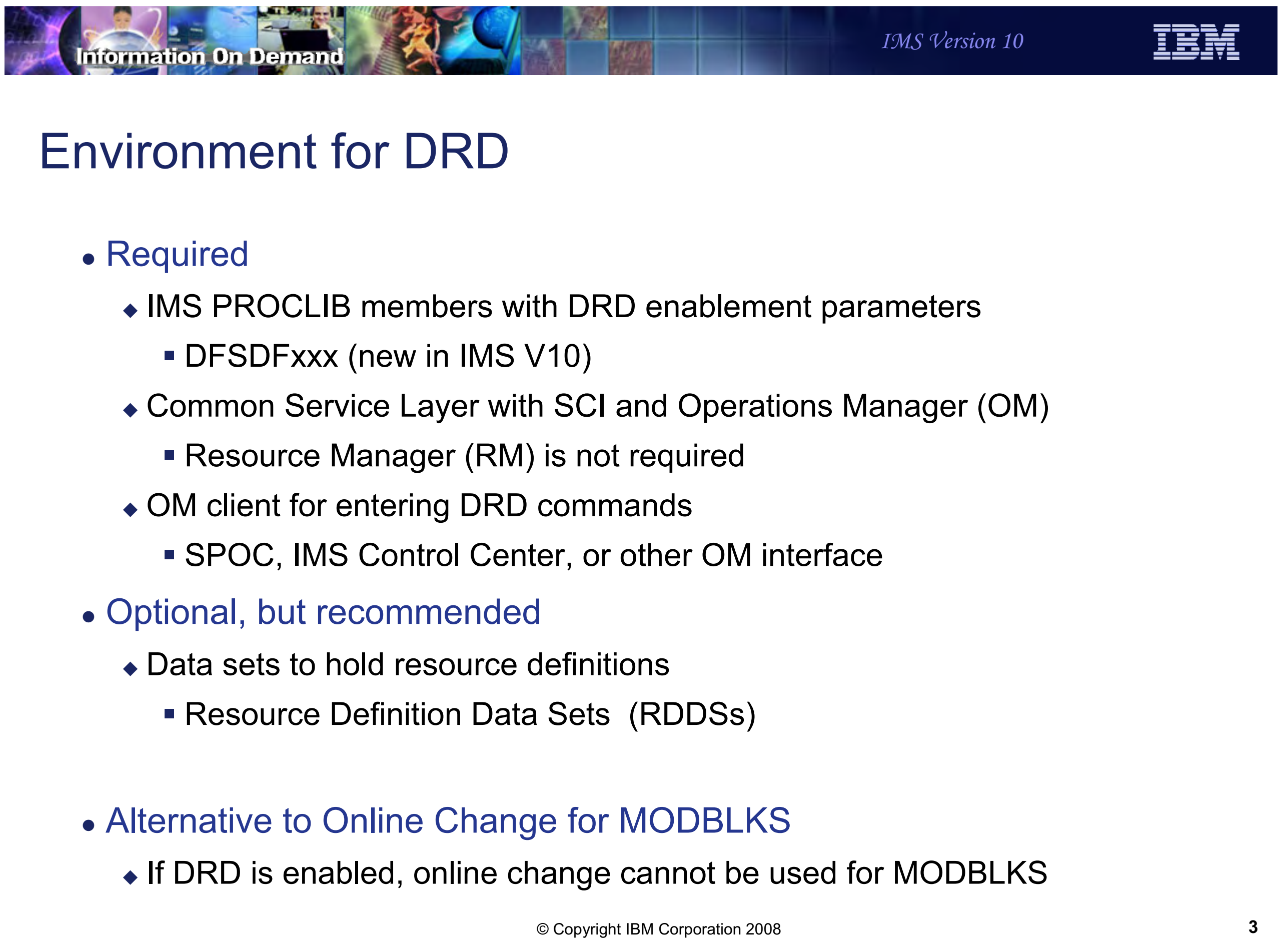
Figure 3-1. Dynamic Resource Definition (DRD) Overview CMA01.0

Notes:

DRD is implemented with new and enhanced type-2 commands. The CREATE command is used to add (create) a new resource control block in a running IMS environment. The DELETE command is used to delete existing control blocks. The UPDATE command is used to update resource attributes.

The definitions added by DRD are typically permanent. They are always permanent across warm and emergency restarts. All definitions are written to the IMS log. This includes those created by the system definition (sysgen) process and those created by DRD. When an IMS system is warm started or emergency restarted, these definitions are read from the log. Definitions added by DRD may be permanent across cold starts. They may be exported for use by subsequent cold starts. Exports are done as part of system checkpoints. They are written to Resource Definition Data Sets (RDDS). When a cold start is done, IMS optionally reads the definitions from an RDDS. We will see later that the use of RDDSs is optional. If they are not used, cold starts will not restore definitions added by DRD.

Finally, QUERY commands may be used to show all resources including those created or updated by DRD.


Information On Demand
IMS Version 10
IBM

Environment for DRD

- Required
  - IMS PROCLIB members with DRD enablement parameters
    - DFSDFxxx (new in IMS V10)
  - Common Service Layer with SCI and Operations Manager (OM)
    - Resource Manager (RM) is not required
  - OM client for entering DRD commands
    - SPOC, IMS Control Center, or other OM interface
- Optional, but recommended
  - Data sets to hold resource definitions
    - Resource Definition Data Sets (RDDSs)
- Alternative to Online Change for MODBLKS
  - If DRD is enabled, online change cannot be used for MODBLKS

© Copyright IBM Corporation 2008

Figure 3-2. Environment for DRD CMA01.0

Notes:

You must define DRD to IMS by including DRD parameters the new DFSDFxxx PROCLIB member.

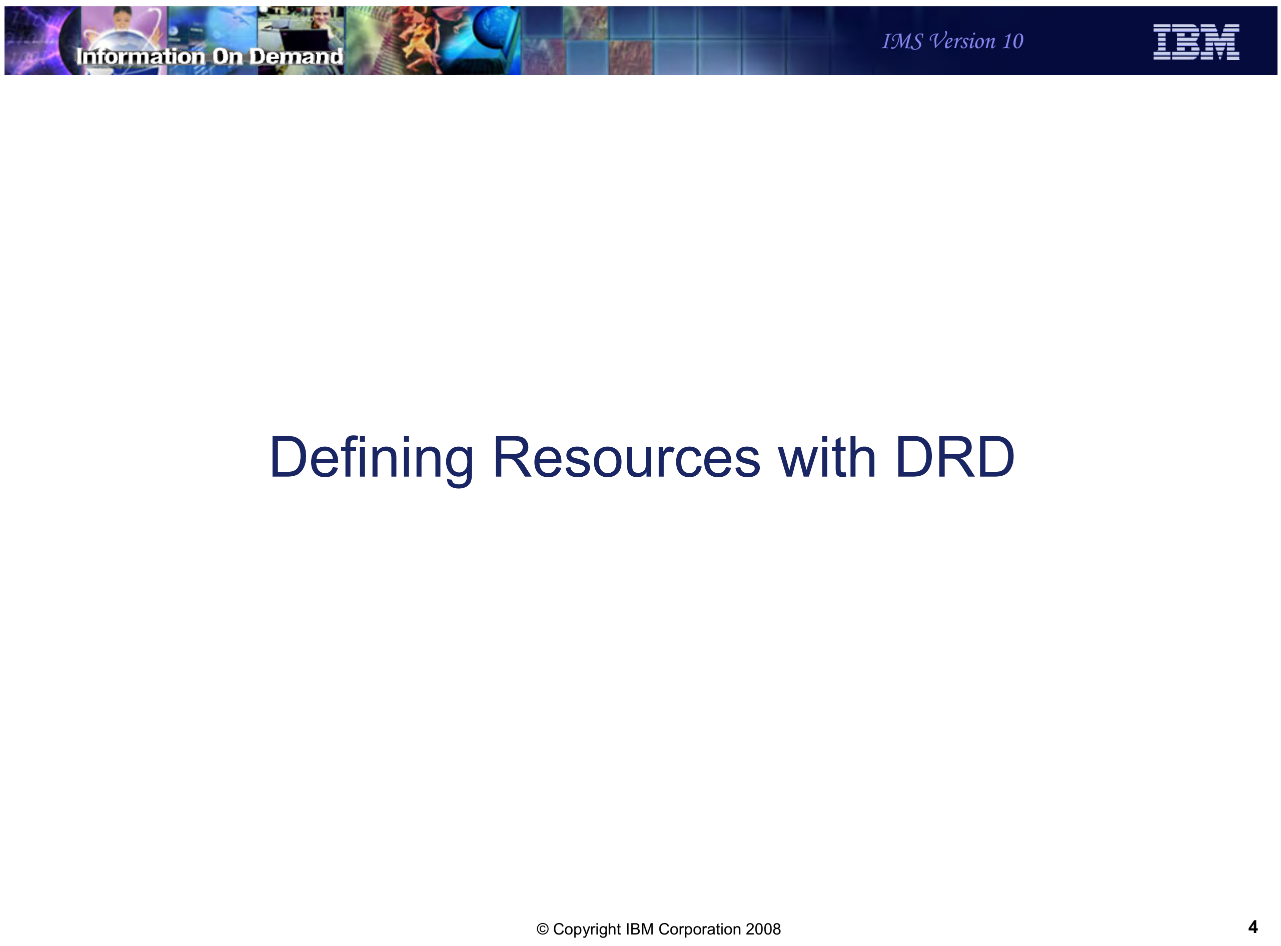
Since type-2 commands are used, you must have a Common Service Layer (CSL) including SCI and OM. RM is not required. You also must have an OM client through which you may enter the commands. This can be the IMS TSO SPOC, the IMS Control Center, or any other OM client.

RDDS data sets are recommended. They are used for exporting and importing definitions across IMS cold starts. They also may be used by the extract utility to move the definitions to another system.

DRD is supported in all IMS environments, including DB/DC, DBCTL, and DCCTL. It is also supported in a data sharing environment and a shared queues environment. Care must be taken in a multi-IMS environment to define resources consistently across all IMSs

When DRD is implemented, online change cannot be used for MODBLKS. Online change remains available for ACBLIB and FORMAT.

When DRD is fully implemented, there is no longer a need for a MODBLKS system definition. MODBLKS system definitions may be used when DRD is implemented, but you may choose to eliminate it and define all MODBLKS resources (programs, transactions, routing codes, and databases) using DRD commands. The MODBLKS libraries associated with the system definition and online change processes may be deleted when only DRD is used to define these resources.

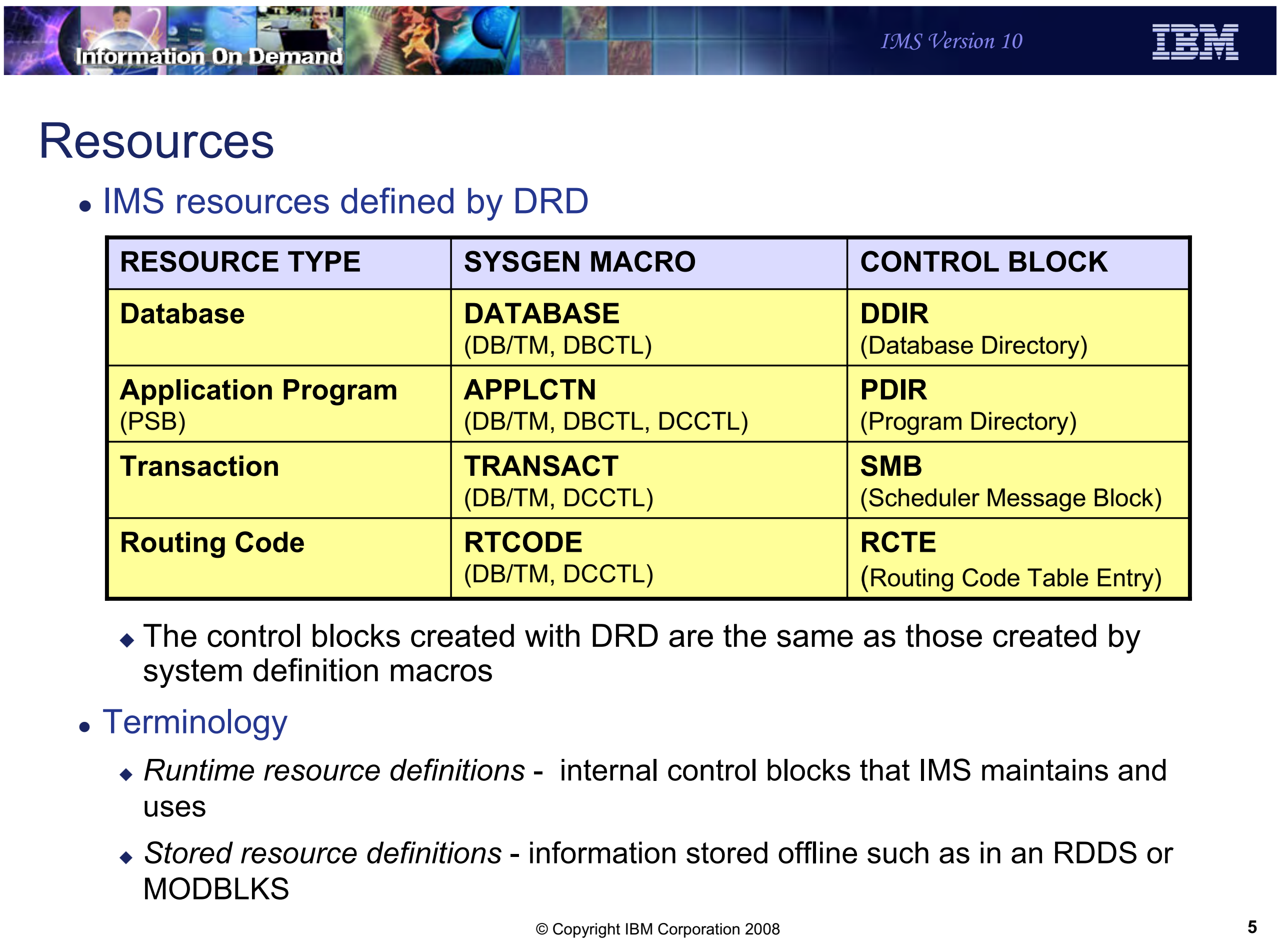

Information On Demand
IMS Version 10
IBM

Defining Resources with DRD

© Copyright IBM Corporation 2008

Figure 3-3. [No title] CMA01.0

Notes:


Information On Demand
IMS Version 10
IBM

Resources

- IMS resources defined by DRD

RESOURCE TYPE	SYSGEN MACRO	CONTROL BLOCK
Database	**DATABASE** (DB/TM, DBCTL)	**DDIR** (Database Directory)
Application Program (PSB)	**APPLCTN** (DB/TM, DBCTL, DCCTL)	**PDIR** (Program Directory)
Transaction	**TRANSACT** (DB/TM, DCCTL)	**SMB** (Scheduler Message Block)
Routing Code	**RTCODE** (DB/TM, DCCTL)	**RCTE** (Routing Code Table Entry)

  - The control blocks created with DRD are the same as those created by system definition macros
- Terminology
  - *Runtime resource definitions* - internal control blocks that IMS maintains and uses
  - *Stored resource definitions* - information stored offline such as in an RDDS or MODBLKS

© Copyright IBM Corporation 2008

Figure 3-4. Resources CMA01.0

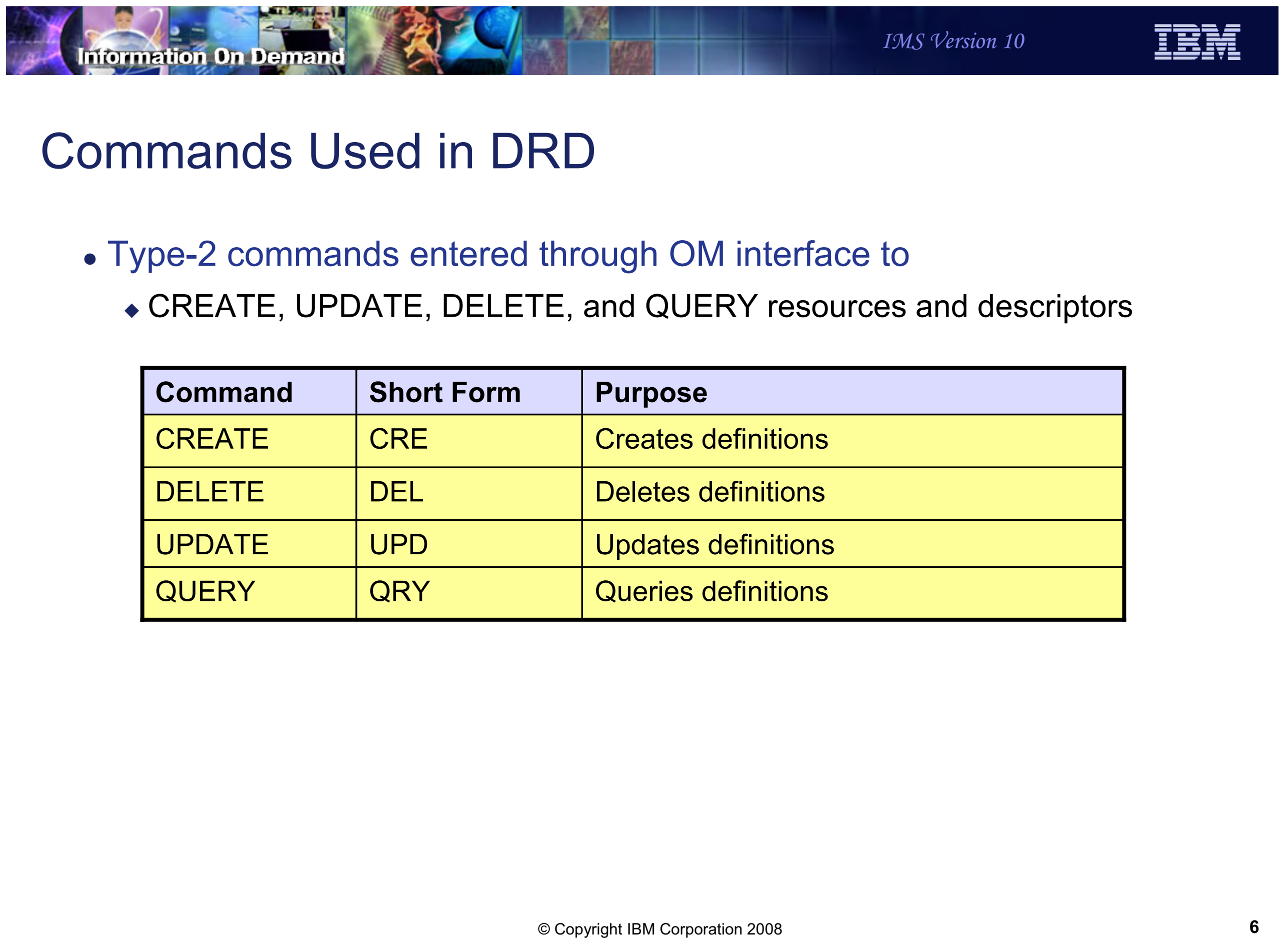
Notes:

The resources that may be defined by DRD are shown here. The control blocks are the same as those that are created by the system definition process.

IMS publications use two new terms to describe the difference between definitions that IMS uses while running versus those that are stored outside of a running IMS system.

- Runtime resource definitions - these are the internal control blocks that IMS maintains and uses
- Stored resource definitions - this is resource information stored offline such as in an RDDS or MODBLKS

These are new terms, however, they describe a difference that has always existed in IMS. For example, the MODBLKS data set contains stored resource definitions that IMS uses to create runtime resource definitions when it is executed. The new terms are not limited to use with DRD.


Information On Demand
IMS Version 10
IBM

Commands Used in DRD

- Type-2 commands entered through OM interface to
  - CREATE, UPDATE, DELETE, and QUERY resources and descriptors

Command	Short Form	Purpose
CREATE	CRE	Creates definitions
DELETE	DEL	Deletes definitions
UPDATE	UPD	Updates definitions
QUERY	QRY	Queries definitions

© Copyright IBM Corporation 2008

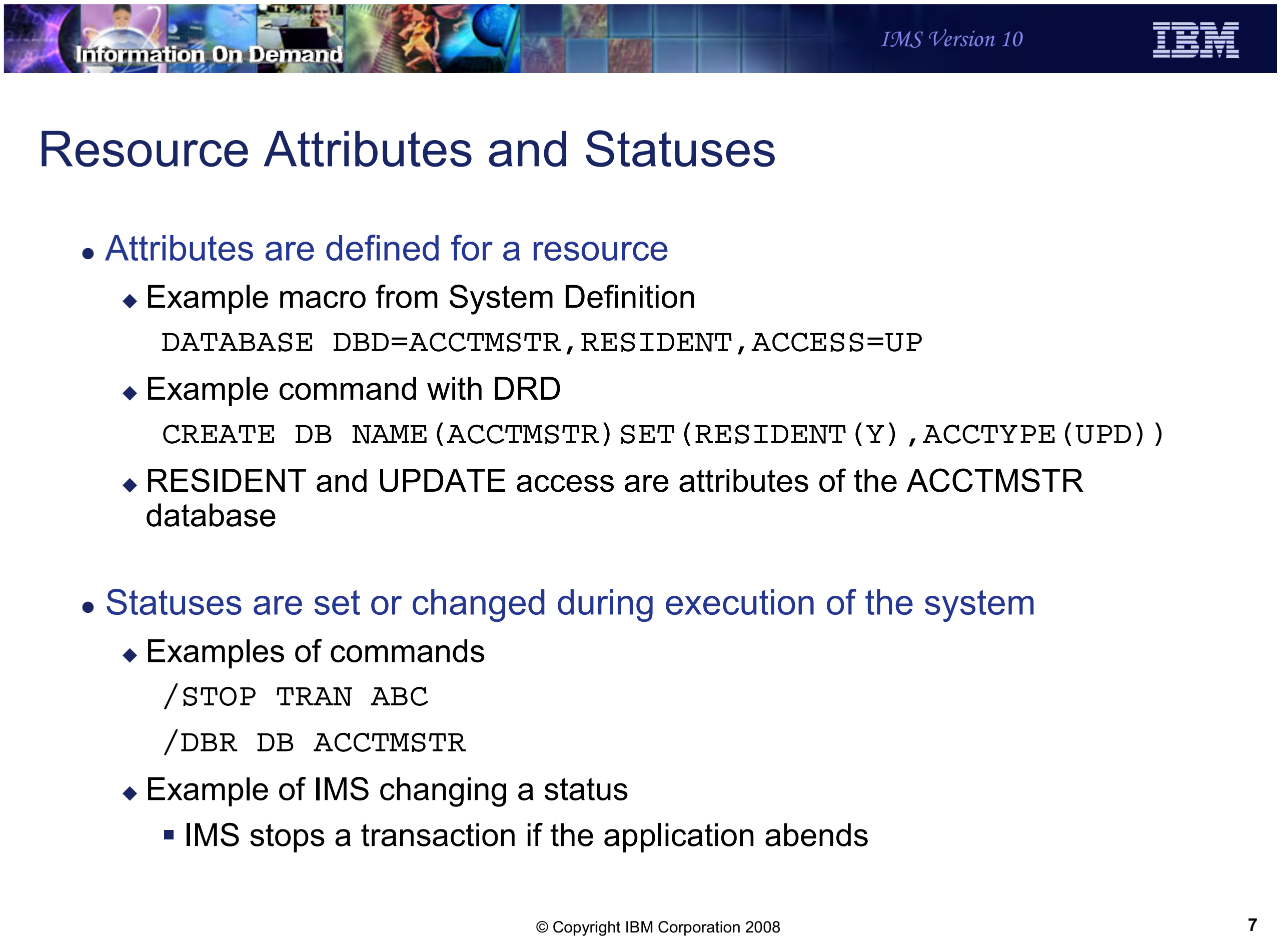
Figure 3-5. Commands Used in DRD CMA01.0

Notes:

There are four commands used with DRD.

- CREATE – This command is used to create definitions
- DELETE – This command is used to delete definitions
- UPDATE – This command is used to update definitions
- QUERY – This command is used to show definitions

The commands may have other uses. For example, the UPDATE command may be used to change a status and the QUERY command may be used to show a status. Statuses are discussed later.



Information On Demand

IMS Version 10

Resource Attributes and Statuses

- Attributes are defined for a resource
  - Example macro from System Definition

```
DATABASE DBD=ACCTMSTR,RESIDENT,ACCESS=UP
```

  - Example command with DRD

```
CREATE DB NAME(ACCTMSTR)SET(RESIDENT(Y),ACCTYPE(UPD))
```

  - RESIDENT and UPDATE access are attributes of the ACCTMSTR database

- Statuses are set or changed during execution of the system
  - Examples of commands

```
/STOP TRAN ABC
/DBR DB ACCTMSTR
```

  - Example of IMS changing a status
    - IMS stops a transaction if the application abends

© Copyright IBM Corporation 2008

Figure 3-6. Resource Attributes and Statuses CMA01.0

Notes:

Resources have attributes and statuses.

Attributes are defined for a resource. They are specified either on a system definition macro or on a DRD command.

Statuses are not part of definitions. They indicate the current availability of a resource. For example, a transaction or database may be stopped or started. The /STA DB, /STOP DB, /DBD DB and /DBR commands are examples of commands that change the status of a database. IMS may change the status of a resource without a command. For example, when an application program abends IMS may stop the transaction.

CREATE Command

- Creates resource definitions or descriptors
  - Programs, transactions, routing codes, and databases
    - Appropriate attributes for each resource type may be set or defaulted
  - SET used to specify attribute values

- Examples:

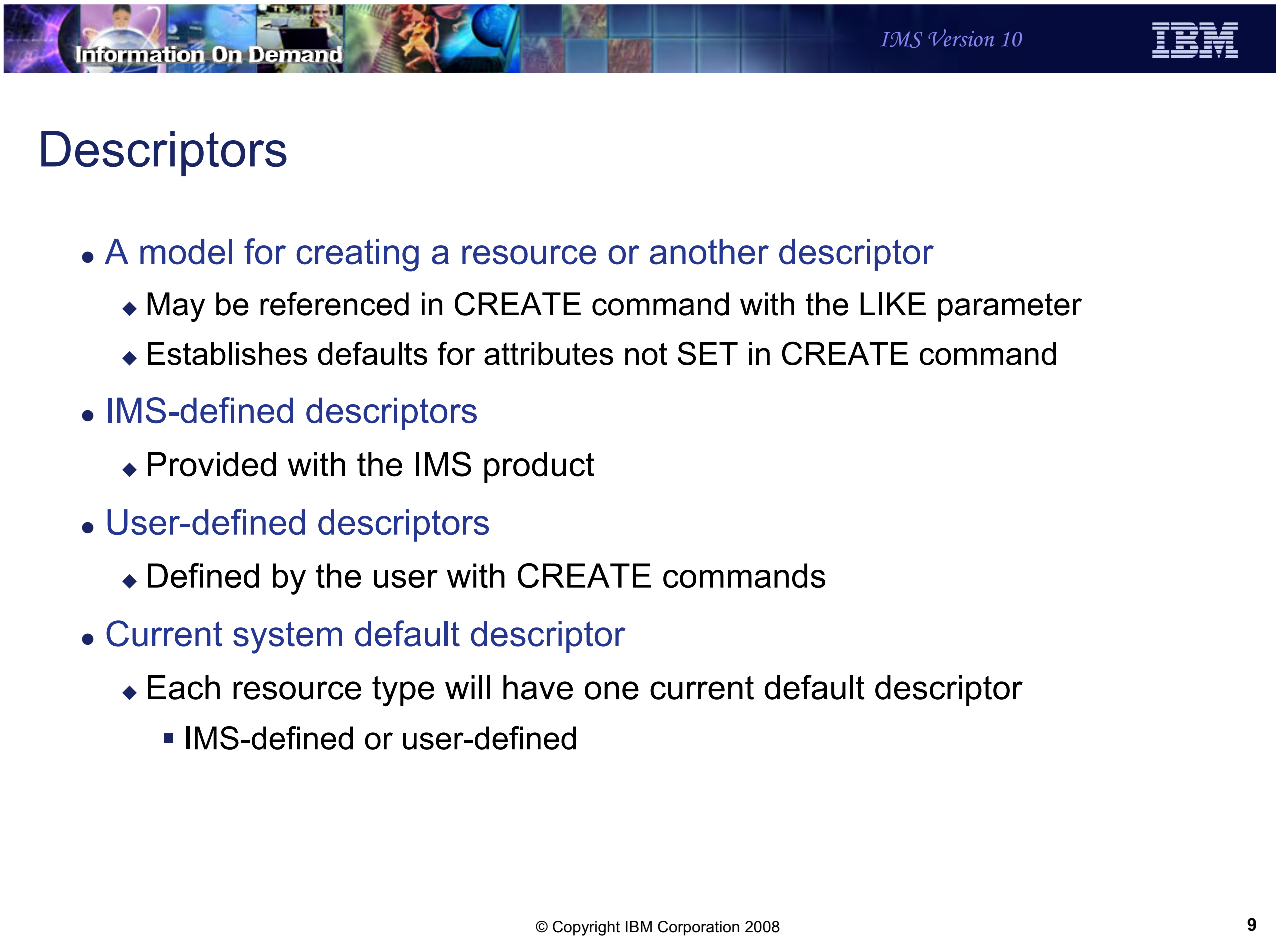
```
CREATE PGM NAME(PGM1) SET(RESIDENT(Y))

CREATE TRAN NAME(TRN1) SET(PGM(PGM1) CLASS(4))

CRE    DB NAME(DB001) SET(RESIDENT(Y))
```

© Copyright IBM Corporation 2008

Figure 3-7. CREATE Command CMA01.0

Notes:

The CREATE command is used to create program, transaction, routing code and database definitions. It is also used to create descriptors for these resources, Descriptors are explained on the next page. The NAME parameter is used to specify the name of the resource or descriptor. You use the SET parameter to set the attribute values. Of course, default values may be used for any or all of the attributes. CREATE may be abbreviated to CRE.


Information On Demand
IMS Version 10
IBM

Descriptors

- A model for creating a resource or another descriptor
    - May be referenced in CREATE command with the LIKE parameter
    - Establishes defaults for attributes not SET in CREATE command
- IMS-defined descriptors
    - Provided with the IMS product
- User-defined descriptors
    - Defined by the user with CREATE commands
- Current system default descriptor
    - Each resource type will have one current default descriptor
        - IMS-defined or user-defined

© Copyright IBM Corporation 2008

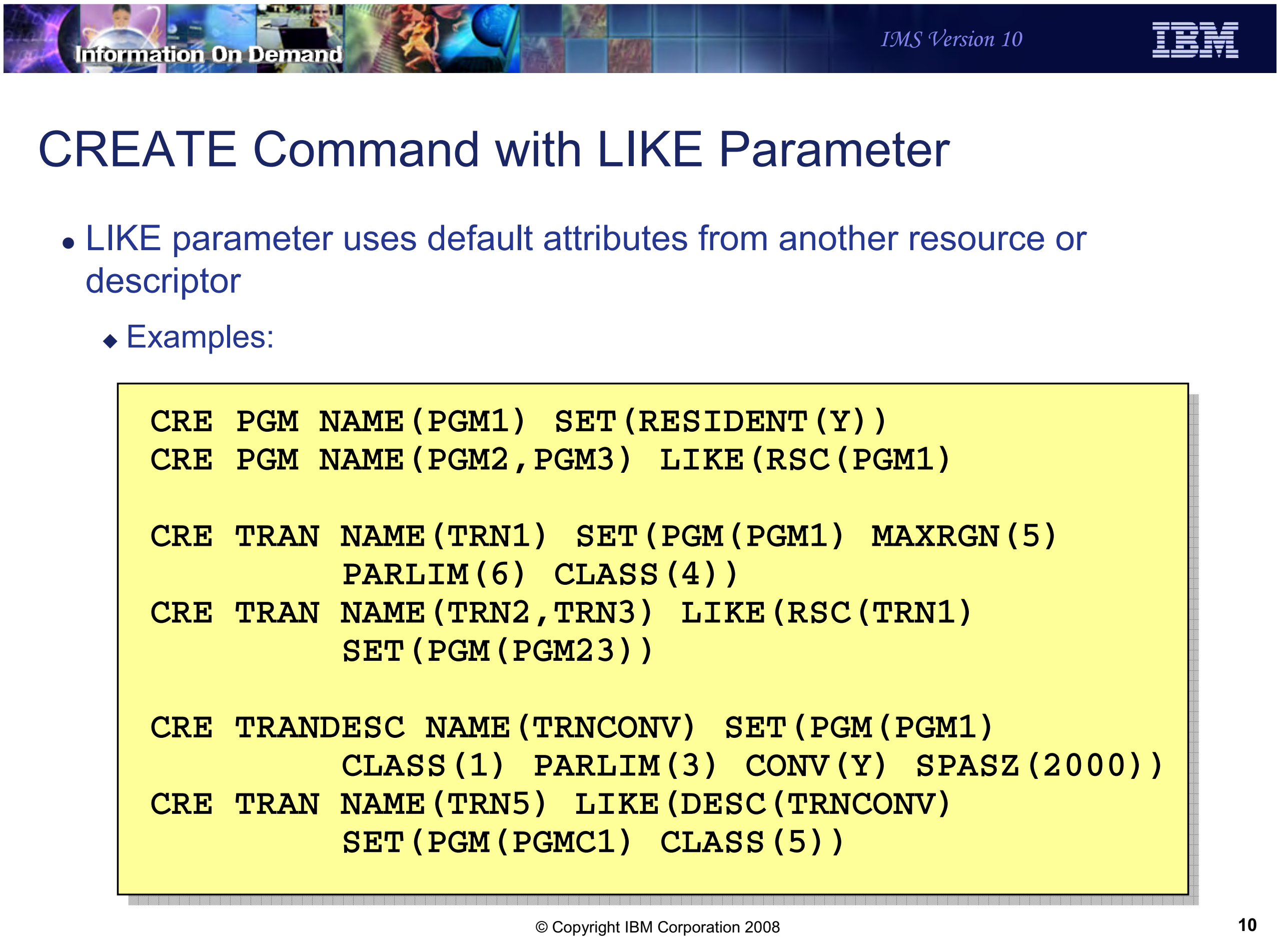
Figure 3-8. Descriptors

CMA01.0

Notes:

A descriptor is a model or template (you will probably see both terms being used) that can be used by the user to define default attributes for resources being created. You can even use a descriptor as a model in creating another descriptor.

IMS provides four descriptors with the product – one for each resource type. Unless the user creates a descriptor with the DEFAULT(Y) attributes, these IMS descriptors are the “current system default descriptors.” The user may create a descriptor which can be specified in the CREATE command or can be set as the current default descriptor, replacing the IMS-defined descriptor. Whichever one is the default is used when the CREATE command does not SET all attributes.



Information On Demand
IMS Version 10

CREATE Command with LIKE Parameter

- LIKE parameter uses default attributes from another resource or descriptor
  - Examples:

```
CRE PGM NAME(PGM1) SET(RESIDENT(Y))
CRE PGM NAME(PGM2,PGM3) LIKE(RSC(PGM1)

CRE TRAN NAME(TRN1) SET(PGM(PGM1) MAXRGN(5)
         PARLIM(6) CLASS(4))
CRE TRAN NAME(TRN2,TRN3) LIKE(RSC(TRN1)
         SET(PGM(PGM23))

CRE TRANDESC NAME(TRNCONV) SET(PGM(PGM1)
         CLASS(1) PARLIM(3) CONV(Y) SPASZ(2000))
CRE TRAN NAME(TRN5) LIKE(DESC(TRNCONV)
         SET(PGM(PGMC1) CLASS(5))
```

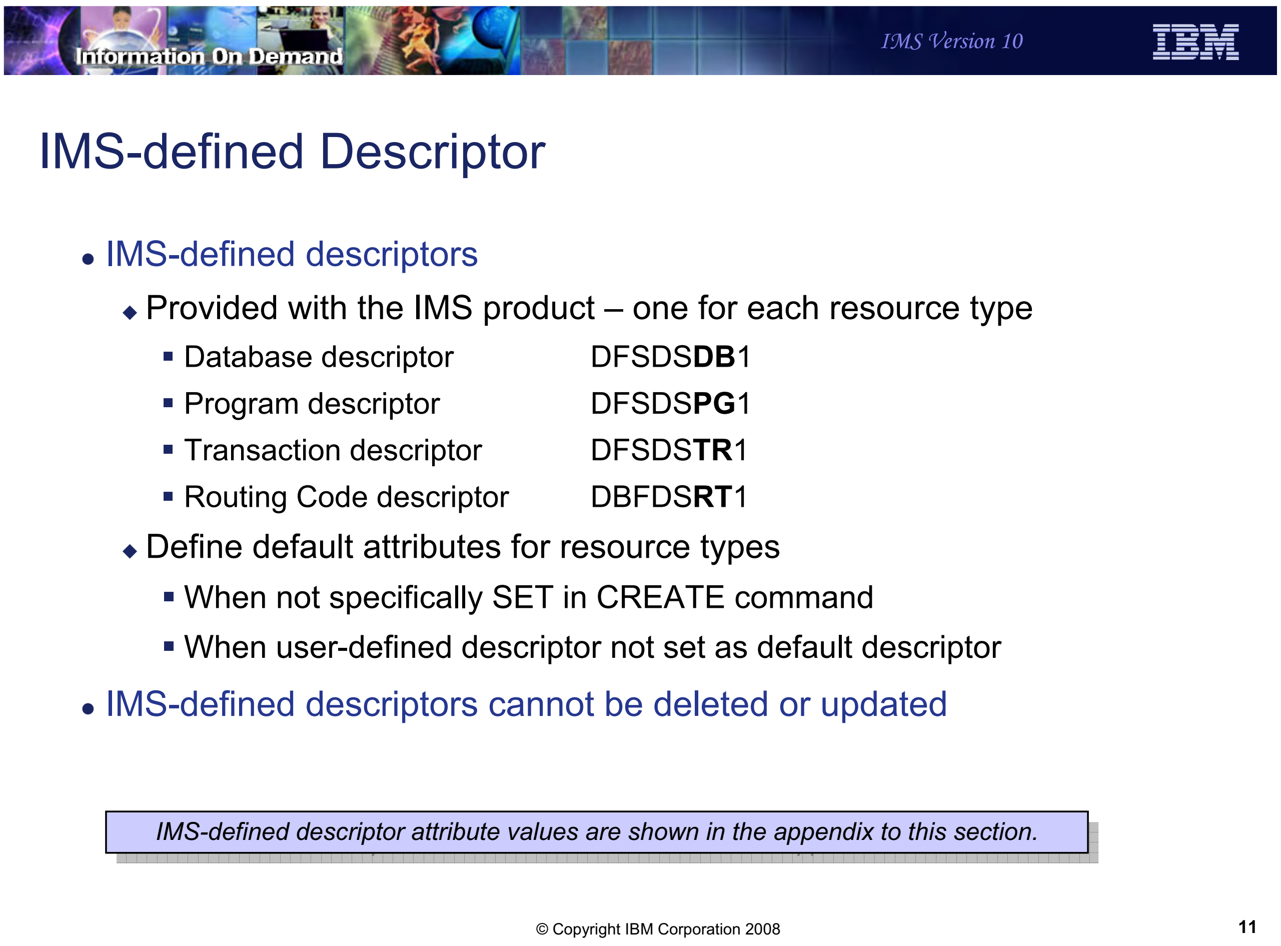
© Copyright IBM Corporation 2008

Figure 3-9. CREATE Command with LIKE Parameter CMA01.0

Notes:

The LIKE parameter may be used to specify a model for the resource or descriptor being created. LIKE refers either to another resource of the same type or to a descriptor for the type. When LIKE is used the new attributes for the new resource or descriptor have the same values as the referenced resource or descriptor unless they specified with the SET parameter. The LIKE parameter allows you to use defaults other than those in the default descriptor.

The first two pairs of commands shown on this page show how the LIKE parameter may refer to another resource. The last pair of commands show how the LIKE parameter may refer to a descriptor.


Information On Demand
IMS Version 10
IBM

IMS-defined Descriptor

- IMS-defined descriptors
  - Provided with the IMS product – one for each resource type

Database descriptor	DFSDS**DB**1
Program descriptor	DFSDS**PG**1
Transaction descriptor	DFSDS**TR**1
Routing Code descriptor	DBFDS**RT**1

  - Define default attributes for resource types
    - When not specifically SET in CREATE command
    - When user-defined descriptor not set as default descriptor
- IMS-defined descriptors cannot be deleted or updated

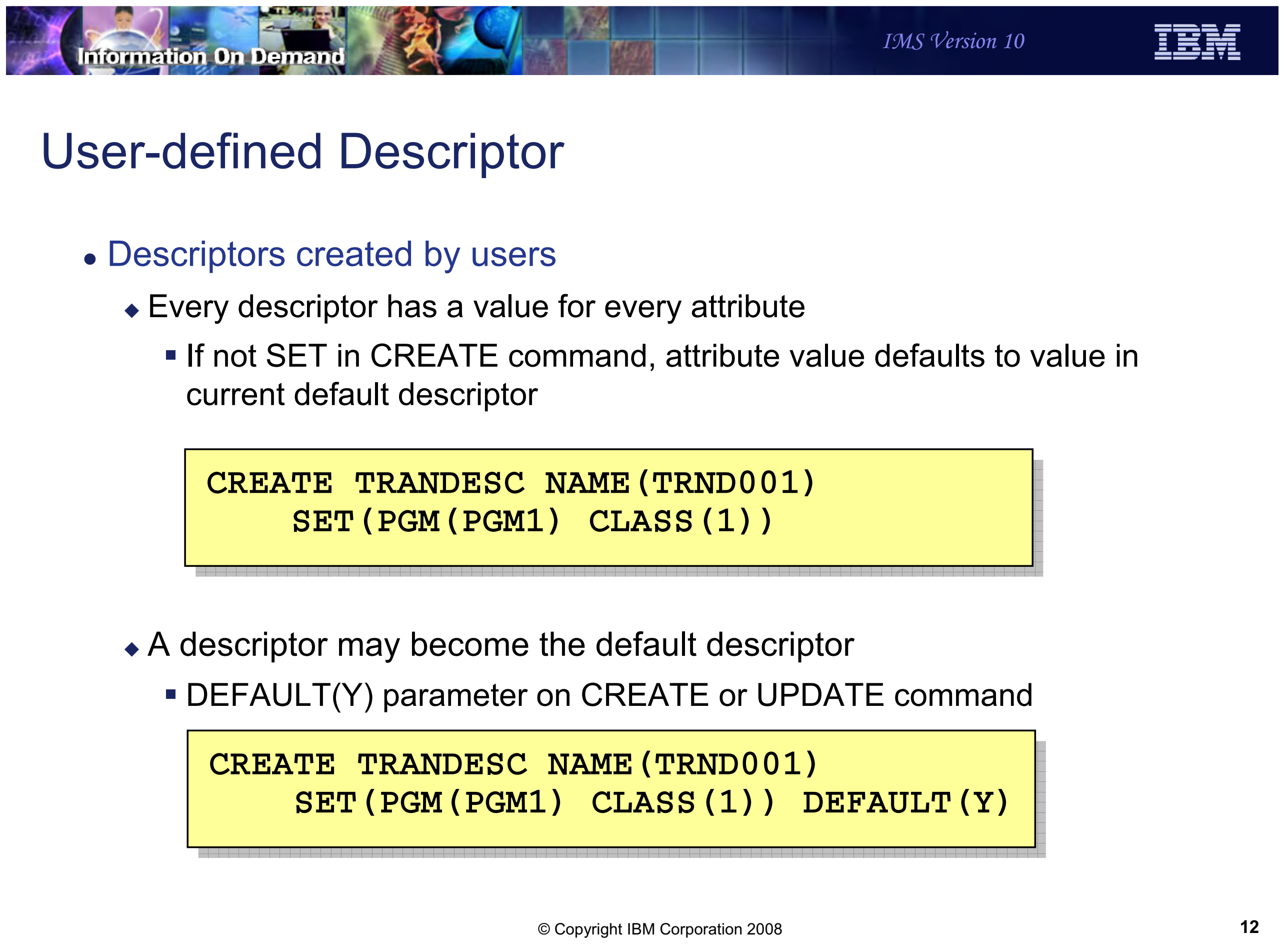
IMS-defined descriptor attribute values are shown in the appendix to this section.

© Copyright IBM Corporation 2008

Figure 3-10. IMS-defined Descriptor CMA01.0

Notes:

These are the four IMS-defined descriptors. They have similar names with two letters identifying the resource type to which the descriptor applies. These are DFSDSxx1, where xx is DB, PG, TR. or RT. These IMS descriptors cannot be updated or deleted except to make it the default if you have previously made a user-defined descriptor the default and want to go back to the IMS descriptor.


Information On Demand
IMS Version 10

User-defined Descriptor

- Descriptors created by users
  - Every descriptor has a value for every attribute
    - If not SET in CREATE command, attribute value defaults to value in current default descriptor

```
CREATE TRANDESC NAME(TRND001)
    SET(PGM(PGM1) CLASS(1))
```

  - A descriptor may become the default descriptor
    - DEFAULT(Y) parameter on CREATE or UPDATE command

```
CREATE TRANDESC NAME(TRND001)
    SET(PGM(PGM1) CLASS(1)) DEFAULT(Y)
```

© Copyright IBM Corporation 2008

Figure 3-11. User-defined Descriptor CMA01.0

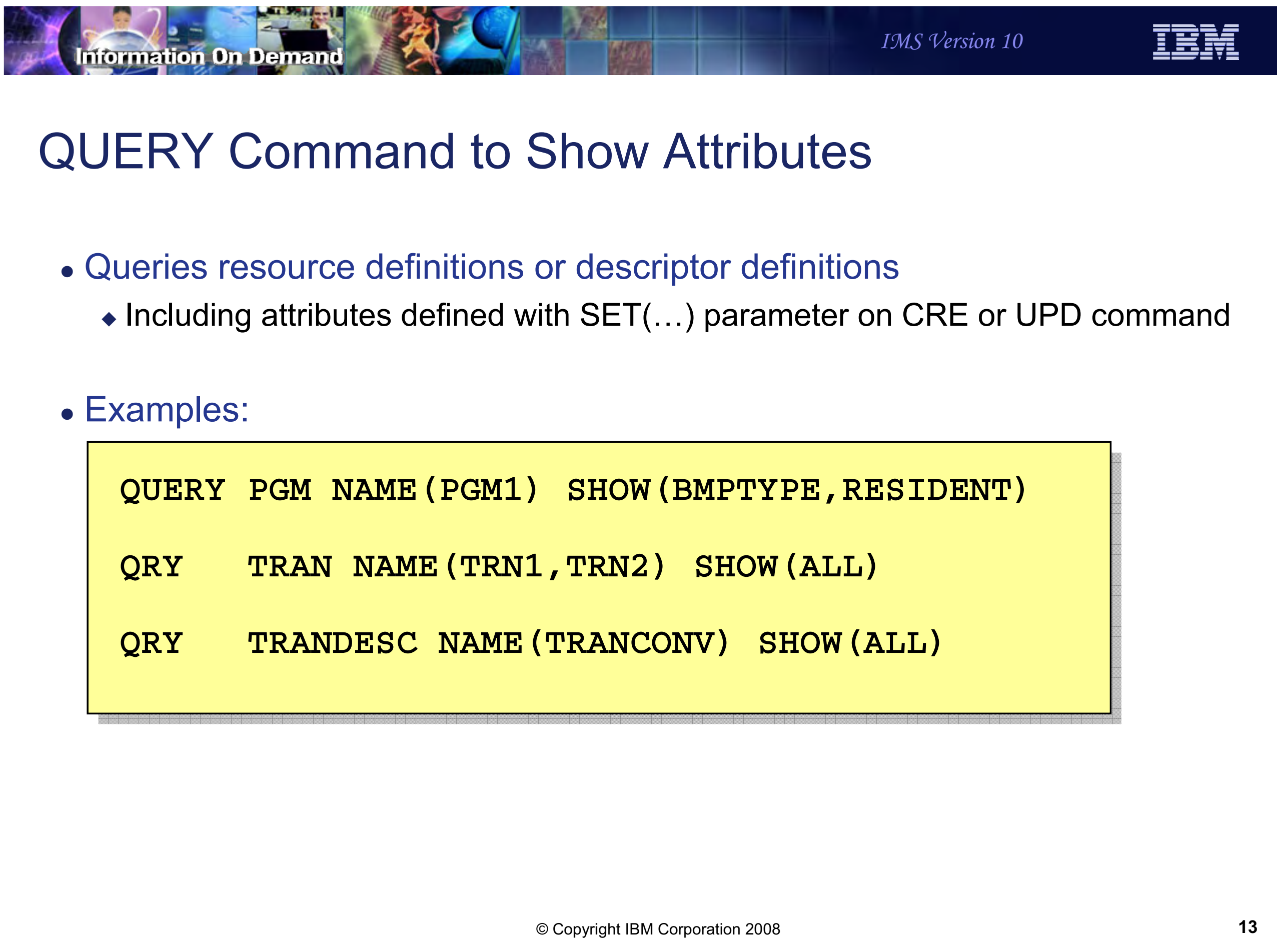
Notes:

You can have as many user-defined descriptors as you like. User descriptors are created, updated, and deleted just like resources. They have the same attributes as resources. If, when creating a user descriptor, you do not SET the value of the attribute, then it will default to whatever that value is in the current default descriptor for that resource type.

When a user descriptor is created, you may make it the default descriptor by including the parameter DEFAULT(Y). Or, you may later make it the default using an UPDATE command.

The keyword for creating, updating, deleting, or querying descriptors are DBDESC, PGMDESC,TRANDESC, and RTCDESC. In the first example on this slide, the user has created a transaction descriptor named TRND001 with an attribute of CLASS(1). The program PGM1 will be invoked for transactions created with this descriptor unless this is overridden when they are created. The other attributes for TRND001 will be taken from whatever the current default transaction descriptor is, perhaps DFSDSTR1.

In the second example, the same descriptor is created, but it has been made the default transaction descriptor.


Information On Demand
IMS Version 10
IBM

QUERY Command to Show Attributes

- Queries resource definitions or descriptor definitions
  - Including attributes defined with SET(…) parameter on CRE or UPD command
- Examples:

```
QUERY PGM NAME(PGM1) SHOW(BMPTYPE,RESIDENT)

QRY   TRAN NAME(TRN1,TRN2) SHOW(ALL)

QRY   TRANDESC NAME(TRANCONV) SHOW(ALL)
```

© Copyright IBM Corporation 2008

13

Figure 3-12. QUERY Command to Show Attributes CMA01.0

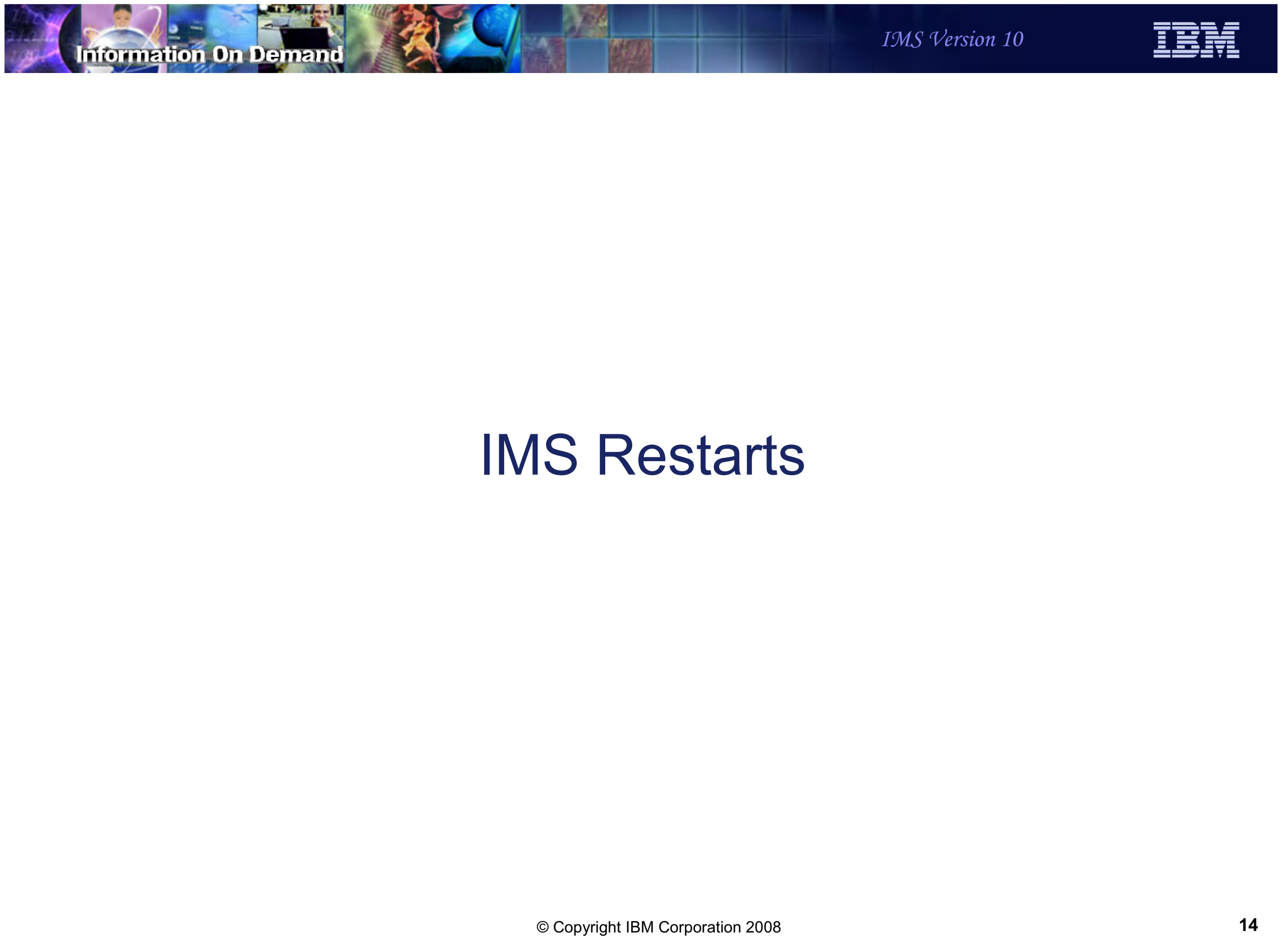
Notes:

QUERY commands may be used to show attributes of resources and descriptors. The SHOW parameter may specify attributes to be shown or SHOW(ALL) may be used to show all of the attributes.

In the first example, only the BMPTYPE and RESIDENT attributes will be returned in the response.

In the second example, all attributes of transactions TRN1 and TRN2 will be returned.

In the third example, all attributes of the TRANCONV transaction descriptor will be returned.

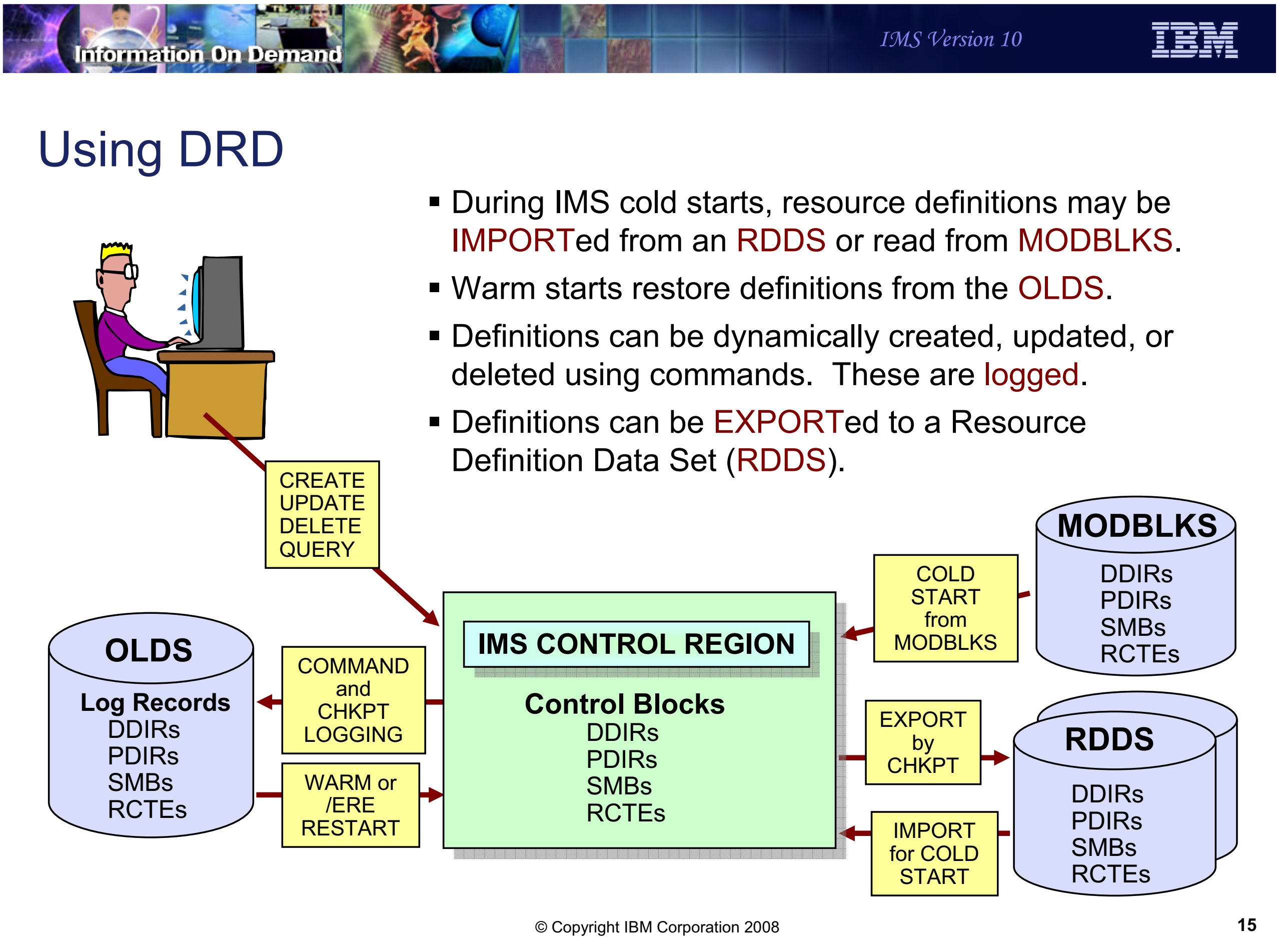

Information On Demand
IMS Version 10

IMS Restarts

© Copyright IBM Corporation 2008

Figure 3-13. [No title] CMA01.0

Notes:



Information On Demand

IMS Version 10

IBM

Using DRD

- During IMS cold starts, resource definitions may be IMPORTed from an RDDS or read from MODBLKS.
- Warm starts restore definitions from the OLDS.
- Definitions can be dynamically created, updated, or deleted using commands. These are logged.
- Definitions can be EXPORTed to a Resource Definition Data Set (RDDS).

CREATE
UPDATE
DELETE
QUERY

OLDS
Log Records
DDIRs
PDIRs
SMBs
RCTEs

COMMAND and CHKPT LOGGING

WARM or /ERE RESTART

IMS CONTROL REGION

Control Blocks
DDIRs
PDIRs
SMBs
RCTEs

COLD START from MODBLKS

MODBLKS
DDIRs
PDIRs
SMBs
RCTEs

EXPORT by CHKPT

IMPORT for COLD START

RDDS
DDIRs
PDIRs
SMBs
RCTEs

© Copyright IBM Corporation 2008

15

Figure 3-14. Using DRD

CMA01.0

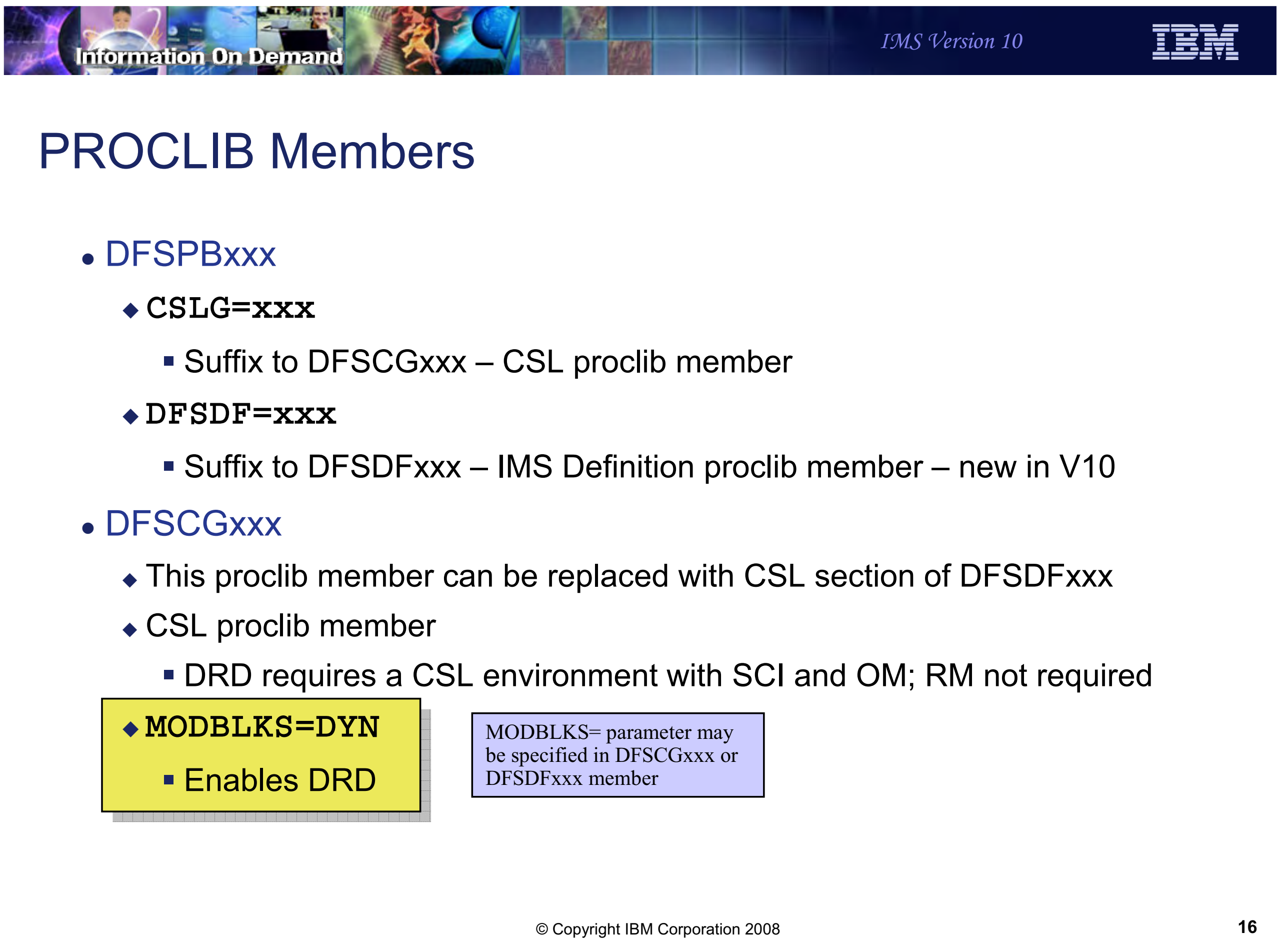
Notes:

During an IMS cold start, IMS imports (loads) all of it resource definitions. In a non-DRD environment, these are always from the active MODBLKS library as determined by MODSTAT or OLCSTAT. In a DRD environment, IMS may “import” resource definitions from MODBLKS, or both resource and descriptor definitions from a new type of data set called the Resource Definition Data Set (RDDS). The process is called AUTOIMPORT and where the definitions are loaded from is determined by an execution time option in DFSDFxxx. MODBLKS, of course, does not contain any descriptors.

During an IMS warm start, definitions are rebuilt from the restart checkpoint on the OLDS. During an IMS emergency restart, definitions are rebuilt from the checkpoint log records plus additional log records created when the attributes or status of those definitions are changed. This is true for both DRD and non-DRD systems.

After restart is complete, in a DRD system, all definitions are “exported” to an RDDS (enabling them to be imported again at the next cold start). This process is called AUTOEXPORT and occurs at every system checkpoint, including the initialization and shut down checkpoints. Definitions are also logged during system checkpoint for restart purposes.

During execution, an operator using the OM interface may enter commands to create, update, or delete resource and descriptor definitions and status. CREATE and DELETE are available only in a DRD environment. UPDATE is available in either environment, although it is not as complete in terms of what can be updated. These updates are also logged for use during restart.



Information On Demand
IMS Version 10
IBM

PROCLIB Members

- DFSPBxxx
  - **`CSLG=xxx`**
    - Suffix to DFSCGxxx – CSL proclib member
  - **`DFSDF=xxx`**
    - Suffix to DFSDFxxx – IMS Definition proclib member – new in V10
- DFSCGxxx
  - This proclib member can be replaced with CSL section of DFSDFxxx
  - CSL proclib member
    - DRD requires a CSL environment with SCI and OM; RM not required
  - **`MODBLKS=DYN`**
    - Enables DRD

MODBLKS= parameter may be specified in DFSCGxxx or DFSDFxxx member

© Copyright IBM Corporation 2008

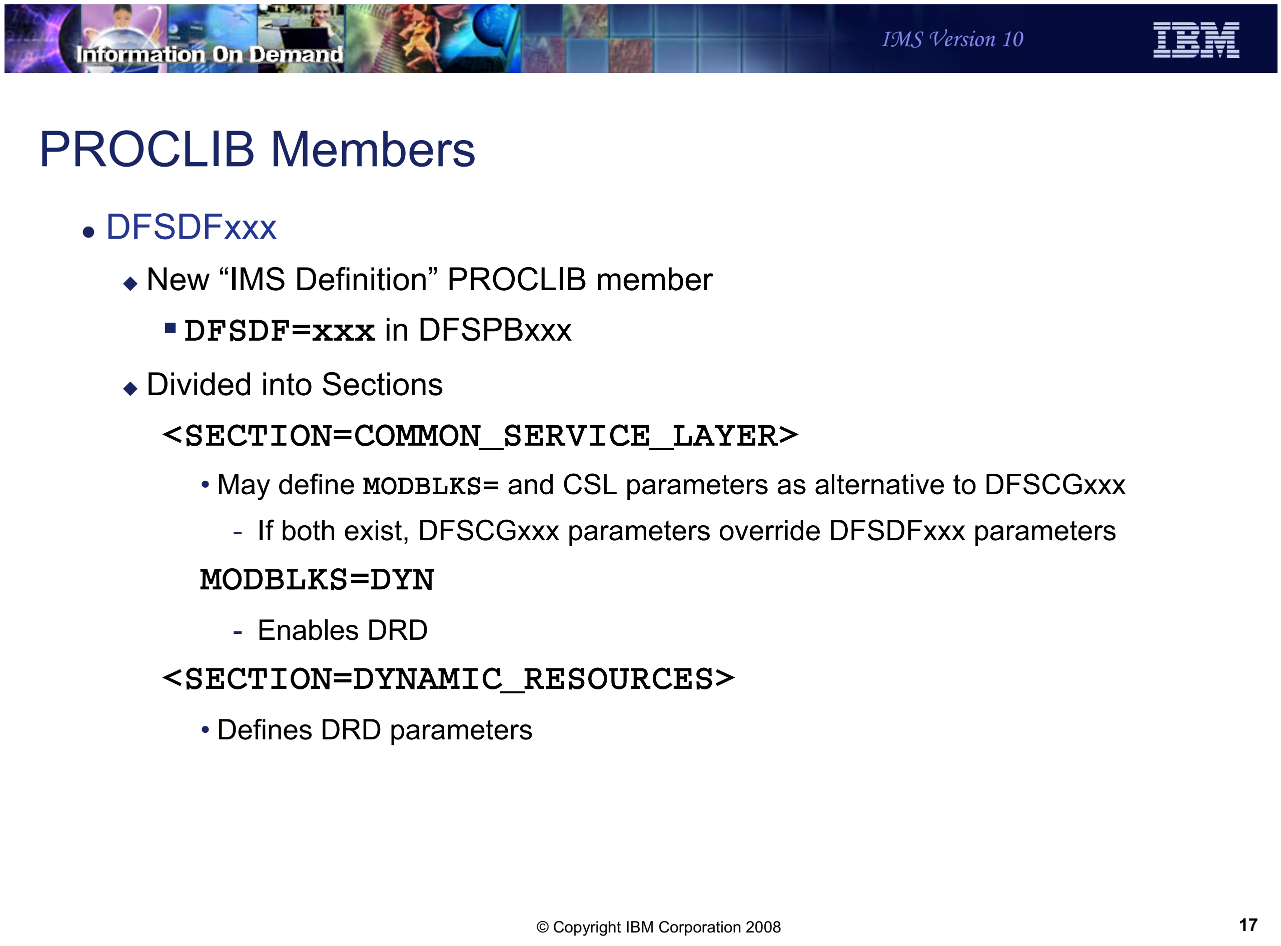
Figure 3-15. PROCLIB Members CMA01.0

Notes:

The CSLG=xxx parameter in DFSPBxxx points to the CSL global proclib member (DFSCGxxx) and the DFSDF=xxx parameter points to the IMS System Definition proclib member which is new in IMS Version 10.

The entire contents of the CSL member can be moved to the CSL section of DFSDFxxx. If both DFSCGxxx and a CSL Section in DFSDFxxx exist, the DFSCGxxx member takes precedence. It is therefore necessary to remove the CSLG parameter from DFSDFxxx if you are coding a CSL section in DFSDFxxx.

DFSCGxxx contains the new parameter MODBLKS=DYN which enables DRD. MODBLKS defaults to MODBLKS=OLC. When MODBLKS=DYN is specified, online change cannot be used for the resources stored in MODBLKS (programs, transactions, routing codes, and databases). Online change for ACBLIB and FORMAT may be used with either MODBLKS=OLC or MODBLKS=OLC.


Information On Demand
IMS Version 10

PROCLIB Members

- DFSDFxxx
  - New “IMS Definition” PROCLIB member
    - `DFSDF=xxx` in DFSPBxxx
  - Divided into Sections

    `<SECTION=COMMON_SERVICE_LAYER>`
    - May define `MODBLKS=` and CSL parameters as alternative to DFSCGxxx
      - If both exist, DFSCGxxx parameters override DFSDFxxx parameters

    `MODBLKS=DYN`
      - Enables DRD

    `<SECTION=DYNAMIC_RESOURCES>`
    - Defines DRD parameters

© Copyright IBM Corporation 2008

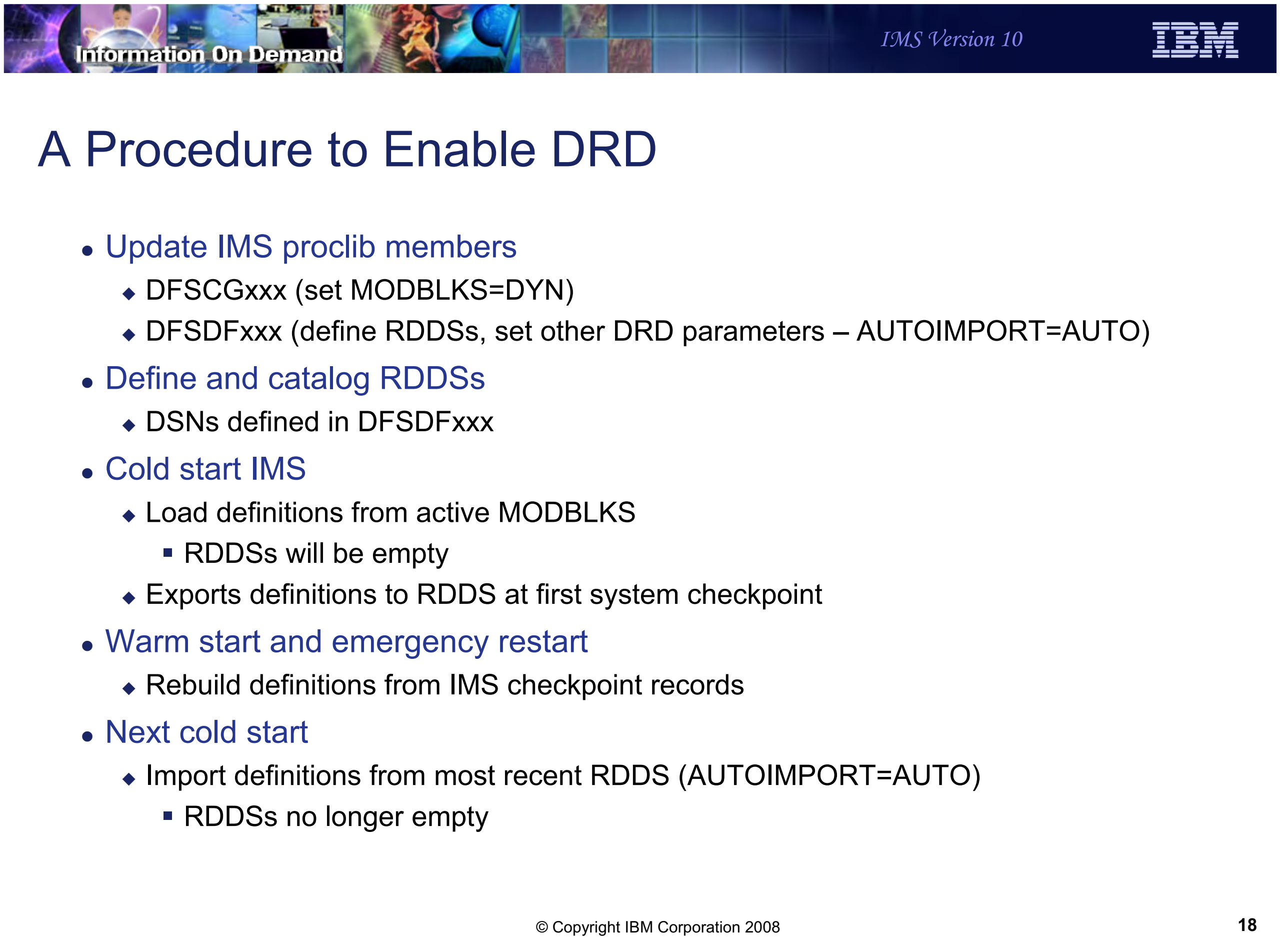
Figure 3-16. PROCLIB Members CMA01.0

Notes:

DFSDFxxx is a new proclib member in IMS Version 10. It is required for DRD but can also be used in place of DFSCGxxx and DFSSQxxx. The user may define four sections in this member.

DRD is enabled by specifying MODBLKS=DYN in the COMMON-SERVICE_LAYER section.

If MODBLKS=DYN is coded in the CSL section, the Dynamic Resources section defines all the DRD parameters .


Information On Demand
IMS Version 10

A Procedure to Enable DRD

- Update IMS proclib members
  - DFSCGxxx (set MODBLKS=DYN)
  - DFSDFxxx (define RDDSs, set other DRD parameters – AUTOIMPORT=AUTO)
- Define and catalog RDDSs
  - DSNs defined in DFSDFxxx
- Cold start IMS
  - Load definitions from active MODBLKS
    - RDDSs will be empty
  - Exports definitions to RDDS at first system checkpoint
- Warm start and emergency restart
  - Rebuild definitions from IMS checkpoint records
- Next cold start
  - Import definitions from most recent RDDS (AUTOIMPORT=AUTO)
    - RDDSs no longer empty

© Copyright IBM Corporation 2008

Figure 3-17. A Procedure to Enable DRD CMA01.0

Notes:

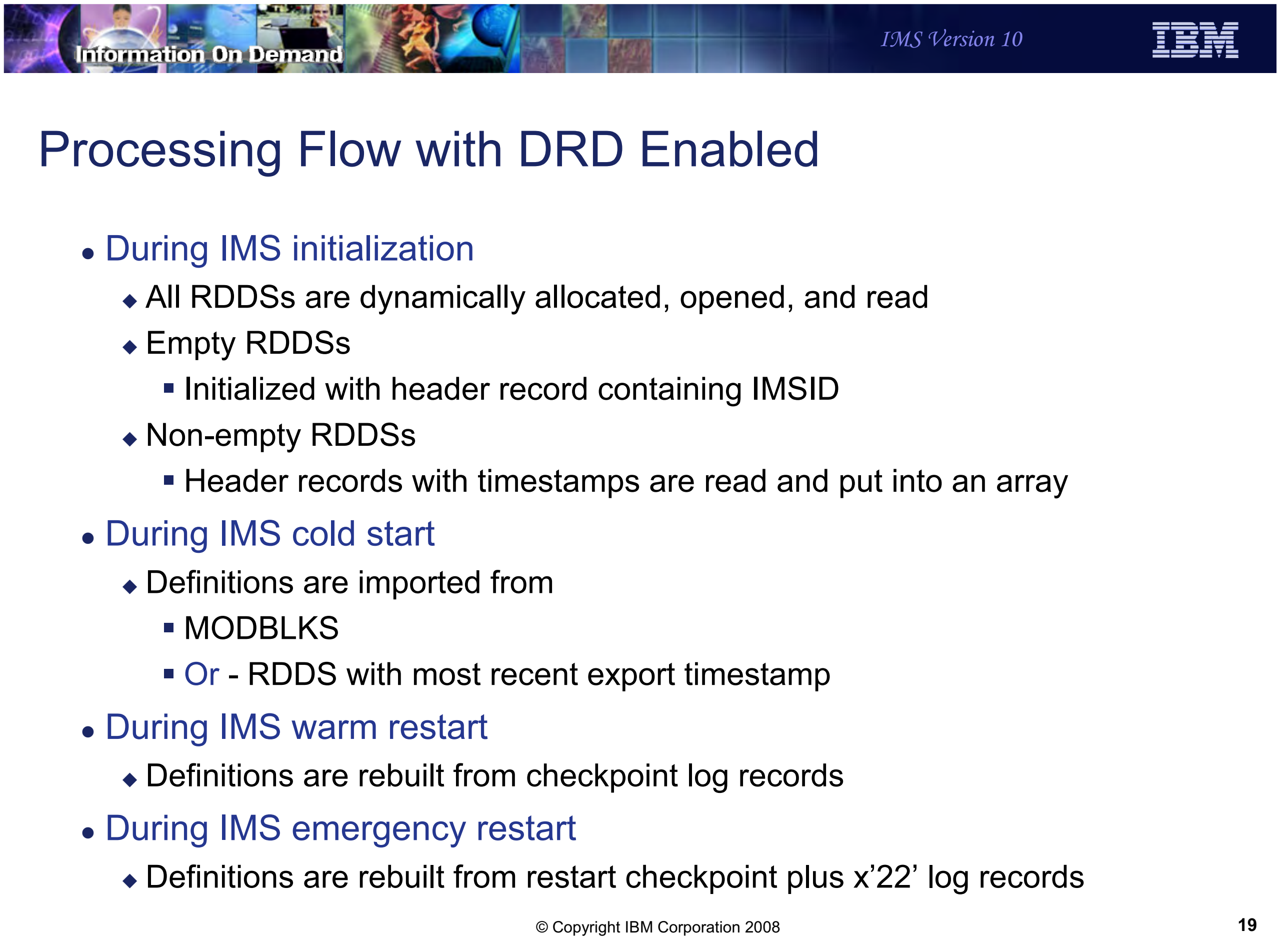
DRD is “enabled” simply by coding MODBLKS=DYN (meaning MODBLKS changes are dynamic – not by online change) in either DFSCGxxx or the CSL Section of DFSDFxxx. Technically it is not necessary to code any other parameters, however, if RDDSs are not defined, export and importing from an RDDS cannot be done. Import could only be done from MODBLKS and OLC would be disabled. Creates, updates, and deletes could be done but they couldn’t be exported.

Assuming this is not what you want, you must create a DFSDFxxx proclib member with a DRD section containing the desired RDDSDSN, AUTOIMPORT, AUTOEXPORT, and DCLWA parameters and the parameters for handling errors. Since all of these parameter except RDDSDSN have reasonable defaults, it may not be necessary to code anything except RDDSDSN.

You can then cold start IMS. Assuming AUTOIMPORT=AUTO, IMS will detect that the RDDSs are empty and import the definitions from the active MODBLKS. At the completion of cold start, AUTOEXPORT=AUTO or RDDS will cause IMS to export these MODBLKS definitions to an RDDS.

After cold starting IMS the first time successfully, and before the next cold start, you can remove the MODBLKS DD statements from the control region JCL. This is not, however, necessary and you may want to leave them in until you are certain you will never want to import from them again.

At the next IMS cold start, the latest version of the definition in RDDS will be imported. At the next warm or emergency restart, the definitions will be rebuilt from the logs.


Information On Demand
IMS Version 10
IBM

Processing Flow with DRD Enabled

- During IMS initialization
  - All RDDSs are dynamically allocated, opened, and read
  - Empty RDDSs
    - Initialized with header record containing IMSID
  - Non-empty RDDSs
    - Header records with timestamps are read and put into an array
- During IMS cold start
  - Definitions are imported from
    - MODBLKS
    - Or - RDDS with most recent export timestamp
- During IMS warm restart
  - Definitions are rebuilt from checkpoint log records
- During IMS emergency restart
  - Definitions are rebuilt from restart checkpoint plus x’22’ log records

© Copyright IBM Corporation 2008

19

Figure 3-18. Processing Flow with DRD Enabled CMA01.0

Notes:

This slide shows the general flow when DRD is enabled.

During IMS initialization:

- All RDDSs defined in DFSDFxxx are dynamically allocated and read
- If they are empty, IMS initializes them with a header record containing the IMSID
- If they are not empty, IMS builds an array in storage of the available (i.e., successfully allocated and read) RDDSs. These RDDSs will be used in round-robin fashion by export

During IMS cold start:

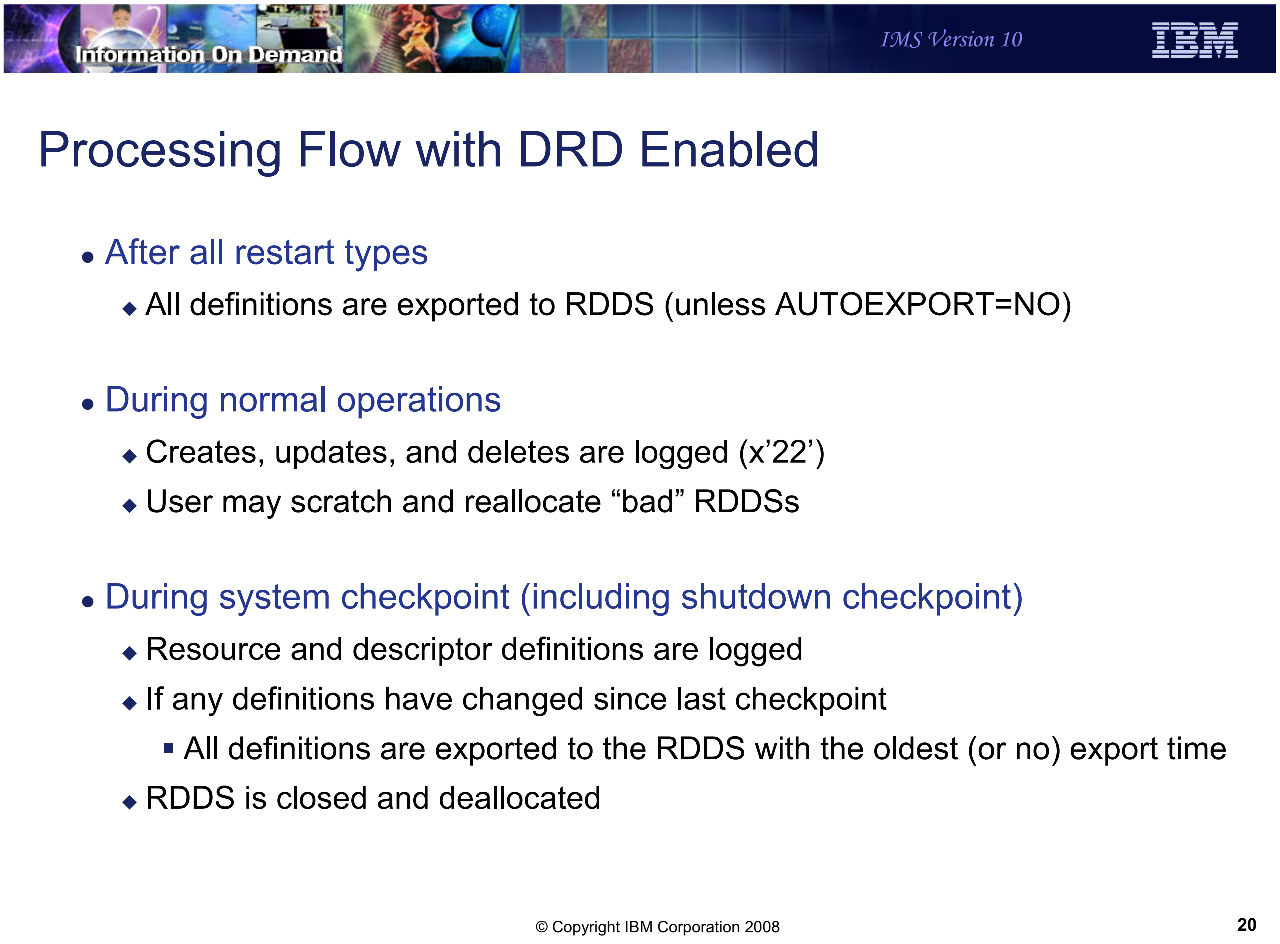
- Definitions are imported from the most recent RDDS or from MODBLKS

During warm start:

- Definitions are rebuilt from the checkpoint log records

During emergency restart:

- Definitions are rebuilt from restart checkpoint log records plus and updates made since that checkpoint (x’22’ log records)



Information On Demand

IMS Version 10

IBM

Processing Flow with DRD Enabled

- After all restart types
  - All definitions are exported to RDDS (unless AUTOEXPORT=NO)
- During normal operations
  - Creates, updates, and deletes are logged (x’22’)
  - User may scratch and reallocate “bad” RDDSs
- During system checkpoint (including shutdown checkpoint)
  - Resource and descriptor definitions are logged
  - If any definitions have changed since last checkpoint
    - All definitions are exported to the RDDS with the oldest (or no) export time
  - RDDS is closed and deallocated

© Copyright IBM Corporation 2008

Figure 3-19. Processing Flow with DRD Enabled CMA01.0

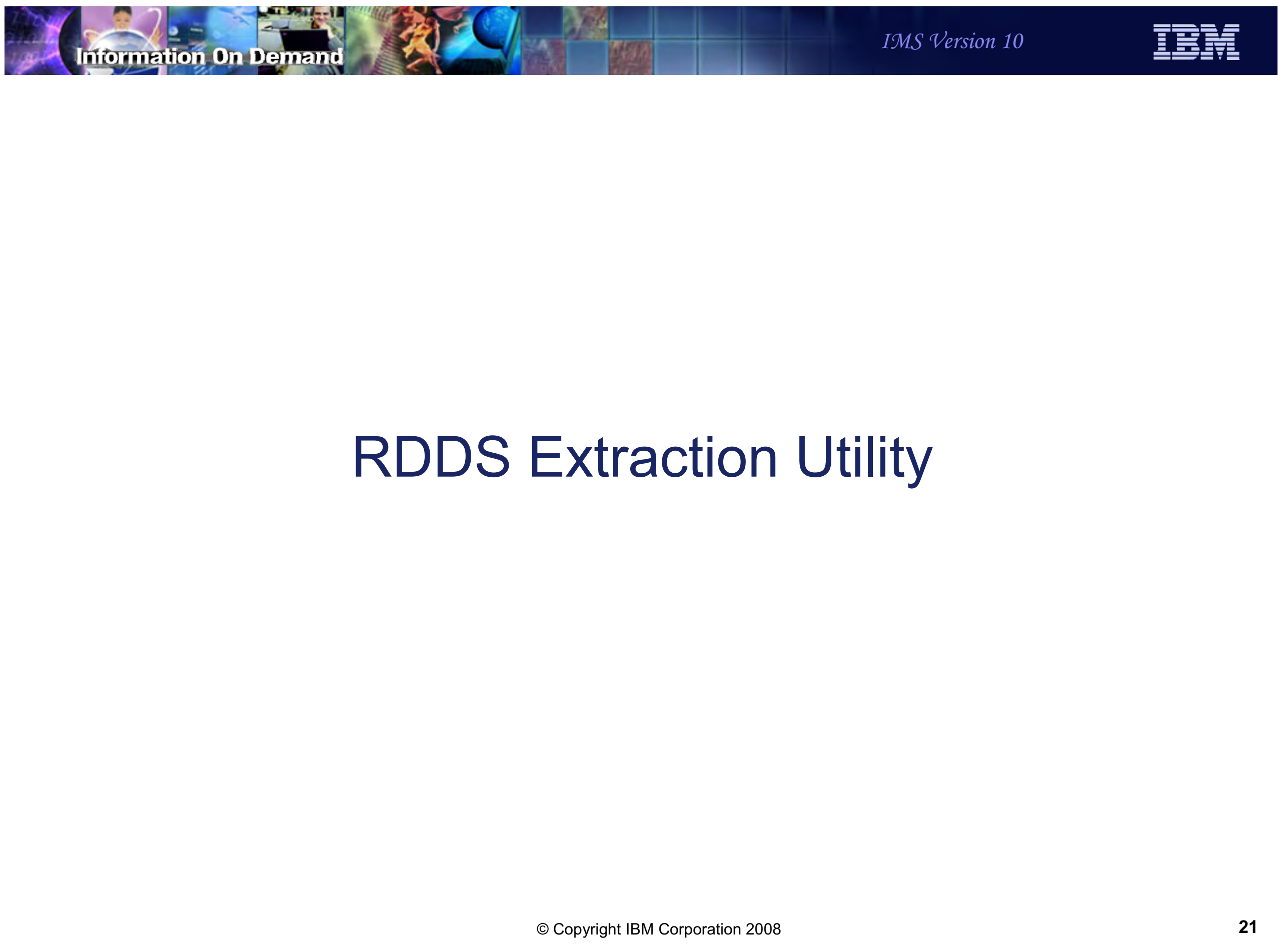
Notes:

After any type of restart completes. AUTOEXPORT will export definitions to the oldest RDDS. These (or later exports) then become available at the next IMS cold start.

During normal operations, all creates, updates, and deletes are logged in x’22’ log records. If an RDDS cannot be read, an error message is issued. You may scratch and reallocate this RDDS. This will add it back to the RDDSs that are used by IMS. You cannot add new RDDS data sets while IMS is executing. Only those defined in the DFSDFxxx member at start up will be used.

At initialization time, IMS reads the RDDSDSN parameter in the DFSDFxxx member. It opens the specified data sets and reads their headers.

At system checkpoint times, if any definitions have change since the last checkpoint, all definitions are exported to the next RDDS. The RDDS is closed and deallocated after the definitions are written to it.


Information On Demand
IMS Version 10
IBM

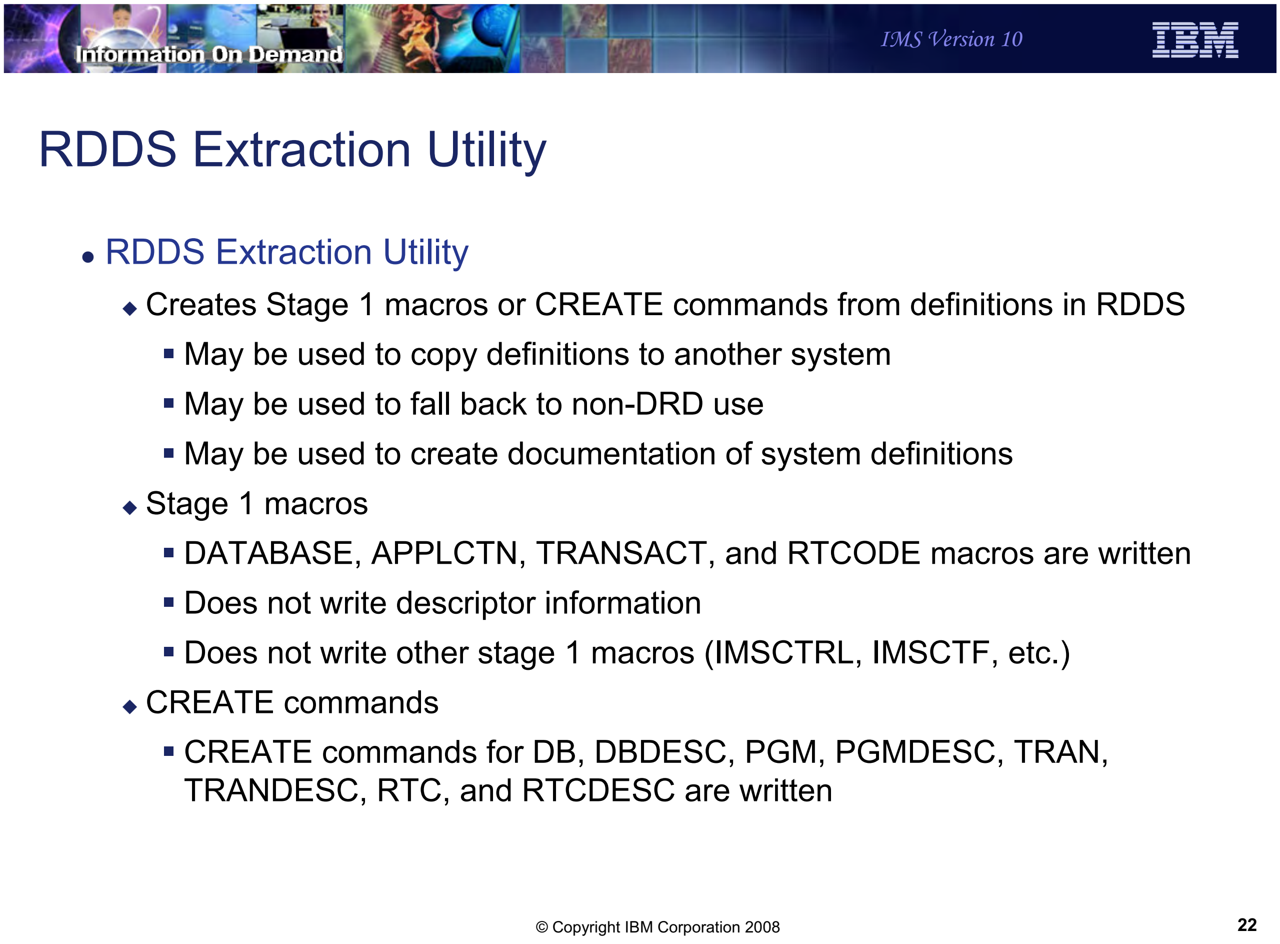
RDDS Extraction Utility

© Copyright IBM Corporation 2008

21

Figure 3-20. [No title] CMA01.0

Notes:



Information On Demand

IMS Version 10

RDDS Extraction Utility

- RDDS Extraction Utility
  - Creates Stage 1 macros or CREATE commands from definitions in RDDS
    - May be used to copy definitions to another system
    - May be used to fall back to non-DRD use
    - May be used to create documentation of system definitions
  - Stage 1 macros
    - DATABASE, APPLCTN, TRANSACT, and RTCODE macros are written
    - Does not write descriptor information
    - Does not write other stage 1 macros (IMSCTRL, IMSCTF, etc.)
  - CREATE commands
    - CREATE commands for DB, DBDESC, PGM, PGMDESC, TRAN, TRANDESC, RTC, and RTCDESC are written

© Copyright IBM Corporation 2008

Figure 3-21. RDDS Extraction Utility CMA01.0

Notes:

The RDDS Extraction Utility (DFSURDD0) is a batch utility which can be used to convert the resource definitions in an RDDS into either stage 1 macro statements or type-2 CREATE commands.

This provides a convenient way to take the definitions for one system and copy them to another system. It also may be used fall back from the use of DRD to the use of MODBLKS. Finally, the creation of stage 1 macros may be used as documentation of the current definitions.

If stage 1 macros are produced, descriptor information in the RDDS is not created. Of course, it only produces stage 1 macros for MODBLKS resources (DATABASE, APPLCTN, TRANSACT, and RTCODE macros).

If CREATE commands are produced, commands for creating descriptors are included.

Security Considerations

- All DRD commands are type-2 and can only be entered through the Operations Manager interface

- Command security for type-2 commands can only be performed by the Operations Manager
  - In OM Initialization proclib member (CSLOIxxx)
    - CMDSEC=N|R|E|A

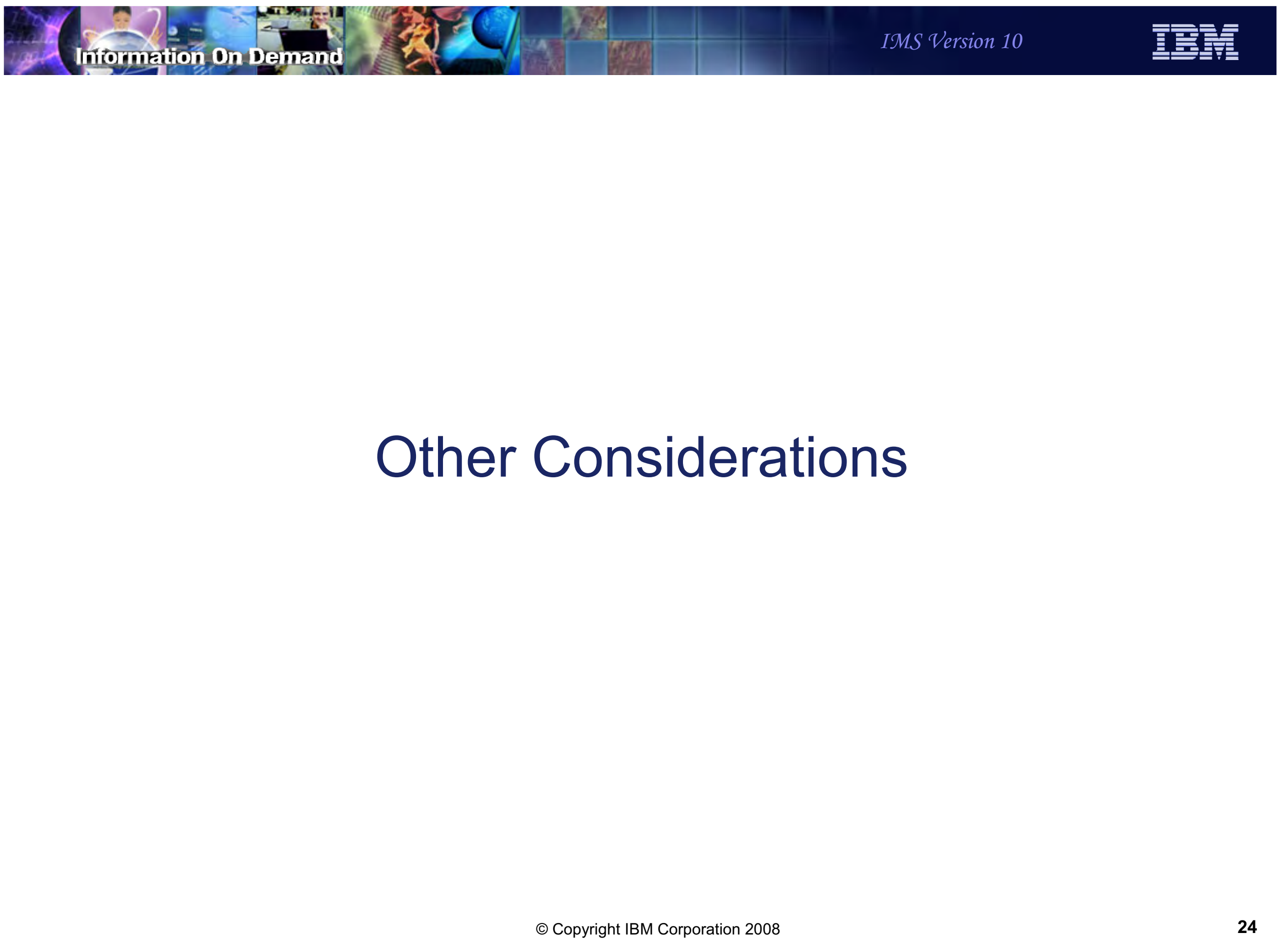
• N	do not perform command authorization in OM (default)
• R	call RACF for authorization
• E	call user security exit for authorization
• A	call both RACF (first) and the user security exit

© Copyright IBM Corporation 2008

Figure 3-22. Security Considerations CMA01.0

Notes:

All DRD commands are type-2 commands. That means they are only submitted through the OM interface. Therefore, only OM can perform authorization processing for these commands. In the OM initialization proclib member (DFSOIxxx), the CMDSEC parameter determines what type of security checking should be done for commands submitted through OM. There is not change in the regard for the DRD commands. This slide is just a quick review of what is already available.


Information On Demand
IMS Version 10
IBM

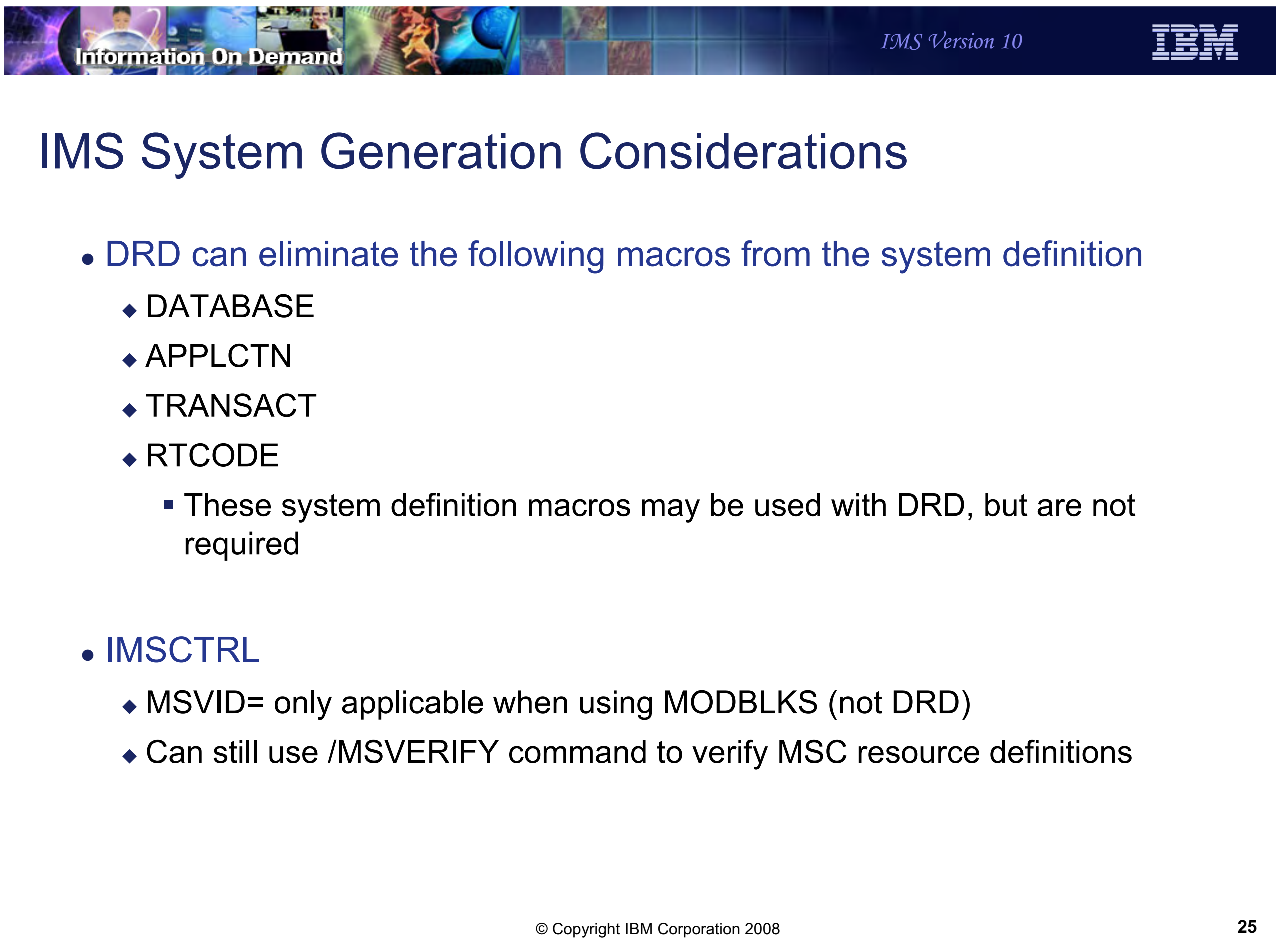
Other Considerations

© Copyright IBM Corporation 2008

Figure 3-23. [No title] CMA01.0

Notes:

There are some other considerations when enabling DRD.



Information On Demand

IMS Version 10

IMS System Generation Considerations

- DRD can eliminate the following macros from the system definition
  - DATABASE
  - APPLCTN
  - TRANSACT
  - RTCODE
    - These system definition macros may be used with DRD, but are not required
- IMSCTRL
  - MSVID= only applicable when using MODBLKS (not DRD)
  - Can still use /MSVERIFY command to verify MSC resource definitions

© Copyright IBM Corporation 2008

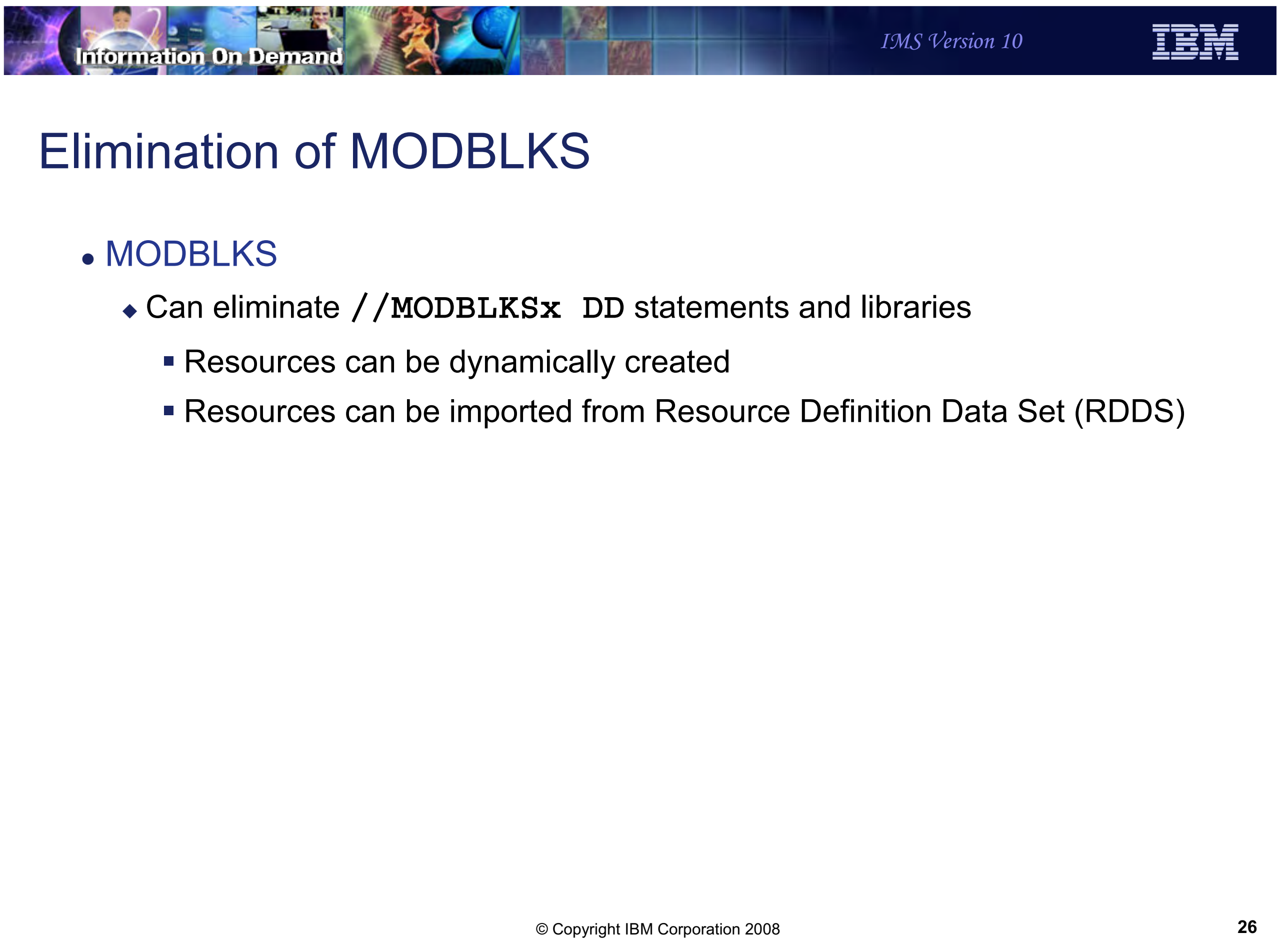
25

Figure 3-24. IMS System Generation Considerations CMA01.0

Notes:

There are a few sysgen considerations when implementing DRD, although nothing is required.

- You can eliminate the DATABASE, APPLCTN, TRANSACT, and RTCODE macros. If you do not eliminate them, their definitions will reside in the MODBLKS data set, but IMS execution will not read them from MODBLKS
- The MSC Verification ID (MSVID) is not applicable when DRD is enabled. You can, however, continue to use /MSV to verify your MSC definitions.


Information On Demand
IMS Version 10
IBM

Elimination of MODBLKS

- MODBLKS
  - Can eliminate `//MODBLKSx DD` statements and libraries
    - Resources can be dynamically created
    - Resources can be imported from Resource Definition Data Set (RDDS)

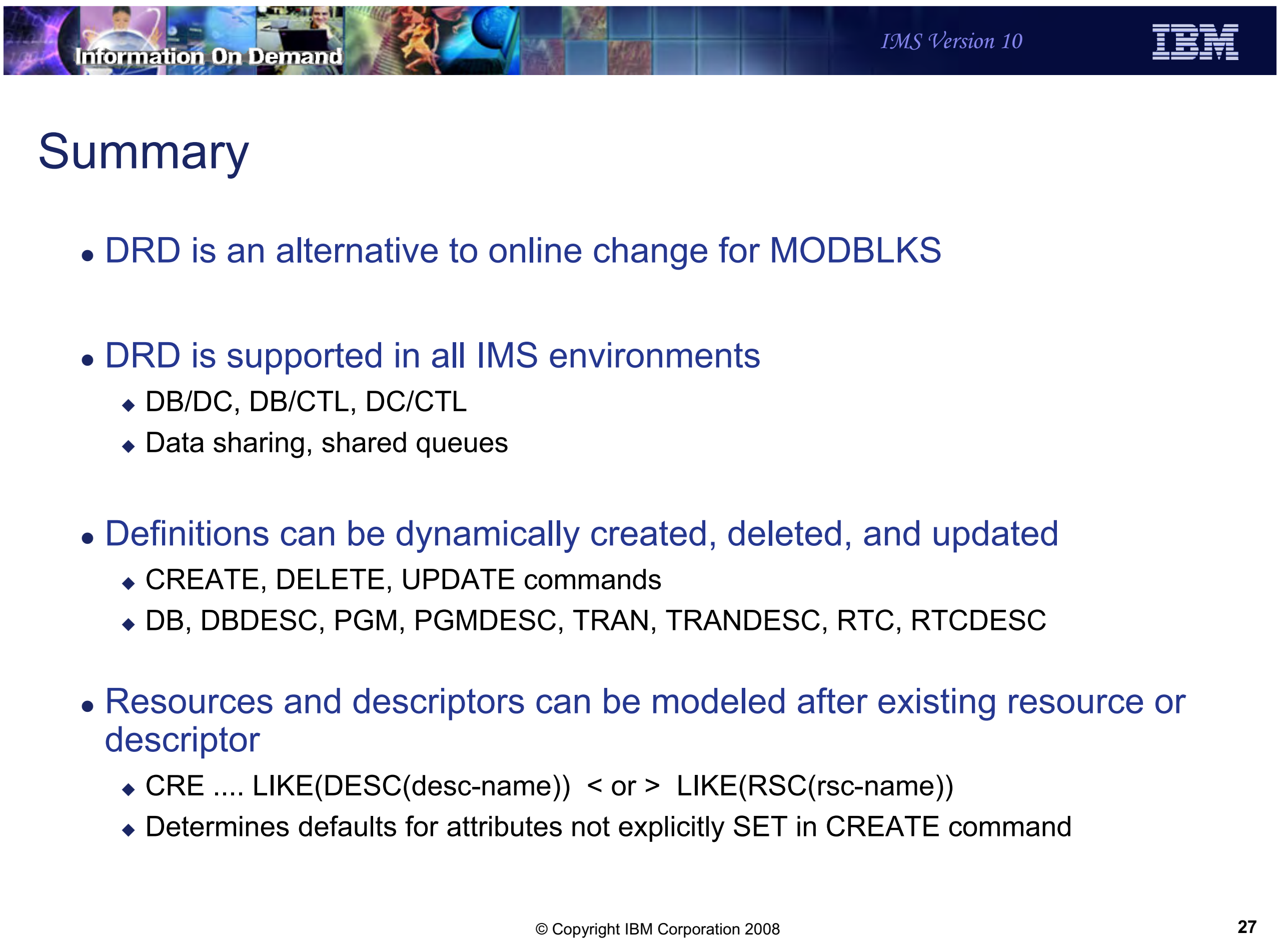
© Copyright IBM Corporation 2008

26

Figure 3-25. Elimination of MODBLKS CMA01.0

Notes:

After successfully importing from MODBLKS the first time, and exporting to an RDDS, the MODBLKS data sets can be eliminated from the control region JCL.



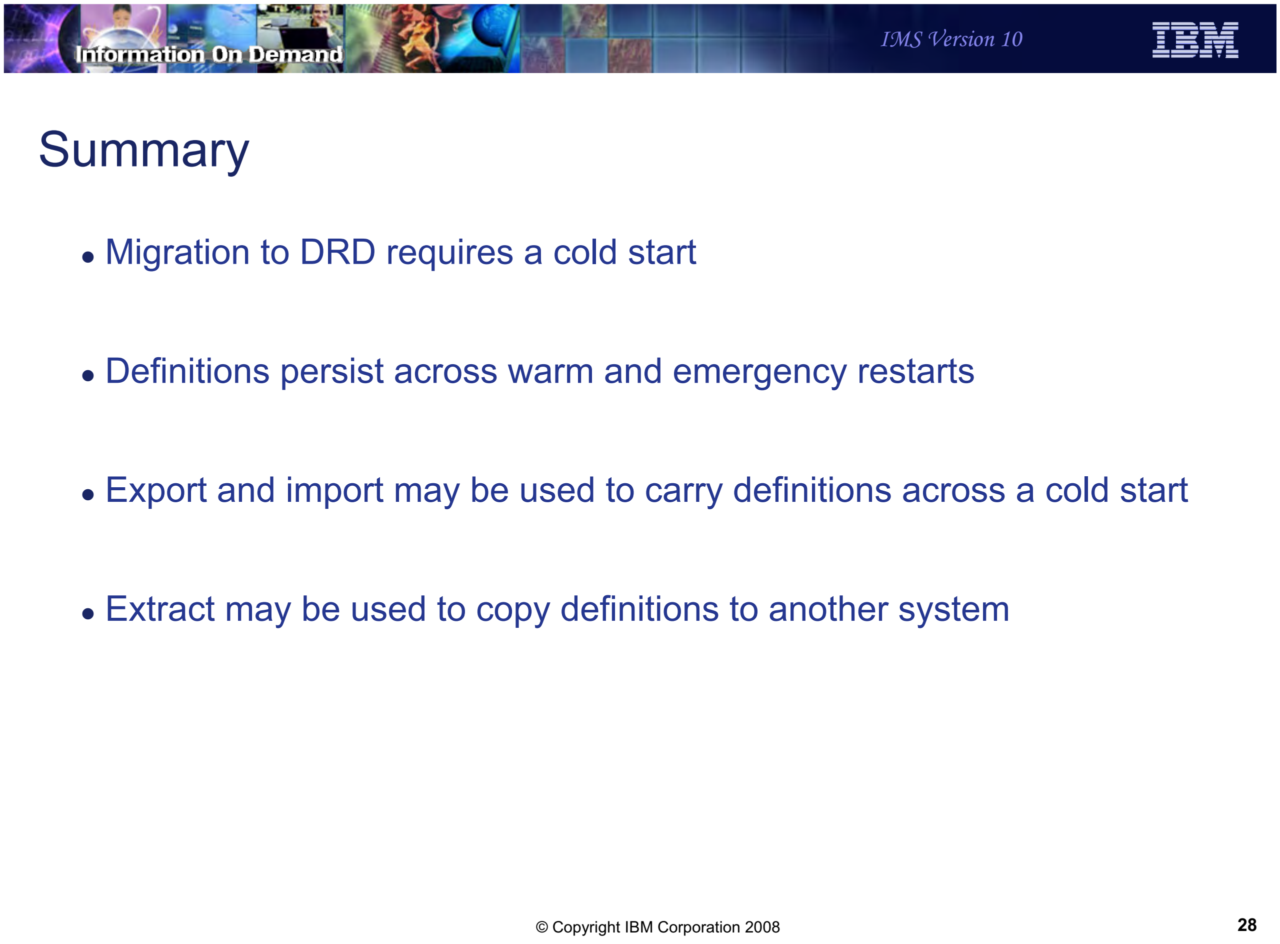
Summary

- DRD is an alternative to online change for MODBLKS
- DRD is supported in all IMS environments
  - DB/DC, DB/CTL, DC/CTL
  - Data sharing, shared queues
- Definitions can be dynamically created, deleted, and updated
  - CREATE, DELETE, UPDATE commands
  - DB, DBDESC, PGM, PGMDESC, TRAN, TRANDESC, RTC, RTCDESC
- Resources and descriptors can be modeled after existing resource or descriptor
  - CRE .... LIKE(DESC(desc-name)) < or > LIKE(RSC(rsc-name))
  - Determines defaults for attributes not explicitly SET in CREATE command

© Copyright IBM Corporation 2008

Figure 3-26. Summary CMA01.0

Notes:



Summary

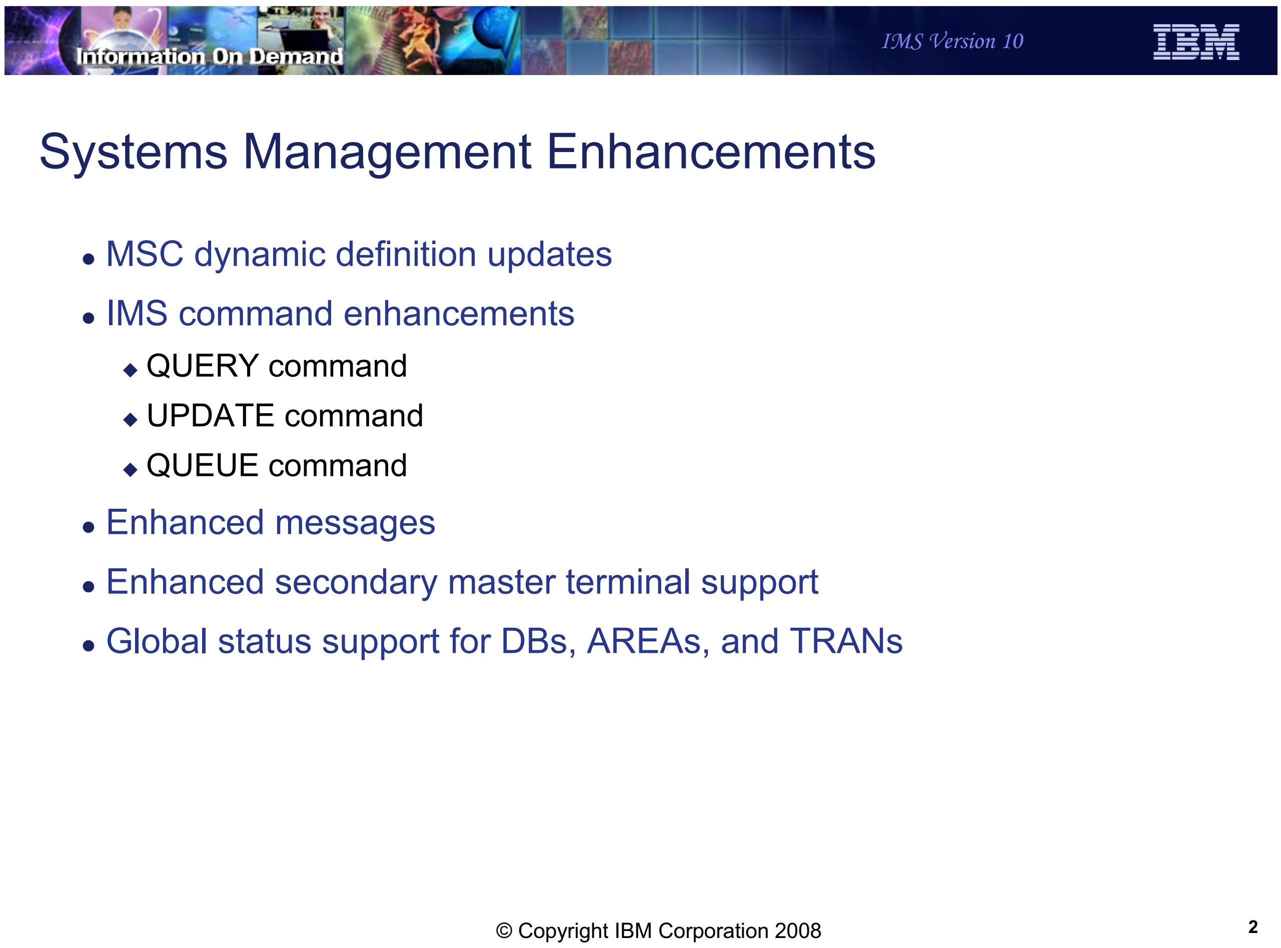
- Migration to DRD requires a cold start
- Definitions persist across warm and emergency restarts
- Export and import may be used to carry definitions across a cold start
- Extract may be used to copy definitions to another system

© Copyright IBM Corporation 2008

Figure 3-27. Summary CMA01.0

Notes:

Unit 4. System Management


Information On Demand
IMS Version 10
IBM

Systems Management Enhancements

- MSC dynamic definition updates
- IMS command enhancements
    - QUERY command
    - UPDATE command
    - QUEUE command
- Enhanced messages
- Enhanced secondary master terminal support
- Global status support for DBs, AREAs, and TRANs

© Copyright IBM Corporation 2008

Figure 4-1. Systems Management Enhancements CMA01.0

Notes:

Information On Demand

IMS Version 10

IBM

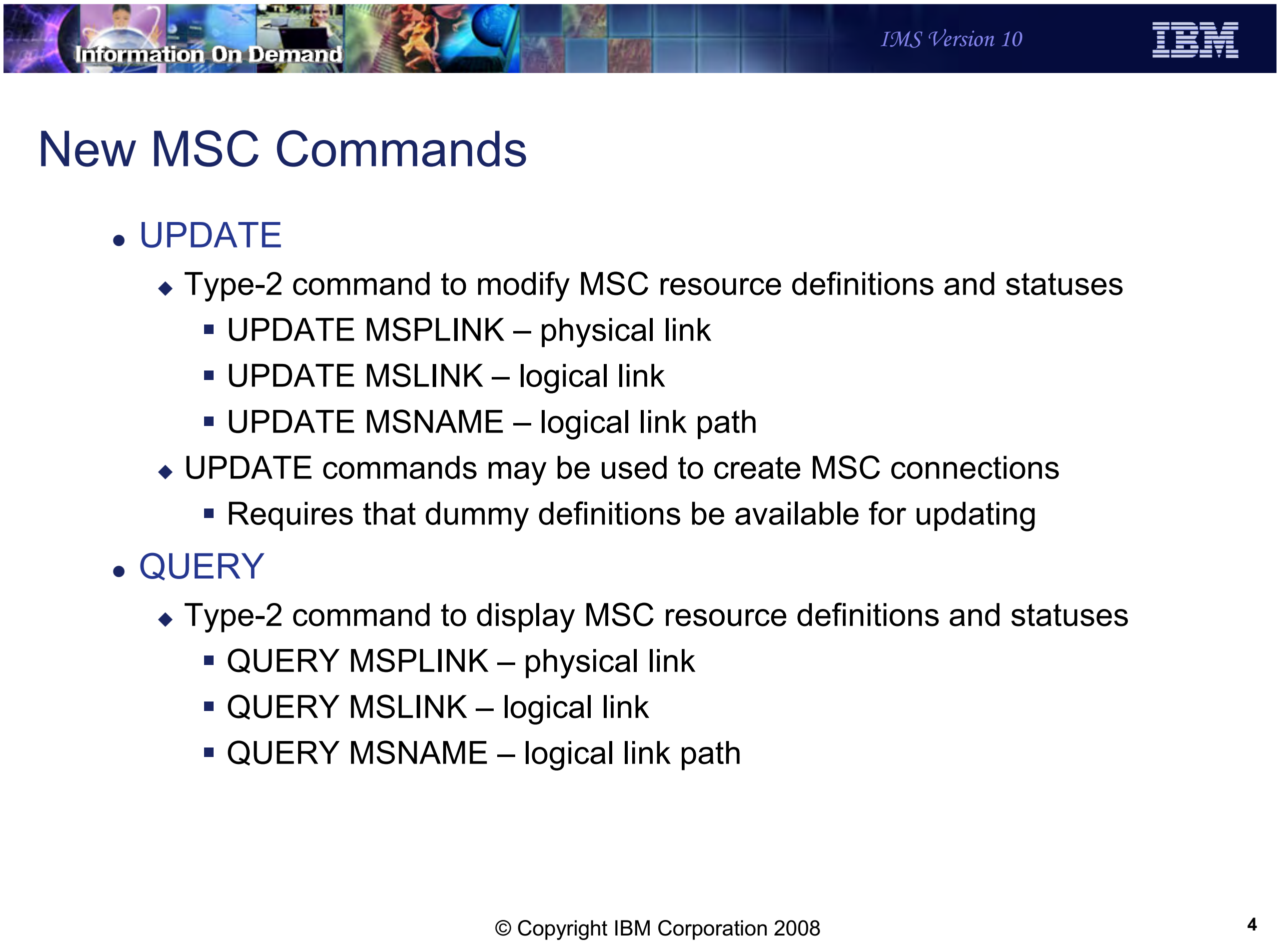
MSC Command Support

© Copyright IBM Corporation 2008

Figure 4-2. [No title] CMA01.0

Notes:

Although the DRD CREATE and DELETE process does not support MSC resources, there is some enhanced support for MSC in V10.



Information On Demand

IMS Version 10

New MSC Commands

- UPDATE
  - Type-2 command to modify MSC resource definitions and statuses
    - UPDATE MSPLINK – physical link
    - UPDATE MSLINK – logical link
    - UPDATE MSNAME – logical link path
  - UPDATE commands may be used to create MSC connections
    - Requires that dummy definitions be available for updating
- QUERY
  - Type-2 command to display MSC resource definitions and statuses
    - QUERY MSPLINK – physical link
    - QUERY MSLINK – logical link
    - QUERY MSNAME – logical link path

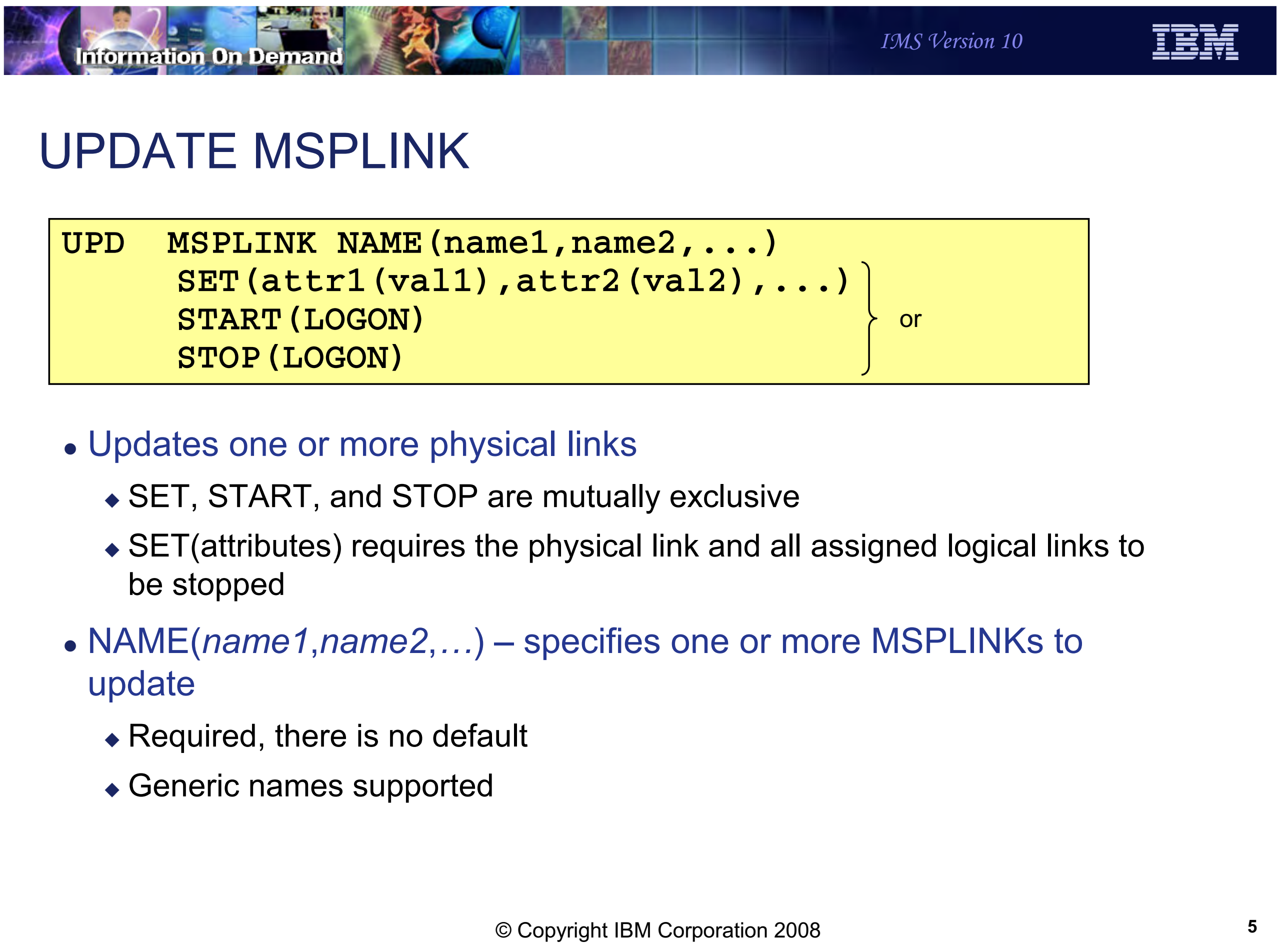
© Copyright IBM Corporation 2008

Figure 4-3. New MSC Commands CMA01.0

Notes:

IMS V10 adds support for MSC resources with the type-2 UPDATE and QUERY commands.

There is a new performance parameter for MSC logical links called BANDWIDTH. You will see this reference in the following slides but a discussion of what this means is in Transaction Manager Enhancements section of the class.


Information On Demand
IMS Version 10
IBM

UPDATE MSPLINK

```
UPD  MSPLINK NAME(name1,name2,...)
      SET(attr1(val1),attr2(val2),...) ⎫
      START(LOGON)                     ⎬ or
      STOP(LOGON)                      ⎭
```

- Updates one or more physical links
  - SET, START, and STOP are mutually exclusive
  - SET(attributes) requires the physical link and all assigned logical links to be stopped
- NAME(*name1*,*name2*,*…*) – specifies one or more MSPLINKs to update
  - Required, there is no default
  - Generic names supported

© Copyright IBM Corporation 2008

Figure 4-4. UPDATE MSPLINK CMA01.0

Notes:

This slide shows the format of the UPDATE MSPLINK command.

Information On Demand
IMS Version 10
IBM

UPDATE MSPLINK (cont.)

```
UPD  MSPLINK NAME(name1,name2,...)
      SET(attr1(val1),attr2(val2),...) }
      START(LOGON)                     }  or
      STOP(LOGON)                      }
```

- SET*(attribute1,attribute2,…)* – specifies the update to perform
  - ASR(*ON*|*OFF*) – sets Automatic Session Restart for each assigned VTAM logical link
  - MODETBL(*new_modetable_name*) – changes VTAM mode table name for each assigned logical link
  - MSPLINK(*new_msplinkname*) – changes the physical link name
  - NODE(*new_nodename*) – changes the VTAM node name for this link
- START*(LOGON)* – Enables logons to the physical link
- STOP*(LOGON)* - Disables logons to the physical link

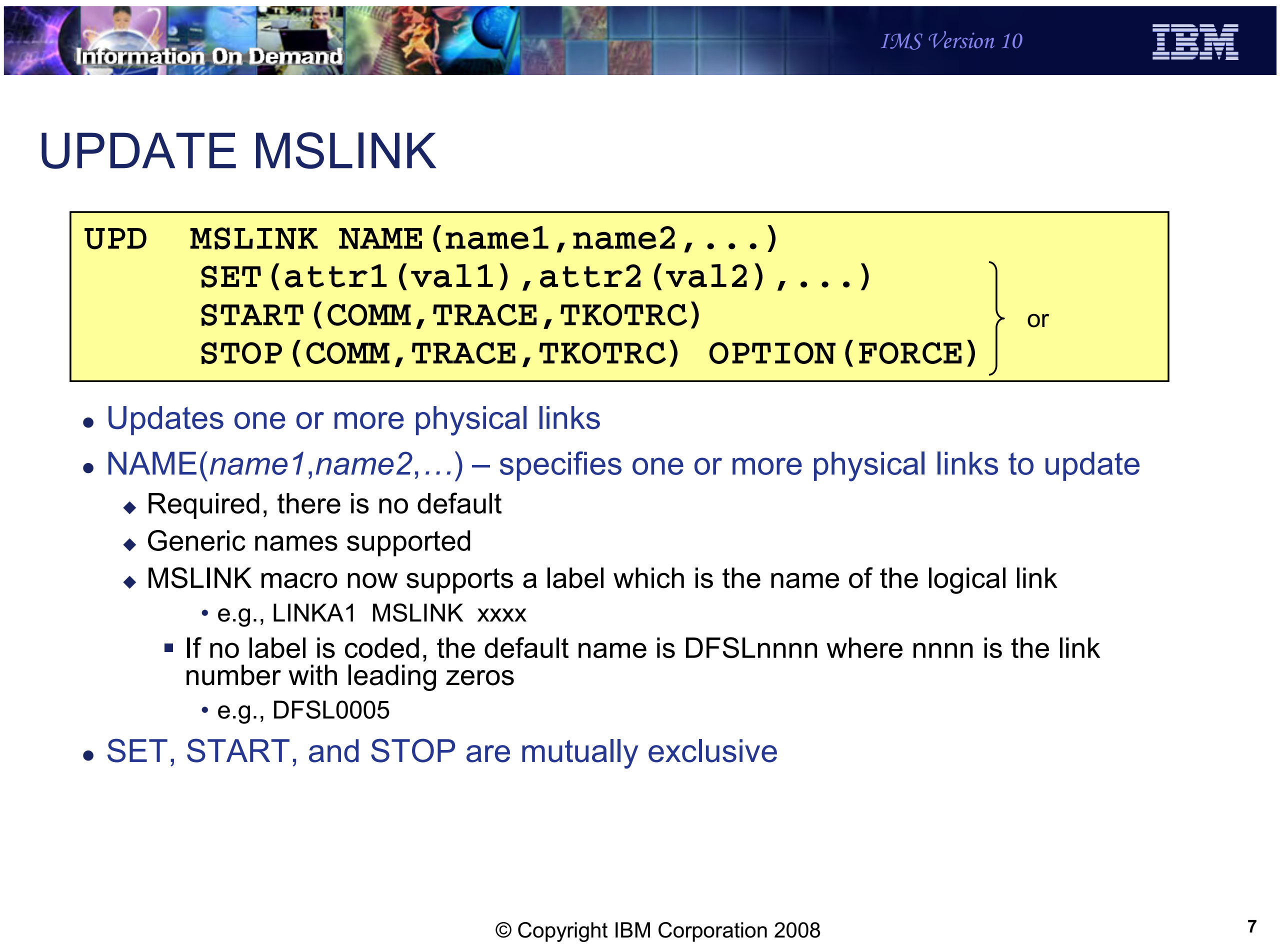
© Copyright IBM Corporation 2008

Figure 4-5. UPDATE MSPLINK (cont.) CMA01.0

Notes:

The UPDATE MSPLINK command can be used to change any of the four attributes identified on the slide. Except for changing the MSPLINK name, they apply only to VTAM links. Before the attributes can be changed, the physical link and all assigned logical links must be stopped.

START and STOP LOGON enables or stops logons to the physical link. STOP does not affect existing sessions.


Information On Demand
IMS Version 10
IBM

UPDATE MSLINK

```
UPD  MSLINK NAME(name1,name2,...)
      SET(attr1(val1),attr2(val2),...)        }
      START(COMM,TRACE,TKOTRC)                } or
      STOP(COMM,TRACE,TKOTRC) OPTION(FORCE)   }
```

- Updates one or more physical links
- NAME(*name1*,*name2*,…) – specifies one or more physical links to update
  - Required, there is no default
  - Generic names supported
  - MSLINK macro now supports a label which is the name of the logical link
    - e.g., LINKA1 MSLINK xxxx
    - If no label is coded, the default name is DFSLnnnn where nnnn is the link number with leading zeros
      - e.g., DFSL0005
- SET, START, and STOP are mutually exclusive

© Copyright IBM Corporation 2008

Figure 4-6. UPDATE MSLINK CMA01.0

Notes:

This slide shows the format of the UPDATE MSLINK command. As with the other UPD commands, you cannot change attributes and the status in the same command. The NAME of the MSLINK is either the label on the MSLINK macro or, if no label exists, a default name of DFSLnnnn where nnnn is the MSLink number with leading zeros. The command does not support MSLink numbers. You must specify a NAME.

Information On Demand
IMS Version 10
IBM

UPDATE MSLINK (cont.)

```
UPD  MSLINK NAME(name1,name2,...)
      SET(attr1(val1),attr2(val2),...)            }
      START(COMM,TRACE,TKOTRC)                     } or
      STOP(COMM,TRACE,TKOTRC) OPTION(FORCE)        }
```

- SET*(attribute1,attribute2,…)* – specifies the update to perform
    - SET requires the logical link to be stopped
    - ASR(*ON*|*OFF*) – sets Automatic Session Restart for VTAM links
    - BANDWIDTH(*ON*|*OFF*) – sets bandwidth mode
    - BUFSIZE(*new_buffer_size*) – changes send and receive buffer sizes
    - MODETBL(*new_modetable_name*) – changes VTAM mode table name
    - MSLINK(*new_mslinkname*) – changes the logical link name
    - MSPLINK(*new_msplinkname*) – reassigns the link to a different physical link
    - PARTNER(*new_partner_id*) – changes the partner id
    - SYNCOPT(*FORCSESS*|*SYNCSESS*|*COLDSESS*) – sets resynchronization options

© Copyright IBM Corporation 2008

Figure 4-7. UPDATE MSLINK (cont.)

Notes:

Any of the attributes can be changed using this command. The link must be stopped before changing the attributes.

UPDATE MSLINK (cont.)

```
UPD  MSLINK NAME(name1,name2,...)
      SET(attr1(val1),attr2(val2),...)       }
      START(COMM,TRACE,TKOTRC)               } or
      STOP(COMM,TRACE,TKOTRC) OPTION(FORCE)  }
```

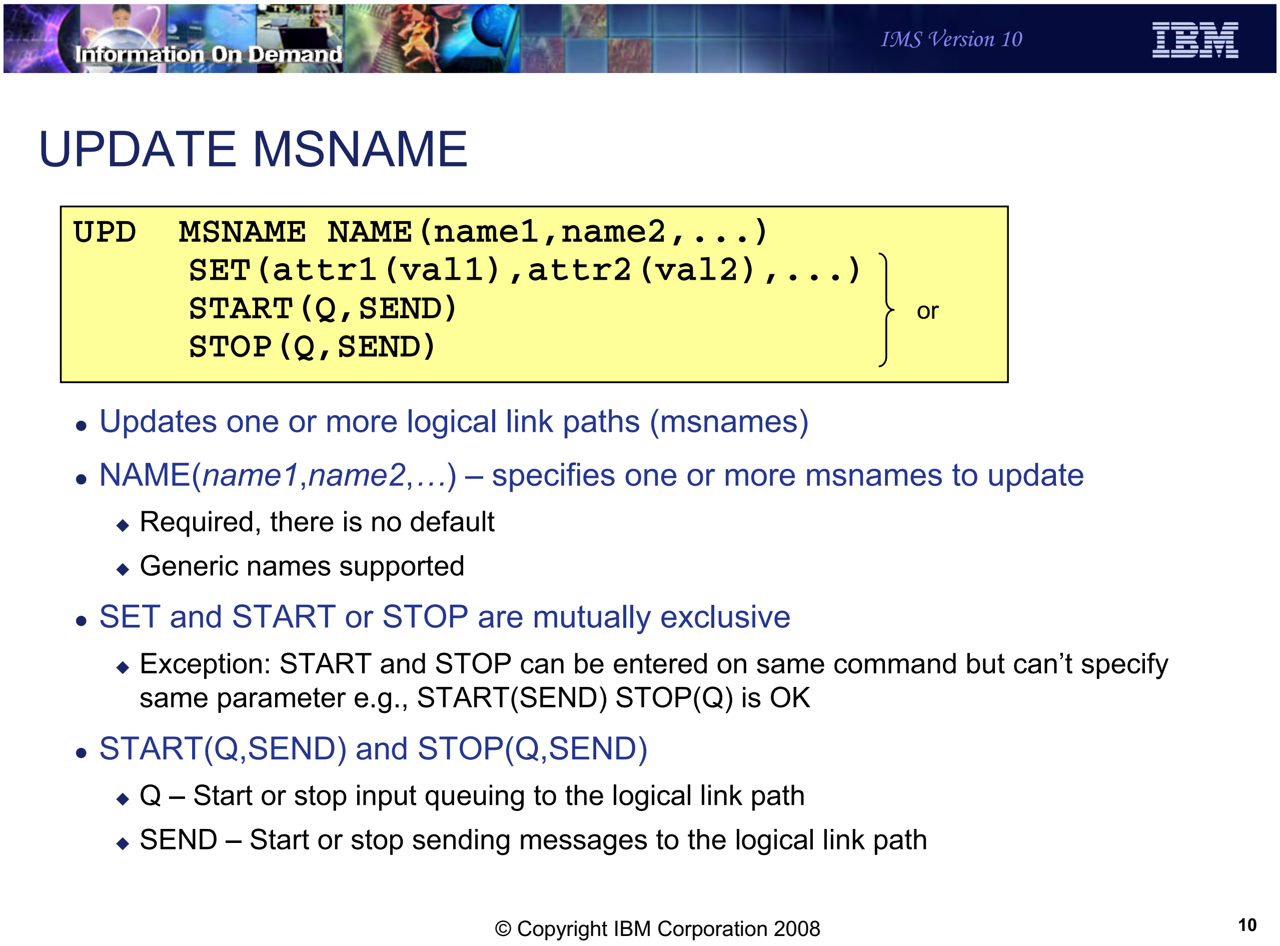
- START*(status1,status2,...)*
  - COMM – Start the link and start sending and receiving messages
  - TRACE – Starts internal tracing
  - TKOTRC – Starts tracing during XRF takeover
- STOP*(status1,status2,...)*
  - COMM – Stops the link and stops sending and receiving messages
  - TRACE – Stops internal tracing
  - TKOTRC – Stops tracing during XRF takeover
- OPTION(*option1*)
  - FORCE -- Force termination of the link if STOP(COMM) fails
    - OPTION(FORCE) is valid only with STOP(COMM)

© Copyright IBM Corporation 2008

Figure 4-8. UPDATE MSLINK (cont.) CMA01.0

Notes:

The START and STOP keywords can be entered on the same command, but you cannot start and stop the same status. OPTION(FORCE) is valid only for VTAM and CTC sessions, and only when used in conjunction with STOP(COMM).



UPDATE MSNAME

```
UPD  MSNAME NAME(name1,name2,...)
      SET(attr1(val1),attr2(val2),...) }
      START(Q,SEND)                    }  or
      STOP(Q,SEND)                     }
```

- Updates one or more logical link paths (msnames)
- NAME(*name1*,*name2*,…) – specifies one or more msnames to update
  - Required, there is no default
  - Generic names supported
- SET and START or STOP are mutually exclusive
  - Exception: START and STOP can be entered on same command but can't specify same parameter e.g., START(SEND) STOP(Q) is OK
- START(Q,SEND) and STOP(Q,SEND)
  - Q – Start or stop input queuing to the logical link path
  - SEND – Start or stop sending messages to the logical link path

© Copyright IBM Corporation 2008

Figure 4-9. UPDATE MSNAME CMA01.0

Notes:

The slide shows the format of the UPDATE MSNAME command. As with the others, you cannot issue a command with SET and with START or STOP.

UPD MSNAME can be used to start or stop queuing and sending messages across that logical link path. You can use START and STOP in the same command, but not with the same status.

Information On Demand
IMS Version 10
IBM

UPDATE MSNAME Attributes

```
UPD  MSNAME NAME(name1,name2,...)
      SET(attr1(val1),attr2(val2),...)  ⎫
      START(Q,SEND)                     ⎬ or
      STOP(Q,SEND)                      ⎭
```

- SET*(attribute1,attribute2,…)* – specifies the update to perform
  - MSLINK(*new_linkname*) – reassigns the MSNAME to a new logical link
  - SIDR(*new_remote_sysid*) – changes the remote sysid for the MSNAME
    - The changes also apply to remote LTERMs associated with this MSNAME
    - Cannot change SID if currently assigned to existing transaction
  - SIDL(*new_local_sysid*) – changes the local sysid for the MSNAME
    - The changes also apply to remote LTERMs and transactions associated with this MSNAME

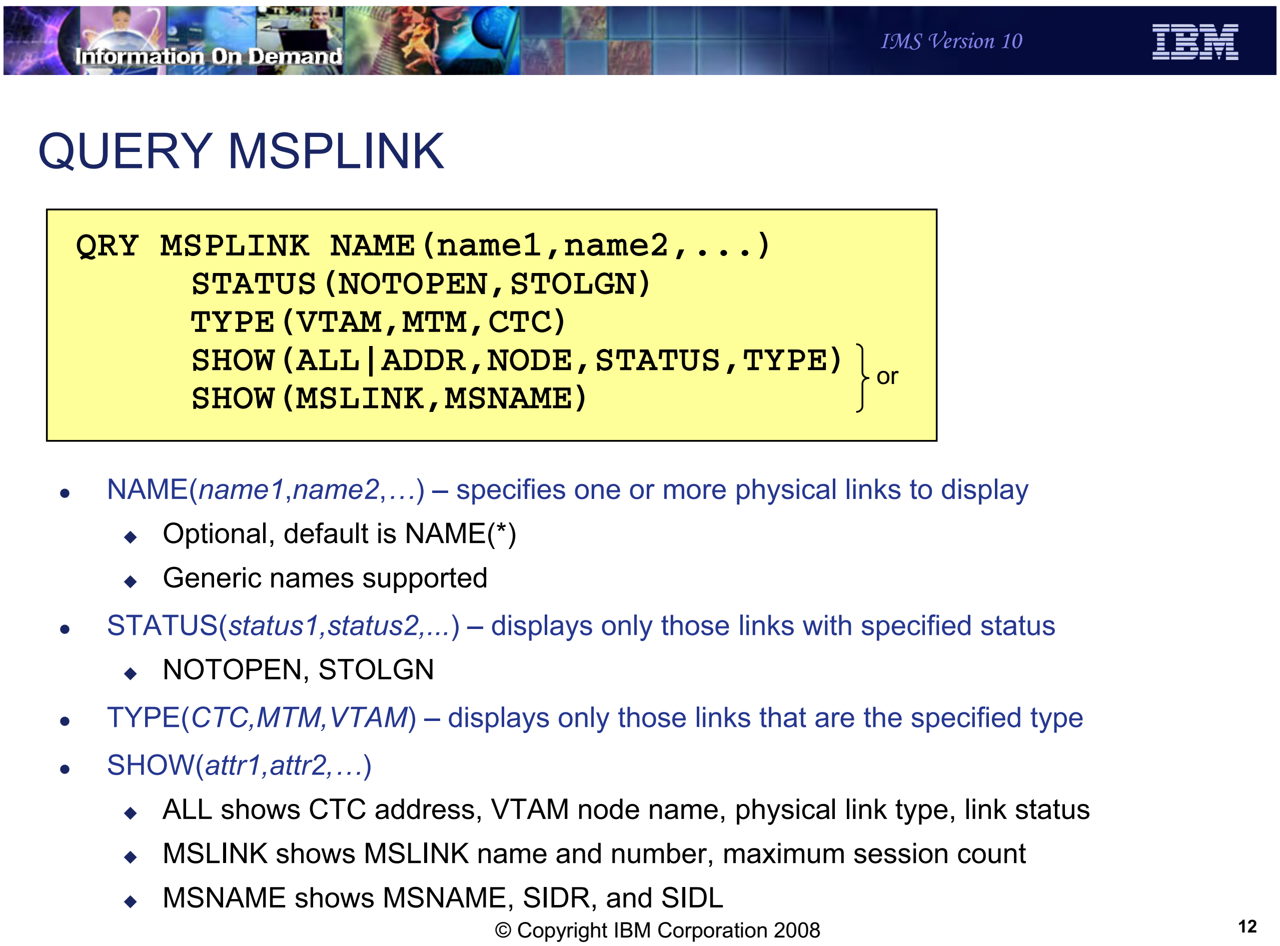
© Copyright IBM Corporation 2008

Figure 4-10. UPDATE MSNAME Attributes CMA01.0

Notes:

To change the logical link (MSLINK) with which this logical link path (MSNAME) is associated, the logical link must be stopped. To change the remote and local sysids, both the logical link and the logical link path must be stopped.

If the SYSIDs are currently assigned to an existing transaction, you must first use the UPDATE TRAN SET(SIDL(),SIDR()) command to reassign that transaction to another valid SYSID. Then you can update SIDR and SIDL. Then you can UPDATE TRAN again to reassign the remote and local sysids for the transaction back to the new sysids.



Information On Demand

IMS Version 10

IBM

QUERY MSPLINK

```
QRY MSPLINK NAME(name1,name2,...)
       STATUS(NOTOPEN,STOLGN)
       TYPE(VTAM,MTM,CTC)
       SHOW(ALL|ADDR,NODE,STATUS,TYPE) } or
       SHOW(MSLINK,MSNAME)             }
```

- NAME(*name1*,*name2*,…) – specifies one or more physical links to display
  - Optional, default is NAME(*)
  - Generic names supported
- STATUS(*status1*,*status2*,...) – displays only those links with specified status
  - NOTOPEN, STOLGN
- TYPE(*CTC*,*MTM*,*VTAM*) – displays only those links that are the specified type
- SHOW(*attr1*,*attr2*,…)
  - ALL shows CTC address, VTAM node name, physical link type, link status
  - MSLINK shows MSLINK name and number, maximum session count
  - MSNAME shows MSNAME, SIDR, and SIDL

© Copyright IBM Corporation 2008

Figure 4-11. QUERY MSPLINK CMA01.0

Notes:

This slide shows the format of the QRY MSPLINK command.

NAME is the name of the physical link. You can specify one or more names, and wild cards are allowed (* = multi-character substitution and % = single character substitution). The default is * which is to display all physical links.

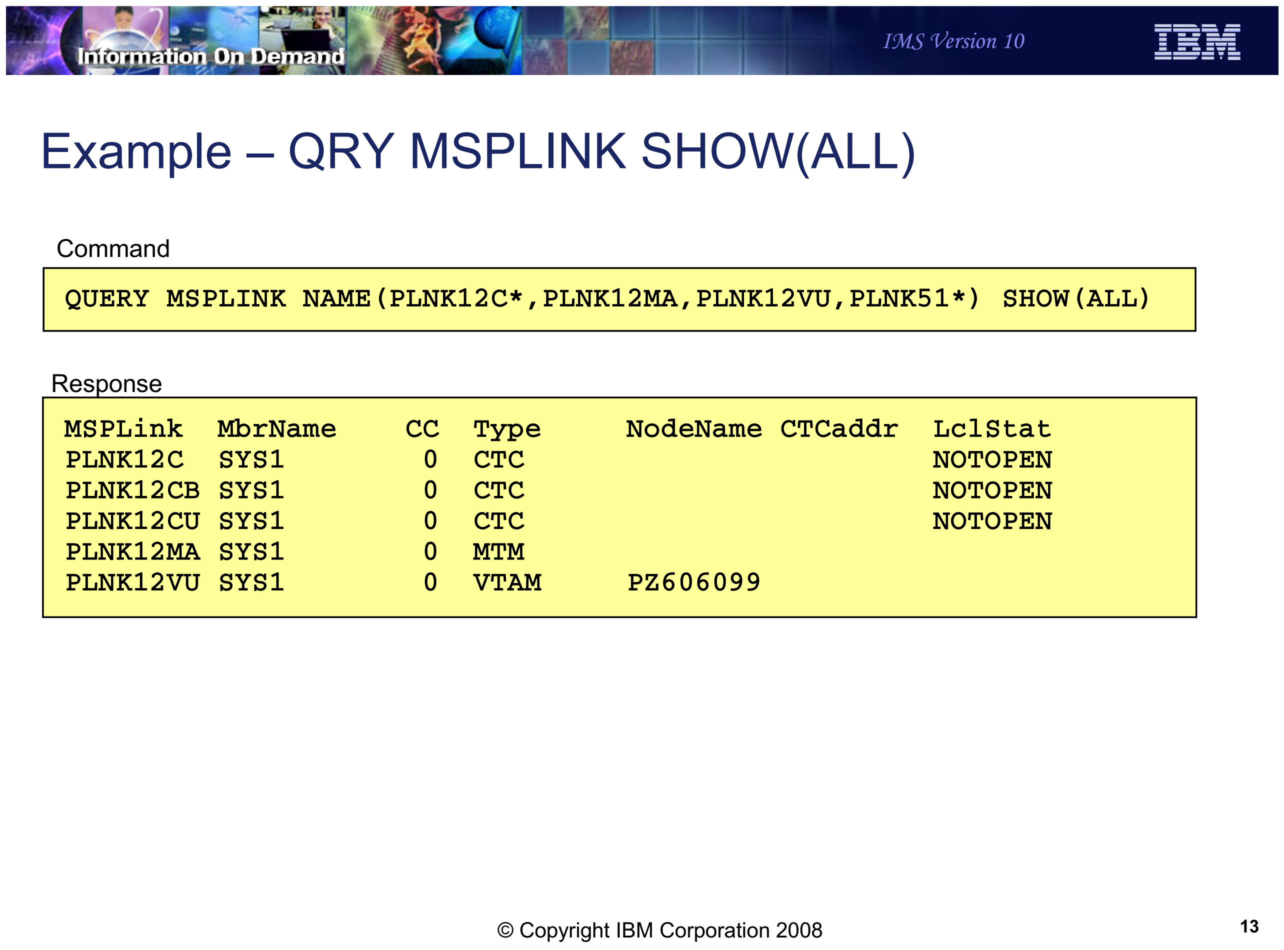
STATUS is a filter to limit the responses to those MSPLINKs with the specified status. Multiple statuses are ORed. For example, if STATUS(STOLGN) is specified, only those physical links with the STOP LOGON status will be displayed. The only other valid status is NOTOPEN.

TYPE is a filter for the link type – VTAM, Memory-to-memory, or Channel-to-channel.

SHOW determines what information is returned by IMS and what the TSO SPOC will show on the screen. SHOW(ALL) includes CRC address. VTAM node name of the remote IMS, the physical link type, and the link status. Each individual attribute may be selected - e.g., SHOW(NODE,STATUS). You can also select SHOW(MSLINK and/or MSNAME). You cannot specify the first and second type of SHOW together on the same command - e.g.

SHOW(NODE,STATUS,MSNAME,MSLINK) is not allowed. MSNAME displays the maximum session count defined on the MSPLINK macro as well as the link name, link number.

Note: V10 permits a label on the MSLINK macro. This label is the MSLINK name. If not specified, the default name is DFSLnnnn, where nnnn is the link number.



Example – QRY MSPLINK SHOW(ALL)

Command

```
QUERY MSPLINK NAME(PLNK12C*,PLNK12MA,PLNK12VU,PLNK51*) SHOW(ALL)
```

Response

```
MSPLink  MbrName    CC  Type     NodeName CTCaddr  LclStat
PLNK12C  SYS1        0  CTC                        NOTOPEN
PLNK12CB SYS1        0  CTC                        NOTOPEN
PLNK12CU SYS1        0  CTC                        NOTOPEN
PLNK12MA SYS1        0  MTM
PLNK12VU SYS1        0  VTAM     PZ606099
```

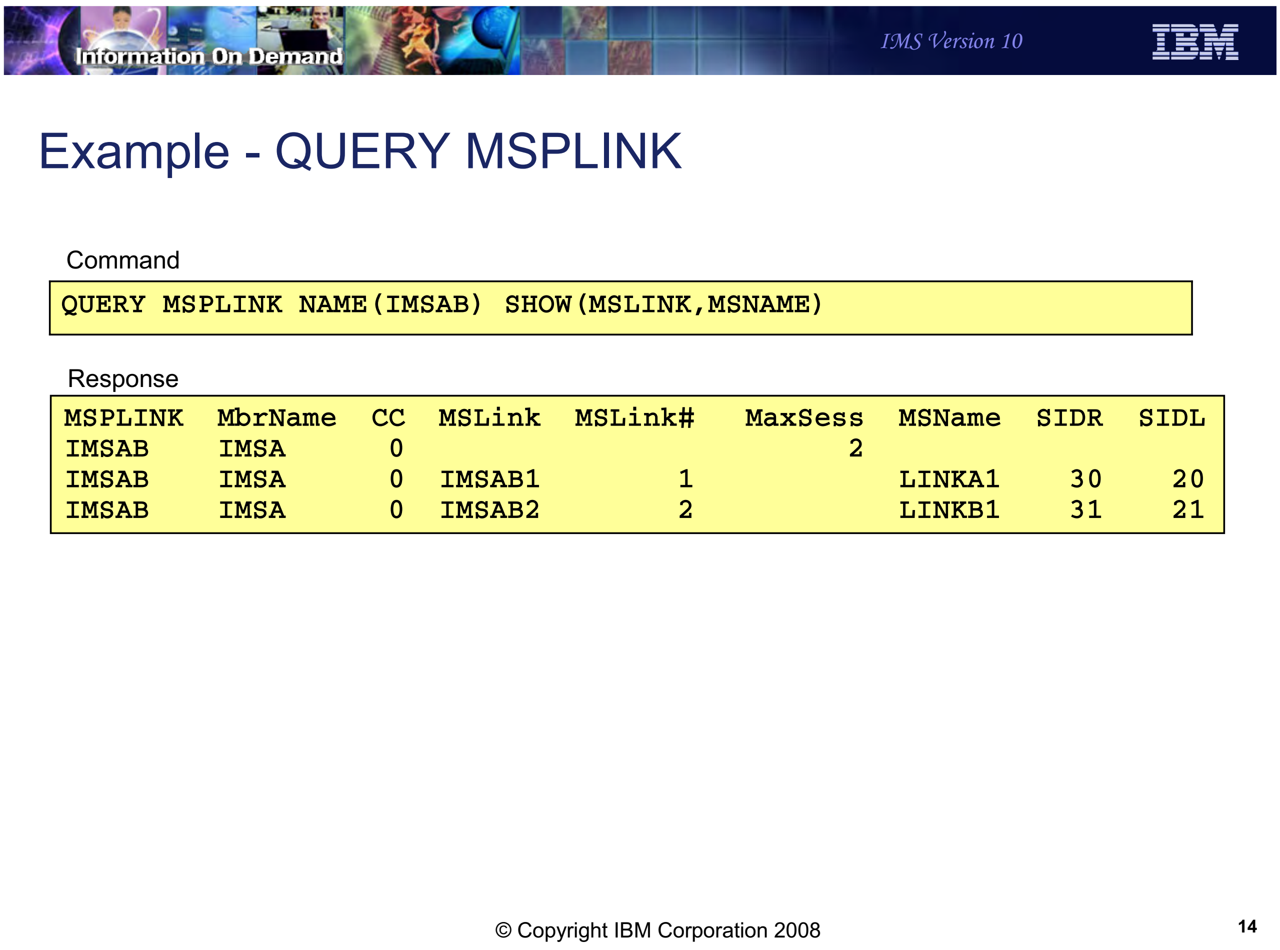
© Copyright IBM Corporation 2008

13

Figure 4-12. Example – QRY MSPLINK SHOW(ALL) CMA01.0

Notes:

This is an example of a QRY MSPLINK NAME(xxx) SHOW(ALL) command. All QRY MSPLINK commands show the MSPLINK name, the member name responding, the condition code, and, if CC is not zero, the CC text explaining why the command failed.



Example - QUERY MSPLINK

Command

```
QUERY MSPLINK NAME(IMSAB) SHOW(MSLINK,MSNAME)
```

Response

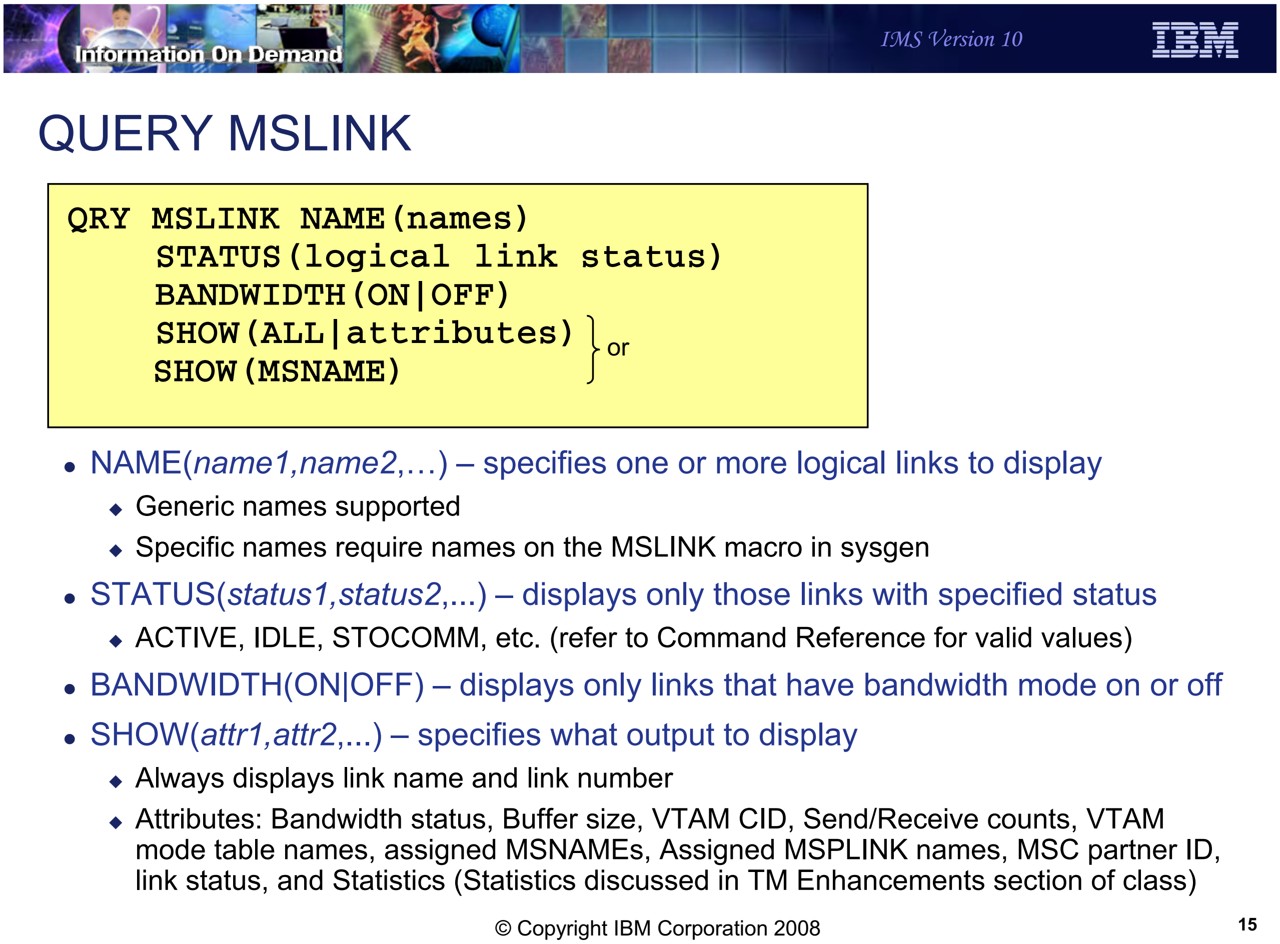
```
MSPLINK  MbrName  CC  MSLink  MSLink#   MaxSess  MSName  SIDR  SIDL
IMSAB    IMSA      0                          2
IMSAB    IMSA      0  IMSAB1        1            LINKA1    30    20
IMSAB    IMSA      0  IMSAB2        2            LINKB1    31    21
```

© Copyright IBM Corporation 2008

Figure 4-13. Example - QUERY MSPLINK CMA01.0

Notes:

This slide shows an example of QRY MSPLINK showing MSLINK and MSNAME. For MSLINK, the MSLink name and number and the MSPLINK maximum sessions are shown. For MSNAME, the MSName and remote and local sysids are shown.



IMS Version 10

QUERY MSLINK

```
QRY MSLINK NAME(names)
     STATUS(logical link status)
     BANDWIDTH(ON|OFF)
     SHOW(ALL|attributes) }
     SHOW(MSNAME)         } or
```

- NAME(*name1,name2,…*) – specifies one or more logical links to display
  - Generic names supported
  - Specific names require names on the MSLINK macro in sysgen
- STATUS(*status1,status2*,...) – displays only those links with specified status
  - ACTIVE, IDLE, STOCOMM, etc. (refer to Command Reference for valid values)
- BANDWIDTH(ON|OFF) – displays only links that have bandwidth mode on or off
- SHOW(*attr1,attr2*,...) – specifies what output to display
  - Always displays link name and link number
  - Attributes: Bandwidth status, Buffer size, VTAM CID, Send/Receive counts, VTAM mode table names, assigned MSNAMEs, Assigned MSPLINK names, MSC partner ID, link status, and Statistics (Statistics discussed in TM Enhancements section of class)

© Copyright IBM Corporation 2008

Figure 4-14. QUERY MSLINK CMA01.0

Notes:

This is the format of the QRY MSLINK command.

NAME is like other query commands. The name of the MSLINK is either the label on the macro (new in V10), or the default name DFSLnnnn, where nnnn is the MSLINK number with leading zeros. The command does not support MSLink numbers. You must specify a NAME.

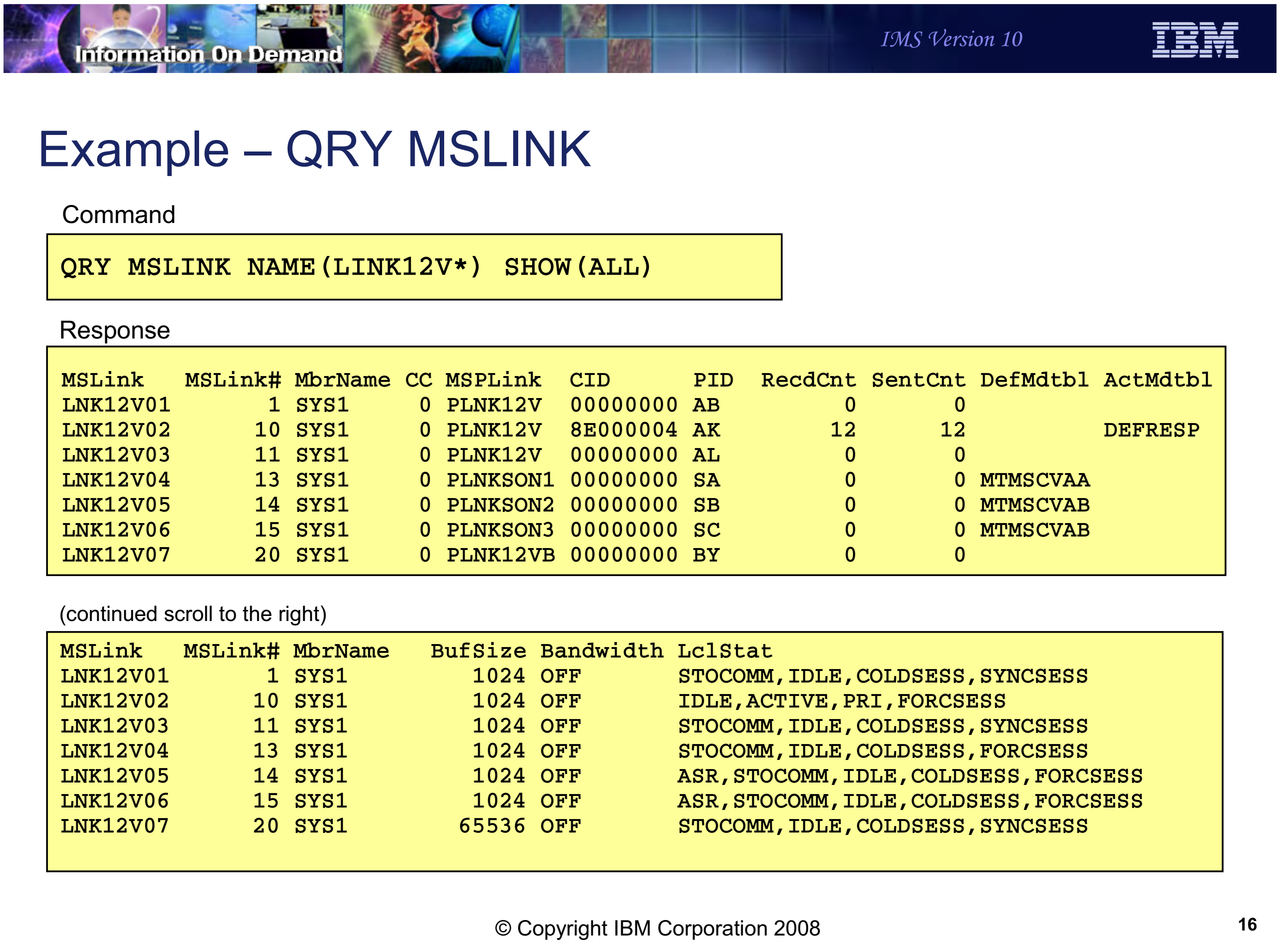
STATUS is a filter to limit the responses to those MSLINKs with the specified status. Multiple statuses are ORed. These statuses may be found in the Command Reference publication.

BANDWIDTH is a filter for whether the new bandwidth performance option is turned OFF or ON.

SHOW always shows the MSNAME name (the label) and the link number to which this MSNAME is assigned.

SHOW(ALL) shows all the definitional attributes and the status of the logical link. The attributes that may be specified can found in the Command Reference publication. You can request that just some of them be shown. Statistics may be requested. This is discussed in the TM Enhancements section of this class. SHOW(ALL) does not return statistics. They must be explicitly requested.

SHOW(MSNAME) cannot be specified with SHOW(ALL) or any of the attributes.


Information On Demand
IMS Version 10
IBM

Example – QRY MSLINK

Command

```
QRY MSLINK NAME(LINK12V*) SHOW(ALL)
```

Response

```
MSLink   MSLink# MbrName CC MSPLink  CID      PID  RecdCnt SentCnt DefMdtbl ActMdtbl
LNK12V01       1 SYS1     0 PLNK12V  00000000 AB         0       0
LNK12V02      10 SYS1     0 PLNK12V  8E000004 AK        12      12          DEFRESP
LNK12V03      11 SYS1     0 PLNK12V  00000000 AL         0       0
LNK12V04      13 SYS1     0 PLNKSON1 00000000 SA         0       0 MTMSCVAA
LNK12V05      14 SYS1     0 PLNKSON2 00000000 SB         0       0 MTMSCVAB
LNK12V06      15 SYS1     0 PLNKSON3 00000000 SC         0       0 MTMSCVAB
LNK12V07      20 SYS1     0 PLNK12VB 00000000 BY         0       0
```

(continued scroll to the right)

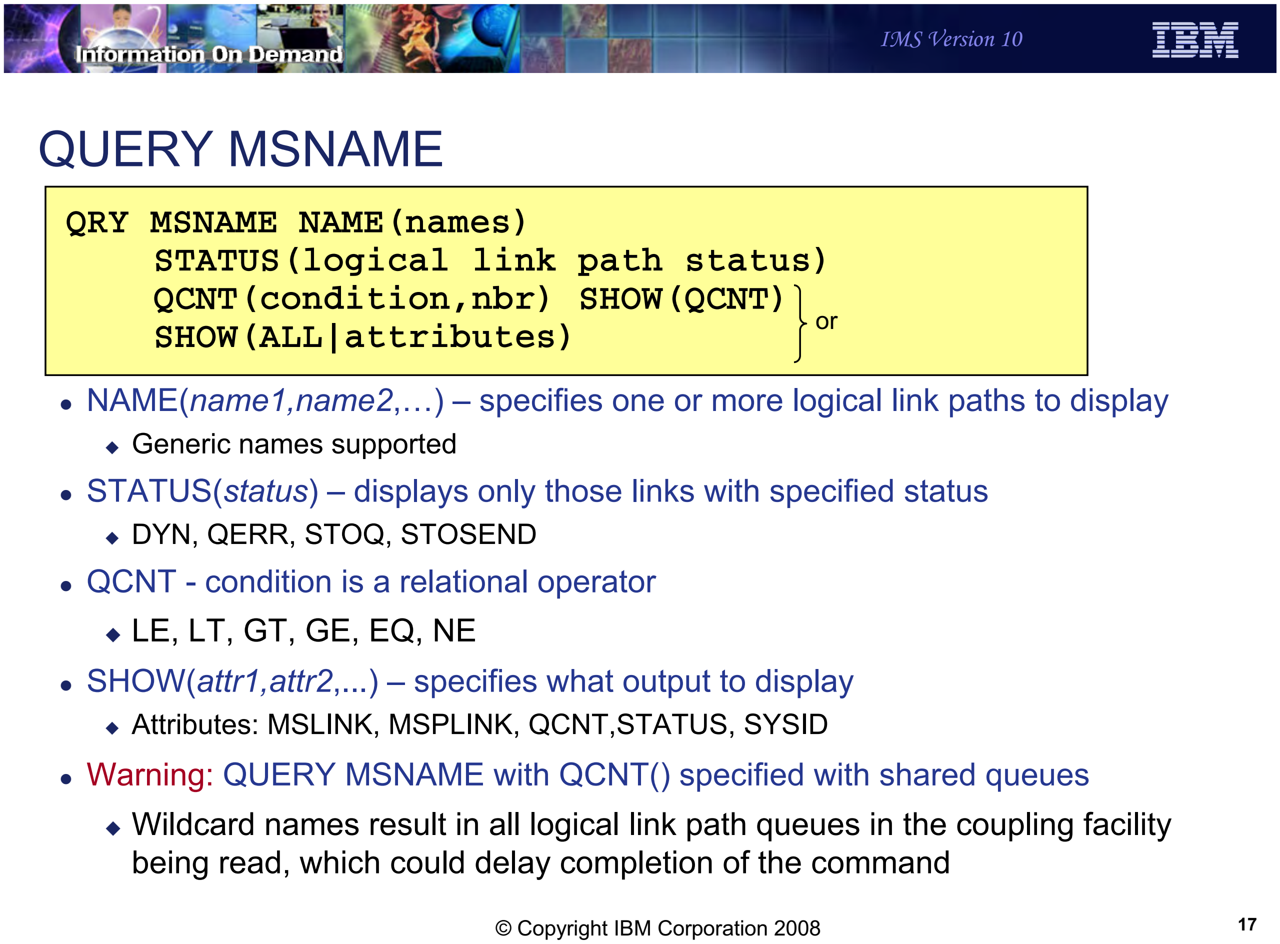
```
MSLink   MSLink# MbrName   BufSize Bandwidth LclStat
LNK12V01       1 SYS1         1024 OFF       STOCOMM,IDLE,COLDSESS,SYNCSESS
LNK12V02      10 SYS1         1024 OFF       IDLE,ACTIVE,PRI,FORCSESS
LNK12V03      11 SYS1         1024 OFF       STOCOMM,IDLE,COLDSESS,SYNCSESS
LNK12V04      13 SYS1         1024 OFF       STOCOMM,IDLE,COLDSESS,FORCSESS
LNK12V05      14 SYS1         1024 OFF       ASR,STOCOMM,IDLE,COLDSESS,FORCSESS
LNK12V06      15 SYS1         1024 OFF       ASR,STOCOMM,IDLE,COLDSESS,FORCSESS
LNK12V07      20 SYS1        65536 OFF       STOCOMM,IDLE,COLDSESS,SYNCSESS
```

© Copyright IBM Corporation 2008

Figure 4-15. Example – QRY MSLINK CMA01.0

Notes:

This slide shows an example of the QRY MSLINK xxx SHOW(ALL) command. The output is too wide for a single screen, so the TSO SPOC will let you scroll right and left, or up and down. The lower part if this screen assumes the user as scrolled right.



Information On Demand

IMS Version 10

QUERY MSNAME

```
QRY MSNAME NAME(names)
    STATUS(logical link path status)
    QCNT(condition,nbr) SHOW(QCNT) }
    SHOW(ALL|attributes)           } or
```

- NAME(*name1*,*name2*,…) – specifies one or more logical link paths to display
  - Generic names supported
- STATUS(*status*) – displays only those links with specified status
  - DYN, QERR, STOQ, STOSEND
- QCNT - condition is a relational operator
  - LE, LT, GT, GE, EQ, NE
- SHOW(*attr1*,*attr2*,...) – specifies what output to display
  - Attributes: MSLINK, MSPLINK, QCNT,STATUS, SYSID
- Warning: QUERY MSNAME with QCNT() specified with shared queues
  - Wildcard names result in all logical link path queues in the coupling facility being read, which could delay completion of the command

© Copyright IBM Corporation 2008

17

Figure 4-16. QUERY MSNAME CMA01.0

Notes:

The logical link path status may be:

DYN - dynamically created in a shared queues environment

QERR - queue error

STOQ - stopped queuing

STOSEND - stopped sending

The "condition" value in the QCNT parameter is a relational operator. It can be:

LE – less than or equal

LT – less than

GE – greater than or equal

GT – greater than

EQ – equal

NE – not equal

QRY MSNAME NAME(*) QCNT(NE,0) SHOW(QCNT) would show all MSNAMEs with a queue count that is not zero.

Be careful when issuing the QUERY MSNAME ... QCNT command in a shared queues environment. The use of wildcard names will result in reading every remote queue “list” to determine whether it satisfies the wildcard name (XES list services does not support wildcards).

Example – QRY MSNAME

Command

```
QUERY MSNAME NAME(LINK12V*,LINK12C1) SHOW(ALL)
```

Response

```
MSName   MbrName  CC MSPLink  MSLink   MSLink# SIDR SIDL  LQCnt LclStat
LINK12C1 SYS1      0 PLNK12C  LNK12C01       2   23   13      0
LINK12V  SYS1      0 PLNK12V  LNK12V02      10  256 1012      0
LINK12V1 SYS1      0 PLNK12V  LNK12V02      10   20   10      0 STOQ,STOSEND
LINK12V2 SYS1      0 PLNK12V  LNK12V03      11   40    9      0
LINK12V3 SYS1      0 PLNK12VB LNK12V07      20   93   73      0
```

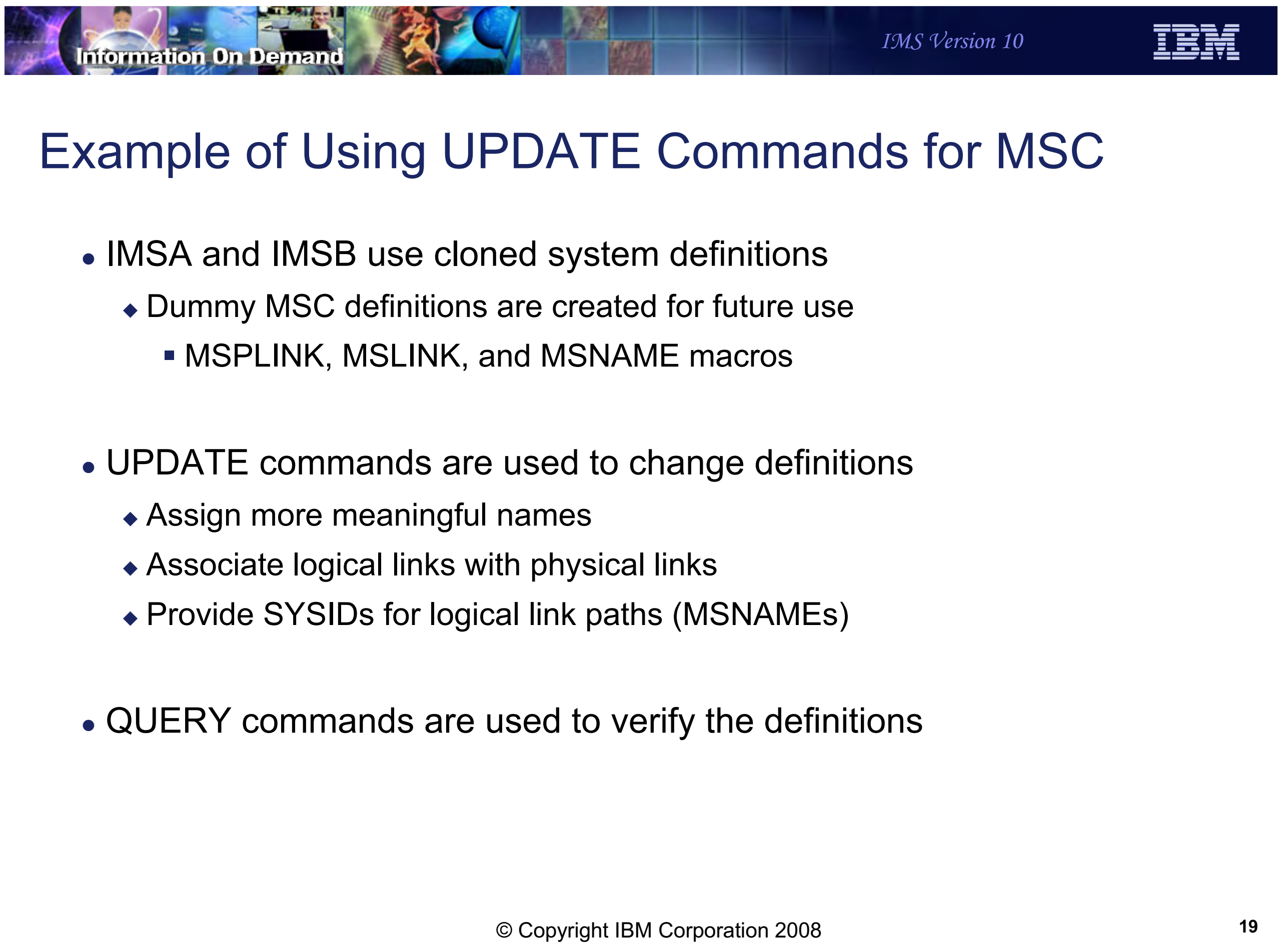
© Copyright IBM Corporation 2008

Figure 4-17. [No title]

CMA01.0

Notes:

This is an example of a QRY MSNAME ... SHOW(ALL) command. In a shared queues environment, there would be two lines for each MSName, with the first line showing the global QCNT and the second line showing the local attributes, queue count, and status as you see in the slide.



Information On Demand
IMS Version 10

Example of Using UPDATE Commands for MSC

- IMSA and IMSB use cloned system definitions
  - Dummy MSC definitions are created for future use
    - MSPLINK, MSLINK, and MSNAME macros
- UPDATE commands are used to change definitions
  - Assign more meaningful names
  - Associate logical links with physical links
  - Provide SYSIDs for logical link paths (MSNAMEs)
- QUERY commands are used to verify the definitions

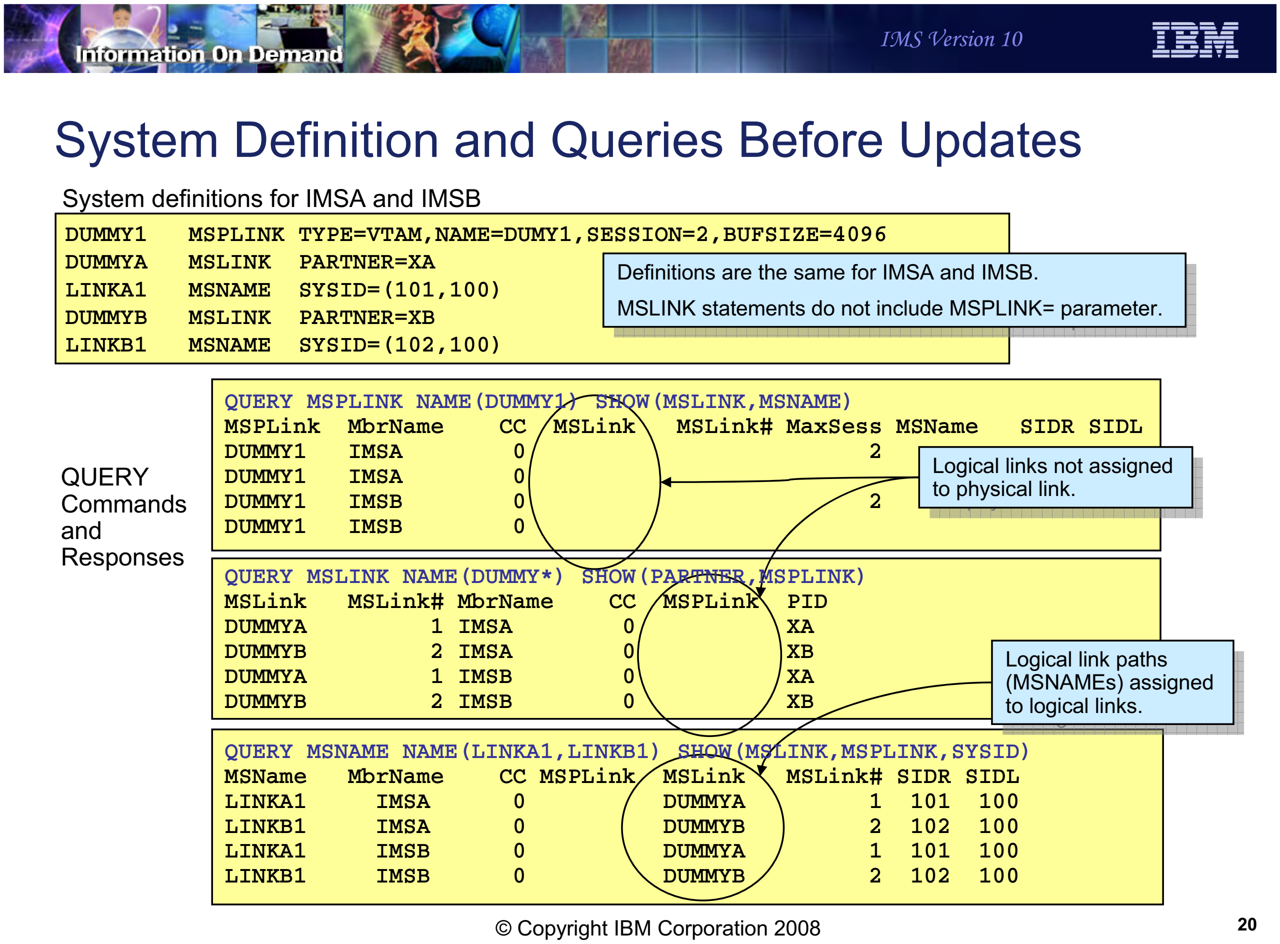
© Copyright IBM Corporation 2008

Figure 4-18. Example of Using UPDATE Commands for MSC CMA01.0

Notes:

Although CREATE does not support the addition of new MSC links, with careful planning, you can define dummy physical links, logical links, and logical link paths in each IMS between which you might want to create links. Then when they are needed, you can use the UPDATE MSPLINK, MSLINK, and MSNAME commands to activate those links.

The following shows an example of this technique. First dummy MSC definitions are included in the system definitions for IMSA and IMSB. The two systems have cloned definitions for future use by UPDATE commands. Second, QUERY commands are used to show these definitions. Third, UPDATE commands are used to assign more meaningful names to the physical links and logical links, to associate logical links with specific physical links, and to provide appropriate SYSIDs for the logical link paths. Finally, QUERY commands are used to show the changed definitions.



System Definition and Queries Before Updates

System definitions for IMSA and IMSB

```
DUMMY1   MSPLINK TYPE=VTAM,NAME=DUMY1,SESSION=2,BUFSIZE=4096
DUMMYA   MSLINK  PARTNER=XA
LINKA1   MSNAME  SYSID=(101,100)
DUMMYB   MSLINK  PARTNER=XB
LINKB1   MSNAME  SYSID=(102,100)
```

Definitions are the same for IMSA and IMSB.

MSLINK statements do not include MSPLINK= parameter.

QUERY
Commands
and
Responses

```
QUERY MSPLINK NAME(DUMMY1) SHOW(MSLINK,MSNAME)
MSPLink   MbrName    CC  MSLink   MSLink# MaxSess MSName   SIDR SIDL
DUMMY1    IMSA        0                          2
DUMMY1    IMSA        0
DUMMY1    IMSB        0                          2
DUMMY1    IMSB        0
```

Logical links not assigned to physical link.

```
QUERY MSLINK NAME(DUMMY*) SHOW(PARTNER,MSPLINK)
MSLink   MSLink# MbrName    CC  MSPLink  PID
DUMMYA         1 IMSA        0           XA
DUMMYB         2 IMSA        0           XB
DUMMYA         1 IMSB        0           XA
DUMMYB         2 IMSB        0           XB
```

Logical link paths (MSNAMEs) assigned to logical links.

```
QUERY MSNAME NAME(LINKA1,LINKB1) SHOW(MSLINK,MSPLINK,SYSID)
MSName   MbrName    CC MSPLink  MSLink   MSLink# SIDR SIDL
LINKA1     IMSA      0          DUMMYA         1  101  100
LINKB1     IMSA      0          DUMMYB         2  102  100
LINKA1     IMSB      0          DUMMYA         1  101  100
LINKB1     IMSB      0          DUMMYB         2  102  100
```

© Copyright IBM Corporation 2008

20

Figure 4-19. [No title] CMA01.0

Notes:

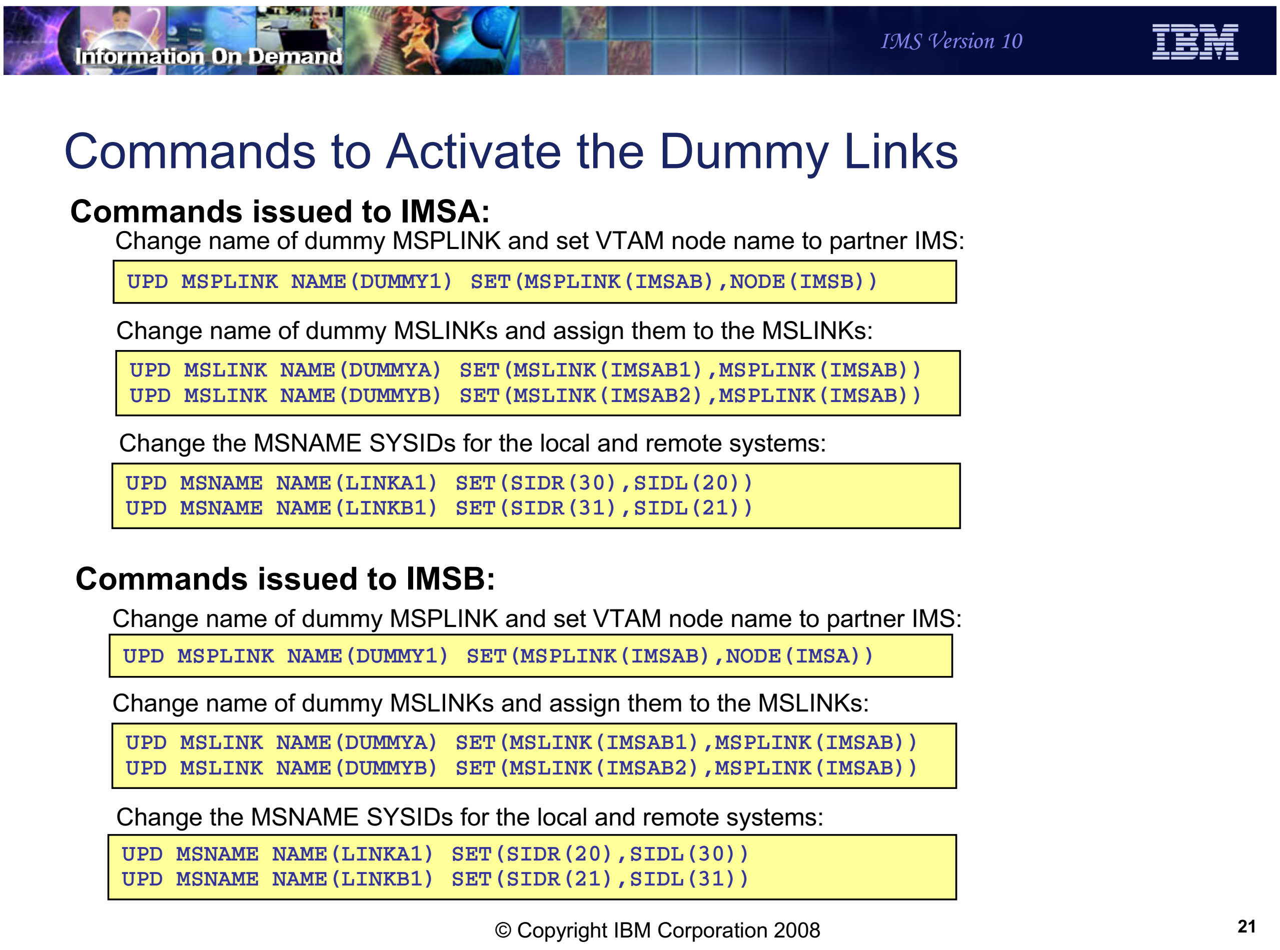
This shows the beginning of a scenario for defining dummy links and then activating them. Here we have defined in the System Definition with one MSPLINK, two MSLINKs, and two MSNAMEs.

The MSPLINK statement uses dummy names for the label and for the node name (NAME=).

The MSLINK statements use dummy names for the label and partner. They do not have a MSPLINK= parameter, so they are not assigned to a physical link.

The MSNAME statements use dummy numbers for the SYSIDs.

The QRY commands show the definitions before any changes are made. The MSNAMEs are assigned to MSLINKs, but since the MSLINKs are not defined to a MSPLINK, neither are the MSNAMEs.


Information On Demand
IMS Version 10
IBM

Commands to Activate the Dummy Links

Commands issued to IMSA:

Change name of dummy MSPLINK and set VTAM node name to partner IMS:

```
UPD MSPLINK NAME(DUMMY1) SET(MSPLINK(IMSAB),NODE(IMSB))
```

Change name of dummy MSLINKs and assign them to the MSLINKs:

```
UPD MSLINK NAME(DUMMYA) SET(MSLINK(IMSAB1),MSPLINK(IMSAB))
UPD MSLINK NAME(DUMMYB) SET(MSLINK(IMSAB2),MSPLINK(IMSAB))
```

Change the MSNAME SYSIDs for the local and remote systems:

```
UPD MSNAME NAME(LINKA1) SET(SIDR(30),SIDL(20))
UPD MSNAME NAME(LINKB1) SET(SIDR(31),SIDL(21))
```

Commands issued to IMSB:

Change name of dummy MSPLINK and set VTAM node name to partner IMS:

```
UPD MSPLINK NAME(DUMMY1) SET(MSPLINK(IMSAB),NODE(IMSA))
```

Change name of dummy MSLINKs and assign them to the MSLINKs:

```
UPD MSLINK NAME(DUMMYA) SET(MSLINK(IMSAB1),MSPLINK(IMSAB))
UPD MSLINK NAME(DUMMYB) SET(MSLINK(IMSAB2),MSPLINK(IMSAB))
```

Change the MSNAME SYSIDs for the local and remote systems:

```
UPD MSNAME NAME(LINKA1) SET(SIDR(20),SIDL(30))
UPD MSNAME NAME(LINKB1) SET(SIDR(21),SIDL(31))
```

© Copyright IBM Corporation 2008

Figure 4-20. [No title] CMA01.0

Notes:

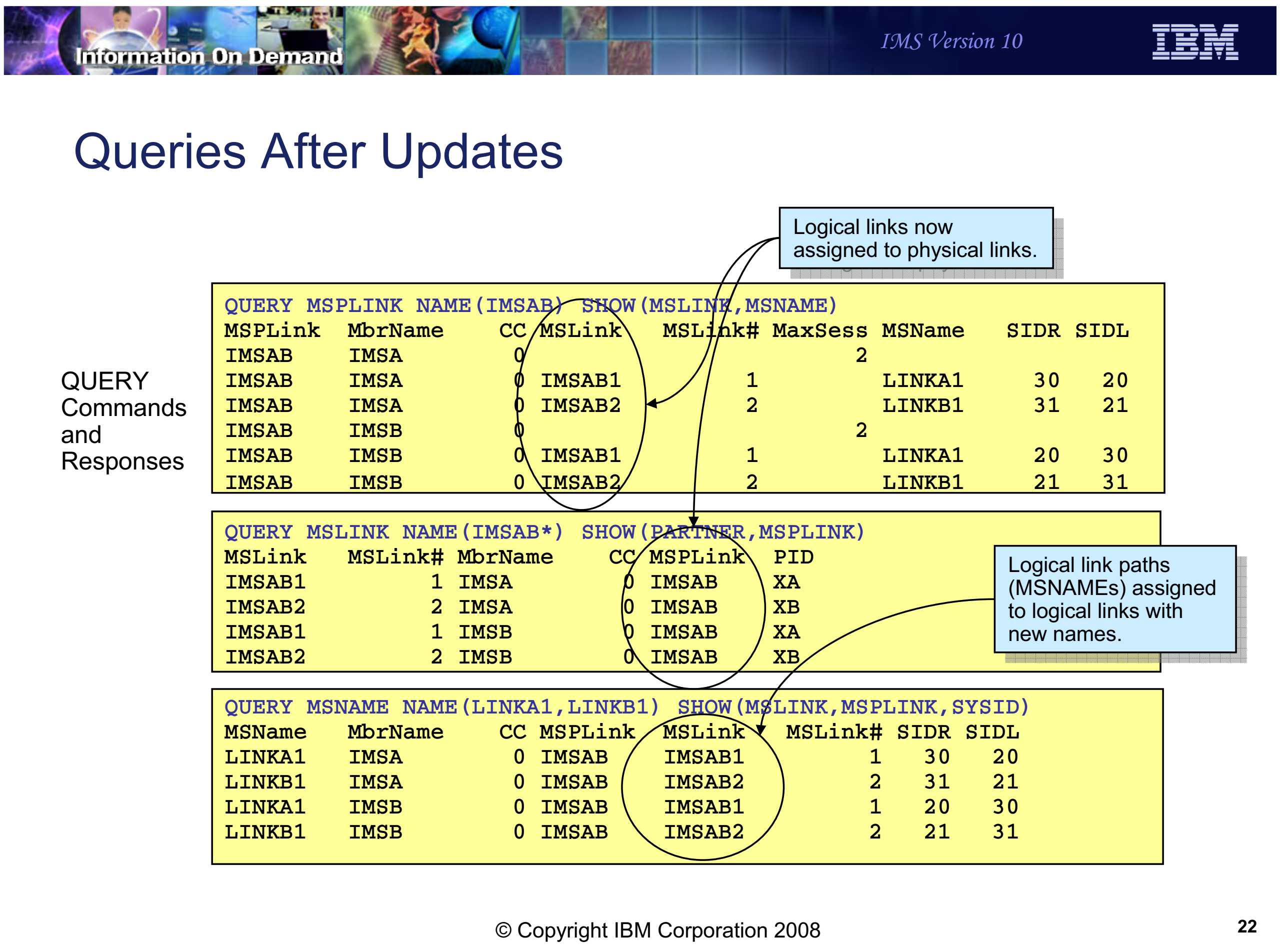
On this slide, we show how to “create” the MSC links between IMSA and IMSB. Similar commands are issued on both IMS systems. The responses to these commands are not shown.

For both systems the MSPLINK is updated to give it a name that is more descriptive and to identify the remote VTAM node. The command in IMSA specifies IMSB as the NODE name. The command in IMSB specifies IMSA as the NODE name.

The MSLINKs are updated to assign them to the renamed MSPLINK. The commands are the same for IMSA and IMSB.

MSNAMEs are updated to specify remote and local sysids. Of course, the remote sysids in IMSA are local sysids in IMSB.

After this we can CREATE remote transactions in IMSA and IMSB with the corresponding sysids.



Queries After Updates

Logical links now assigned to physical links.

QUERY Commands and Responses

```
QUERY MSPLINK NAME(IMSAB) SHOW(MSLINK,MSNAME)
MSPLink  MbrName     CC MSLink   MSLink# MaxSess MSName   SIDR SIDL
IMSAB    IMSA         0                        2
IMSAB    IMSA         0 IMSAB1         1         LINKA1     30   20
IMSAB    IMSA         0 IMSAB2         2         LINKB1     31   21
IMSAB    IMSB         0                        2
IMSAB    IMSB         0 IMSAB1         1         LINKA1     20   30
IMSAB    IMSB         0 IMSAB2         2         LINKB1     21   31
```

```
QUERY MSLINK NAME(IMSAB*) SHOW(PARTNER,MSPLINK)
MSLink   MSLink# MbrName     CC MSPLink  PID
IMSAB1         1 IMSA         0 IMSAB    XA
IMSAB2         2 IMSA         0 IMSAB    XB
IMSAB1         1 IMSB         0 IMSAB    XA
IMSAB2         2 IMSB         0 IMSAB    XB
```

Logical link paths (MSNAMEs) assigned to logical links with new names.

```
QUERY MSNAME NAME(LINKA1,LINKB1) SHOW(MSLINK,MSPLINK,SYSID)
MSName   MbrName     CC MSPLink  MSLink   MSLink# SIDR SIDL
LINKA1   IMSA         0 IMSAB    IMSAB1         1   30   20
LINKB1   IMSA         0 IMSAB    IMSAB2         2   31   21
LINKA1   IMSB         0 IMSAB    IMSAB1         1   20   30
LINKB1   IMSB         0 IMSAB    IMSAB2         2   21   31
```

© Copyright IBM Corporation 2008

22

Figure 4-21. [No title]

CMA01.0

Notes:

The QUERY commands show the results of the updates.

Information On Demand

IMS Version 10

MSC Commands Summary

- Updates provide capability to dynamically add and modify MSC links
  - Addition of links requires available dummy definitions
  - Restrictions:
    - Cannot change link type: CTC, VTAM, or MTM
    - Cannot modify CTC DDNAME or address
    - Cannot change number of sessions for VTAM link
- Queries show current definitions

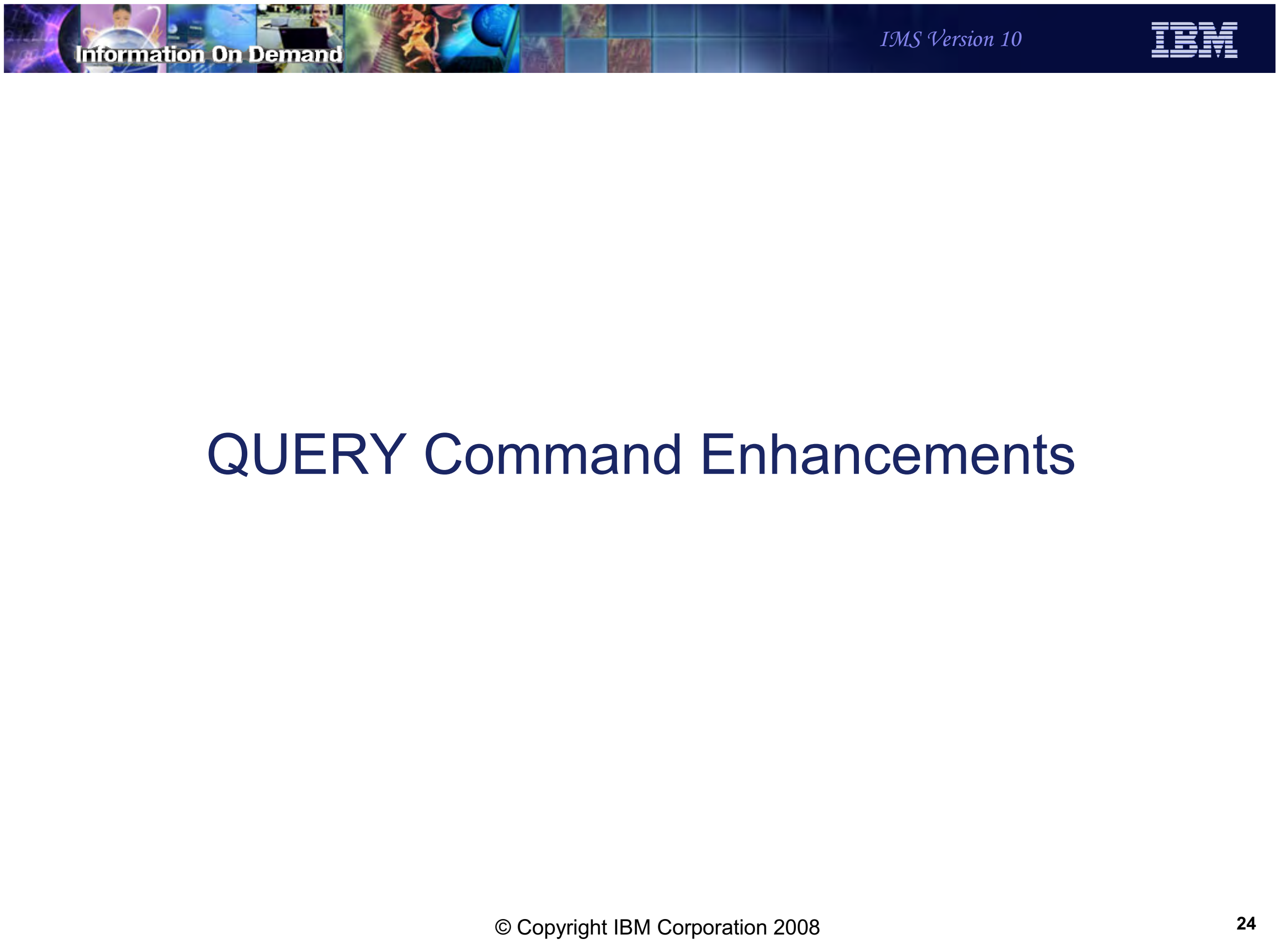
© Copyright IBM Corporation 2008

Figure 4-22. MSC Commands Summary CMA01.0

Notes:

UPDATE commands for MSPLINK, MSLINK, and MSNAME may be used to dynamically add and modify MSC links. The addition of links requires available dummy definitions. This capability is most easily used with VTAM links. Dummy definitions must be of the correct link type. That is, CTC, VTAM, and MTM definitions cannot be changed to another type. For example, you cannot change a CTC MSPLINK definition to a VTAM MSPLINK. The UPDATE command cannot be used to change the DDNAME or address associated with a CTC MSPLINK definition. The SESSION= parameter on the MSPLINK system definition macro for VTAM physical links cannot be modified with the UPDATE command.

QUERY commands may be used to show the current definitions.


Information On Demand
IMS Version 10
IBM

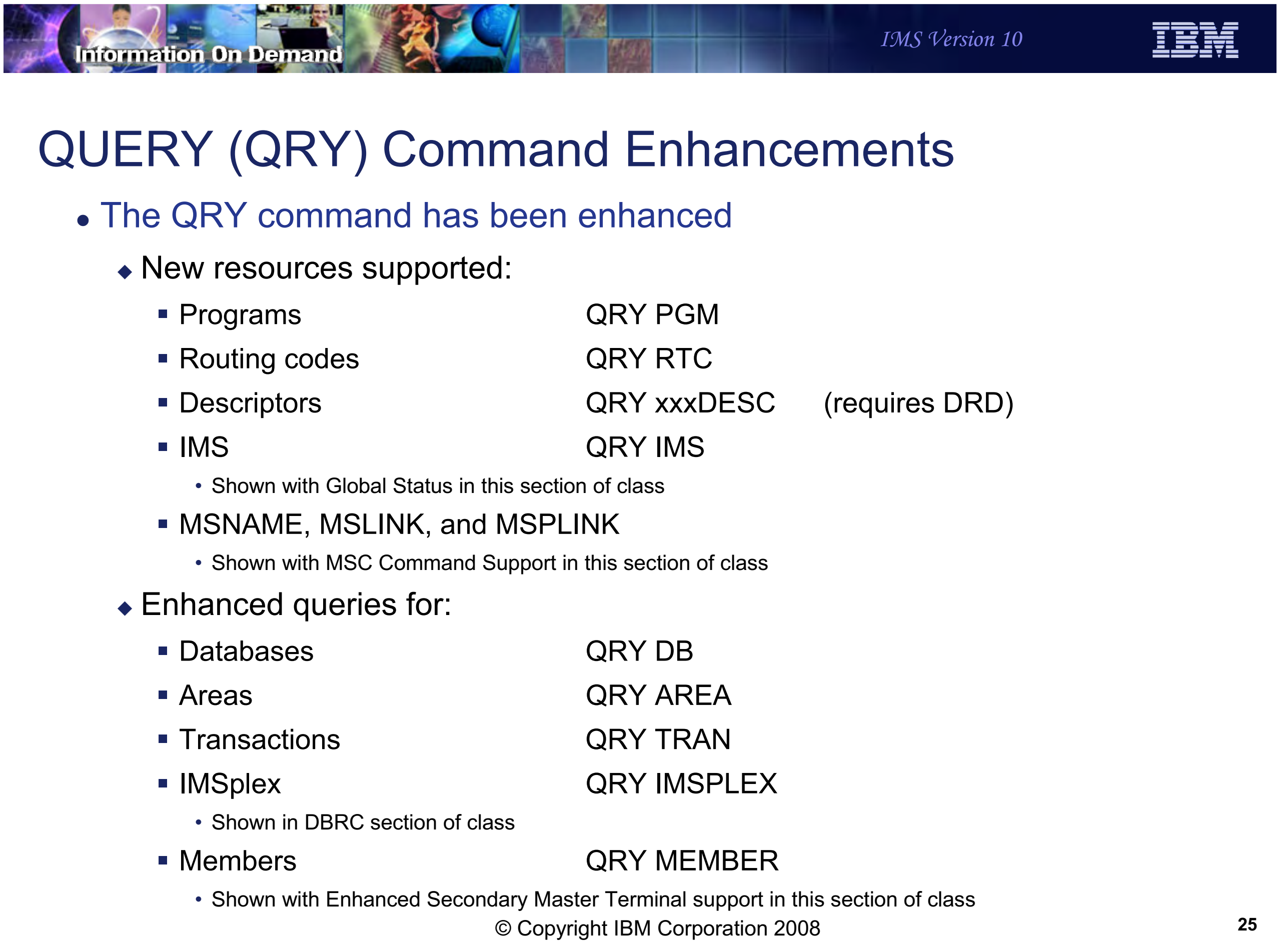
QUERY Command Enhancements

© Copyright IBM Corporation 2008

Figure 4-23. [No title]

Notes:

This section addresses enhancements to the QRY command.



Information On Demand
IMS Version 10
IBM

QUERY (QRY) Command Enhancements

- The QRY command has been enhanced
  - New resources supported:

Programs	QRY PGM	
Routing codes	QRY RTC	
Descriptors	QRY xxxDESC	(requires DRD)
IMS	QRY IMS	

      - Shown with Global Status in this section of class
    - MSNAME, MSLINK, and MSPLINK
      - Shown with MSC Command Support in this section of class
  - Enhanced queries for:

Databases	QRY DB
Areas	QRY AREA
Transactions	QRY TRAN
IMSplex	QRY IMSPLEX

      - Shown in DBRC section of class

Members	QRY MEMBER

      - Shown with Enhanced Secondary Master Terminal support in this section of class

© Copyright IBM Corporation 2008

25

Figure 4-24. QUERY (QRY) Command Enhancements CMA01.0

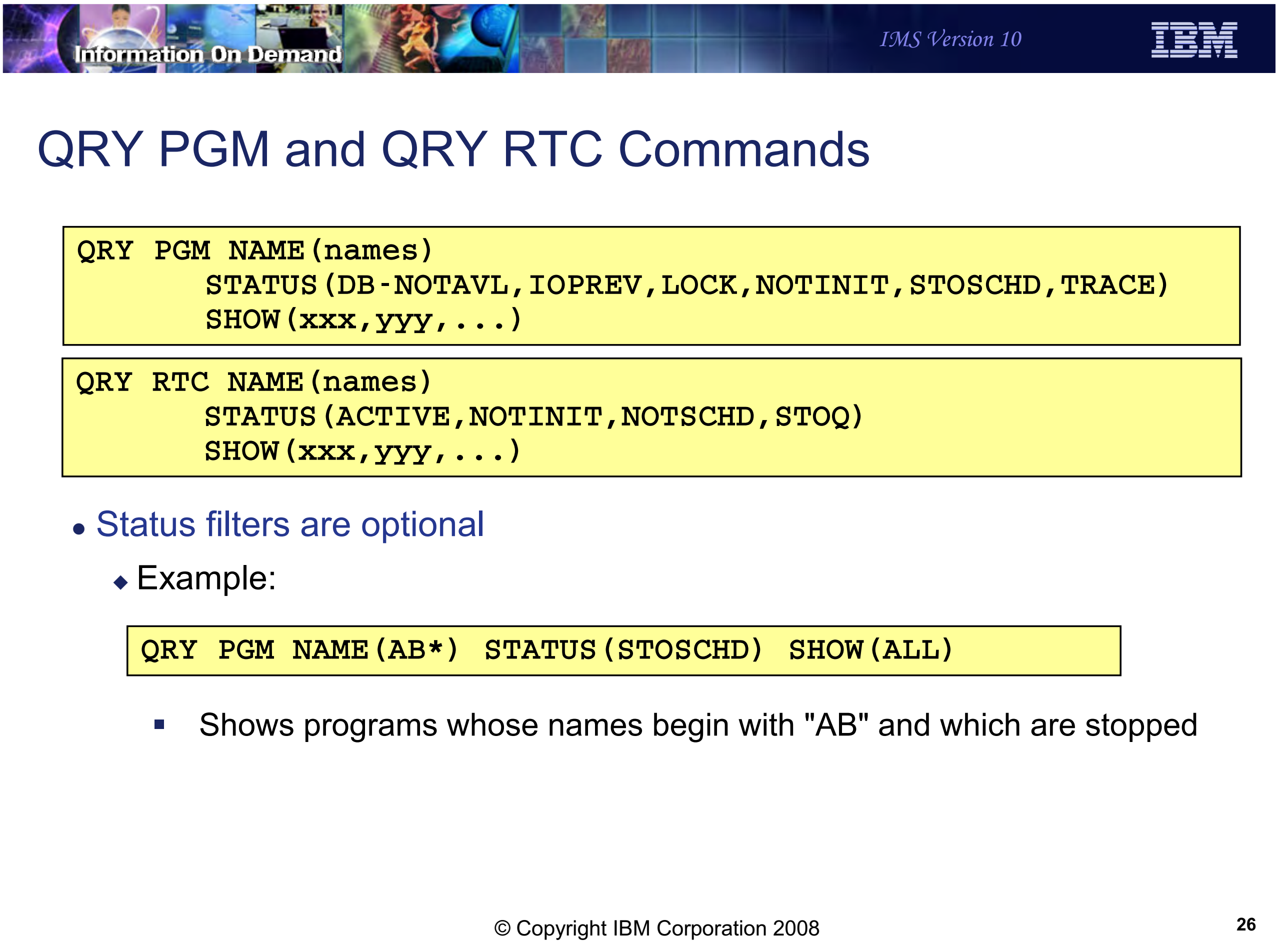
Notes:

The QUERY (or QRY) command has been enhanced to support new resources and extend the command for resources supported in previous releases.

The QRY command for descriptors is rejected if DRD is not active. The other QRY commands may be used with or without DRD.

The only change in the QUERY TRAN command is the way of filtering requests for conversational, fast path, remote, and response mode. Previously, these were done with STATUS(xxxxxxx) where xxxxxxxx was CONV, FPE, FPN, RMT, or RESP. There are new alternatives in IMS V10. The new alternatives are CONV(Y|N), FP(N|E|P), REMOTE(Y|N)m and RESP(Y|N).

Some of the QRY command enhancements are discussed in other sections of the class.


Information On Demand
IMS Version 10
IBM

QRY PGM and QRY RTC Commands

```
QRY PGM NAME(names)
        STATUS(DB-NOTAVL,IOPREV,LOCK,NOTINIT,STOSCHD,TRACE)
        SHOW(xxx,yyy,...)
```

```
QRY RTC NAME(names)
        STATUS(ACTIVE,NOTINIT,NOTSCHD,STOQ)
        SHOW(xxx,yyy,...)
```

- Status filters are optional
  - Example:

```
QRY PGM NAME(AB*) STATUS(STOSCHD) SHOW(ALL)
```

- Shows programs whose names begin with "AB" and which are stopped

© Copyright IBM Corporation 2008

Figure 4-25. QRY PGM and QRY RTC Commands CMA01.0

Notes:

QRY PGM and QRY RTC are new in IMS V10.

The following statuses may be specified for QRY PGM:

DB-NOTAVL	At least one database is not available
IOPREV	BMP with GSAM can't complete due to IO Prevention
LOCK	Program is LOCKED
NOTINIT	Program is not initialized - PSB or DMB not in ACBLIB
STOSCHD	Program scheduling has been stopped - UPD PGM xxx STOP(SCHD)
TRACE	Program trace is on - UPD PGM xxx START(TRACE)

The following statuses may specified for QRY RTC:

ACTIVE	RTC is active
NOTINIT	RTC is not initialized and can't be used
NOTSCHD	RTC is not scheduled
STOQ	Routing code is stopped

QRY Resource or Descriptor – Definition Attributes

- QRY commands can return definition attributes
  - DB, TRAN, PGM, RTC, DBDESC, TRANDESC, PGMDESC, and RTCDESC
- Examples:

```
QRY TRAN NAME(ZG*) SHOW(SPATRUNC,SPASZ)
```

```
Trancode MbrName CC SPATrunc SPASz
ZG0700RG IMS1     0  S         200
ZG2314WL IMS1     0  R         500
```

```
QRY PGMDESC NAME(PD100A) SHOW(SCHDTYPE)
```

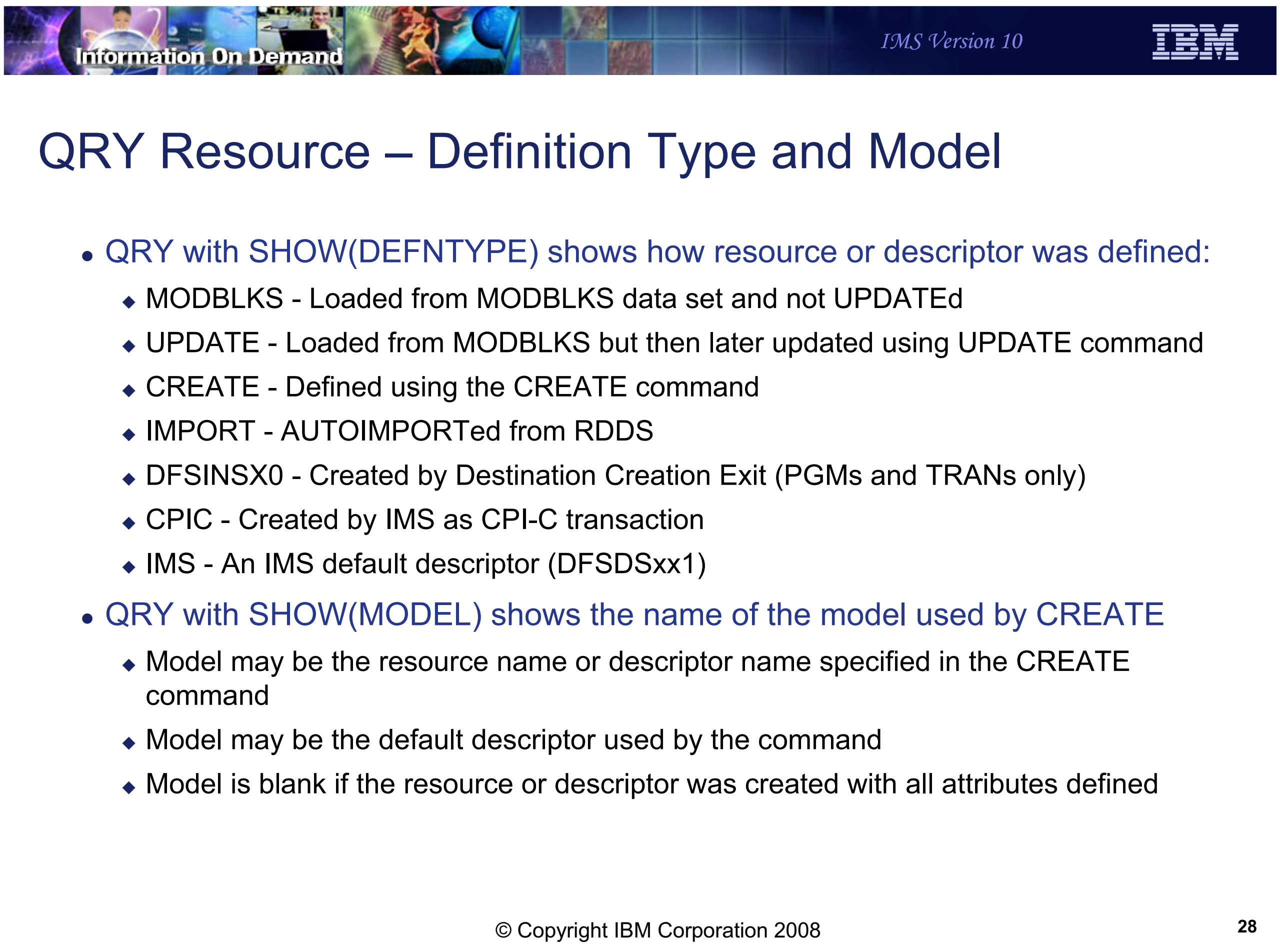
```
DescName MbrName CC SchdType
PD100A   IMS1     0  PARALLEL
```

© Copyright IBM Corporation 2008

Figure 4-26. QRY Resource or Descriptor – Definition Attributes CMA01.0

Notes:

QUERY commands may be used to show definition attributes for databases, programs, transactions, routing codes and their descriptors. These attributes may have been defined through system definition or through the use of DRD.


Information On Demand
IMS Version 10
IBM

QRY Resource – Definition Type and Model

- QRY with SHOW(DEFNTYPE) shows how resource or descriptor was defined:
  - MODBLKS - Loaded from MODBLKS data set and not UPDATEd
  - UPDATE - Loaded from MODBLKS but then later updated using UPDATE command
  - CREATE - Defined using the CREATE command
  - IMPORT - AUTOIMPORTed from RDDS
  - DFSINSX0 - Created by Destination Creation Exit (PGMs and TRANs only)
  - CPIC - Created by IMS as CPI-C transaction
  - IMS - An IMS default descriptor (DFSDSxx1)
- QRY with SHOW(MODEL) shows the name of the model used by CREATE
  - Model may be the resource name or descriptor name specified in the CREATE command
  - Model may be the default descriptor used by the command
  - Model is blank if the resource or descriptor was created with all attributes defined

© Copyright IBM Corporation 2008

Figure 4-27. QRY Resource – Definition Type and Model CMA01.0

Notes:

For all resources and descriptors, there are several new attributes that can be shown.

The first is definition type which shows how that resource or descriptor was created. The slide shows the possibilities. CREATE and IMPORT may be returned for both resources and descriptors. MODBLKS, UPDATE, DFSINSX0, and CPIC may be returned only for resources. IMS may be returned only for descriptors.

The second is the name of the model, if any, used by the CREATE command which created the definition.

QRY Resource – Timestamps

- QRY with SHOW(TIMESTAMP)
  - Four timestamps are displayed in format: YYYY.JJJ HH:MM:SS.TH
    - TIMECREATE – time resource or descriptor was created
      - Includes import time for resources defined by system definition
    - TIMEUPDATE – time resource or descriptor definition was last updated
    - TIMEACCESS – time resource or descriptor was last accessed or used as a model
      - For example, last DL/I call for a database
    - TIMEIMPORT – time resource was imported from RDDS
      - Create, update, and access times are remembered across warm restarts, emergency restarts, and EXPORT/IMPORT
      - Import times are remembered across warm restarts and emergency restarts

© Copyright IBM Corporation 2008

Figure 4-28. QRY Resource – Timestamps

Notes:

Four timestamps can be displayed. These are the time the resource was created, the time it was last updated, the time is was last accessed, and the time it was imported from an RDDS.

TIMECREATE is the time of the CREATE command, the time the definition was IMPORTed from RDDS, the time the definition was loaded from MODBLKS, or the time a transaction or program definition was created by DFSINSX0.

TIMEUPDATE is the time that an UPDATE command that updated a definitional attribute, not a status.

TIMEACCESS may be used to understand the last time a resource or descriptor was accessed. For resources and descriptors the meaning of TIMEACCESS is:

DB	time DB was last accessed by an application program (DL/I call)
PGM	time program (PSB) was last scheduled
TRAN	time message was last enqueued or dequeued by a program

RTCtime message was last enqueued to BALG using this routing code

DESCtime descriptor last used as a model in CREATE command

The create, update and access times are remembered across warm start, emergency restart, EXPORT and IMPORT. Import times are remembered across warm and emergency restarts.

The updating of the last access time is not logged between system checkpoints. After a restart, the last access time reflects the time recorded in the restart checkpoint log records. For warm starts this is the time recorded in the shutdown checkpoint. For emergency restarts, any access between the last system checkpoint and the failure of the IMS system will not be reflected in the information available to the restarted IMS system. For cold starts that import definitions from the RDDS, the times will be those in the RDDS. This will be the last access times at the time of the last export.

Information On Demand
IMS Version 10

QRY Resource Example

- Example:

Command

```
QRY TRAN NAME(INVTRA*) SHOW(TIMESTAMP,MODEL,DEFNTYPE)
```

Response

```
Trancode MbrName  CC  TimeCreate           TimeUpdate
INVTRAR  IMS1     0   2006.185 12:33:15.16 2006.186 09:15:26.17
INVTRA7  IMS1     0
```

```
TimeAccess            TimeImport            ModelName  DefnType
2006.195 17:30:04.56                        INVDESC    CREATE
2006.195 17:30:05.22  2006.183 07:11:43.44             IMPORT
```

© Copyright IBM Corporation 2008

Figure 4-29. QRY Resource Example CMA01.0

Notes:

Timestamps, model, and definition type have been added to the QRY command SHOW parameters. The example above is what would be shown by the TSO SPOC. The second part of the example above would be displayed by scrolling to the right.

Information On Demand

IMS Version 10

QUEUE Command

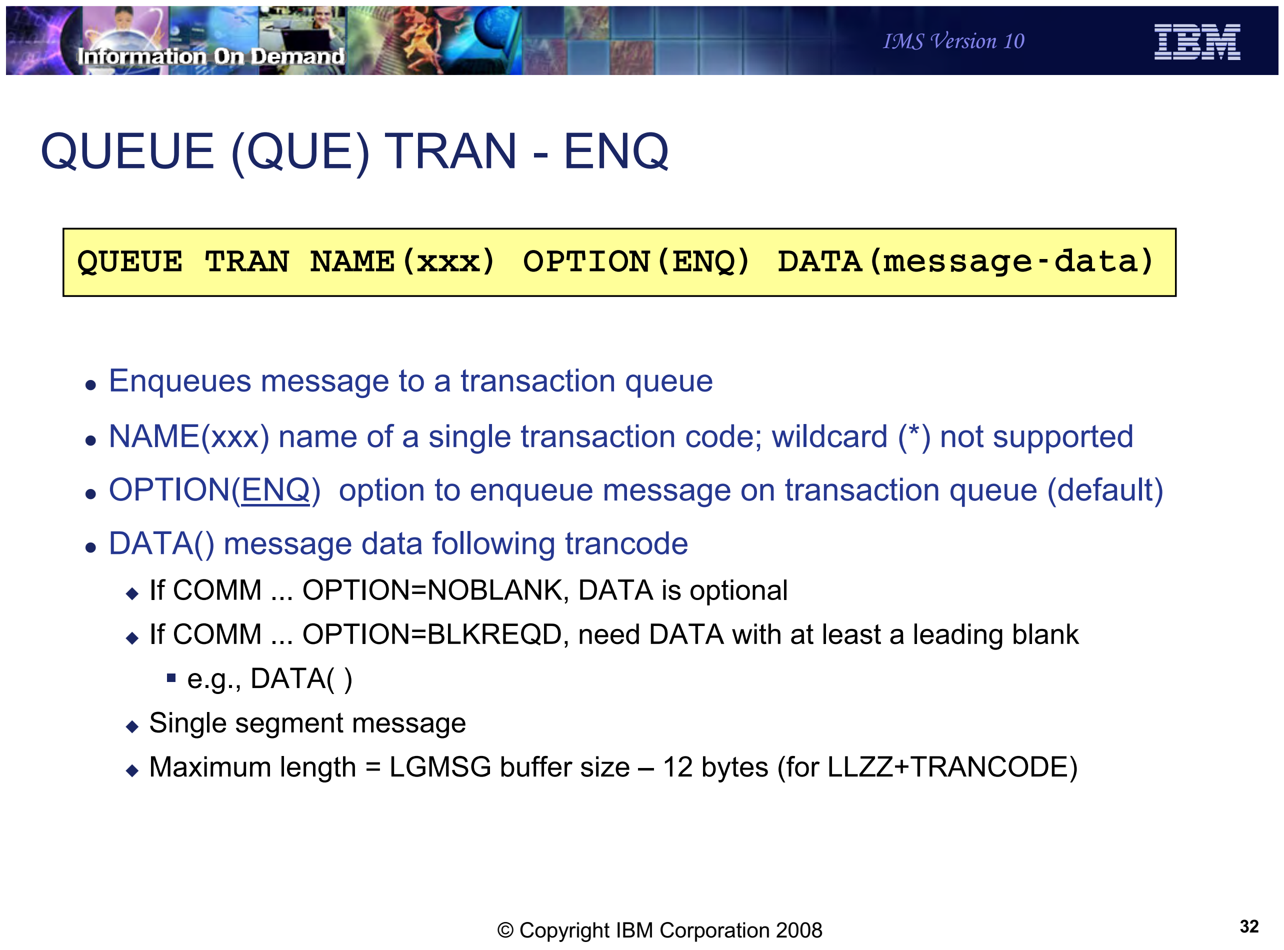
© Copyright IBM Corporation 2008

Figure 4-30. [No title] CMA01.0

Notes:

The QUEUE command in new in IMS V10.

Security for the QUQUE command is provided by OM with an exit, RACF, or both. The commands must be placed in the OPERCMDS resource class with a prefix of the CSL plex name. The short form of the command must be registered and may be followed by either TRAN or LTERM. For example, if IMSPLEX=PLEX1, then to provide security for the QUEUE TRAN command, you would RDEFINE CSLPLEX1.QUE.TRAN in the OPERCMDS class. Then PERMIT authorized users UPDATE access.



Information On Demand
IMS Version 10

QUEUE (QUE) TRAN - ENQ

```
QUEUE TRAN NAME(xxx) OPTION(ENQ) DATA(message-data)
```

- Enqueues message to a transaction queue
- NAME(xxx) name of a single transaction code; wildcard (*) not supported
- OPTION(ENQ) option to enqueue message on transaction queue (default)
- DATA() message data following trancode
  - If COMM ... OPTION=NOBLANK, DATA is optional
  - If COMM ... OPTION=BLKREQD, need DATA with at least a leading blank
    - e.g., DATA( )
  - Single segment message
  - Maximum length = LGMSG buffer size – 12 bytes (for LLZZ+TRANCODE)

© Copyright IBM Corporation 2008

Figure 4-31. QUEUE (QUE) TRAN - ENQ CMA01.0

Notes:

This shows the syntax when using the QUE command to enqueue a transaction to the transaction queue.

- NAME(xxx) identifies the transaction code – there can be only a single name in this command
- OPTION(ENQ) indicates that the trancode and DATA should be enqueued to the transaction queue (local or global); ENQ is the default
- DATA(xxx) is optional; if supplied, the trancode is followed by the data within the parentheses. Note that whatever is in DATA is moved character for character following the trancode. If OPTION=BLANK is specified on the COMM macro, then DATA must have a leading blank. All messages are single segment, and the maximum size of the DATA can be the size of the LGMSG buffer minus 12 bytes for LLZZ and the trancode.

QUEUE TRAN - ENQ

```
QUEUE TRAN NAME(xxx) OPTION(ENQ) DATA(message-data)
```

- Shared queues: Command processed only by command master
- Local queues: Command processed by all IMSs to which it is sent
- Command rejected if
    - Transaction is response mode, fast path, or conversational
    - Transaction is REMOTE(Y) and MSNAME is stopped
- LTERM name for message is DFSOMAPI
- Userid for message is the userid of the command client

© Copyright IBM Corporation 2008

Figure 4-32. QUEUE TRAN - ENQ CMA01.0

Notes:

The command is processed only by the command master in the shared queues environment. It is processed by all the IMSs to which it is sent in a local queues environment.

The message data does not include the LLZZ fields for the segment. IMS adds these fields.

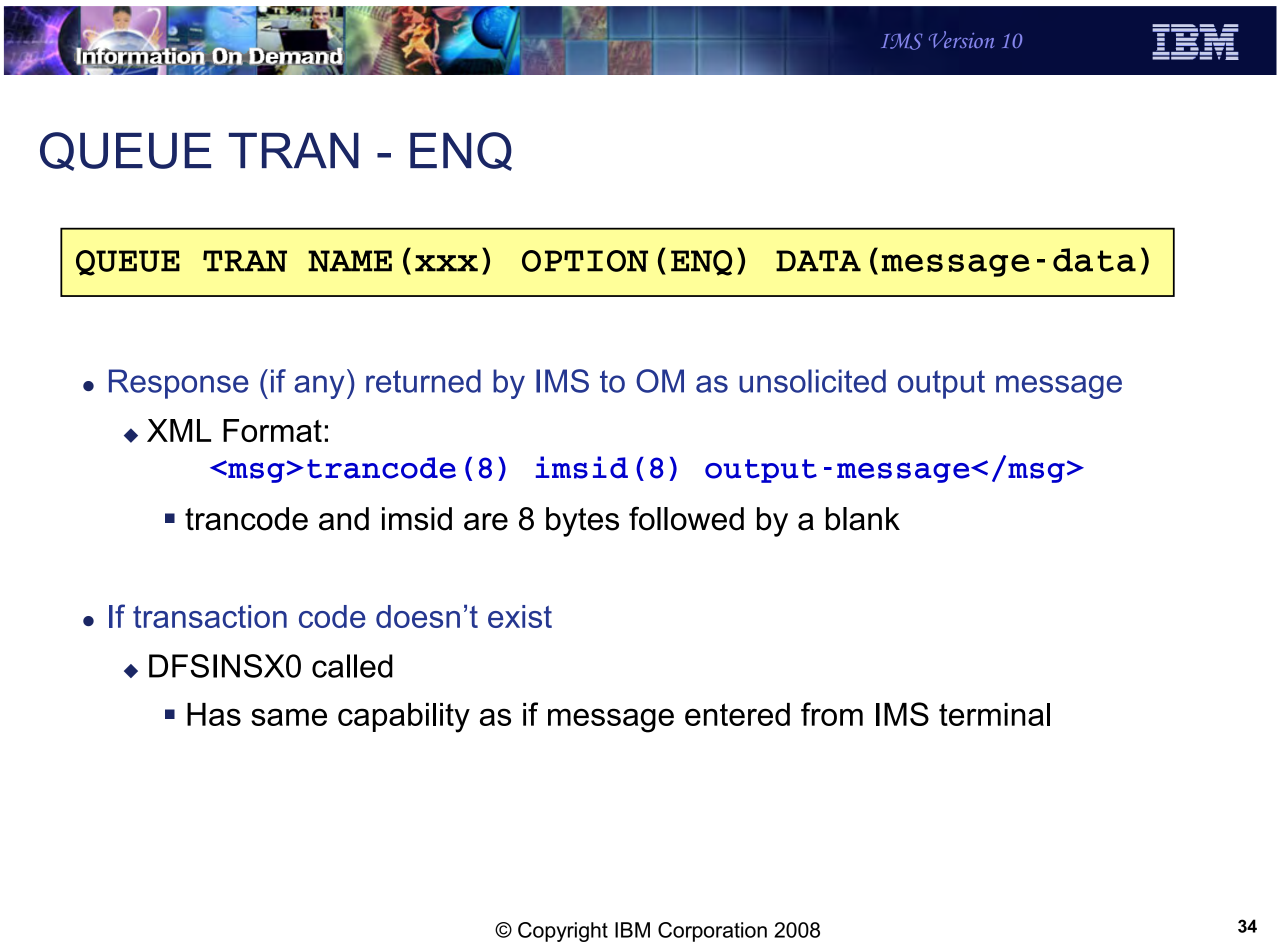
Response mode, fast path, and conversational transactions are rejected if entered. Remote transactions are rejected if the logical link is stopped.

If there is not an RM structure and therefore, there is no global status, the command is rejected if the local status of the transaction on the command master is STOPPED.

If there is an RM structure and if the global status in a shared queues environment is STOQ, the command will be rejected even if the local status is not stopped. If there is an RM structure and if the global status in a shared queues environment is STAQ, the command will be accepted even if the local status is stopped. In other words, if there is an

RM structure with shared queues, the global status controls whether the command is accepted or rejected.

The LTERM name used for messages enqueued with this command is DFSOMAPI. The userid associated with the message is the userid of the command client, such as TSO SPOC.


Information On Demand
IMS Version 10
IBM

QUEUE TRAN - ENQ

```
QUEUE TRAN NAME(xxx) OPTION(ENQ) DATA(message-data)
```

- Response (if any) returned by IMS to OM as unsolicited output message
  - XML Format:
    `<msg>trancode(8) imsid(8) output-message</msg>`
    - trancode and imsid are 8 bytes followed by a blank

- If transaction code doesn't exist
  - DFSINSX0 called
    - Has same capability as if message entered from IMS terminal

© Copyright IBM Corporation 2008

Figure 4-33. QUEUE TRAN - ENQ CMA01.0

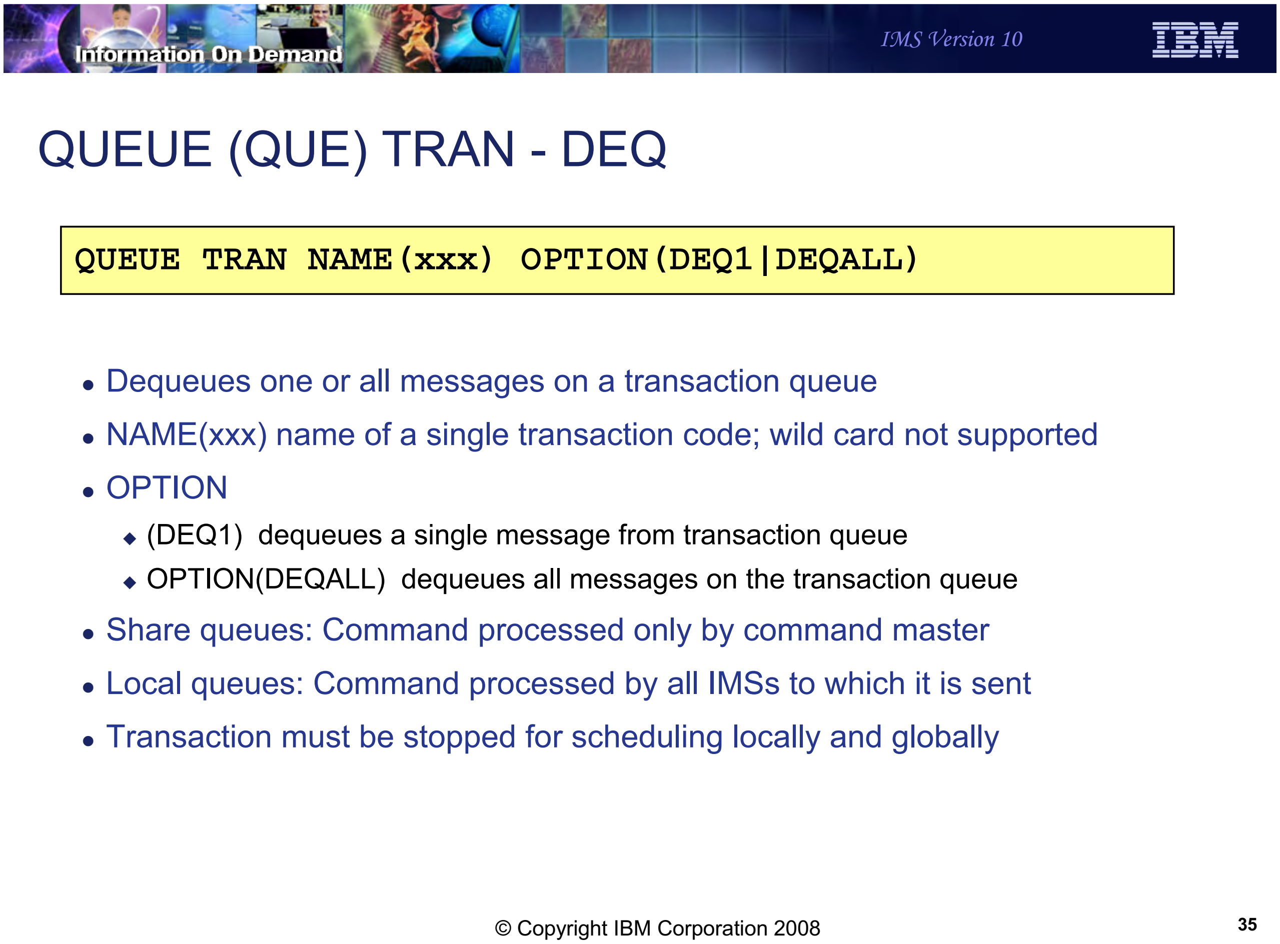
Notes:

Any response from this transaction is returned by IMS to OM in XML format as an unsolicited message. The XML tags are shown on the slide. The new OM2 Unsolicited Output Message support will place this message in the audit trail if enabled, and the SPOC (or a REXX program) can be used to display the contents of the message.

In the above XML format, 8 bytes is reserved for the trancode, followed by a blank, then 8 bytes for the IMSID, followed by a blank, then the message text.

If an IMS V8 or V9 system processes the message and inserts to the IO-PCB, the program will get an AD status code. This requires APARS PK30188 for IMS V8 and PK30189 for IMS V9.

If the transaction name is unknown, DFSINSX0 is invoked. The exit has the same capabilities as if a message from the network arrived with an unknown destination.


Information On Demand
IMS Version 10
IBM

QUEUE (QUE) TRAN - DEQ

```
QUEUE TRAN NAME(xxx) OPTION(DEQ1|DEQALL)
```

- Dequeues one or all messages on a transaction queue
- NAME(xxx) name of a single transaction code; wild card not supported
- OPTION
  - (DEQ1) dequeues a single message from transaction queue
  - OPTION(DEQALL) dequeues all messages on the transaction queue
- Share queues: Command processed only by command master
- Local queues: Command processed by all IMSs to which it is sent
- Transaction must be stopped for scheduling locally and globally

© Copyright IBM Corporation 2008

Figure 4-34. QUEUE (QUE) TRAN - DEQ CMA01.0

Notes:

Transactions can be dequeued from either the local or shared message queue by specifying OPTION(DEQ|DEQALL). DEQ deletes one message. DEQALL deletes all the messages on the queue. In a shared queues environment, the transactions are deleted by the command master. In a local queue environment, each IMS to which the command is routed deletes transactions from its local queue.

The transaction must be stopped for queuing and scheduling on the IMS processing the command.

The QUEUE TRAN … OPTION(DEQ1|DEQALL) command is not supported for Fast Path exclusive transactions.

If the transaction is unknown, DFSINSX0 can create the transaction for purposes of dequeuing it. This is useful only in a SQ environment since, if it is unknown in a local queues environment, there won't be any messages on the queue.

Information On Demand
IMS Version 10
IBM

QUEUE LTERM - ENQ

```
QUEUE LTERM NAME(xxx) OPTION(ENQ) DATA(message-data)
```

- Enqueues message to an LTERM queue
- NAME() name of a single logical terminal; wildcard (*) not supported
- OPTION(ENQ) option to enqueue message on LTERM queue
- DATA message data
  - Single segment message
  - Maximum length = LGMSG buffer size – 4 bytes (for LLZZ)
  - If EDIT=UC, message translated to upper case
- Share queues: Command processed only by command master
- Local queues: Command processed by all IMSs to which it is sent
- Command rejected if
  - LTERM is stopped on processing IMS
  - LTERM is remote and MSNAME is stopped

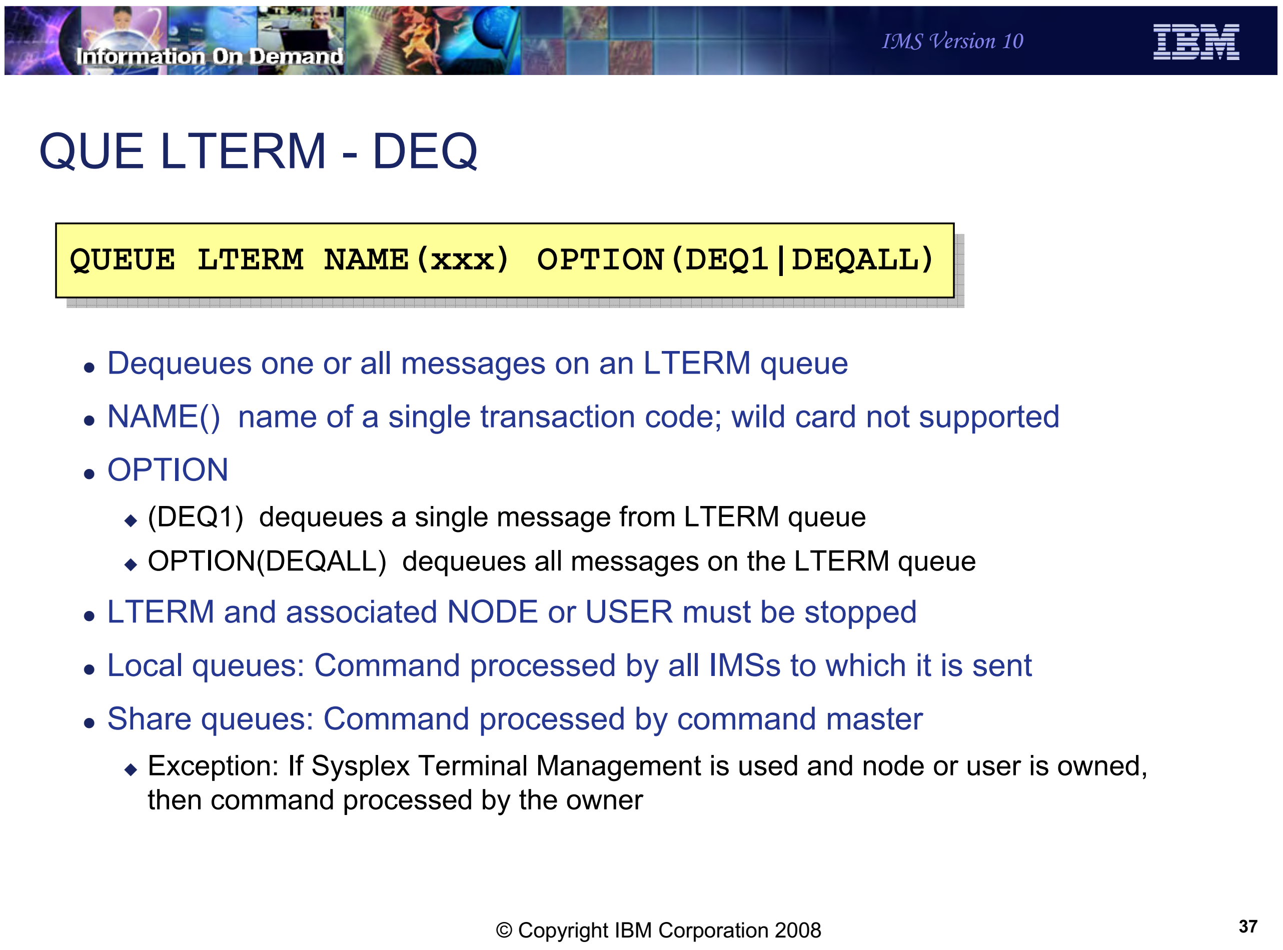
© Copyright IBM Corporation 2008

Figure 4-35. QUEUE LTERM - ENQ CMA01.0

Notes:

This chart shows the syntax of the QUEUE LTERM command. There can be only one name specified in the command. This is similar to the QUEUE TRAN command, except that the data can be 8 bytes longer since there is no trancode. Messages must be single segment.

In a SQ environment, only the command master queues the message. In a local queues environment, each IMS queues the message. The command is rejected by any IMS where the LTERM is stopped or if it is a remote LTERM and the MSNAME is stopped.



QUE LTERM - DEQ

```
QUEUE LTERM NAME(xxx) OPTION(DEQ1|DEQALL)
```

- Dequeues one or all messages on an LTERM queue
- NAME() name of a single transaction code; wild card not supported
- OPTION
  - (DEQ1) dequeues a single message from LTERM queue
  - OPTION(DEQALL) dequeues all messages on the LTERM queue
- LTERM and associated NODE or USER must be stopped
- Local queues: Command processed by all IMSs to which it is sent
- Share queues: Command processed by command master
  - Exception: If Sysplex Terminal Management is used and node or user is owned, then command processed by the owner

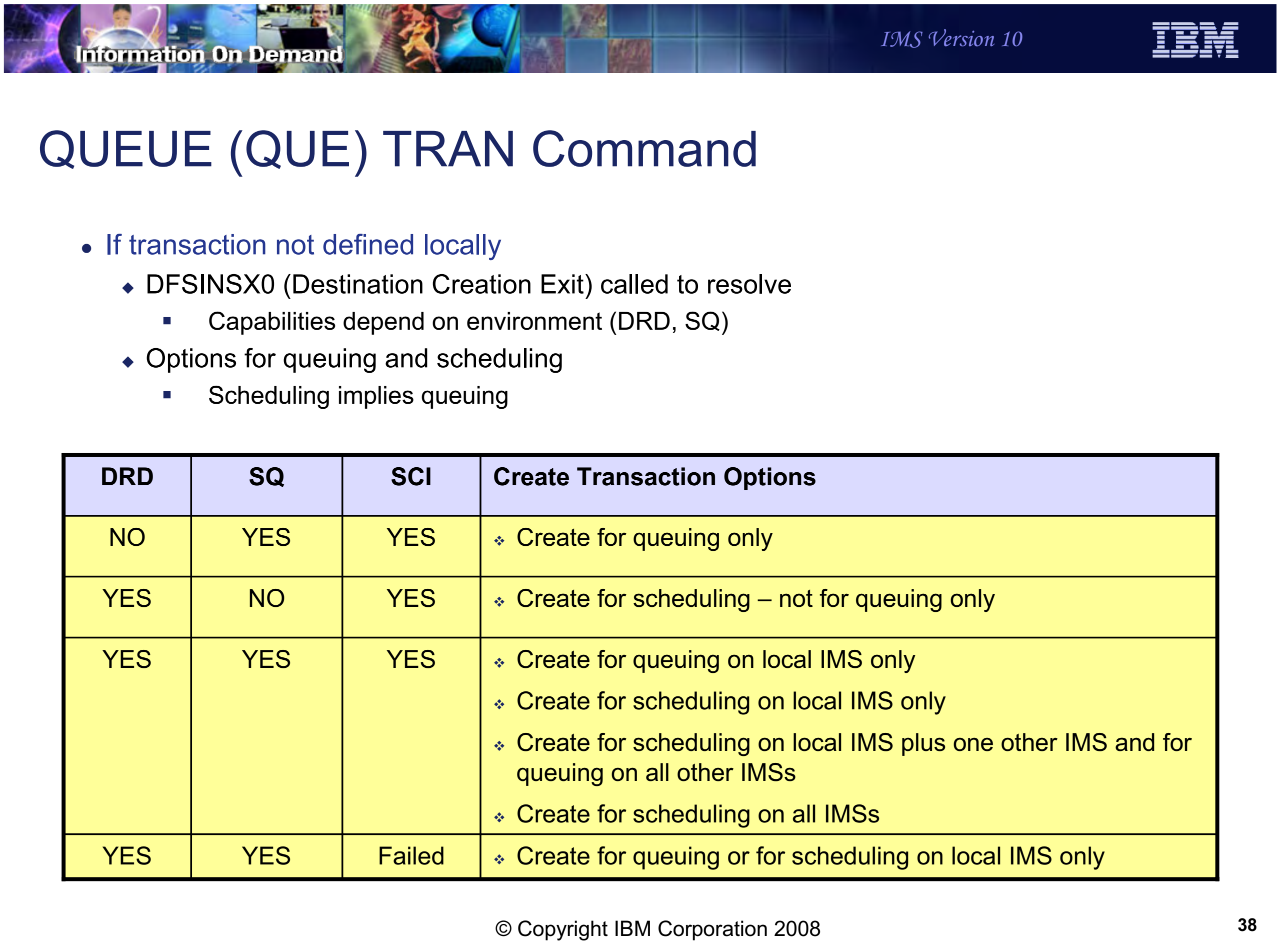
© Copyright IBM Corporation 2008

Figure 4-36. QUE LTERM - DEQ CMA01.0

Notes:

Messages can be dequeued from the local or shared LTERM queues. For this command to be accepted, the LTERM and associated NODE or USER must be stopped on the IMS processing the command.

This command is a little different when Sysplex Terminal Management is active (which means that SQ is also active). If STM is not enabled, the command is processed only by the command master. If STM is enabled, the command is processed by either the NODE or USER owner or by the command master if the node or user is not owned.


Information On Demand
IMS Version 10
IBM

QUEUE (QUE) TRAN Command

- If transaction not defined locally
  - DFSINSX0 (Destination Creation Exit) called to resolve
    - Capabilities depend on environment (DRD, SQ)
  - Options for queuing and scheduling
    - Scheduling implies queuing

DRD	SQ	SCI	Create Transaction Options
NO	YES	YES	❖ Create for queuing only
YES	NO	YES	❖ Create for scheduling – not for queuing only
YES	YES	YES	❖ Create for queuing on local IMS only ❖ Create for scheduling on local IMS only ❖ Create for scheduling on local IMS plus one other IMS and for queuing on all other IMSs ❖ Create for scheduling on all IMSs
YES	YES	Failed	❖ Create for queuing or for scheduling on local IMS only

© Copyright IBM Corporation 2008

Figure 4-37. QUEUE (QUE) TRAN Command CMA01.0

Notes:

If QUE TRAN is entered and the transaction is not defined to IMS, then IMS will invoke the Destination Determination Exit (DFSINSX0) to make a decision about how to process the command. The options are the same as for a message with an unknown destination arriving from the network. The table shows the options depending on the environment – DRD, SQ, and SCI. SCI should always be there, since CSL is required for Type-2 commands, but the last row of the table identifies the options if, for some reason, SCI has failed.

Information On Demand
IMS Version 10
IBM

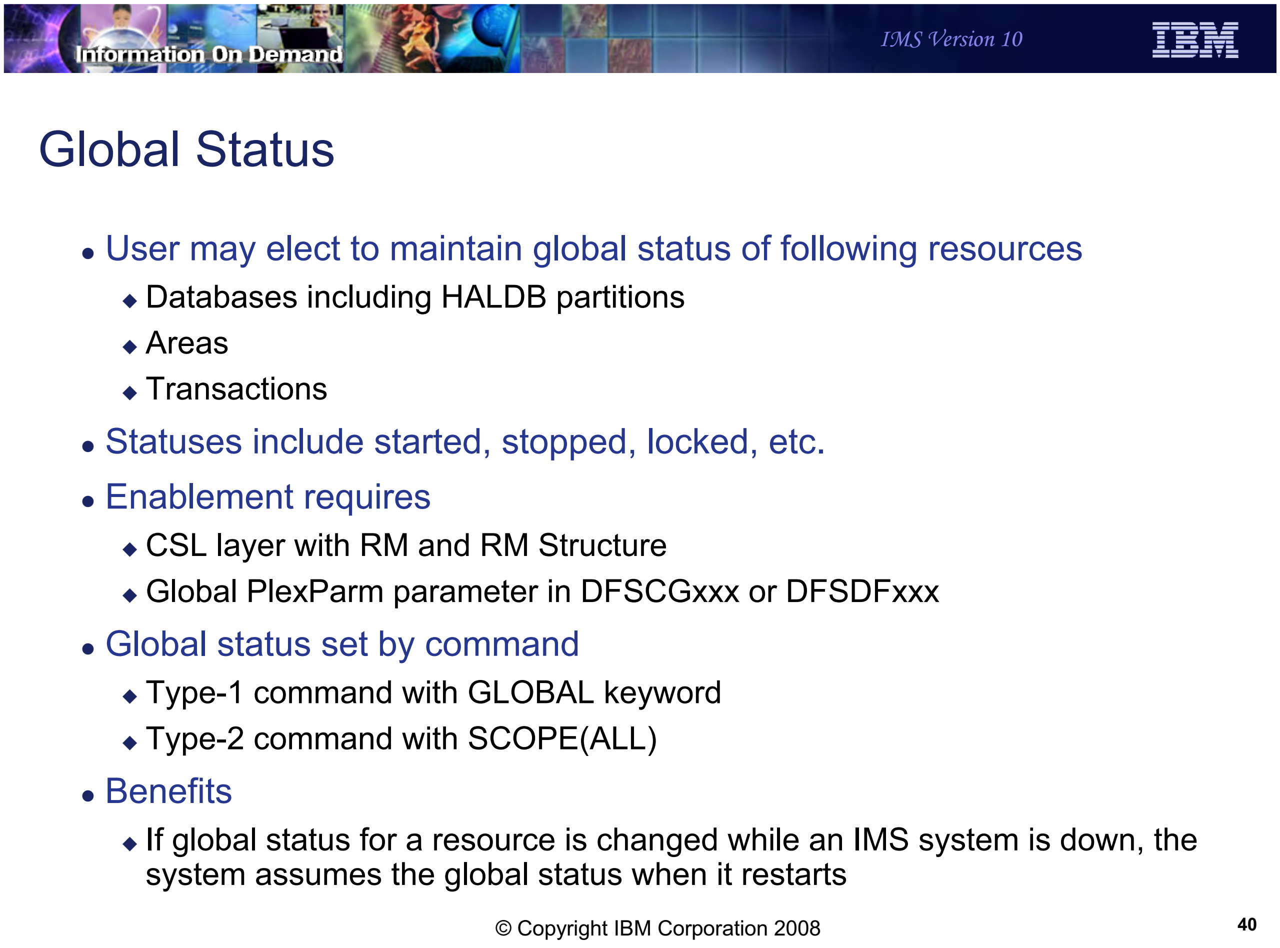
Global Status

© Copyright IBM Corporation 2008

Figure 4-38. [No title] CMA01.0

Notes:

This section discusses the concept of “global status” which is enabled by defining a Global PLEXPARM parameter.


Information On Demand
IMS Version 10

Global Status

- User may elect to maintain global status of following resources
  - Databases including HALDB partitions
  - Areas
  - Transactions
- Statuses include started, stopped, locked, etc.
- Enablement requires
  - CSL layer with RM and RM Structure
  - Global PlexParm parameter in DFSCGxxx or DFSDFxxx
- Global status set by command
  - Type-1 command with GLOBAL keyword
  - Type-2 command with SCOPE(ALL)
- Benefits
  - If global status for a resource is changed while an IMS system is down, the system assumes the global status when it restarts

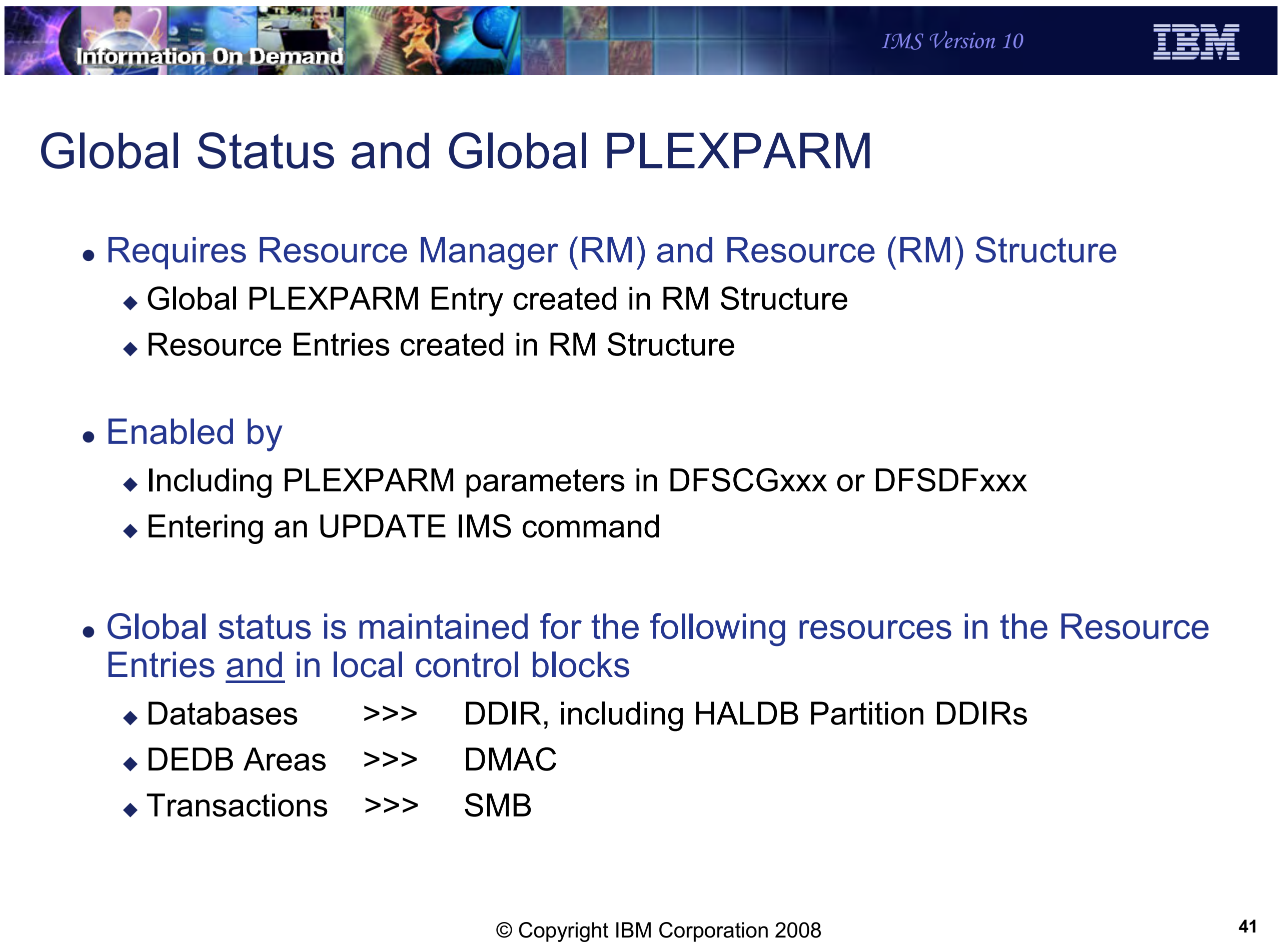
© Copyright IBM Corporation 2008

Figure 4-39. Global Status CMA01.0

Notes:

The user can elect to maintain global status for databases, areas, and transactions in a Resource Structure. Global status is enabled initially by coding the GSTSxxx parameters in the DFSCGxxx or DFSDFxxx PROCLIB member, and may be updated (or added) later by the UPD IMS command. Global status for a resource is set with a global command. Global commands are type-1 commands with the GLOBAL keyword or type-2 commands with SCOPE(ALL) specified.

The benefit of global status is that it allows users to more easily manage sysplex environments where some IMS systems are not always up. The user may change the status of a resource, such as starting it or stopping it, while some IMS systems are not up. When these systems are started, the resources in them assume the statuses that were changed while they were not running.


Information On Demand
IMS Version 10

Global Status and Global PLEXPARM

- Requires Resource Manager (RM) and Resource (RM) Structure
  - Global PLEXPARM Entry created in RM Structure
  - Resource Entries created in RM Structure
- Enabled by
  - Including PLEXPARM parameters in DFSCGxxx or DFSDFxxx
  - Entering an UPDATE IMS command
- Global status is maintained for the following resources in the Resource Entries and in local control blocks
  - Databases >>> DDIR, including HALDB Partition DDIRs
  - DEDB Areas >>> DMAC
  - Transactions >>> SMB

© Copyright IBM Corporation 2008

Figure 4-40. Global Status and Global PLEXPARM CMA01.0

Notes:

Maintaining global status on databases, DEDB areas, and transactions requires the CSL environment with a Resource Manager and Resource Structure. Resource entries are created in the structure and hold the global status. Enabling parameters are coded in DFSCGxxx, or the CSL section of DFSDFxxx. They may also be enabled after IMS is up and running using the UPDATE command.

Resources for which global status is kept are databases (all kinds except MSDBs), HALDB partitions, DEDB areas, and transactions. The global status is kept in both the resource entry and the local control block.

What Is Global Status?

- Global status is a resource status which has been set by a “global” command
  - Requires global status be enabled
  - Commands that set global status
    - Type-1 commands with GLOBAL keyword
    - Type-2 UPDATE command with SCOPE(ALL)
      - SCOPE(ALL) is the default for the UPDATE command
      - SCOPE(ACTIVE) does not set global status
  - Global status exists ONLY if set by one of these commands
  - Global status is never set by events in IMS
- Global status is maintained in a Resource Entry in the RM Structure
- A restarting IMS may change its local resource status to the global status of that resource
  - Cold start
  - Warm start
  - Emergency restart

© Copyright IBM Corporation 2008

Figure 4-41. What Is Global Status? CMA01.0

Notes:

Global status is status which has been set by a “global command” or more accurately, a command with global scope.

Both type-1 and type-2 command may be global in scope. For type-1 commands, this means including the keyword GLOBAL on the command - for example /DBR DATABASE XXX GLOBAL. For type-2 commands, this means including SCOPE(ALL) on the command - for example, UPD DB NAME(XXX) STOP(ACCESS) SCOPE(ALL). Global status is only set by one of these commands. It is never set by events in IMS.

When IMS is restarted, in may change its local status for a resource to the global status found in the structure. This is always true for cold starts, and sometimes true for warm or emergency restart. The conditions under which global status will override local status will be discussed a little later.

Information On Demand IMS Version 10 IBM

Global Status vs Local Status

- A resource control block contains both local and global status
  - Local status is current status of resource *in that IMS*
    - Maintained in local control blocks
  - Global status is status contained in Resource Entry *in RM structure*
    - Global status and timestamp of when global status was established also maintained in local resource control blocks
    - Global status is only used to set status when an IMS system is (re)started
  - *Local status does not always equal to global status*
    - e.g., Can stop database globally, then start it locally

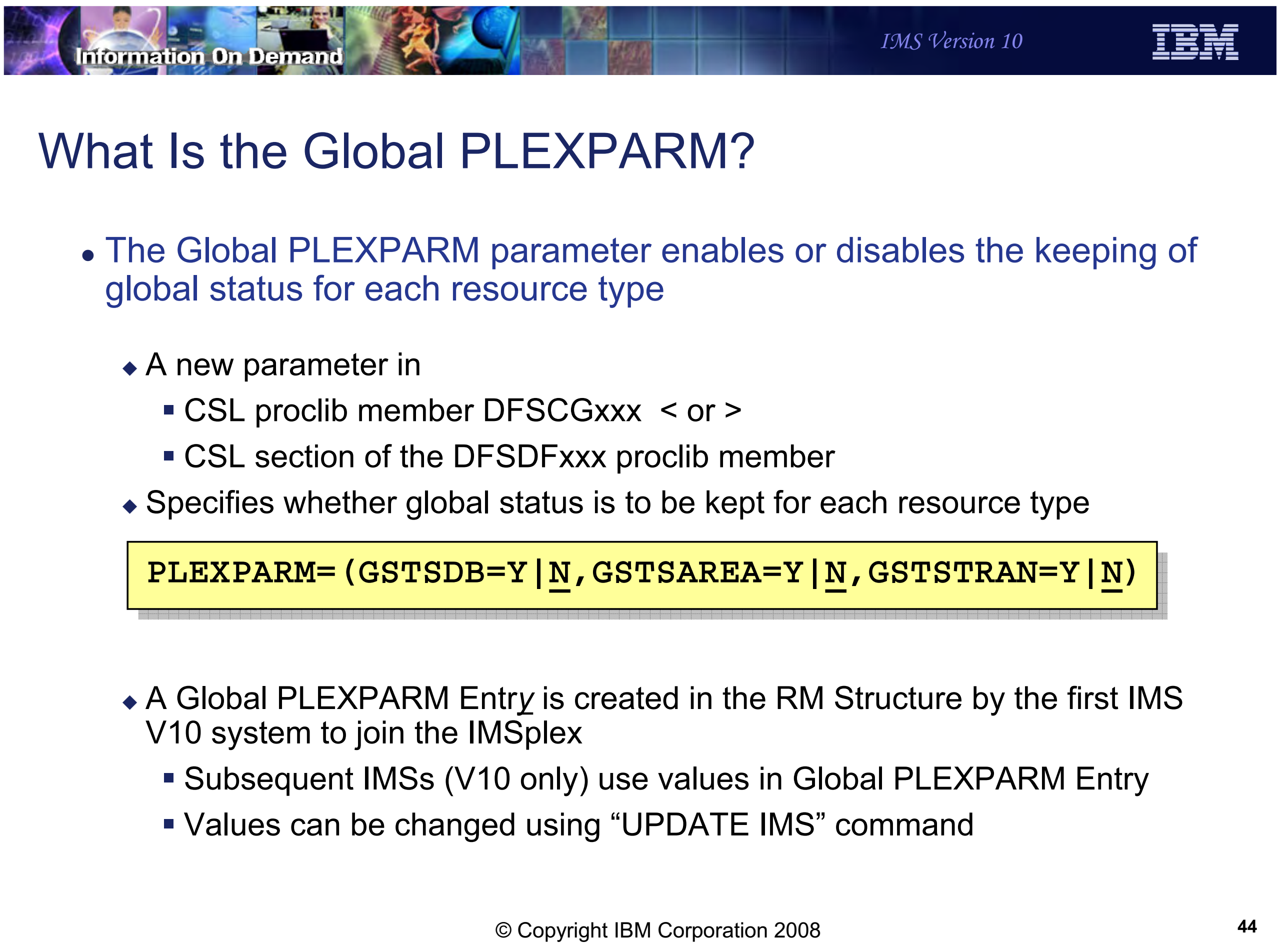
© Copyright IBM Corporation 2008

Figure 4-42. Global Status vs Local Status CMA01.0

Notes:

Resource controls blocks (DDIRs, DMACs, and SMBs) contain BOTH the local status and the last known global status. Global status is also contained in the resource entry of the structure. Both the control block and the structure also have the timestamp of when that status was set.

Local status does not always equal global status. For example, a database can be stopped globally, but then later started locally on one IMS. We will talk more later about this concept of local vs global status.


Information On Demand
IMS Version 10
IBM

What Is the Global PLEXPARM?

- The Global PLEXPARM parameter enables or disables the keeping of global status for each resource type
  - A new parameter in
    - CSL proclib member DFSCGxxx < or >
    - CSL section of the DFSDFxxx proclib member
  - Specifies whether global status is to be kept for each resource type

```
PLEXPARM=(GSTSDB=Y|N,GSTSAREA=Y|N,GSTSTRAN=Y|N)
```

  - A Global PLEXPARM Entry is created in the RM Structure by the first IMS V10 system to join the IMSplex
    - Subsequent IMSs (V10 only) use values in Global PLEXPARM Entry
    - Values can be changed using "UPDATE IMS" command

© Copyright IBM Corporation 2008

44

Figure 4-43. What Is the Global PLEXPARM? CMA01.0

Notes:

Whether or not global status is maintained is determined by the "global plexparm" parameters in CSL (either DFSCGxxx or the CSL section of DFSDFxxx). The PLEXPARM parameter has values for databases, DEDB areas, and transactions. In each case, the default is N (no), meaning that global status is not to be maintained for that resource type.

When any IMS V10 in the IMSplex joins the IMSplex, it looks for this parameter in the RM structure. If it does not find it, it creates a global plexparm entry with values for GSTSDB, GSTSAREA, and GSTSTRAN.

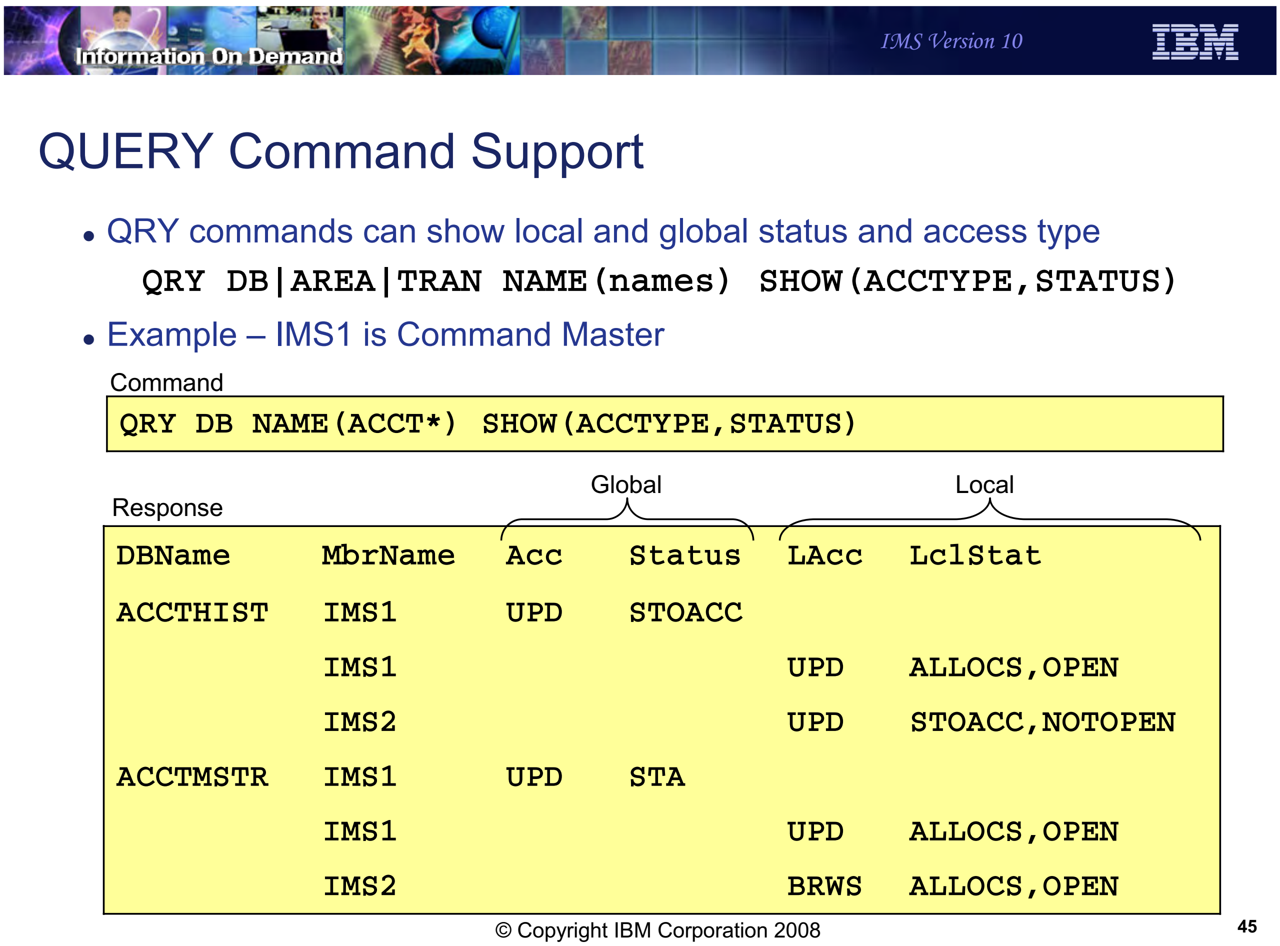
The user can change these parameters later using the UPDATE IMS command. The UPDATE IMS command is new in V10.

When an IMS system starts, it uses the values from the PLEXPARM entry in the RM structure if they exist. If they do not exist, it sets them from its parameters (those specified in DFSCGxxx or DFSDFxxx). If the values in the structure entry and those from the IMS system differ, message DFS3425I is issued. The message is:

DFS3425I PLEXPARM INIT VALUE rrrrrrrr=xxxx OVERRIDDEN WITH GLOBAL PLEXPARM VALUE rrrrrrrr=y

where rrrrrrr is GSTSDB, GSTSAREA, or GSTSTRAN, x is Y, N, or NULL and y is Y or N.

DBCTL systems do not have transaction resources. DCCTL systems do not have database or area resources. The PLEXPARM value for the resource type they do not support is NULL within these systems. Nevertheless, they can set values for all three parameters in their PROCLIB member or with the UPDATE IMS command.



Information On Demand
IMS Version 10
IBM

QUERY Command Support

- QRY commands can show local and global status and access type

```
QRY DB|AREA|TRAN NAME(names) SHOW(ACCTYPE,STATUS)
```

- Example – IMS1 is Command Master

Command

```
QRY DB NAME(ACCT*) SHOW(ACCTYPE,STATUS)
```

Response

```
                             Global                 Local
DBName     MbrName    Acc    Status        LAcc    LclStat
ACCTHIST   IMS1       UPD    STOACC
           IMS1                            UPD     ALLOCS,OPEN
           IMS2                            UPD     STOACC,NOTOPEN
ACCTMSTR   IMS1       UPD    STA
           IMS1                            UPD     ALLOCS,OPEN
           IMS2                            BRWS    ALLOCS,OPEN
```

© Copyright IBM Corporation 2008

Figure 4-44. QUERY Command Support CMA01.0

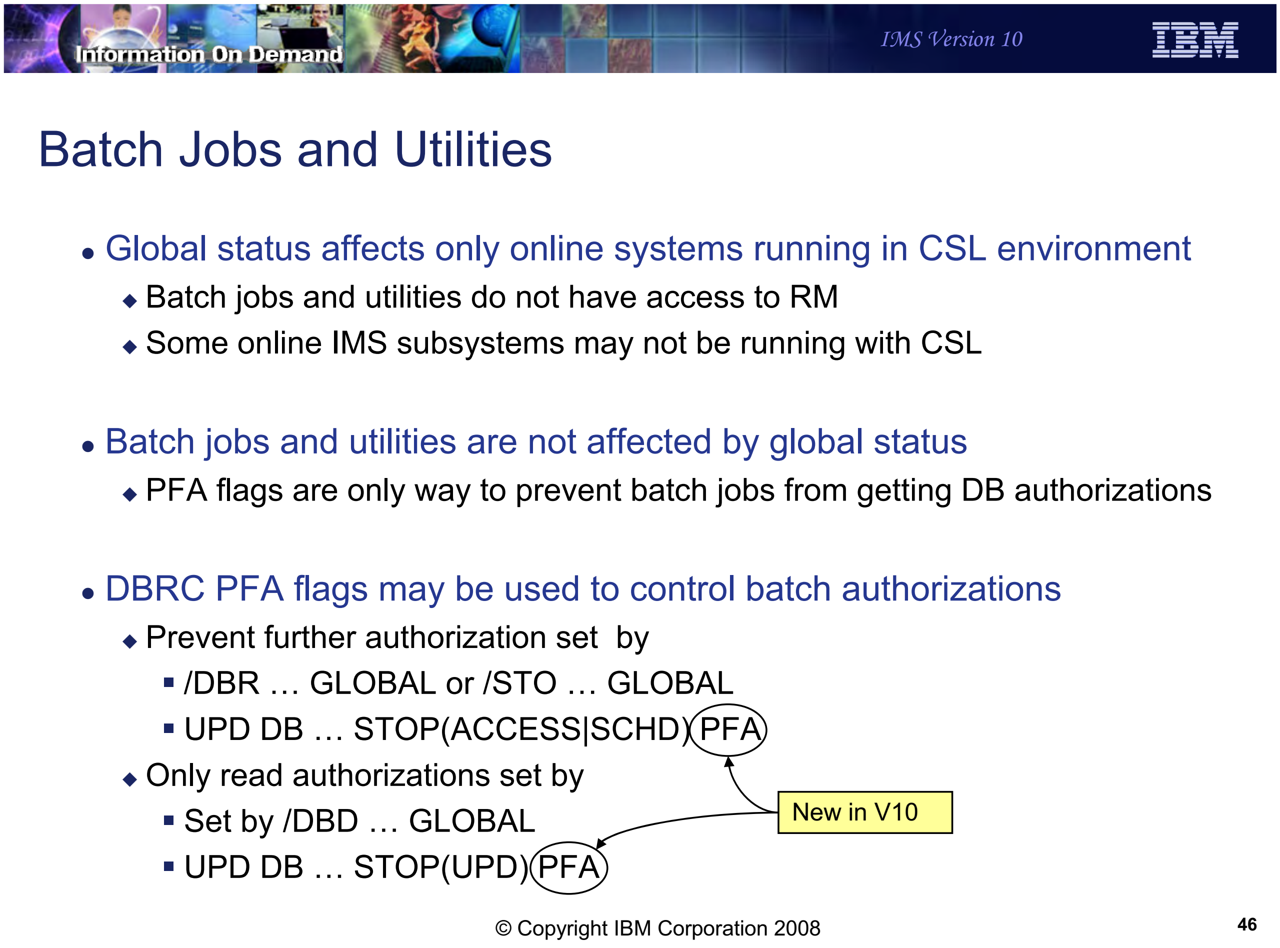
Notes:

This is an example of a command showing local and global DB status and local and global DB access type. Because neither LOCAL nor GLOBAL was specified, both are shown in the response.

The response could have been created by the following actions.

A /DBR DB ACCTHIST GLOBAL or a UPD DB NAME(ACCTHIST) STOP(ACCESS) SCOPE(ALL) command was issued. If the UPD DB command was used, it was routed to both IMS systems. This set the global status for the database to STOACC. It also caused the database to be closed on both IMS1 and IMS2. Later a /START DB ACCTHIST or an UPD DB NAME(ACCTHIST) START(ACCESS) SCOPE(ACTIVE) command was issued only on IMS1. This started the database locally on IMS1. The database was then allocated and opened. The QRY command shows the results for ACCTHIST.

Database ACCTMSTR was started on both systems with a global status of STA and a global access type UPD. A /DBD DB ACCTMSTR or a UPD DB NAME(ACCTMSTR) SET(ACCTYPE(BRWS)) was sent only to IMS2. This caused IMS2 to change its local access type to BRWS which is read-only. It also closed and reopened the database data sets for read. The QRY command shows the results for ACCTMSTR.



Information On Demand

IMS Version 10

Batch Jobs and Utilities

- Global status affects only online systems running in CSL environment
  - Batch jobs and utilities do not have access to RM
  - Some online IMS subsystems may not be running with CSL
- Batch jobs and utilities are not affected by global status
  - PFA flags are only way to prevent batch jobs from getting DB authorizations
- DBRC PFA flags may be used to control batch authorizations
  - Prevent further authorization set by
    - /DBR … GLOBAL or /STO … GLOBAL
    - UPD DB … STOP(ACCESS|SCHD) PFA
  - Only read authorizations set by
    - Set by /DBD … GLOBAL
    - UPD DB … STOP(UPD) PFA

New in V10

© Copyright IBM Corporation 2008

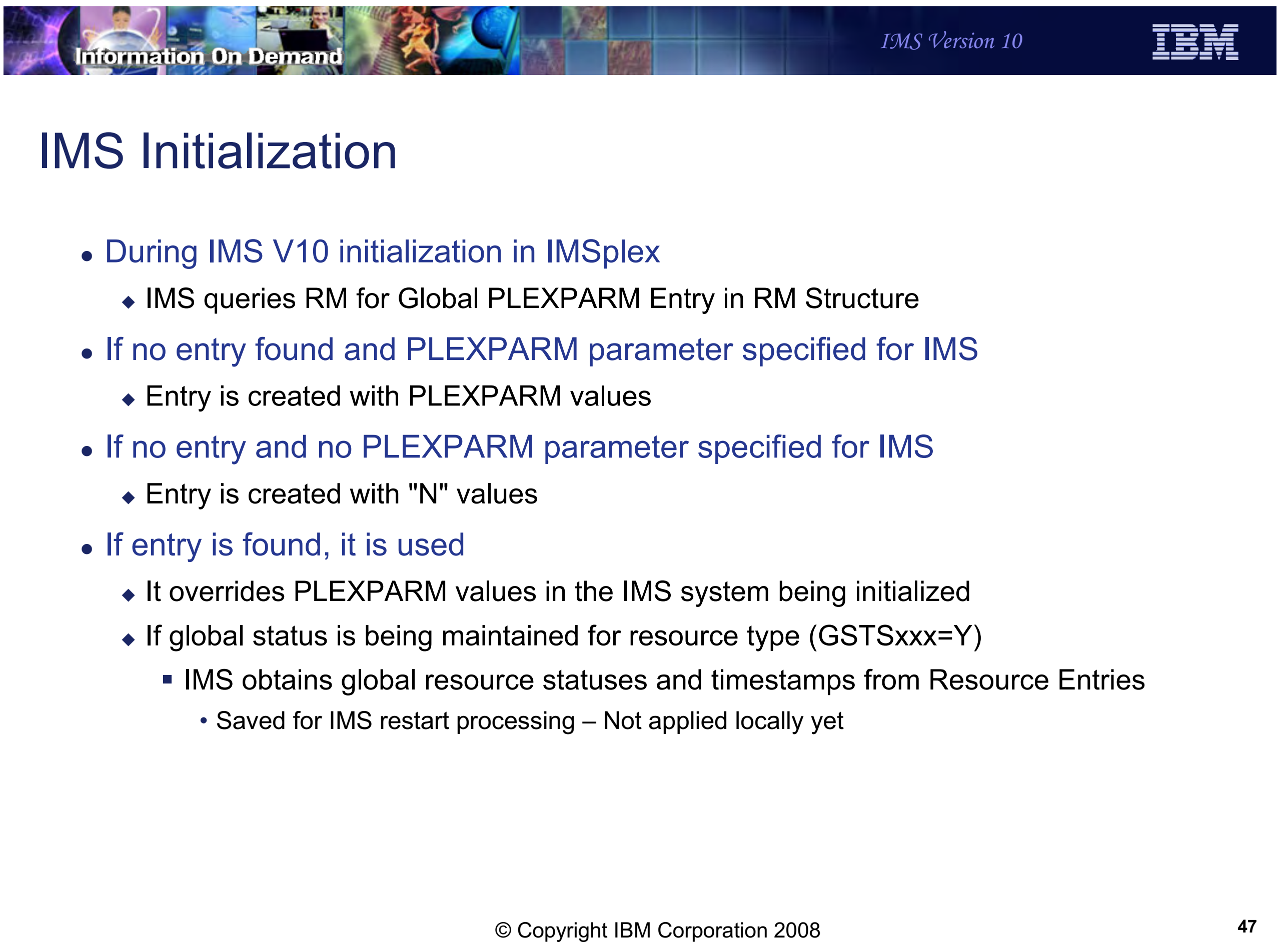
Figure 4-45. Batch Jobs and Utilities CMA01.0

Notes:

The PFA (prohibit further authorization) flag in DBRC prevents further authorization of a database to another IMS subsystem, including batch jobs. Because non-online IMS systems do not run with CSL and do not have access to the global status of a database, a global command such as UPD DB NAME(XYZ) STOP(ACCESS) SCOPE(ALL) would not prevent a batch job from getting authorization to that database. Only the PFA flag can do that. It also prevents online IMS systems that are not running with CSL and global status support from being able to authorize the DB. The point is global status is effective only within an IMSplex running with CSL and global status enabled.

DEDB areas also have a PFA flag, but cannot be accessed by batch jobs so it is seldom as much of an issue.

As we will see in the commands section of the class, the type-2 UPD DB class has been enhanced in V10 to set PFA flags.



Information On Demand
IMS Version 10
IBM

IMS Initialization

- During IMS V10 initialization in IMSplex
  - ◆ IMS queries RM for Global PLEXPARM Entry in RM Structure
- If no entry found and PLEXPARM parameter specified for IMS
  - ◆ Entry is created with PLEXPARM values
- If no entry and no PLEXPARM parameter specified for IMS
  - ◆ Entry is created with "N" values
- If entry is found, it is used
  - ◆ It overrides PLEXPARM values in the IMS system being initialized
  - ◆ If global status is being maintained for resource type (GSTSxxx=Y)
    - ▪ IMS obtains global resource statuses and timestamps from Resource Entries
      - Saved for IMS restart processing – Not applied locally yet

© Copyright IBM Corporation 2008

Figure 4-46. IMS Initialization CMA01.0

Notes:

During initialization, IMS queries RM for the global PLEXPARM entry and takes action depending on whether or not the entry exists.

If the entry does not exist, IMS will create one according to the type of IMS it is and according to the PLEXPARM parameter in DFSCGxxx of DFSDFxxx. Note that the default for each of the parameters is N (no global status to be maintained).

- DB/DC will add an entry with values from PLEXPARM for all three resource types.
- DBCTL will add an entry with values from PLEXPARM for databases and areas, but will leave the transaction parameter NULL
- DCCTL will add an entry with values for transactions but leave databases and areas NULL

If there is no PLEXPARM parameter in DFSCGxxx of DFSDFxxx, IMS will set the GSTSxxx values to N or NULL, depending on subsystem type.

IMS Restart

- When IMS is restarted
  - Global statuses from RM structure have already been obtained during initialization
- Cold starts apply global statuses to local statuses
  - Cold started systems assume the global status for each resource
- Warm and emergency restarts only apply global statuses to local statuses if global statuses were set while IMS was down
  - Warm started systems assume the global status only if it was changed while IMS was down

© Copyright IBM Corporation 2008

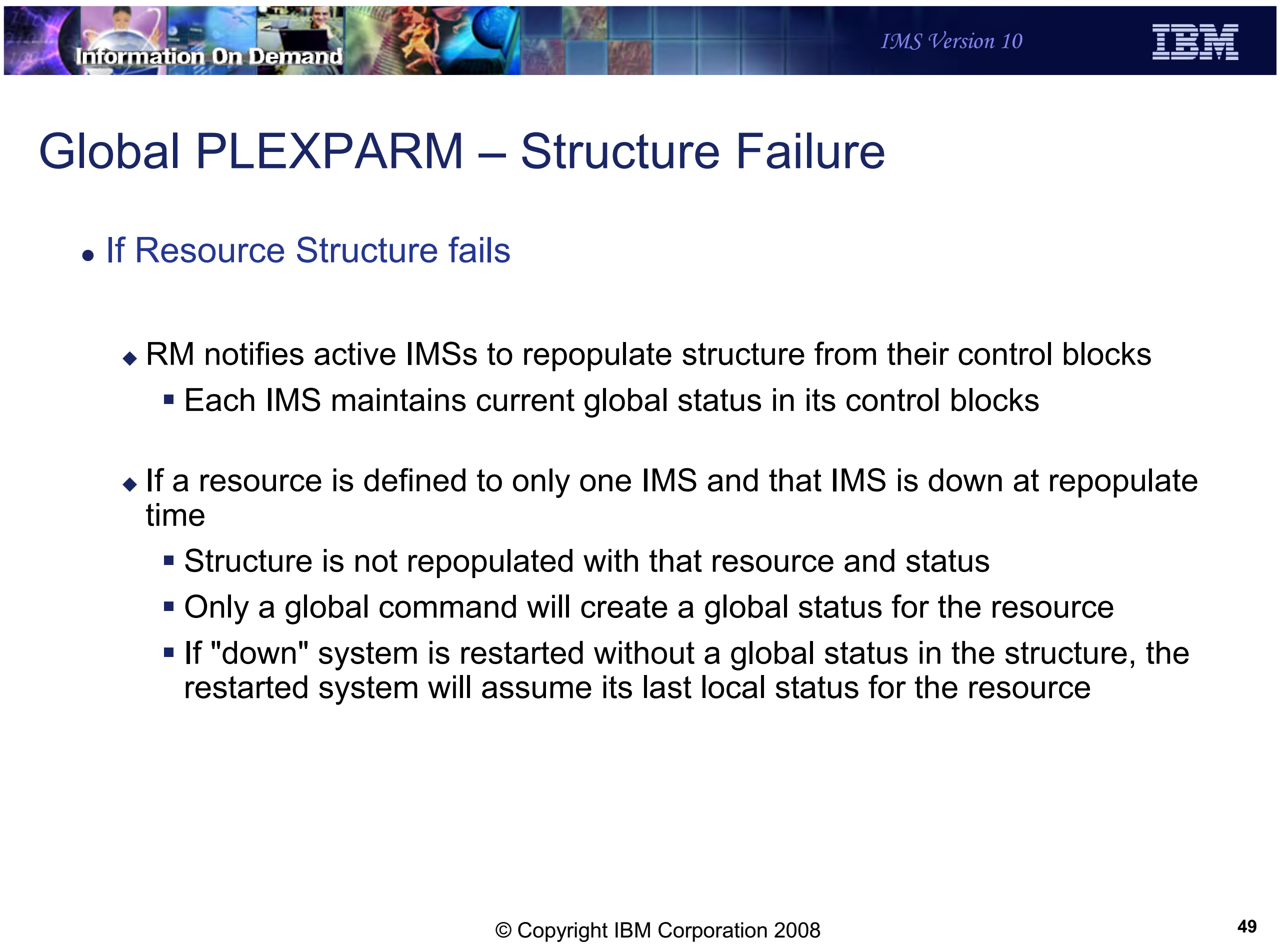
Figure 4-47. IMS Restart CMA01.0

Notes:

When IMS is restarted, it has already obtained the global values from the structure. It is during restart that IMS decides whether or not to apply this global status to the local control blocks.

- For cold starts IMS always sets the local status to the global status.
- For warm and emergency restart, the local status from the log records used for restart are first applied. (1) If the global status read from the structure was set at a time when IMS was down, the global status is applied to the control block. This occurs because someone changed the status globally while IMS was down. If IMS had been up, it would have changed its status at the time. (2) If the global status was set before IMS went down, then the local status is retained. The status on this IMS must have been changed after the global status was set.
- For an /ERE COLDBASE restart, the database and area statuses are set according to the rules for cold start. Transaction statuses are set according to the rules for emergency restart.

- For an /ERE COLDCOMM restart, the transaction statuses are set according to the rules of cold start. Database and area statuses are set according to the rules of emergency restart.



Information On Demand
IMS Version 10
IBM

Global PLEXPARM – Structure Failure

- If Resource Structure fails
    - RM notifies active IMSs to repopulate structure from their control blocks
        - Each IMS maintains current global status in its control blocks
    - If a resource is defined to only one IMS and that IMS is down at repopulate time
        - Structure is not repopulated with that resource and status
        - Only a global command will create a global status for the resource
        - If "down" system is restarted without a global status in the structure, the restarted system will assume its last local status for the resource

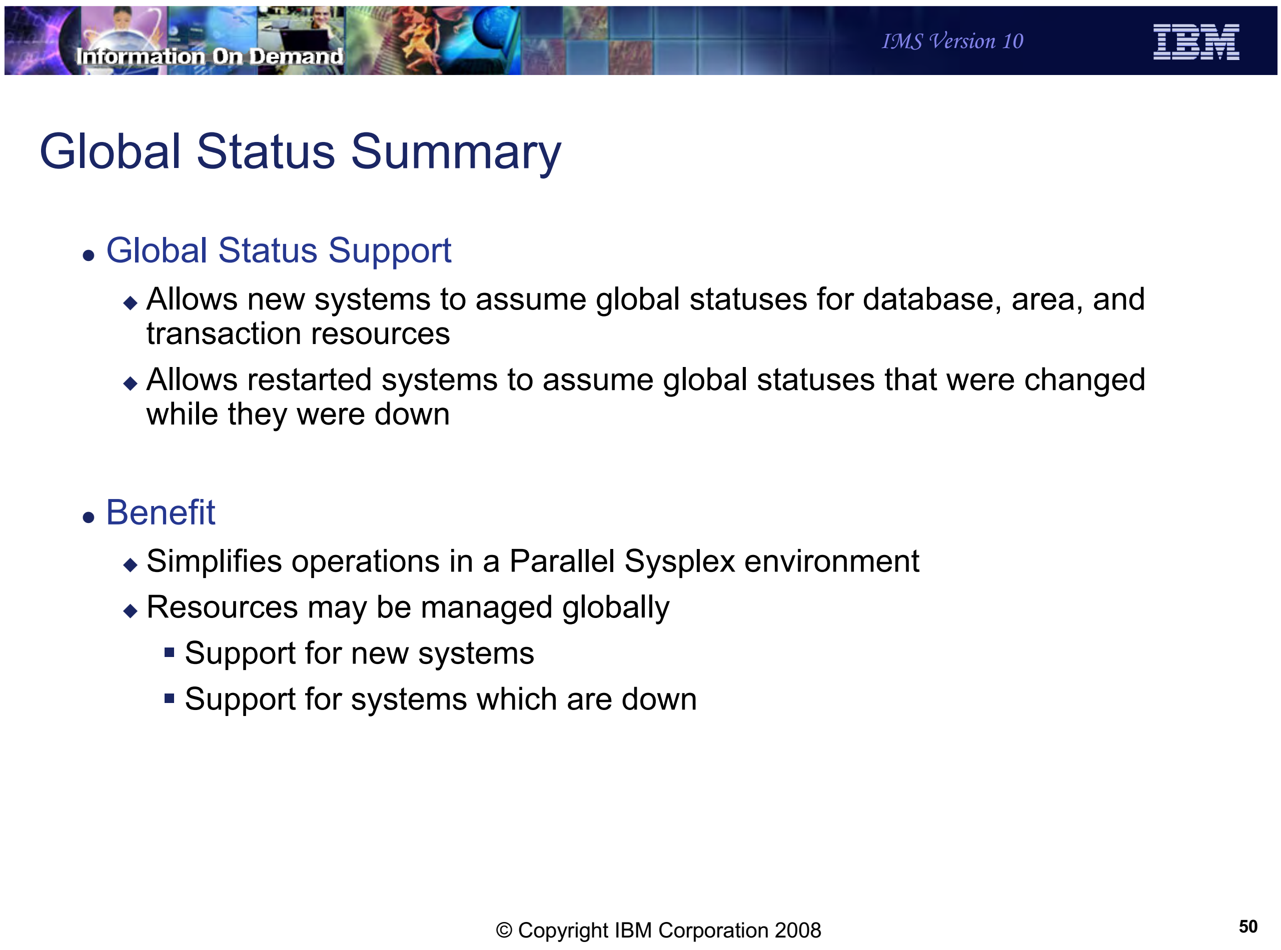
© Copyright IBM Corporation 2008

Figure 4-48. Global PLEXPARM – Structure Failure CMA01.0

Notes:

If the resource structure fails, RM notifies each IMSplex member to repopulate the structure from their control blocks. Remember, the control blocks have both the local and global status. Whichever IMS gets to the entry first sets the global status.

If a resource is defined to just one IMS, and that IMS is down at repopulate time (or all IMSs to which the resource is defined are down), then it cannot be repopulated. If a DB or AREA resource is later dynamically added to another IMS, there is no global status. If a transaction is dynamically entered by another IMS, a transaction entry is created with NULL status. If the DB or AREA resource then acquires global status, a DB or AREA entry is created with the appropriate global status. A transaction entry is updated with the global status. When the down IMS is restarted, it takes its global status from the structure.



IMS Version 10

Global Status Summary

- Global Status Support
  - Allows new systems to assume global statuses for database, area, and transaction resources
  - Allows restarted systems to assume global statuses that were changed while they were down
- Benefit
  - Simplifies operations in a Parallel Sysplex environment
  - Resources may be managed globally
    - Support for new systems
    - Support for systems which are down

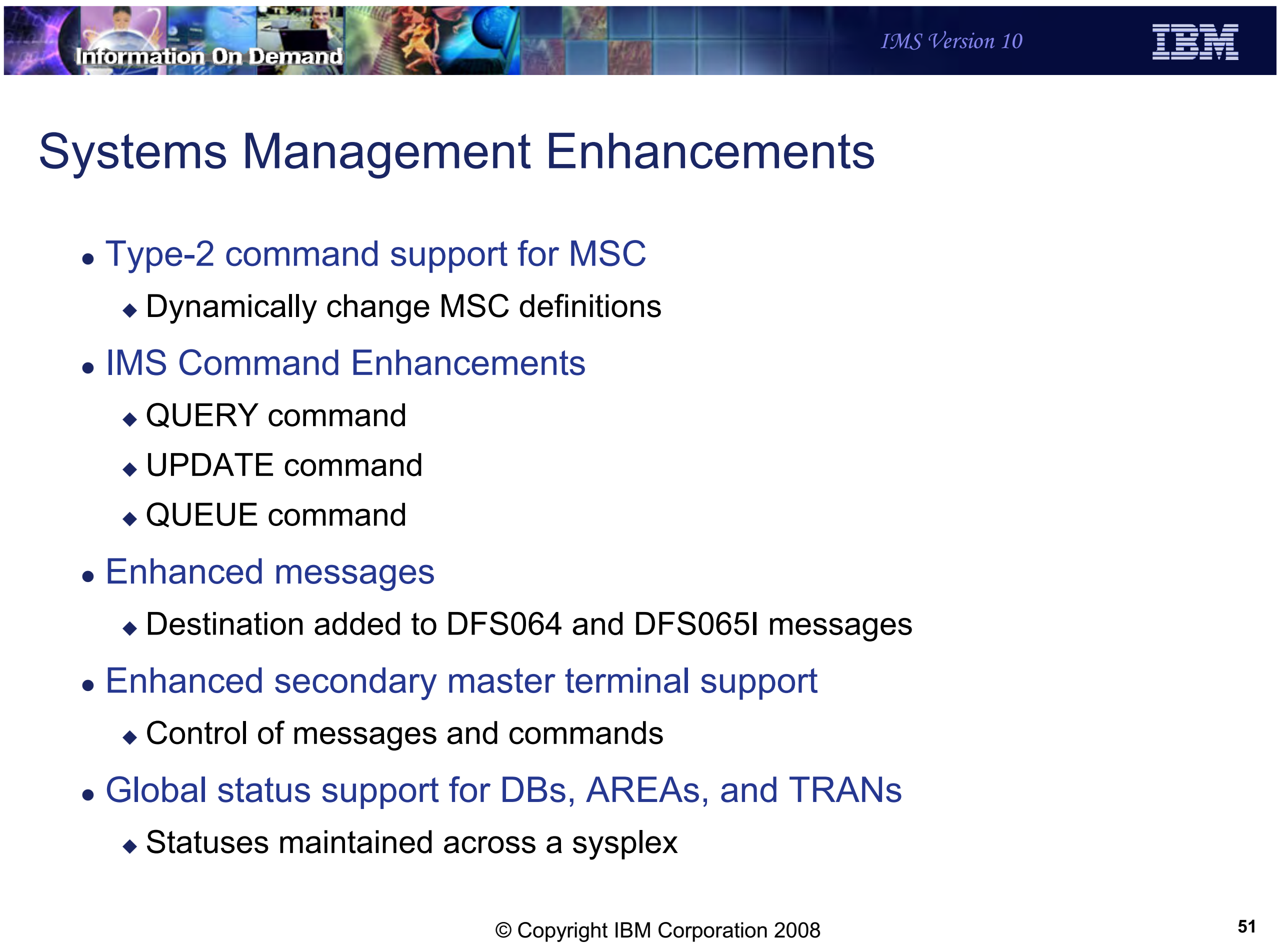
© Copyright IBM Corporation 2008

50

Figure 4-49. Global Status Summary CMA01.0

Notes:

The user can elect to maintain global statuses for databases, areas, and transactions in a Resource Structure. Global status is enabled initially by coding the GSTSxxx parms in the DFSCGxxx or DFSDFxxx PROCLIB member. It may be updated or added later by the UPD IMS command. Both type-1 and type-2 commands can set global statuses for databases and areas, Only type-2 commands can set global statuses for transactions. The benefit is that it allows the user to set a global status while some IMSs are down and then have an IMS accept that status when it is restarted.


Information On Demand
IMS Version 10

Systems Management Enhancements

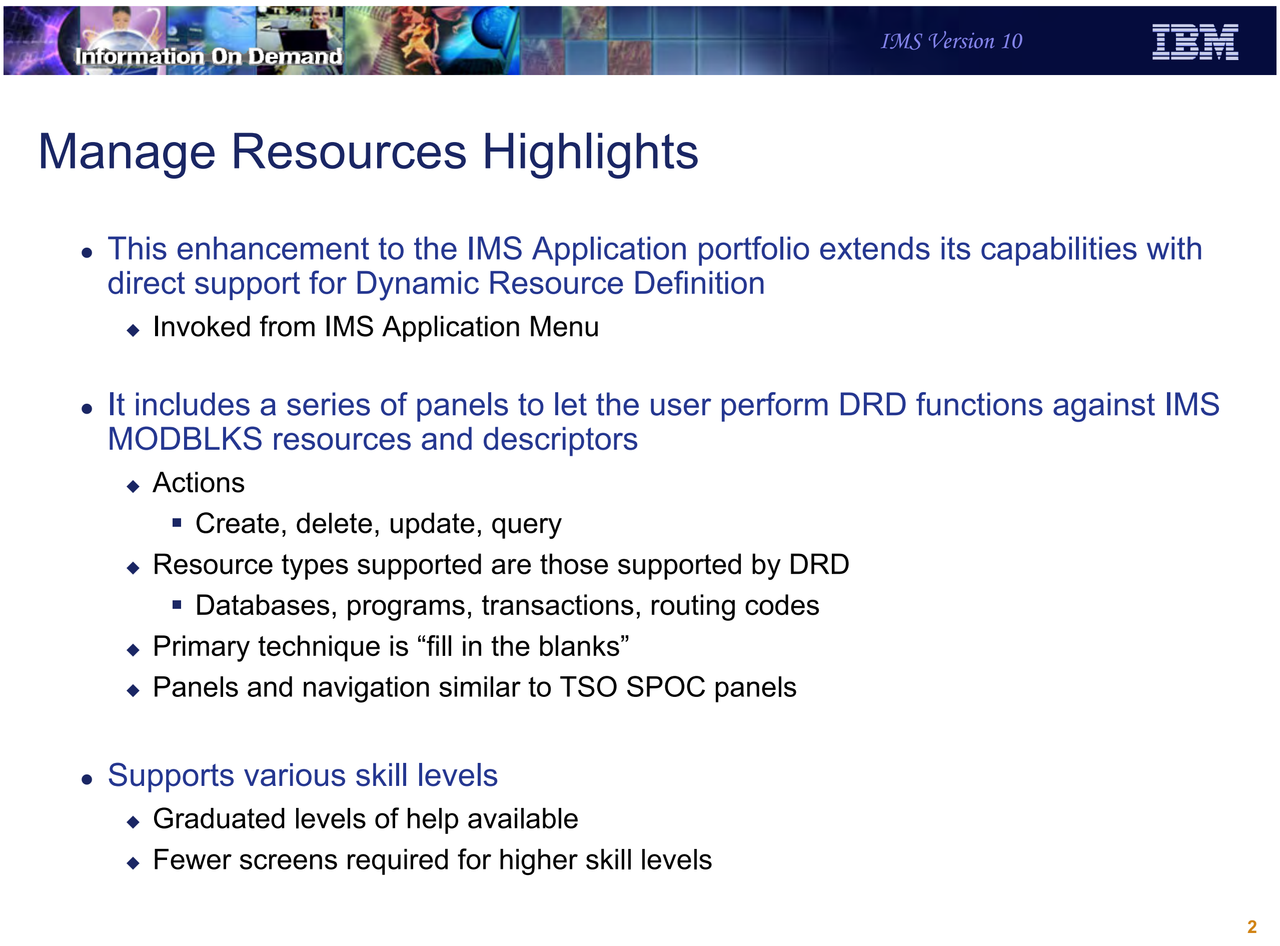
- Type-2 command support for MSC
  - Dynamically change MSC definitions
- IMS Command Enhancements
  - QUERY command
  - UPDATE command
  - QUEUE command
- Enhanced messages
  - Destination added to DFS064 and DFS065I messages
- Enhanced secondary master terminal support
  - Control of messages and commands
- Global status support for DBs, AREAs, and TRANs
  - Statuses maintained across a sysplex

© Copyright IBM Corporation 2008

Figure 4-50. Systems Management Enhancements CMA01.0

Notes:

Unit 5. Managing Resources



Information On Demand
IMS Version 10
IBM

Manage Resources Highlights

- This enhancement to the IMS Application portfolio extends its capabilities with direct support for Dynamic Resource Definition
  - Invoked from IMS Application Menu
- It includes a series of panels to let the user perform DRD functions against IMS MODBLKS resources and descriptors
  - Actions
    - Create, delete, update, query
  - Resource types supported are those supported by DRD
    - Databases, programs, transactions, routing codes
  - Primary technique is “fill in the blanks”
  - Panels and navigation similar to TSO SPOC panels
- Supports various skill levels
  - Graduated levels of help available
  - Fewer screens required for higher skill levels

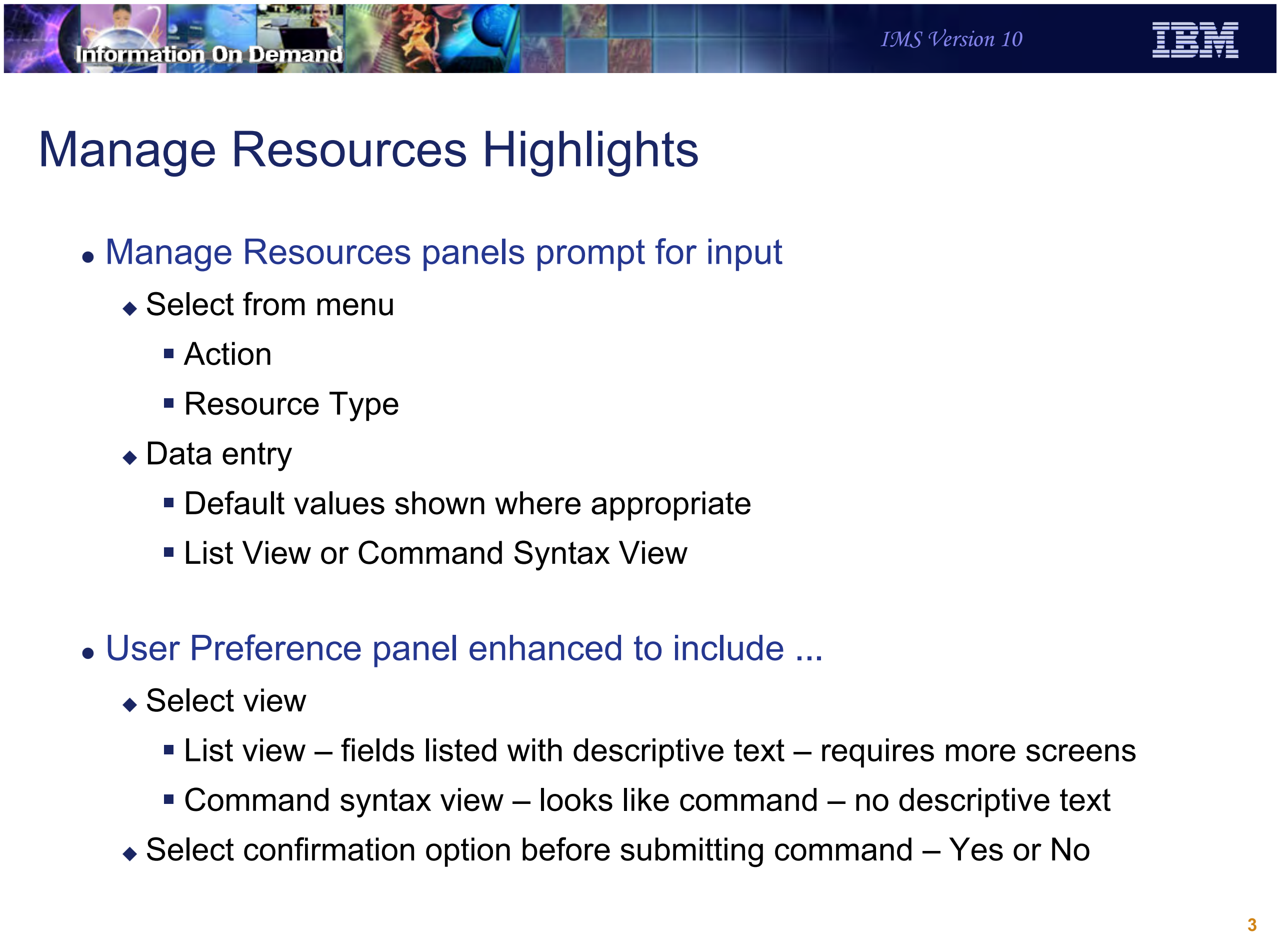
Figure 5-1. Manage Resources Highlights CMA01.0

Notes:

Manage Resources is invoked from the Application Menu. This is the menu that includes several other programs, including SPOC, KBLA, and the HALDB Partition Definition Utility.

Manage Resources presents a series of panels to simplify the task of creating, deleting, updating and querying the definitions used with DRD. The primary technique for creating a resource or descriptor is to present the user with a screen with a field for every attribute which the user can then fill in or let default. The defaults will be shown to the right of the field.

More skilled users can choose a different format that requires them to know the meaning and valid values for every field. This streamlines the process by presenting everything in a single screen.


Information On Demand
IMS Version 10

Manage Resources Highlights

- Manage Resources panels prompt for input
  - Select from menu
    - Action
    - Resource Type
  - Data entry
    - Default values shown where appropriate
    - List View or Command Syntax View

- User Preference panel enhanced to include ...
  - Select view
    - List view – fields listed with descriptive text – requires more screens
    - Command syntax view – looks like command – no descriptive text
  - Select confirmation option before submitting command – Yes or No

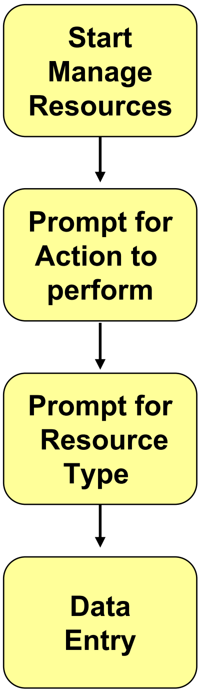
Figure 5-2. Manage Resources Highlights CMA01.0

Notes:

Manage Resources will always prompt the user for the next input. It will, for example, ask you what action you want to perform against what resource or descriptor. The SPOC Preference screen lets you select a default for the view. You may select either List View which is designed for the less skilled user and the Command System View which is intended for the more skilled user. You can tell the program that you do not want to confirm every entry that would update a resource or descriptor.

Information On Demand
IMS Version 10
IBM

Hierarchical Panel Structure



Start Manage Resources — From IMS Application Menu

Prompt for Action to perform — Create, Delete, Query, Update

Prompt for Resource Type — Database, Program, Transaction, Routing Code

Data Entry — Varies Depending On Action

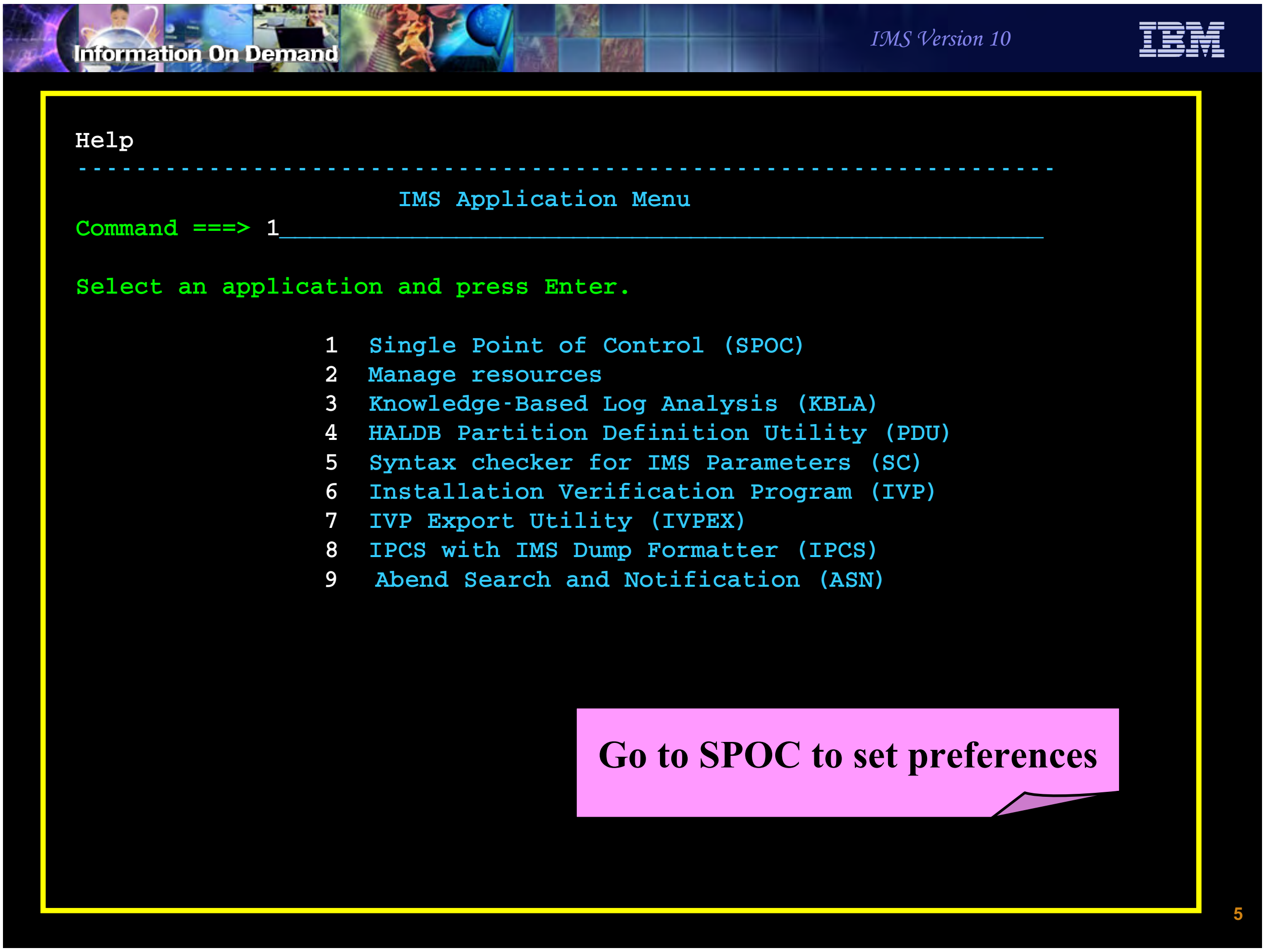
General Flow through Manage Resources

Figure 5-3. Hierarchical Panel Structure CMA01.0

Notes:

The basic flow is:

- Start Manage Resources from the IMS Application Menu
- Select the action you want to perform (e.g., create a definition)
- Select the resource type (e.g., create a DATABASE)
- Enter the data (e.g., enter the database name and attributes)



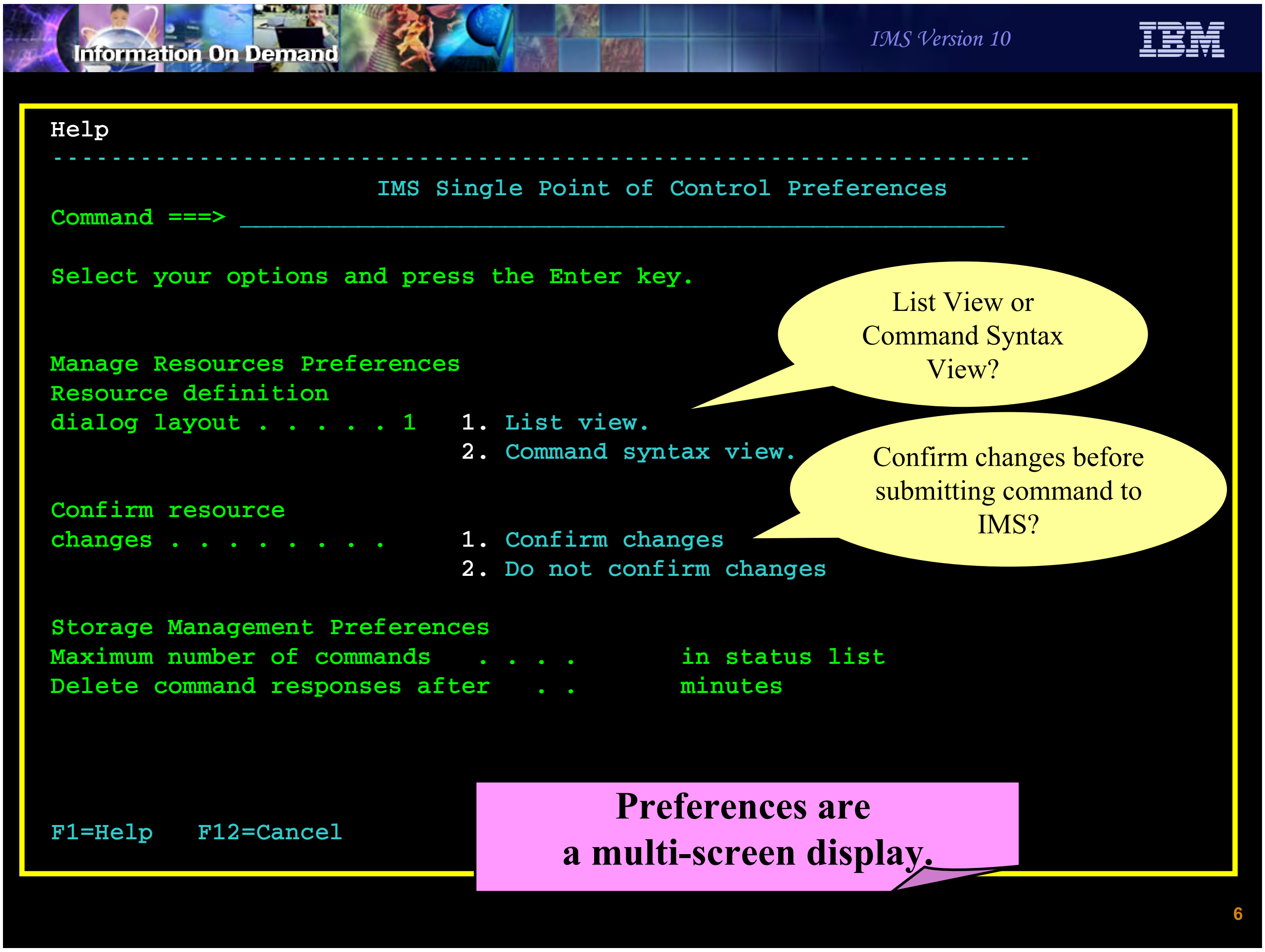
```
Help
-----------------------------------------------------------------
                     IMS Application Menu
Command ===> 1__________________________________________________

Select an application and press Enter.

                 1  Single Point of Control (SPOC)
                 2  Manage resources
                 3  Knowledge-Based Log Analysis (KBLA)
                 4  HALDB Partition Definition Utility (PDU)
                 5  Syntax checker for IMS Parameters (SC)
                 6  Installation Verification Program (IVP)
                 7  IVP Export Utility (IVPEX)
                 8  IPCS with IMS Dump Formatter (IPCS)
                 9   Abend Search and Notification (ASN)
```

Go to SPOC to set preferences

Figure 5-4. [No title] CMA01.0

Notes:

Start the IMS Application Menu and select Option 1 – SPOC. You should go to SPOC first to set your preferences for Manage Resources.



```
Help
-------------------------------------------------------------------
                      IMS Single Point of Control Preferences
Command ===> ____________________________________________________

Select your options and press the Enter key.


Manage Resources Preferences
Resource definition
dialog layout . . . . . 1    1. List view.
                             2. Command syntax view.

Confirm resource
changes . . . . . . . .      1. Confirm changes
                             2. Do not confirm changes

Storage Management Preferences
Maximum number of commands    . . . .       in status list
Delete command responses after    . .       minutes




F1=Help   F12=Cancel
```

List View or Command Syntax View?

Confirm changes before submitting command to IMS?

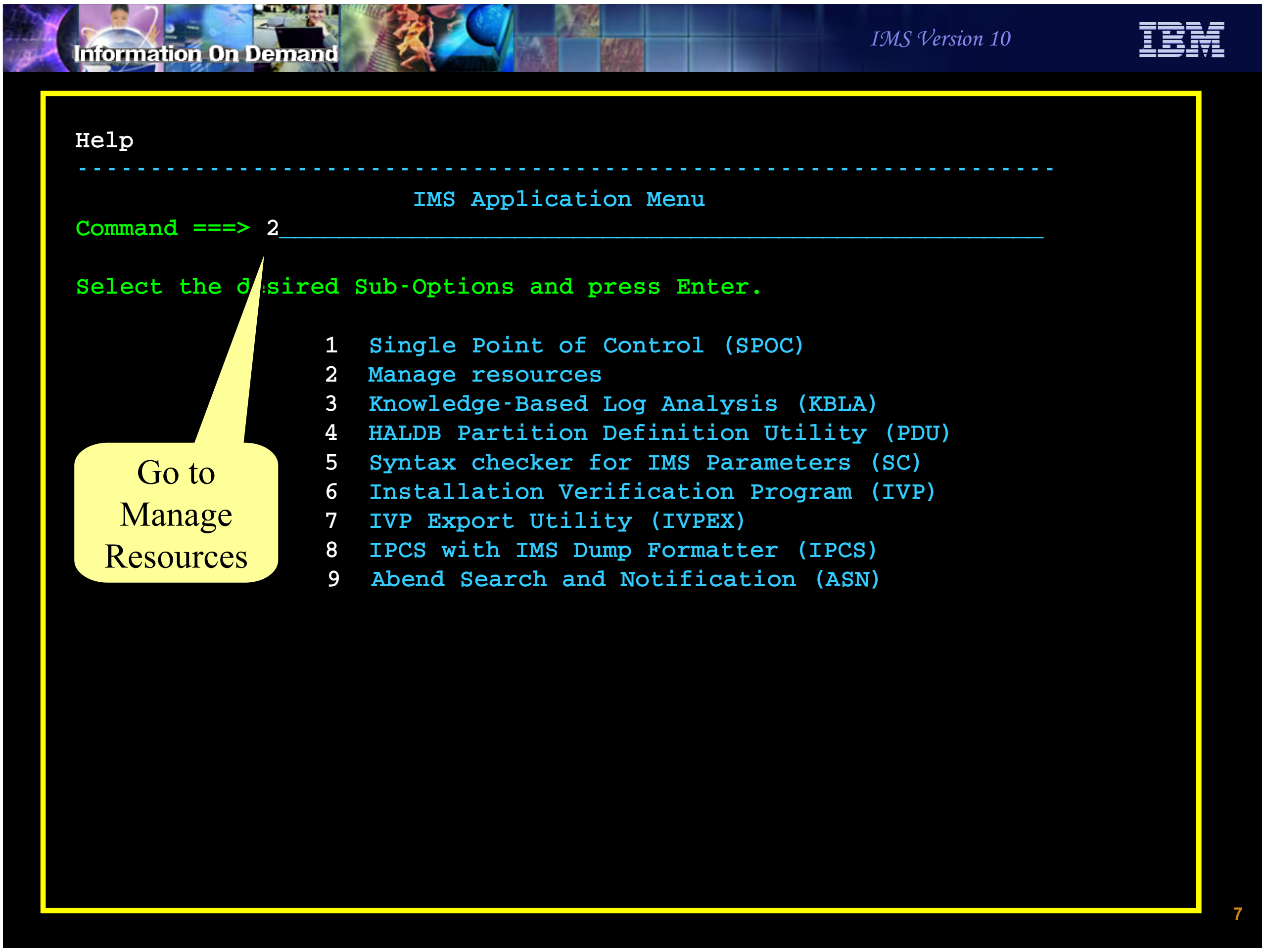
Preferences are a multi-screen display.

Figure 5-5. [No title] CMA01.0

Notes:

There are two Manage Resources preferences you can set:

- Do you want Manage Resources to present screens to you in List for or Command Syntax form? We will see examples of both shortly. The default is List View.
- Do you want Manage Resources to ask you to confirm everything before sending a command to IMS that will update a resource? The default is to Confirm.



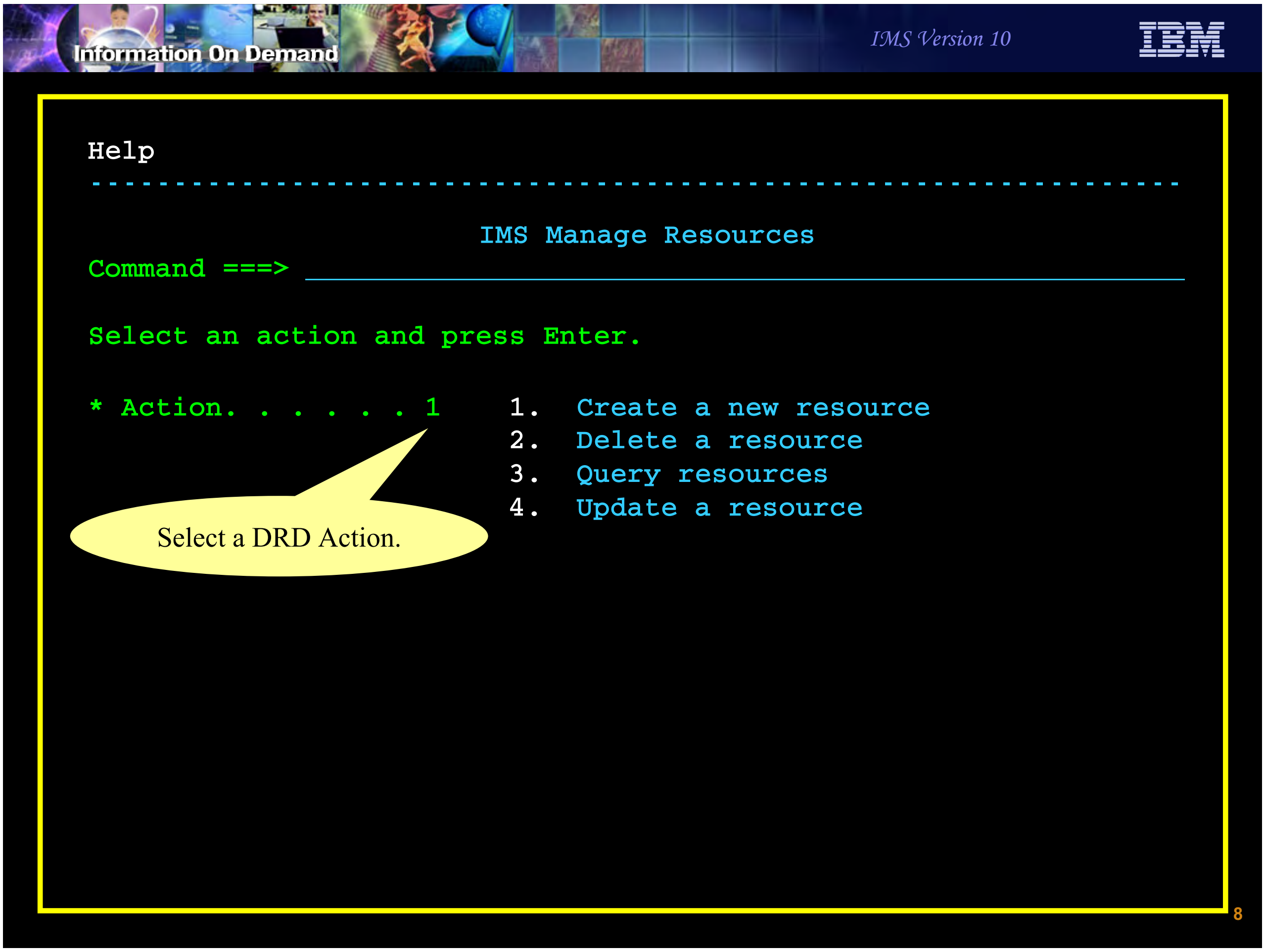
```
Help
------------------------------------------------------------------
                       IMS Application Menu
Command ===> 2____________________________________________________

Select the desired Sub-Options and press Enter.

                1  Single Point of Control (SPOC)
                2  Manage resources
                3  Knowledge-Based Log Analysis (KBLA)
                4  HALDB Partition Definition Utility (PDU)
                5  Syntax checker for IMS Parameters (SC)
                6  Installation Verification Program (IVP)
                7  IVP Export Utility (IVPEX)
                8  IPCS with IMS Dump Formatter (IPCS)
                9  Abend Search and Notification (ASN)
```

Go to Manage Resources

Figure 5-6. [No title] CMA01.0

Notes:

Go back to Application Menu and select Manage Resources



```
Help
------------------------------------------------------------------

                       IMS Manage Resources
Command ===> _____________________________________________________

Select an action and press Enter.

* Action. . . . . . 1     1.  Create a new resource
                          2.  Delete a resource
                          3.  Query resources
                          4.  Update a resource
```

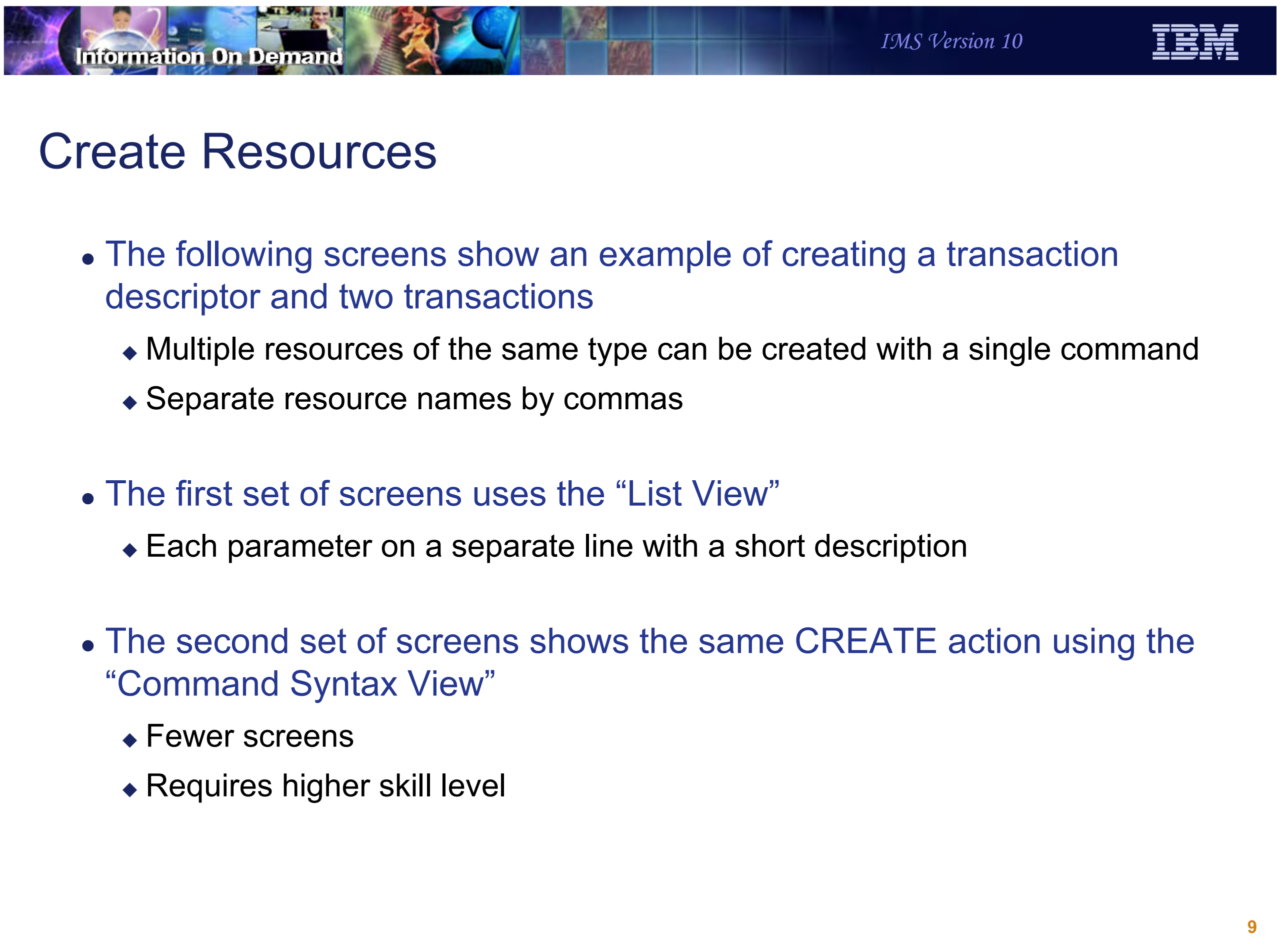
Select a DRD Action.

Figure 5-7. [No title] CMA01.0

Notes:

Tell Manage Resources what action you want to perform.

Here we say we want to Create a new resource. On this screen, “resource” means either resource or descriptor.



Information On Demand

IMS Version 10

Create Resources

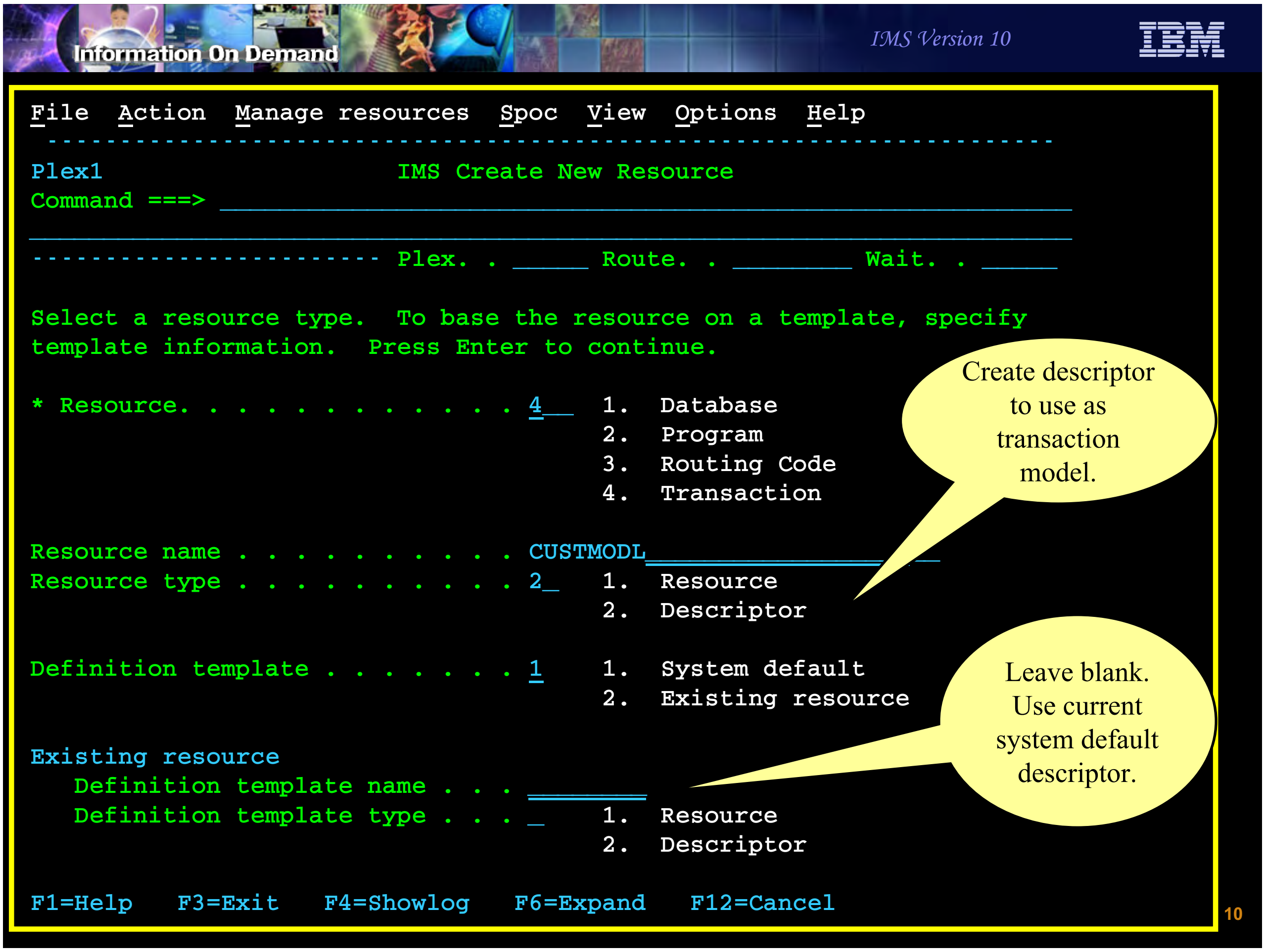
- The following screens show an example of creating a transaction descriptor and two transactions
  - Multiple resources of the same type can be created with a single command
  - Separate resource names by commas
- The first set of screens uses the “List View”
  - Each parameter on a separate line with a short description
- The second set of screens shows the same CREATE action using the “Command Syntax View”
  - Fewer screens
  - Requires higher skill level

9

Figure 5-8. Create Resources CMA01.0

Notes:

The next few screens are an example of creating a transaction descriptor then using that descriptor to create two transactions. Both the List View and Command Syntax View are shown.



```
File  Action  Manage resources  Spoc  View  Options  Help
 ---------------------------------------------------------------------
Plex1                     IMS Create New Resource
Command ===> ________________________________________________________
_________________________________________________________________________
------------------------ Plex. . _____ Route. . ________ Wait. . _____

Select a resource type.  To base the resource on a template, specify
template information.  Press Enter to continue.

* Resource. . . . . . . . . . . . . 4__   1.  Database
                                          2.  Program
                                          3.  Routing Code
                                          4.  Transaction

Resource name . . . . . . . . . . . CUSTMODL____________________
Resource type . . . . . . . . . . . 2_    1.  Resource
                                          2.  Descriptor

Definition template . . . . . . . . 1     1.  System default
                                          2.  Existing resource

Existing resource
   Definition template name . . .  ________
   Definition template type . . .  _      1.  Resource
                                          2.  Descriptor

F1=Help   F3=Exit   F4=Showlog   F6=Expand   F12=Cancel
```

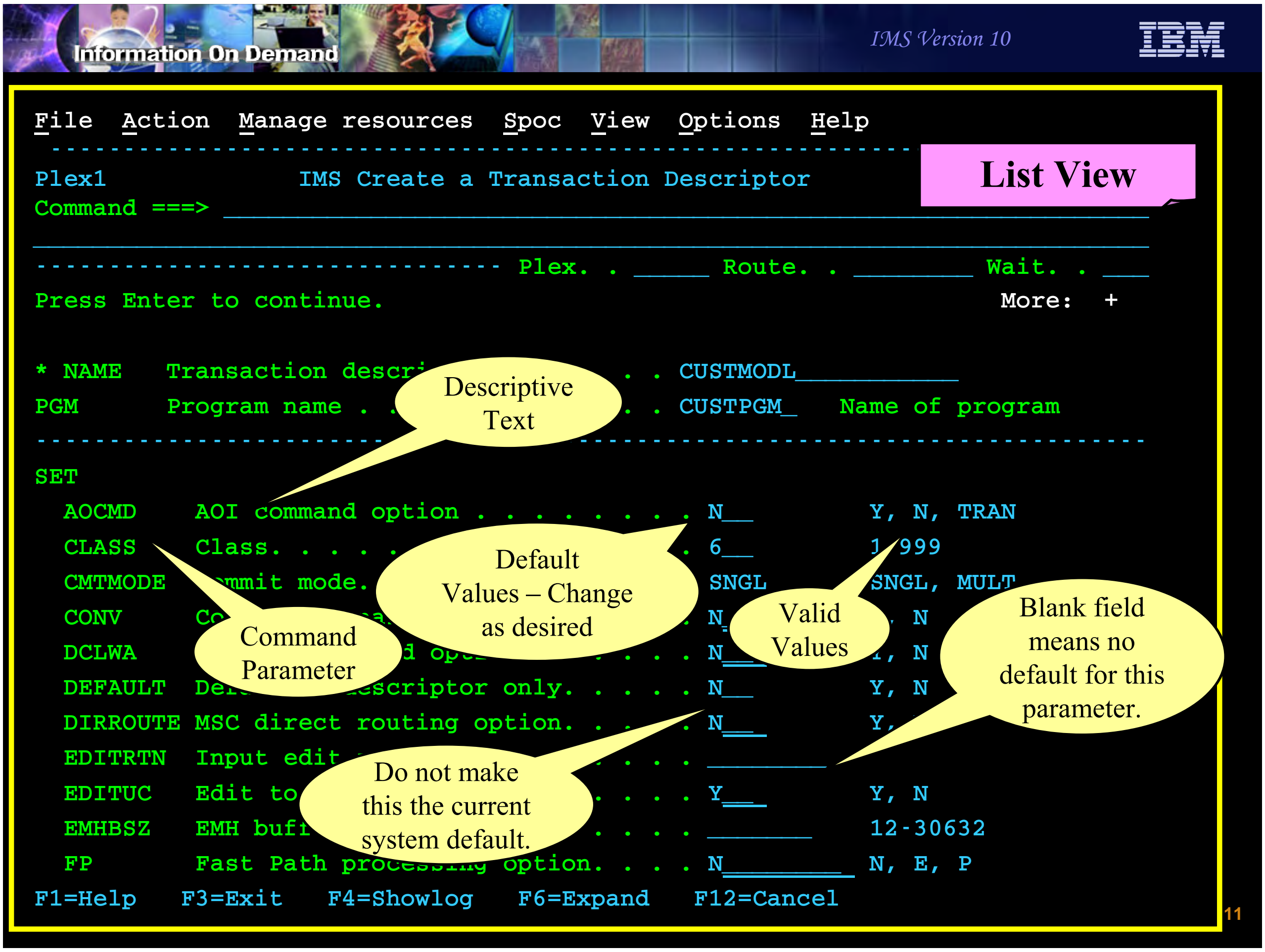
Create descriptor to use as transaction model.

Leave blank. Use current system default descriptor.

Figure 5-9. [No title] CMA01.0

Notes:

Here we say we want to create a transaction descriptor using the current default transaction descriptor for attributes which we don't supply.


Information On Demand
IMS Version 10
IBM
List View
Descriptive Text
Default Values – Change as desired
Command Parameter
Valid Values
Blank field means no default for this parameter.
Do not make this the current system default.
11

```
File  Action  Manage resources  Spoc  View  Options  Help
 ------------------------------------------------------------
Plex1              IMS Create a Transaction Descriptor
Command ===> _______________________________________________________
____________________________________________________________________
-------------------------------- Plex. . _____ Route. . ________ Wait. . ___
Press Enter to continue.                                       More:   +

* NAME   Transaction descri               . . CUSTMODL___________
PGM      Program name . .                 . . CUSTPGM_   Name of program
-----------------------                      --------------------------------
SET
  AOCMD    AOI command option . . . . . . . . N__        Y, N, TRAN
  CLASS    Class. . . . .                   . 6__        1 999
  CMTMODE       mmit mode.                    SNGL       SNGL, MULT
  CONV     Co                                 N_           N
  DCLWA                    d ope          . . N___         N
  DEFAULT  De              escriptor only. . . . N__        Y, N
  DIRROUTE MSC direct routing option. .  . . N___       Y,
  EDITRTN  Input edit                    . . . ________
  EDITUC   Edit  to                     . . . . Y___       Y, N
  EMHBSZ   EMH buff                     . . . . ________   12-30632
  FP       Fast Path proce        option. . . . N_________ N, E, P
F1=Help   F3=Exit   F4=Showlog   F6=Expand   F12=Cancel
```

Figure 5-10. [No title] CMA01.0

Notes:

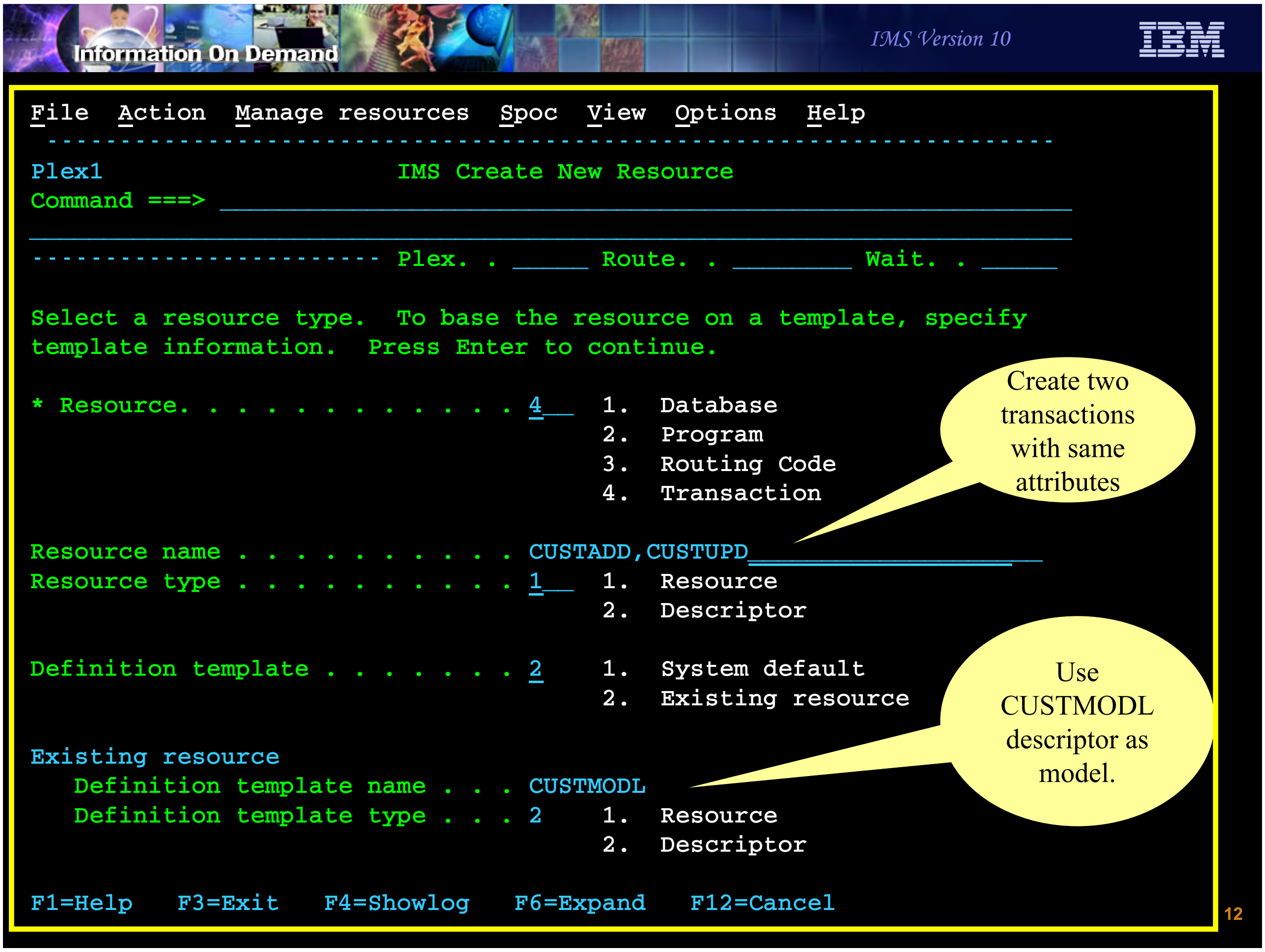
Manage Resources sends us screens (List View – that was our preference) with all the possible attributes for transaction descriptors. There will be multiple screens since there are more attributes than will fit on one screen in List View.

At the top of the screen, enter the name or names of the descriptors you want to create and the name of the program.

Since we opted to use the system default transaction descriptor, the third column will have these defaults already filled in. If the field is blank, there is no default. You can change those you want to change and leave the rest as they are. The far right column shows the valid values for the field.

Press F8 to scroll forward and F7 to scroll backward through all the screens.

After entering all the data, pressing Enter will either send a CREATE TRANDESC command to IMS or will ask you to confirm.



```
File  Action  Manage resources  Spoc  View  Options  Help
 ---------------------------------------------------------------------
Plex1                        IMS Create New Resource
Command ===> __________________________________________________________
 ______________________________________________________________________
------------------------ Plex. . _____ Route. . ________ Wait. . _____

Select a resource type.  To base the resource on a template, specify
template information.  Press Enter to continue.

* Resource. . . . . . . . . . . . . 4__   1.  Database
                                          2.  Program
                                          3.  Routing Code
                                          4.  Transaction

Resource name . . . . . . . . . . . CUSTADD,CUSTUPD____________________
Resource type . . . . . . . . . . . 1__   1.  Resource
                                          2.  Descriptor

Definition template . . . . . . . . 2     1.  System default
                                          2.  Existing resource

Existing resource
   Definition template name . . . CUSTMODL
   Definition template type . . . 2       1.  Resource
                                          2.  Descriptor

F1=Help   F3=Exit   F4=Showlog   F6=Expand   F12=Cancel
```

Create two transactions with same attributes

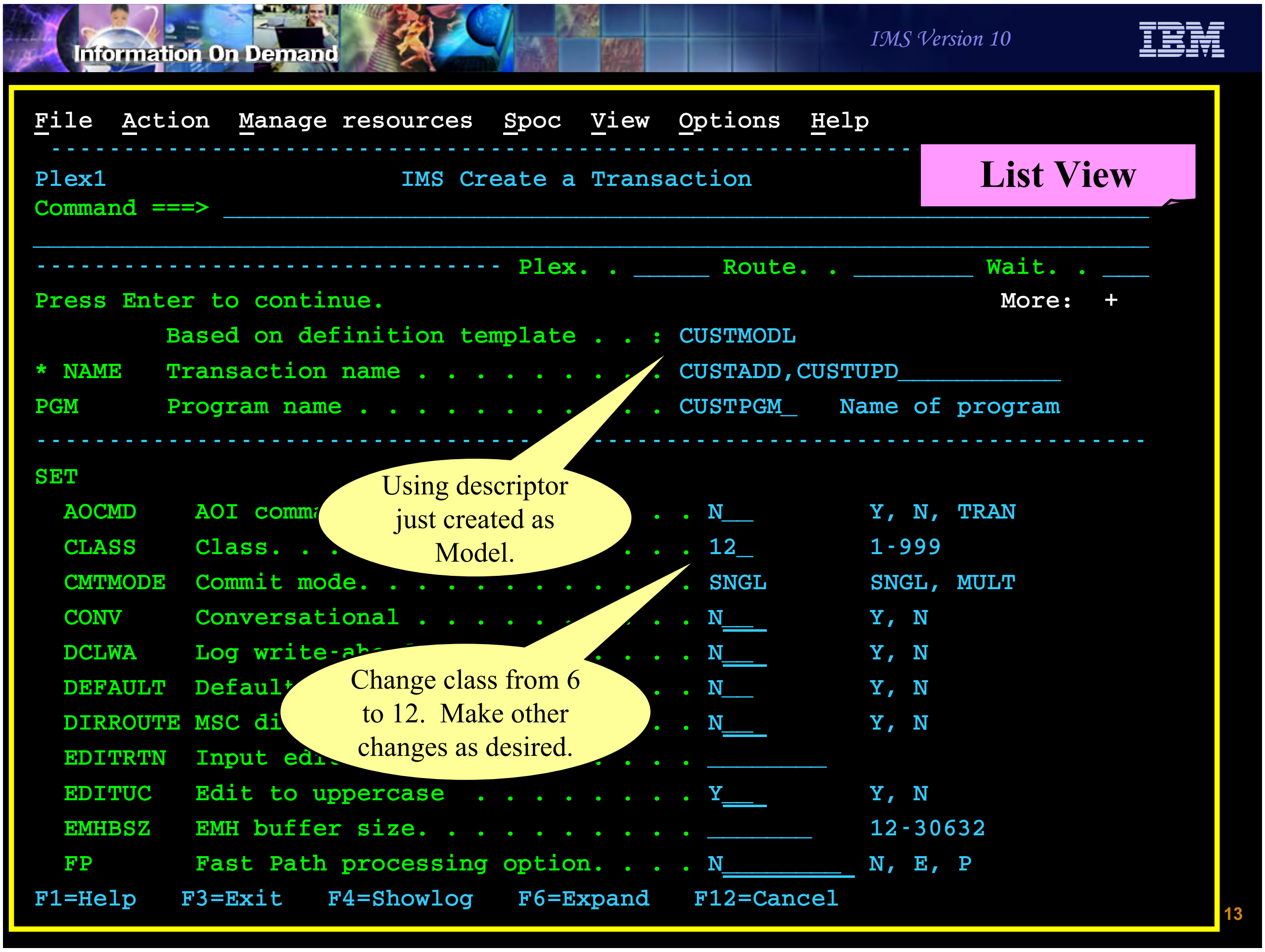
Use CUSTMODL descriptor as model.

Figure 5-11. [No title] CMA01.0

Notes:

After the transaction descriptor has been created you can create transactions using this descriptor as a template.

On this screen, we are creating two transactions, CUSTADD and CUSTUPD, using the descriptor we just created, CUSTMODL, as a template.



List View

```
File  Action  Manage resources  Spoc  View  Options  Help
 ------------------------------------------------------------
Plex1                        IMS Create a Transaction
Command ===> _____________________________________________________

______________________________________________________________________
-------------------------------- Plex. . _____ Route. . ________ Wait. . ___
Press Enter to continue.                                        More:   +
         Based on definition template . . : CUSTMODL
* NAME   Transaction name . . . . . . . . . CUSTADD,CUSTUPD___________
PGM      Program name . . . . . . . . . . . CUSTPGM_   Name of program
-----------------------------------  ------------------------------------
SET
  AOCMD    AOI comma                    . . N__         Y, N, TRAN
  CLASS    Class. . .                 . . . 12_         1-999
  CMTMODE  Commit mode. . . . . . . . . . . SNGL        SNGL, MULT
  CONV     Conversational . . . . .   . . . N___        Y, N
  DCLWA    Log write-ah                 . . N___        Y, N
  DEFAULT  Default                      . . N__         Y, N
  DIRROUTE MSC di                       . . N___        Y, N
  EDITRTN  Input edi               . . . .  ________
  EDITUC   Edit to uppercase  . . . . . . . Y___        Y, N
  EMHBSZ   EMH buffer size. . . . . . . . . _______     12-30632
  FP       Fast Path processing option. . . N_________  N, E, P
F1=Help   F3=Exit   F4=Showlog   F6=Expand   F12=Cancel
```

Using descriptor just created as Model.

Change class from 6 to 12. Make other changes as desired.

Figure 5-12. [No title] CMA01.0

Notes:

Change any attributes that should be different from the descriptor. In this example, we are changing the class from 6 to 12.

Create Resource

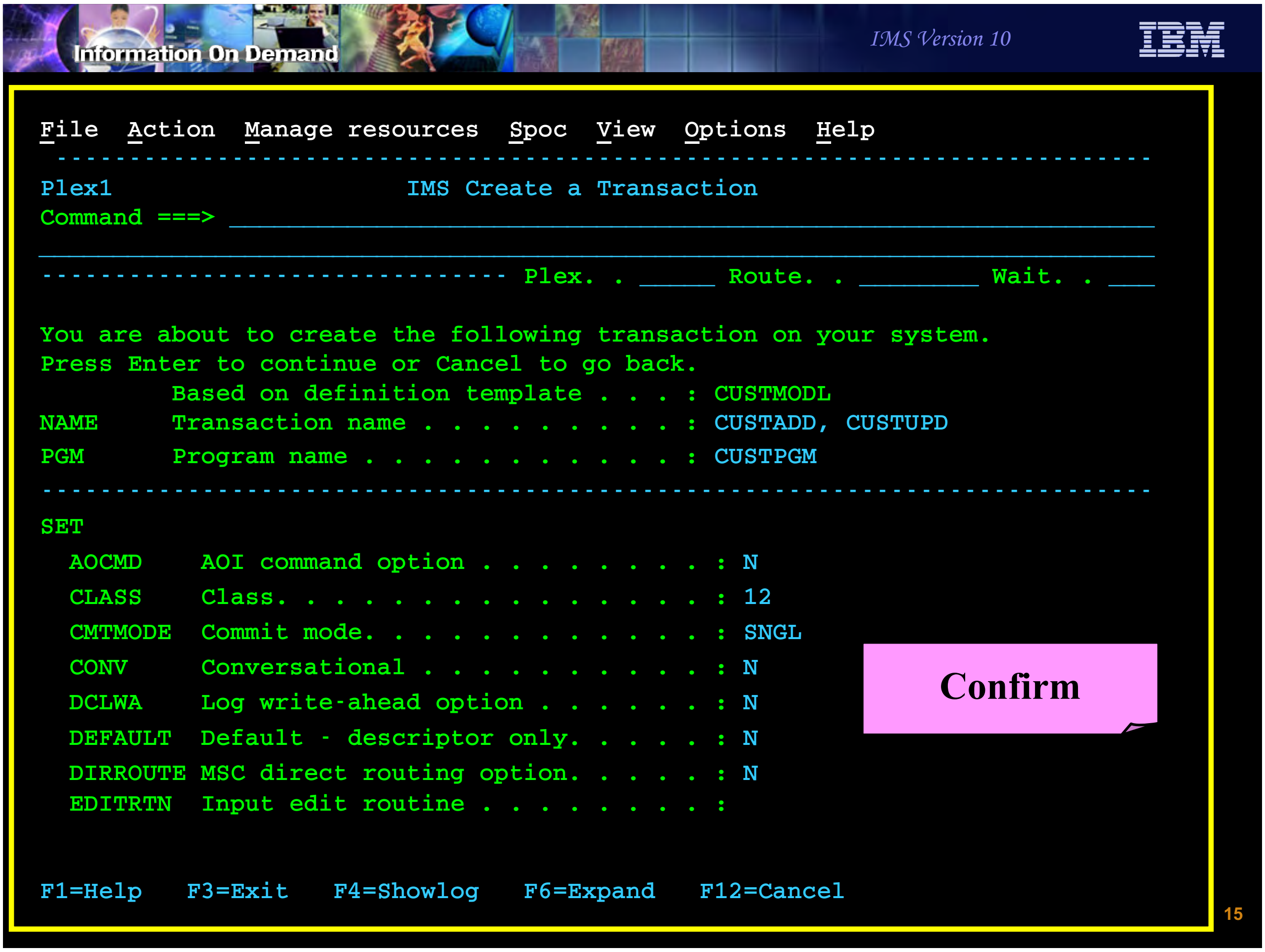
- Scroll through the screens using ISPF function key assignments
  - PF8 – scroll forward
  - PF7 – scroll back
- Make changes to defaults on the screen and press Enter to submit
- A confirmation screen will be presented if elected in Preferences
- Press enter again and Manage Resources will submit command to IMS(s)
  - Manage Resources will edit input for valid field values
    - Does not cross check multiple fields
  - IMS edits input and returns success or failure

Figure 5-13. Create Resource CMA01.0

Notes:

If your preference is to confirm all updates, a confirmation screen will be displayed. Check the definitions and press Enter again. Manage Resources will submit a CREATE TRAN command for you.

Note that Manage Resources will edit fields for valid values but will not cross-check to be sure definitions are consistent. For example, Manage Resources will not verify that a RTC is assigned to an FPE program, or that a FP transaction is not conversational. This is done only by IMS.



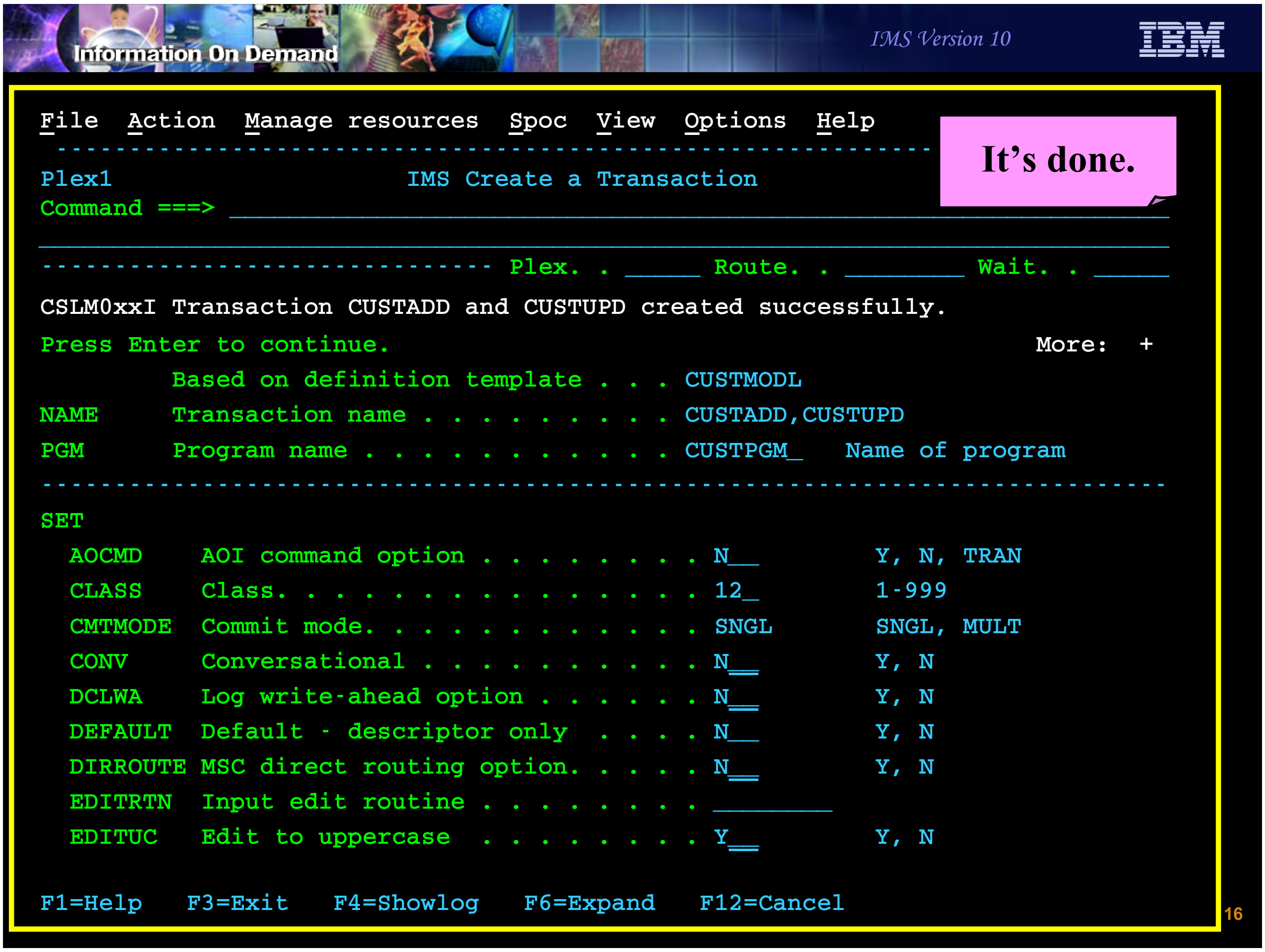
```
File  Action  Manage resources  Spoc  View  Options  Help
 --------------------------------------------------------------------------
Plex1                    IMS Create a Transaction
Command ===> ______________________________________________________________
___________________________________________________________________________
-------------------------------- Plex. . _____ Route. . ________ Wait. . ___

You are about to create the following transaction on your system.
Press Enter to continue or Cancel to go back.
         Based on definition template . . . : CUSTMODL
NAME     Transaction name . . . . . . . . . : CUSTADD, CUSTUPD
PGM      Program name . . . . . . . . . . . : CUSTPGM
---------------------------------------------------------------------------
SET
  AOCMD    AOI command option . . . . . . . . : N
  CLASS    Class. . . . . . . . . . . . . . . : 12
  CMTMODE  Commit mode. . . . . . . . . . . . : SNGL
  CONV     Conversational . . . . . . . . . . : N
  DCLWA    Log write-ahead option . . . . . . : N
  DEFAULT  Default - descriptor only. . . . . : N
  DIRROUTE MSC direct routing option. . . . . : N
  EDITRTN  Input edit routine . . . . . . . . :


F1=Help   F3=Exit   F4=Showlog   F6=Expand   F12=Cancel
```

Confirm

Figure 5-14. [No title] CMA01.0

Notes:

If you asked to confirm, you will be presented with a confirmation screen. Check it out and press enter again.



```
File  Action  Manage resources  Spoc  View  Options  Help
-------------------------------------------------------------
Plex1                     IMS Create a Transaction
Command ===> _____________________________________________________________

_____________________________________________________________________________
------------------------------- Plex. . _____ Route. . ________ Wait. . _____
CSLM0xxI Transaction CUSTADD and CUSTUPD created successfully.
Press Enter to continue.                                          More:   +
         Based on definition template . . . CUSTMODL
NAME     Transaction name . . . . . . . . . CUSTADD,CUSTUPD
PGM      Program name . . . . . . . . . . . CUSTPGM_   Name of program
-----------------------------------------------------------------------------
SET
  AOCMD    AOI command option . . . . . . . . N__         Y, N, TRAN
  CLASS    Class. . . . . . . . . . . . . . . 12_         1-999
  CMTMODE  Commit mode. . . . . . . . . . . . SNGL        SNGL, MULT
  CONV     Conversational . . . . . . . . . . N__         Y, N
  DCLWA    Log write-ahead option . . . . . . N__         Y, N
  DEFAULT  Default - descriptor only  . . . . N__         Y, N
  DIRROUTE MSC direct routing option. . . . . N__         Y, N
  EDITRTN  Input edit routine . . . . . . . . ________
  EDITUC   Edit to uppercase  . . . . . . . . Y__         Y, N

F1=Help   F3=Exit   F4=Showlog   F6=Expand   F12=Cancel
```

It’s done.

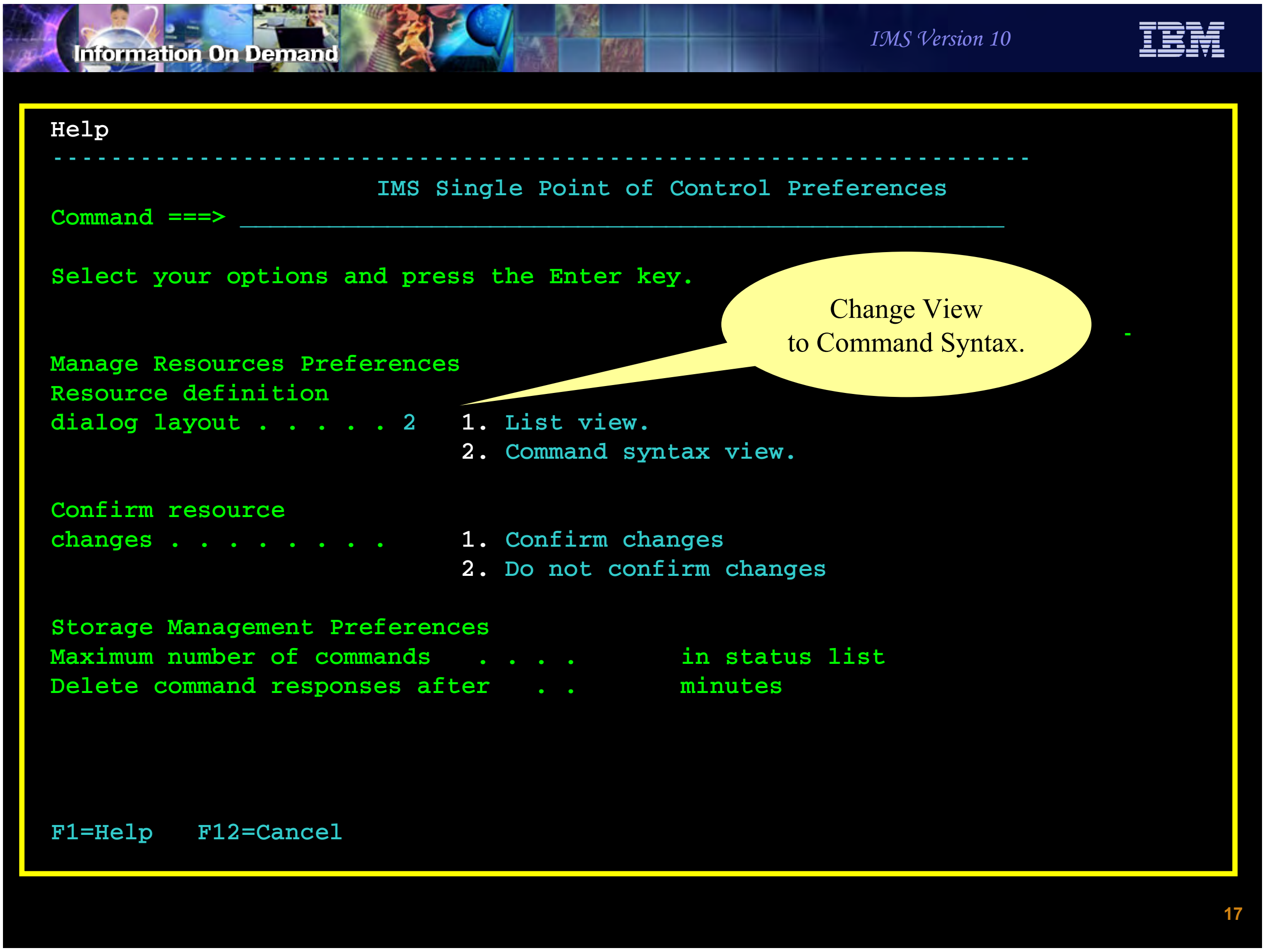
Figure 5-15. [No title]

CMA01.0

Notes:

If everything went well, you will get a message indicating success. If the command failed, you will get an error message.

With this command we are creating two transactions with exactly the same attributes.



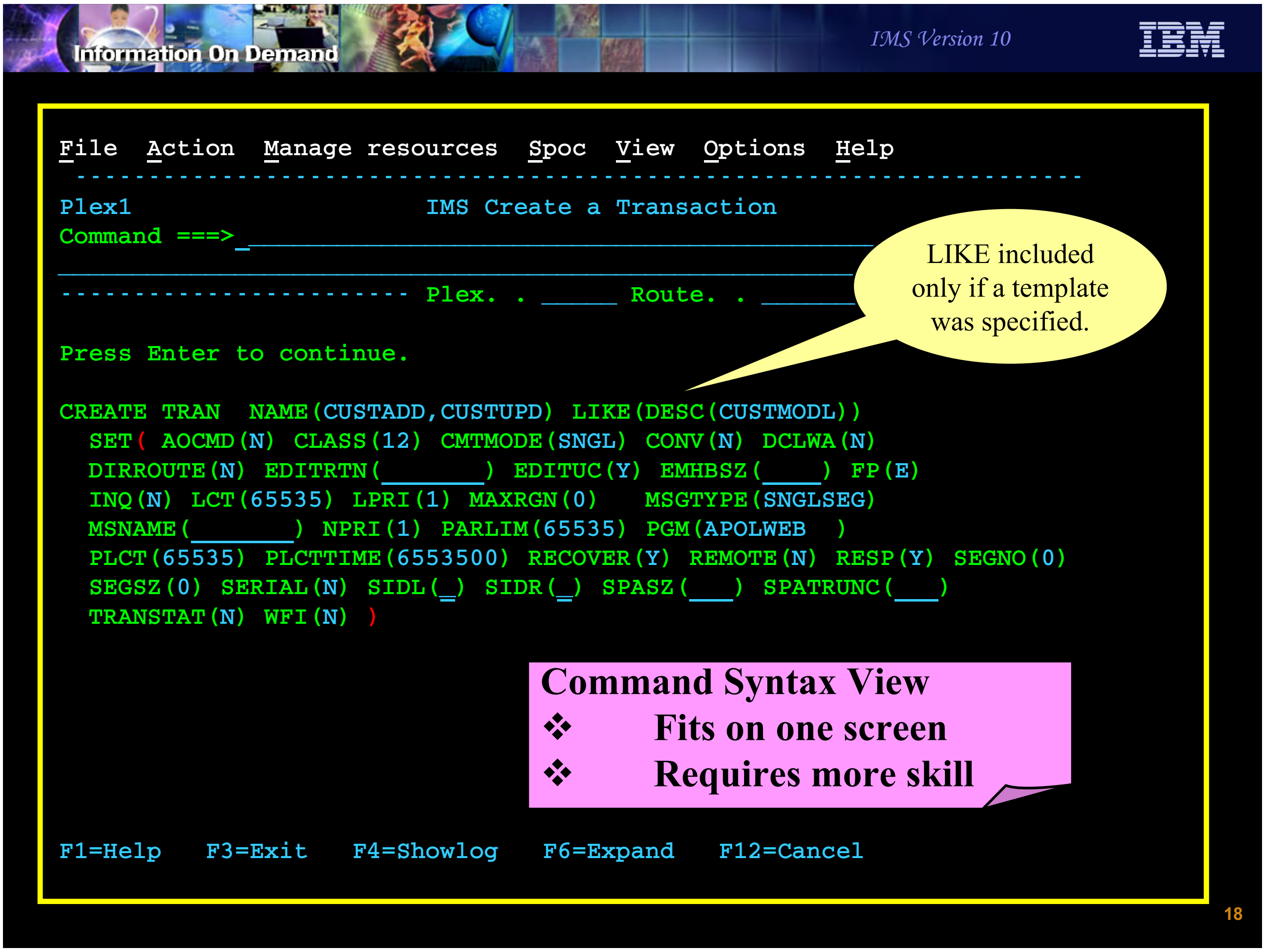
```
Help
-------------------------------------------------------------------
                      IMS Single Point of Control Preferences
Command ===> ____________________________________________________

Select your options and press the Enter key.


Manage Resources Preferences
Resource definition
dialog layout . . . . . 2   1. List view.
                            2. Command syntax view.

Confirm resource
changes . . . . . . . .     1. Confirm changes
                            2. Do not confirm changes

Storage Management Preferences
Maximum number of commands    . . . .       in status list
Delete command responses after    . .       minutes



F1=Help   F12=Cancel
```

Change View
to Command Syntax.

Figure 5-16. [No title] CMA01.0

Notes:

Alternatively, you can use Command Syntax View. You may return to the SPOC screen which was shown earlier to change this selection.



```
File  Action  Manage resources  Spoc  View  Options  Help
 ----------------------------------------------------------------------
Plex1                        IMS Create a Transaction
Command ===>_____________________________________________

_______________________________________________________
------------------------ Plex. . _____ Route. . _______

Press Enter to continue.

CREATE TRAN  NAME(CUSTADD,CUSTUPD) LIKE(DESC(CUSTMODL))
  SET( AOCMD(N) CLASS(12) CMTMODE(SNGL) CONV(N) DCLWA(N)
  DIRROUTE(N) EDITRTN(_______) EDITUC(Y) EMHBSZ(____) FP(E)
  INQ(N) LCT(65535) LPRI(1) MAXRGN(0)   MSGTYPE(SNGLSEG)
  MSNAME(_______) NPRI(1) PARLIM(65535) PGM(APOLWEB   )
  PLCT(65535) PLCTTIME(6553500) RECOVER(Y) REMOTE(N) RESP(Y) SEGNO(0)
  SEGSZ(0) SERIAL(N) SIDL(_) SIDR(_) SPASZ(___) SPATRUNC(___)
  TRANSTAT(N) WFI(N) )


F1=Help   F3=Exit   F4=Showlog   F6=Expand   F12=Cancel
```

LIKE included only if a template was specified.

Command Syntax View

- ❖ Fits on one screen
- ❖ Requires more skill

Figure 5-17. [No title] CMA01.0

Notes:

This is the same input as the previous CREATE TRAN but this shows the exact syntax of the command without all the help (field descriptions and valid values).

Information On Demand
IMS Version 10

Querying Resources

- Support for the QUERY command has been enhanced
  - Short cut to displaying ALL attributes of a resource
    - QRY xxxx SHOW(ALL)
  - Short cut to displaying resources with “exceptional” status
    - QRY xxxx <user-default filters>
  - Build customized queries
    - QRY xxxx <customized filters>
  - Field level help on all displayed fields
    - Put cursor on column heading and press F1

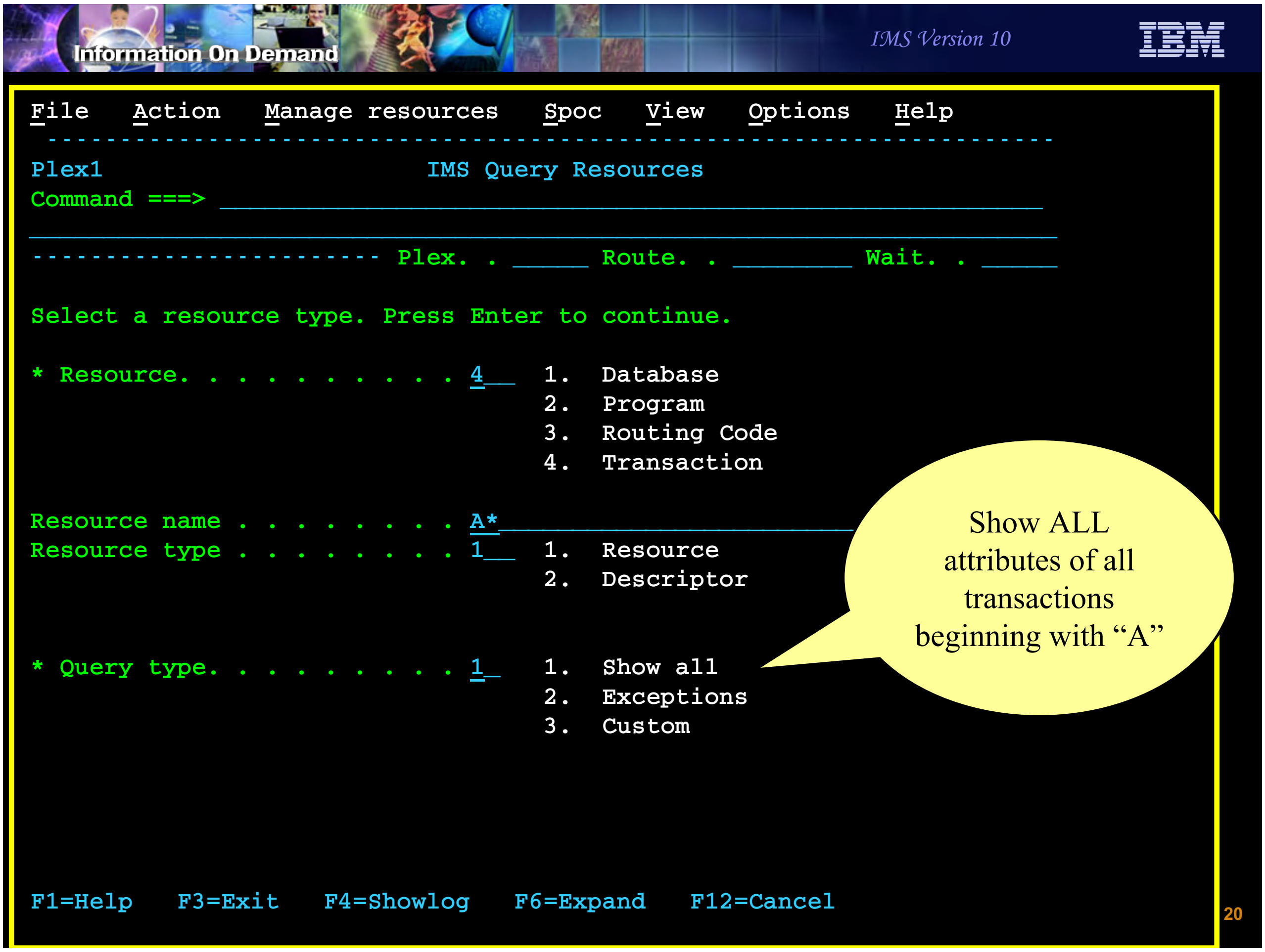
Figure 5-18. Querying Resources CMA01.0

Notes:

Manage Resources also supports a QRY capability. This capability has a shortcut for displaying all attributes of a resource. You can choose to display only resources with exceptional status. You decide what is exceptional. You can also build custom queries which show only a subset of the attributes.

The QRY results screen has a column where you can enter line commands.

Help is available for individual fields on each screen.



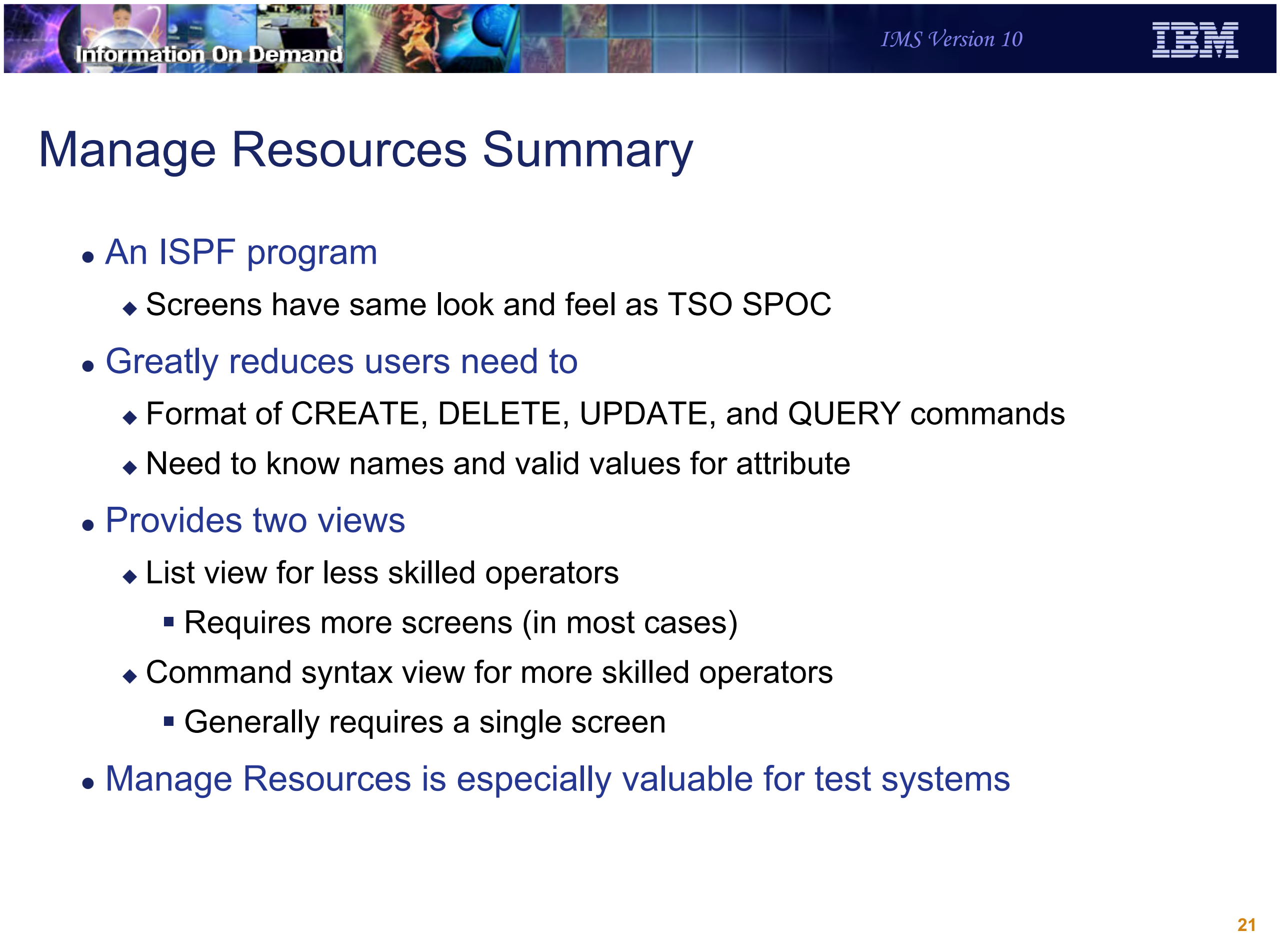
```
File   Action   Manage resources   Spoc   View   Options   Help
 ----------------------------------------------------------------------
Plex1                           IMS Query Resources
Command ===> ________________________________________________________
______________________________________________________________________
------------------------ Plex. . _____ Route. . ________ Wait. . _____

Select a resource type. Press Enter to continue.

* Resource. . . . . . . . . . 4__   1.  Database
                                    2.  Program
                                    3.  Routing Code
                                    4.  Transaction

Resource name . . . . . . . . A*________________________
Resource type . . . . . . . . 1__   1.  Resource
                                    2.  Descriptor


* Query type. . . . . . . . . 1_    1.  Show all
                                    2.  Exceptions
                                    3.  Custom




F1=Help   F3=Exit   F4=Showlog   F6=Expand   F12=Cancel
```

Show ALL
attributes of all
transactions
beginning with “A”

Figure 5-19. [No title] CMA01.0

Notes:

From this screen you can enter the resource or descriptor type and the its name or names, select either resource or descriptor, and choose the query type. In this case, we have chosen to query all transaction resources beginning with A and to show all of the attributes.



Information On Demand
IMS Version 10
IBM

Manage Resources Summary

- An ISPF program
  - Screens have same look and feel as TSO SPOC
- Greatly reduces users need to
  - Format of CREATE, DELETE, UPDATE, and QUERY commands
  - Need to know names and valid values for attribute
- Provides two views
  - List view for less skilled operators
    - Requires more screens (in most cases)
  - Command syntax view for more skilled operators
    - Generally requires a single screen
- Manage Resources is especially valuable for test systems

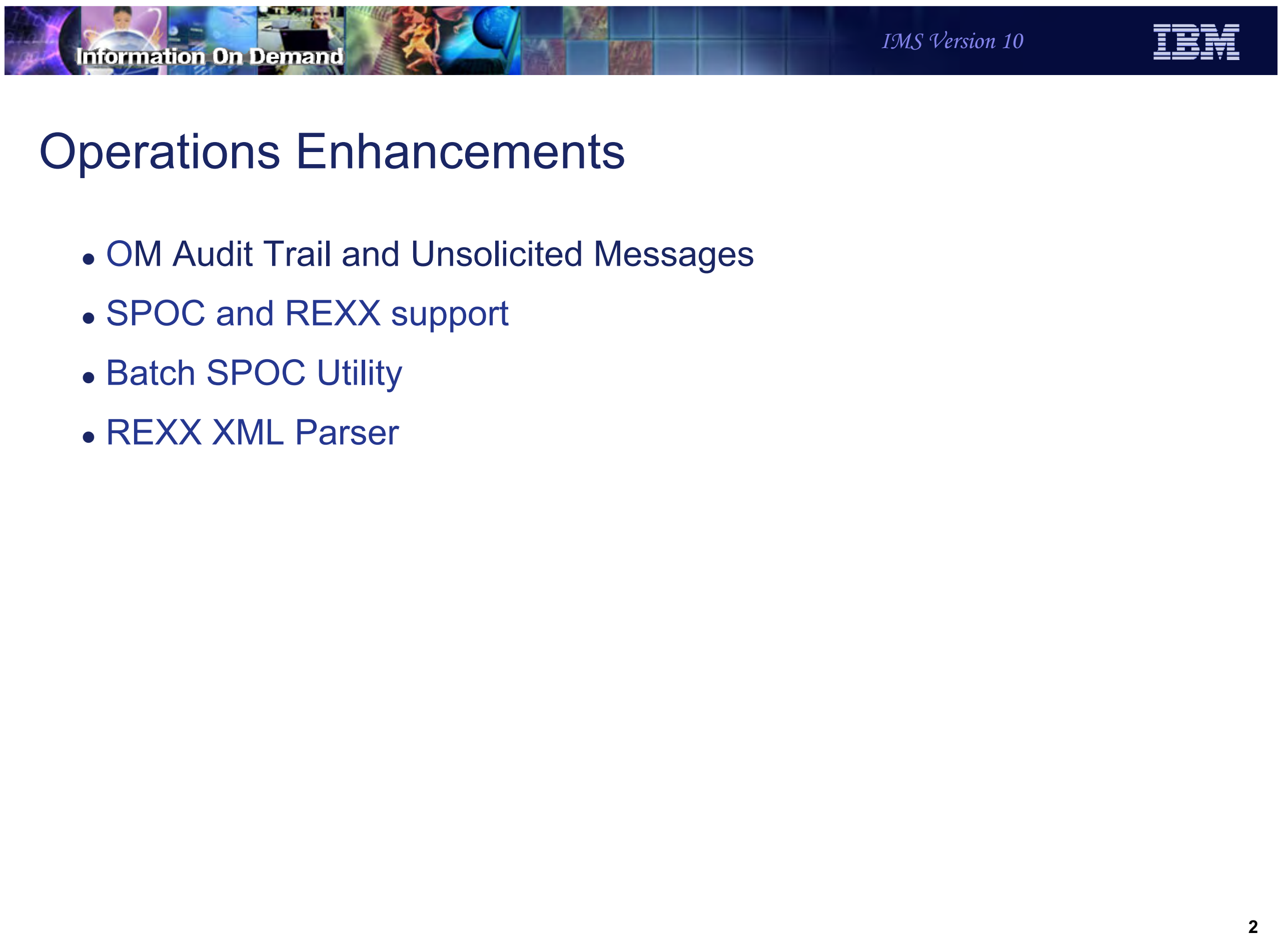
Figure 5-20. Manage Resources Summary CMA01.0

Notes:

Manage Resources greatly reduces the need for the user to know the names of all the attributes and their valid values. It also eliminates the need to know the exact format of the commands. For those that do know the names and values for the attributes, a Command Syntax View is available to reduce the number of screens required for some commands – especially those related to transactions which have lots of attributes.

Some installations will find Manage Resources to be especially valuable for test systems. With production systems they may prefer the use of standard commands for DRD. This may be more easily audited. For test systems Manage Resources provides a convenient and flexible interface for quickly changing resource definitions.

Unit 6. Operations

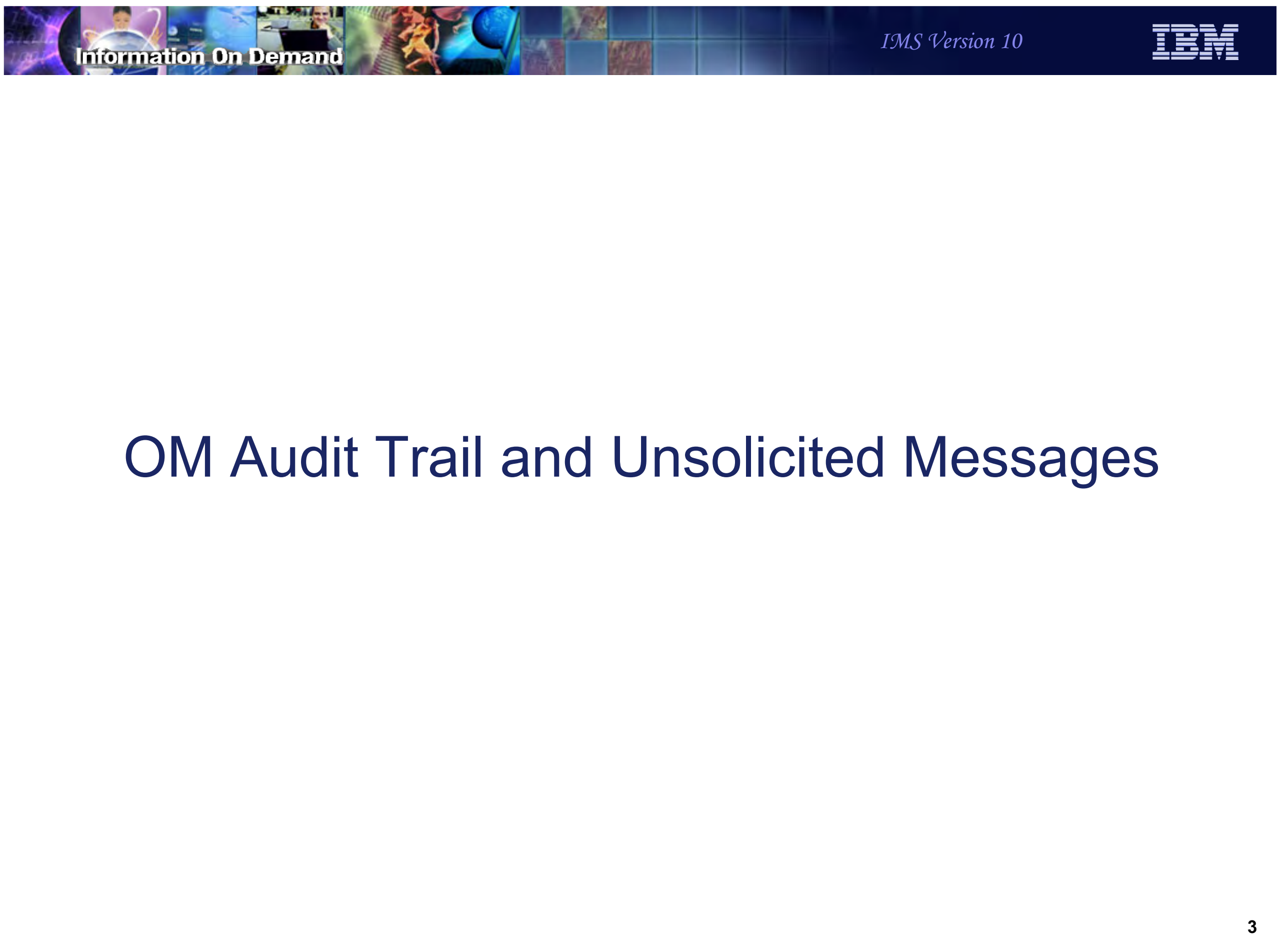

Information On Demand
IMS Version 10

Operations Enhancements

- OM Audit Trail and Unsolicited Messages
- SPOC and REXX support
- Batch SPOC Utility
- REXX XML Parser

Figure 6-1. Operations Enhancements CMA01.0

Notes:


Information On Demand
IMS Version 10
IBM

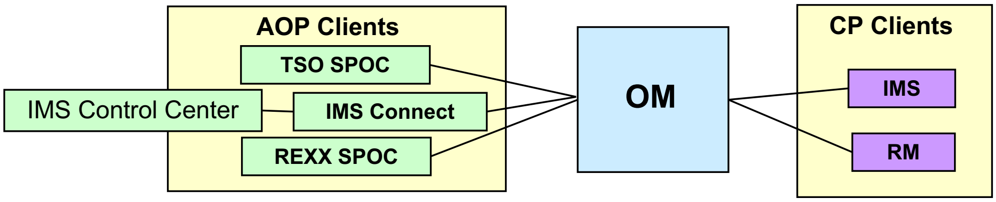
OM Audit Trail and Unsolicited Messages

Figure 6-2. [No title] CMA01.0

Notes:

OM Terminology Review

- A command processing (CP) client
  - An IMSplex member that accepts and processes commands entered by another member through the OM API
    - Examples are the IMS control region and RM
- An automated operator program (AOP) client
  - An IMSplex member that submits commands through the OM API to a CP client
    - Most common example is TSO SPOC
    - Other examples are REXX programs using the REXX SPOC API, IMS Connect (for IMS Control Center), vendor or user-written SPOC



AOP Clients

TSO SPOC

IMS Control Center

IMS Connect

REXX SPOC

OM

CP Clients

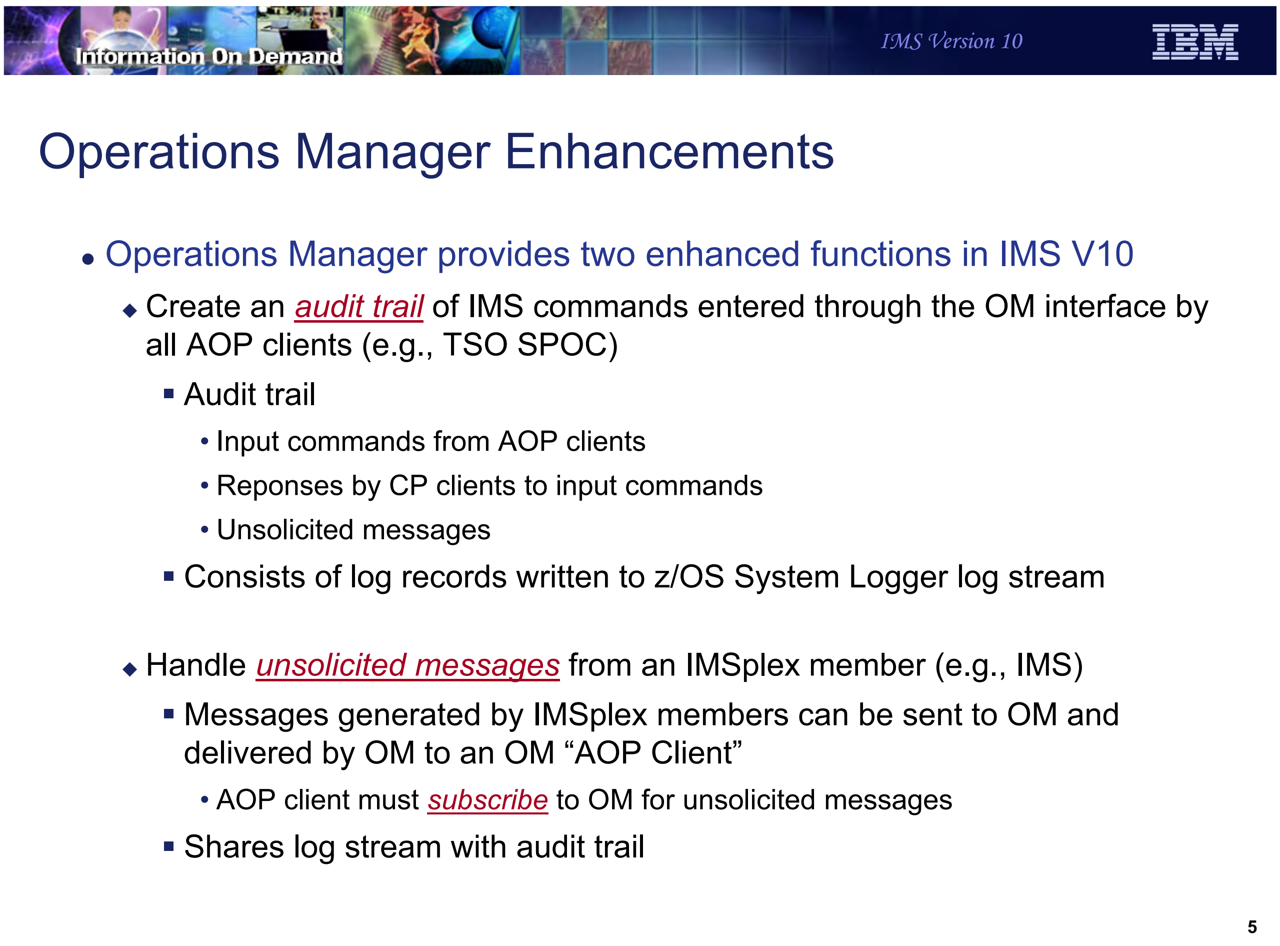
IMS

RM

Figure 6-3. OM Terminology Review CMA01.0

Notes:

As a review, OM has command processing clients and automated operator program clients.


Information On Demand
IMS Version 10

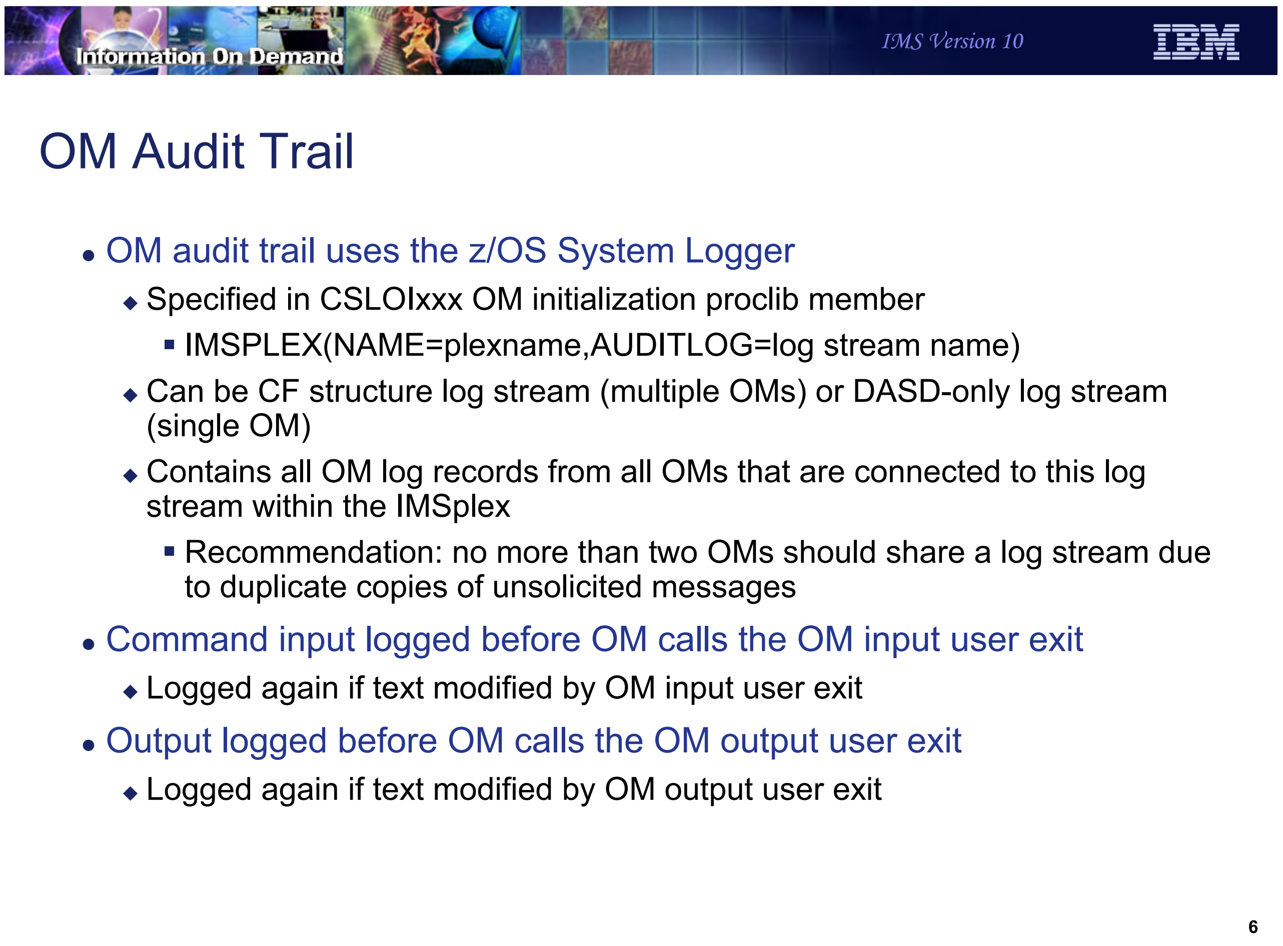
Operations Manager Enhancements

- Operations Manager provides two enhanced functions in IMS V10
  - Create an *audit trail* of IMS commands entered through the OM interface by all AOP clients (e.g., TSO SPOC)
    - Audit trail
      - Input commands from AOP clients
      - Reponses by CP clients to input commands
      - Unsolicited messages
    - Consists of log records written to z/OS System Logger log stream
  - Handle *unsolicited messages* from an IMSplex member (e.g., IMS)
    - Messages generated by IMSplex members can be sent to OM and delivered by OM to an OM “AOP Client”
      - AOP client must *subscribe* to OM for unsolicited messages
    - Shares log stream with audit trail

Figure 6-4. Operations Manager Enhancements CMA01.0

Notes:

In IMS V10, OM provides two new functions: (1) an audit trail of commands and (2) support for unsolicited messages.



Information On Demand

IMS Version 10

IBM

OM Audit Trail

- OM audit trail uses the z/OS System Logger
  - Specified in CSLOIxxx OM initialization proclib member
    - IMSPLEX(NAME=plexname,AUDITLOG=log stream name)
  - Can be CF structure log stream (multiple OMs) or DASD-only log stream (single OM)
  - Contains all OM log records from all OMs that are connected to this log stream within the IMSplex
    - Recommendation: no more than two OMs should share a log stream due to duplicate copies of unsolicited messages
- Command input logged before OM calls the OM input user exit
  - Logged again if text modified by OM input user exit
- Output logged before OM calls the OM output user exit
  - Logged again if text modified by OM output user exit

Figure 6-5. OM Audit Trail CMA01.0

Notes:

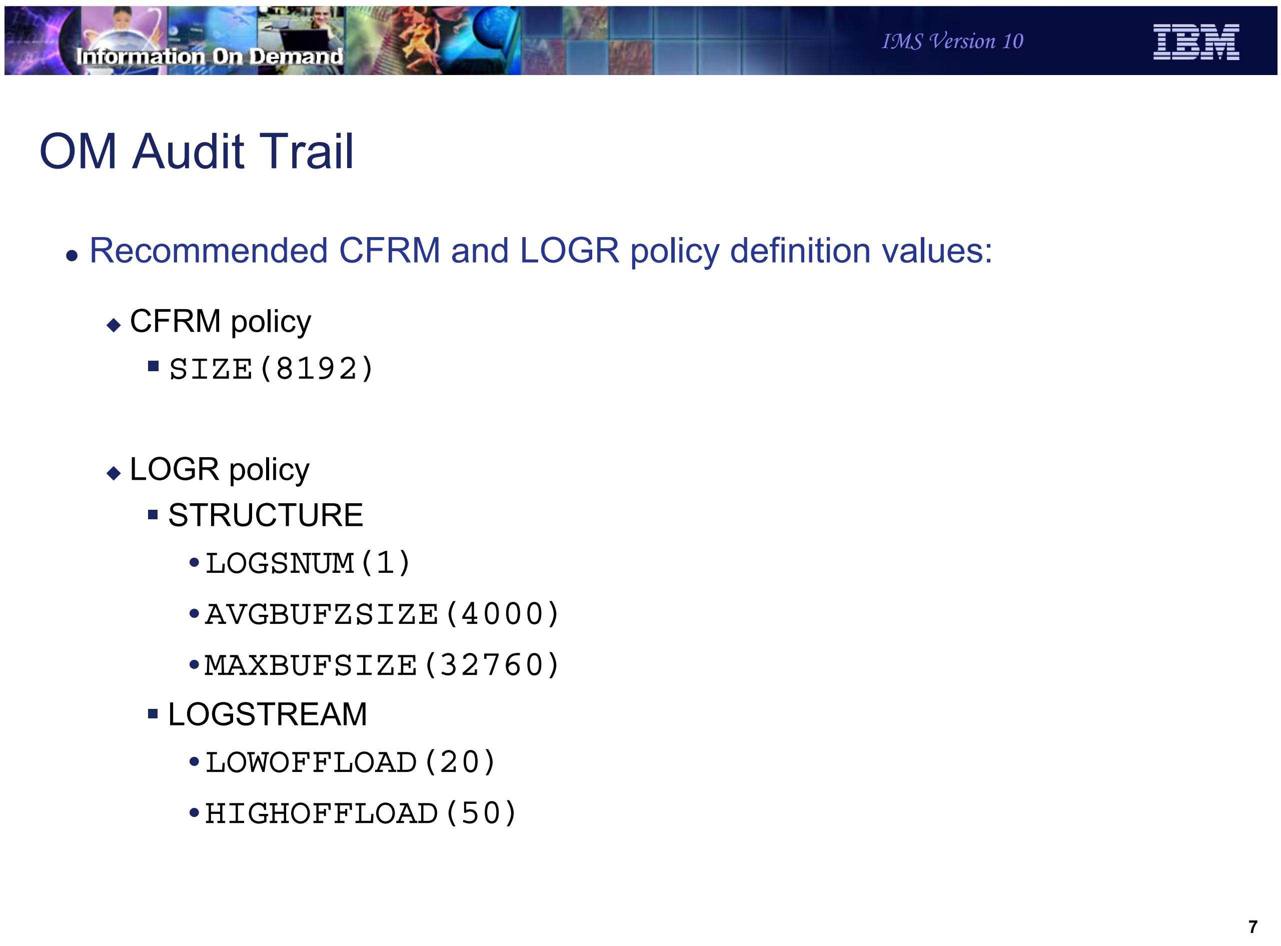
The OM Audit Trail is implemented with an MVS logger log stream. It is specified via the AUDITLOG=log stream name in CSLOIxxx. The log stream can be either in a CF (for multiple IMSs) or DASD-only (for a single IMS). OM does not require that the same MVS log stream be used within an IMSplex. You may have multiple OMs use the same log stream or they may each have their own log stream. When multiple OMs use the same log stream a CF is required. The audit trail will be maintained in chronological sequence.

Each OM will receive a copy of unsolicited messages. If multiple OMs share a log stream, it will contain multiple copies of unsolicited messages. For this reason, IBM recommends that no more two OMs share a log stream to avoid many duplicate messages. On the other hand, the advantage of a shared log stream is that all messages are in one stream.

The OM input user exit is an optional exit that allows a user to view and manipulate command input from an automation client. This exit is called before OM processes the command, which allows the command to be modified or rejected.

The OM output user exit is an optional exit that allows a user to view and manipulate output from OM. This exit is called when a command has been processed and is ready to be delivered to the originator of the command; the exit can modify the response text before it is delivered, or when an unsolicited message is received from a client using the CSLOMOUT API.

Based on whether or not the optional OM input and OM output exits modify the text, each input command or output response will have one or two entries in the audit trail.



Information On Demand

IMS Version 10

OM Audit Trail

- Recommended CFRM and LOGR policy definition values:
  - CFRM policy
    - `SIZE(8192)`
  - LOGR policy
    - STRUCTURE
      - `LOGSNUM(1)`
      - `AVGBUFZSIZE(4000)`
      - `MAXBUFSIZE(32760)`
    - LOGSTREAM
      - `LOWOFFLOAD(20)`
      - `HIGHOFFLOAD(50)`

7

Figure 6-6. OM Audit Trail CMA01.0

Notes:

The recommended CFRM and LOGR policy definition values are shown.

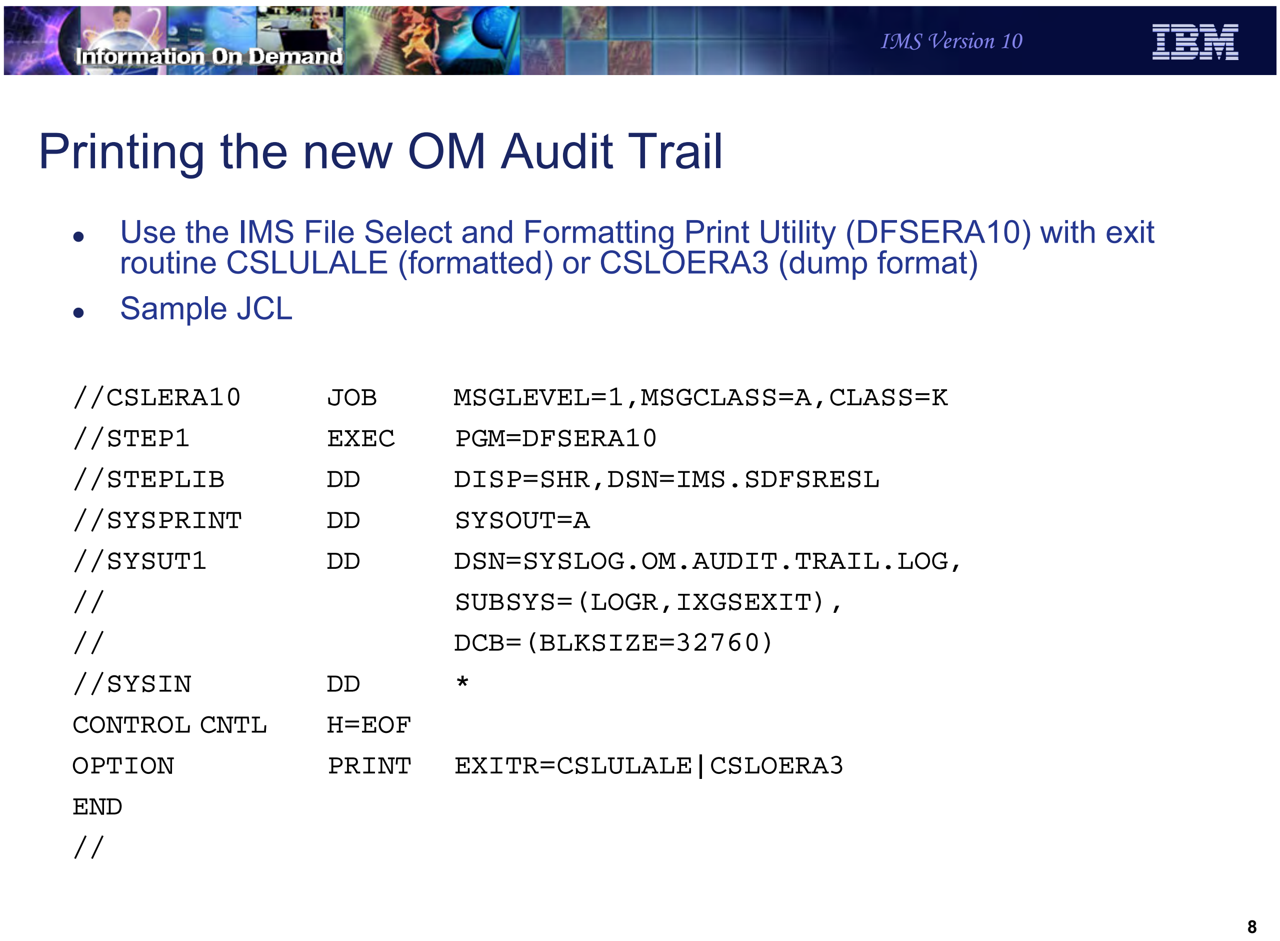
The recommended structure size definition in the CFRM policy is SIZE(8192). This is specified in Kbytes, so it creates an 8M structure.

The DEFINE STRUCTURE statement in the LOGR policy should include:

```
LOGSNUM(1)
AVGBUFZSIZE(4000)
MAXBUFSIZE(32760)
```

The DEFINE LOGSTREAM statement in the LOGR policy should include:

```
LOWOFFLOAD(20)
HIGHOFFLOAD(50)
```


Information On Demand
IMS Version 10
IBM

Printing the new OM Audit Trail

- Use the IMS File Select and Formatting Print Utility (DFSERA10) with exit routine CSLULALE (formatted) or CSLOERA3 (dump format)
- Sample JCL

```
//CSLERA10      JOB     MSGLEVEL=1,MSGCLASS=A,CLASS=K
//STEP1         EXEC    PGM=DFSERA10
//STEPLIB       DD      DISP=SHR,DSN=IMS.SDFSRESL
//SYSPRINT      DD      SYSOUT=A
//SYSUT1        DD      DSN=SYSLOG.OM.AUDIT.TRAIL.LOG,
//                      SUBSYS=(LOGR,IXGSEXIT),
//                      DCB=(BLKSIZE=32760)
//SYSIN         DD      *
CONTROL CNTL    H=EOF
OPTION          PRINT   EXITR=CSLULALE|CSLOERA3
END
//
```

Figure 6-7. Printing the new OM Audit Trail CMA01.0

Notes:

This is an example of how to use DFSERA10, the IMS File Select and Formatting Print Utility, to print the OM Audit Trail. The JCL requirements are:

STEPLIB - DSN points to IMS.SDFSRESL, which contains DFSERA10

SYSUT1 - DSN= points to the log stream name that was specified in the CSLOIxxx PROCLIB member on the AUDITLOG parameter

H= - specifies the number of log records to print (H=EOF to print all records)

EXITR=CSLxxxxx – specifies the log record routine that is called to format each log record

OM log record formats are available by assembling mapping macro CSLOLGRC.

Each OM log record contains a log record prefix, followed by data that is unique to the record. Macro CSLZLGPF maps the log record prefix.

The audit trail log records are:

Log Record

TypeSubtypeMapping MacroConditions for writing log record

X’06’X’01’CSLOLGCMCommand input, prior to calling the OM Input user exit

X’06’X’02’CSLOLGCMCommand input, after being modified by the OM Input user exit

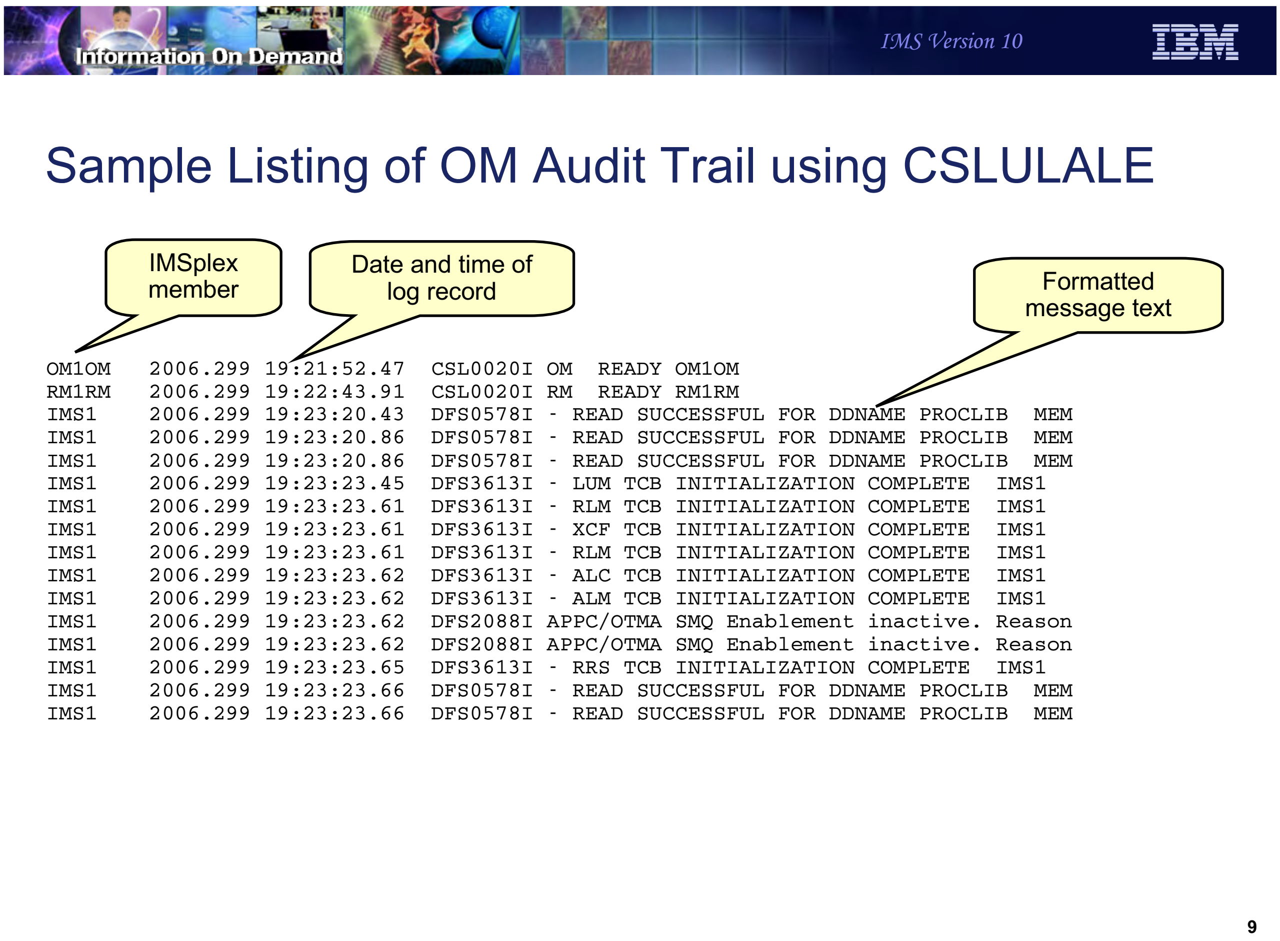
X’08’X’01’CSLOLGCRCommand response output, prior to calling the OM Output user exit

X’08’X’02’CSLOLGCRCommand response output after being modified by the OM Output user exit

X’09’X’01’CSLOLGOUUnsolicited output message to an AOP client before calling OM Output user exit

X’09’X’02’CSLOLGOUUnsolicited output message to an AOP client after calling OM Output user exit and OM output user exit modified the unsolicited output message

The documentation for printing the OM audit trail using CSLULALE with DFSERA10 for formatted output is in the IMS 10 System Utilities Reference; the documentation for printing the OM audit trail using CSLOERA3 with DFSERA10 for dump output is in the IMS 10 Diagnosis Guide.



Sample Listing of OM Audit Trail using CSLULALE

IMSplex member

Date and time of log record

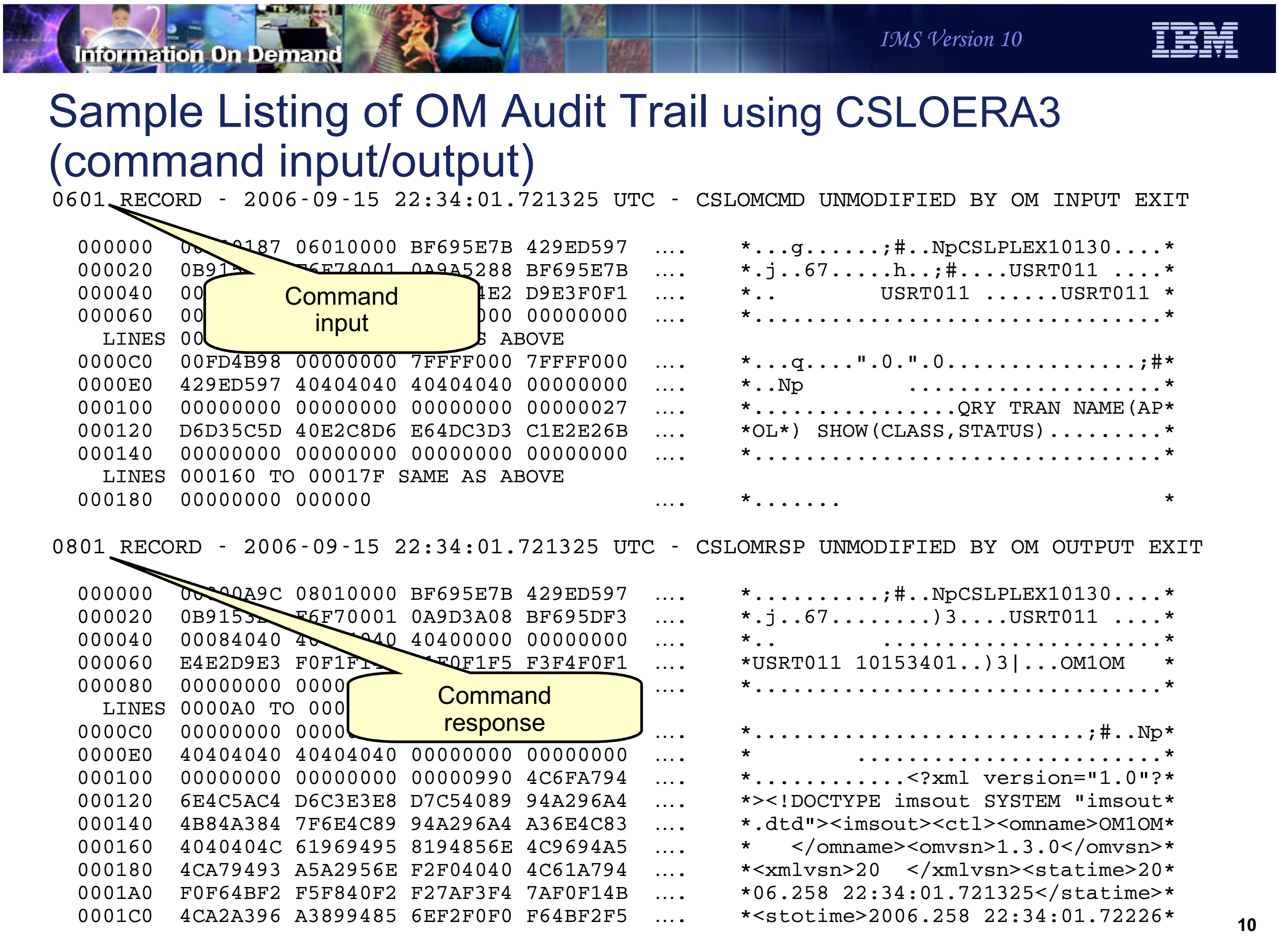
Formatted message text

```
OM1OM   2006.299 19:21:52.47  CSL0020I OM  READY OM1OM
RM1RM   2006.299 19:22:43.91  CSL0020I RM  READY RM1RM
IMS1    2006.299 19:23:20.43  DFS0578I - READ SUCCESSFUL FOR DDNAME PROCLIB  MEM
IMS1    2006.299 19:23:20.86  DFS0578I - READ SUCCESSFUL FOR DDNAME PROCLIB  MEM
IMS1    2006.299 19:23:20.86  DFS0578I - READ SUCCESSFUL FOR DDNAME PROCLIB  MEM
IMS1    2006.299 19:23:23.45  DFS3613I - LUM TCB INITIALIZATION COMPLETE  IMS1
IMS1    2006.299 19:23:23.61  DFS3613I - RLM TCB INITIALIZATION COMPLETE  IMS1
IMS1    2006.299 19:23:23.61  DFS3613I - XCF TCB INITIALIZATION COMPLETE  IMS1
IMS1    2006.299 19:23:23.61  DFS3613I - RLM TCB INITIALIZATION COMPLETE  IMS1
IMS1    2006.299 19:23:23.62  DFS3613I - ALC TCB INITIALIZATION COMPLETE  IMS1
IMS1    2006.299 19:23:23.62  DFS3613I - ALM TCB INITIALIZATION COMPLETE  IMS1
IMS1    2006.299 19:23:23.62  DFS2088I APPC/OTMA SMQ Enablement inactive. Reason
IMS1    2006.299 19:23:23.62  DFS2088I APPC/OTMA SMQ Enablement inactive. Reason
IMS1    2006.299 19:23:23.65  DFS3613I - RRS TCB INITIALIZATION COMPLETE  IMS1
IMS1    2006.299 19:23:23.66  DFS0578I - READ SUCCESSFUL FOR DDNAME PROCLIB  MEM
IMS1    2006.299 19:23:23.66  DFS0578I - READ SUCCESSFUL FOR DDNAME PROCLIB  MEM
```

Figure 6-8. Sample Listing of OM Audit Trail using CSLULALE CMA01.0

Notes:

This shows the listing created by DFSERA10 with exit routine CSLULALE (formatted output) for messages in the OM audit trail.



Sample Listing of OM Audit Trail using CSLOERA3 (command input/output)

```
0601 RECORD - 2006-09-15 22:34:01.721325 UTC - CSLOMCMD UNMODIFIED BY OM INPUT EXIT

  000000   00  0187 06010000 BF695E7B 429ED597  ....    *...g......;#..NpCSLPLEX10130....*
  000020   0B91     6F78001 0A9A5288 BF695E7B  ....    *.j..67.....h..;#....USRT011 ....*
  000040   00                  4E2 D9E3F0F1  ....    *..         USRT011 ......USRT011 *
  000060   00                  000 00000000  ....    *................................*
    LINES 00                  S ABOVE
  0000C0   00FD4B98 00000000 7FFFF000 7FFFF000  ....    *...q....".0.".0...............;#*
  0000E0   429ED597 40404040 40404040 00000000  ....    *..Np        ....................*
  000100   00000000 00000000 00000000 00000027  ....    *................QRY TRAN NAME(AP*
  000120   D6D35C5D 40E2C8D6 E64DC3D3 C1E2E26B  ....    *OL*) SHOW(CLASS,STATUS).........*
  000140   00000000 00000000 00000000 00000000  ....    *................................*
    LINES 000160 TO 00017F SAME AS ABOVE
  000180   00000000 000000                     ....    *.......                         *

0801 RECORD - 2006-09-15 22:34:01.721325 UTC - CSLOMRSP UNMODIFIED BY OM OUTPUT EXIT

  000000   00  00A9C 08010000 BF695E7B 429ED597  ....    *..........;#..NpCSLPLEX10130....*
  000020   0B9153    F6F70001 0A9D3A08 BF695DF3  ....    *.j..67........)3....USRT011 ....*
  000040   00084040 40   040 40400000 00000000  ....    *..        ......................*
  000060   E4E2D9E3 F0F1F1    F0F1F5 F3F4F0F1  ....    *USRT011 10153401..)3|...OM1OM   *
  000080   00000000 0000                        ....    *................................*
    LINES 0000A0 TO 000
  0000C0   00000000 0000                        ....    *........................;#..Np*
  0000E0   40404040 40404040 00000000 00000000  ....    *        ........................*
  000100   00000000 00000000 00000990 4C6FA794  ....    *............<?xml version="1.0"?*
  000120   6E4C5AC4 D6C3E3E8 D7C54089 94A296A4  ....    *><!DOCTYPE imsout SYSTEM "imsout*
  000140   4B84A384 7F6E4C89 94A296A4 A36E4C83  ....    *.dtd"><imsout><ctl><omname>OM1OM*
  000160   4040404C 61969495 8194856E 4C9694A5  ....    *   </omname><omvsn>1.3.0</omvsn>*
  000180   4CA79493 A5A2956E F2F04040 4C61A794  ....    *<xmlvsn>20  </xmlvsn><statime>20*
  0001A0   F0F64BF2 F5F840F2 F27AF3F4 7AF0F14B  ....    *06.258 22:34:01.721325</statime>*
  0001C0   4CA2A396 A3899485 6EF2F0F0 F64BF2F5  ....    *<stotime>2006.258 22:34:01.72226*
```

Figure 6-9. Sample Listing of OM Audit Trail using CSLOERA3 (command input/output) CMA01.0

Notes:

This shows the listing created by DFSERA10 with exit routine CSLOERA3 (dump format) for a QRY TRAN command where the OM input and OM output exits did not modify the messages. Since this is in XML format, it is very large and only the beginning of the output record is shown here.

Information On Demand

IMS Version 10

Unsolicited Message Support

- An unsolicited message is a message that is issued by a CP client such as IMS but is not associated with a specific command input
- OM support for unsolicited messages
  - Logs unsolicited messages to “audit trail log stream” defined in CSLOIxxx
  - If requested, will send these messages to a user-provided AOP client
- AOP client must use new OM requests to Subscribe and Unsubscribe to unsolicited messages
  - Implemented by REXX SPOC API
  - Not yet implemented by IMS Control Center

11

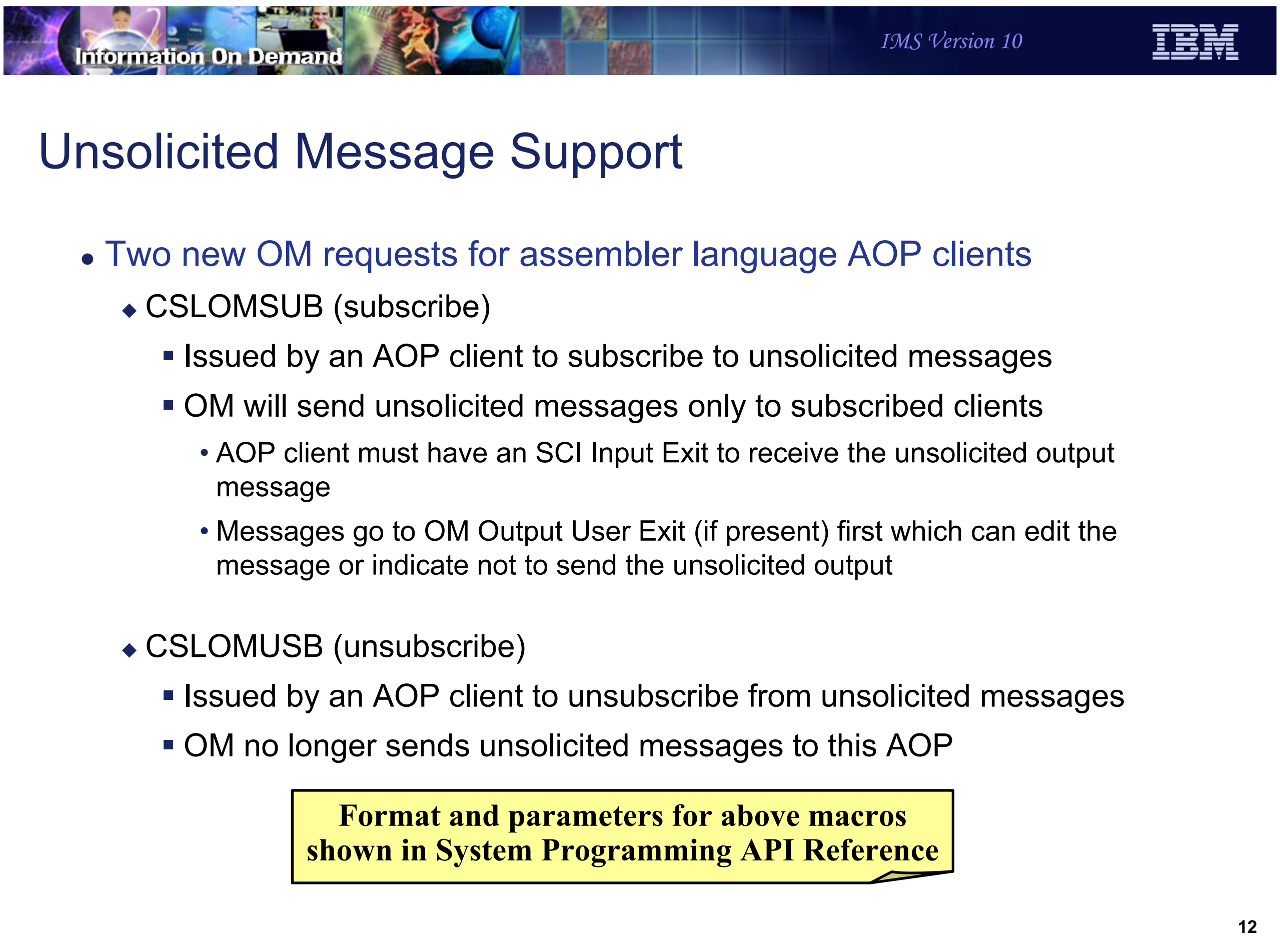
Figure 6-10. Unsolicited Message Support CMA01.0

Notes:

A command processing client will use the CSLOMOUT API to send an unsolicited message.

The IMS-provided TSO SPOC does not use Subscribe/Unsubscribe support.

Command responses are sent as unsolicited messages to any AOP clients that have asked for them (subscribed), in addition to sending the command response back to the client that issued the command.



Unsolicited Message Support

- Two new OM requests for assembler language AOP clients
  - CSLOMSUB (subscribe)
    - Issued by an AOP client to subscribe to unsolicited messages
    - OM will send unsolicited messages only to subscribed clients
      - AOP client must have an SCI Input Exit to receive the unsolicited output message
      - Messages go to OM Output User Exit (if present) first which can edit the message or indicate not to send the unsolicited output
  - CSLOMUSB (unsubscribe)
    - Issued by an AOP client to unsubscribe from unsolicited messages
    - OM no longer sends unsolicited messages to this AOP

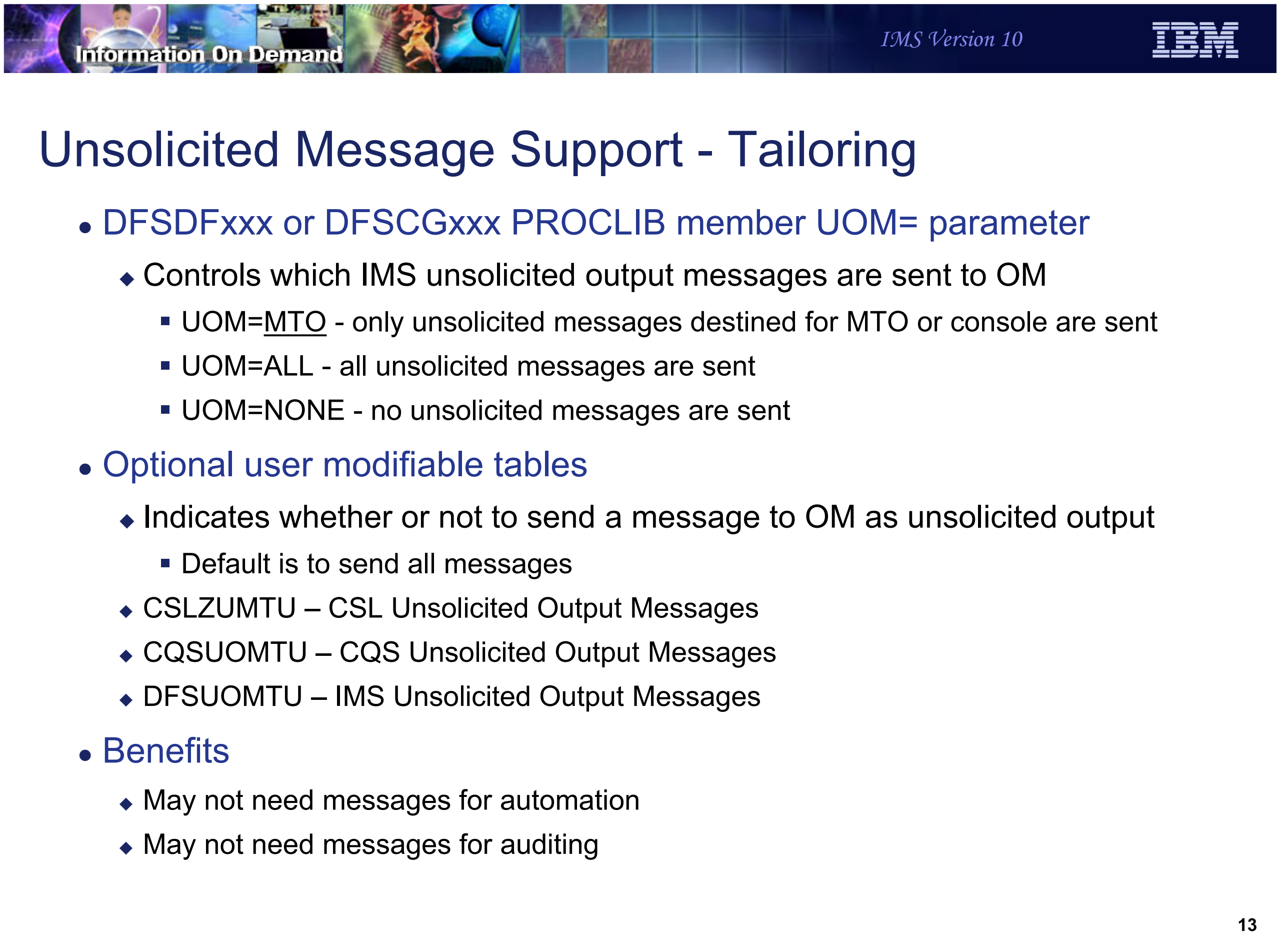
Format and parameters for above macros shown in System Programming API Reference

Figure 6-11. Unsolicited Message Support CMA01.0

Notes:

OM provides two new requests for unsolicited message support for assembler language programs: CSLOMSUB and CSLOMUSB.

REXX support is shown in the next section.


Information On Demand
IMS Version 10
IBM

Unsolicited Message Support - Tailoring

- DFSDFxxx or DFSCGxxx PROCLIB member UOM= parameter
  - Controls which IMS unsolicited output messages are sent to OM
    - UOM=MTO - only unsolicited messages destined for MTO or console are sent
    - UOM=ALL - all unsolicited messages are sent
    - UOM=NONE - no unsolicited messages are sent
- Optional user modifiable tables
  - Indicates whether or not to send a message to OM as unsolicited output
    - Default is to send all messages
  - CSLZUMTU – CSL Unsolicited Output Messages
  - CQSUOMTU – CQS Unsolicited Output Messages
  - DFSUOMTU – IMS Unsolicited Output Messages
- Benefits
  - May not need messages for automation
  - May not need messages for auditing

13

Figure 6-12. Unsolicited Message Support - Tailoring CMA01.0

Notes:

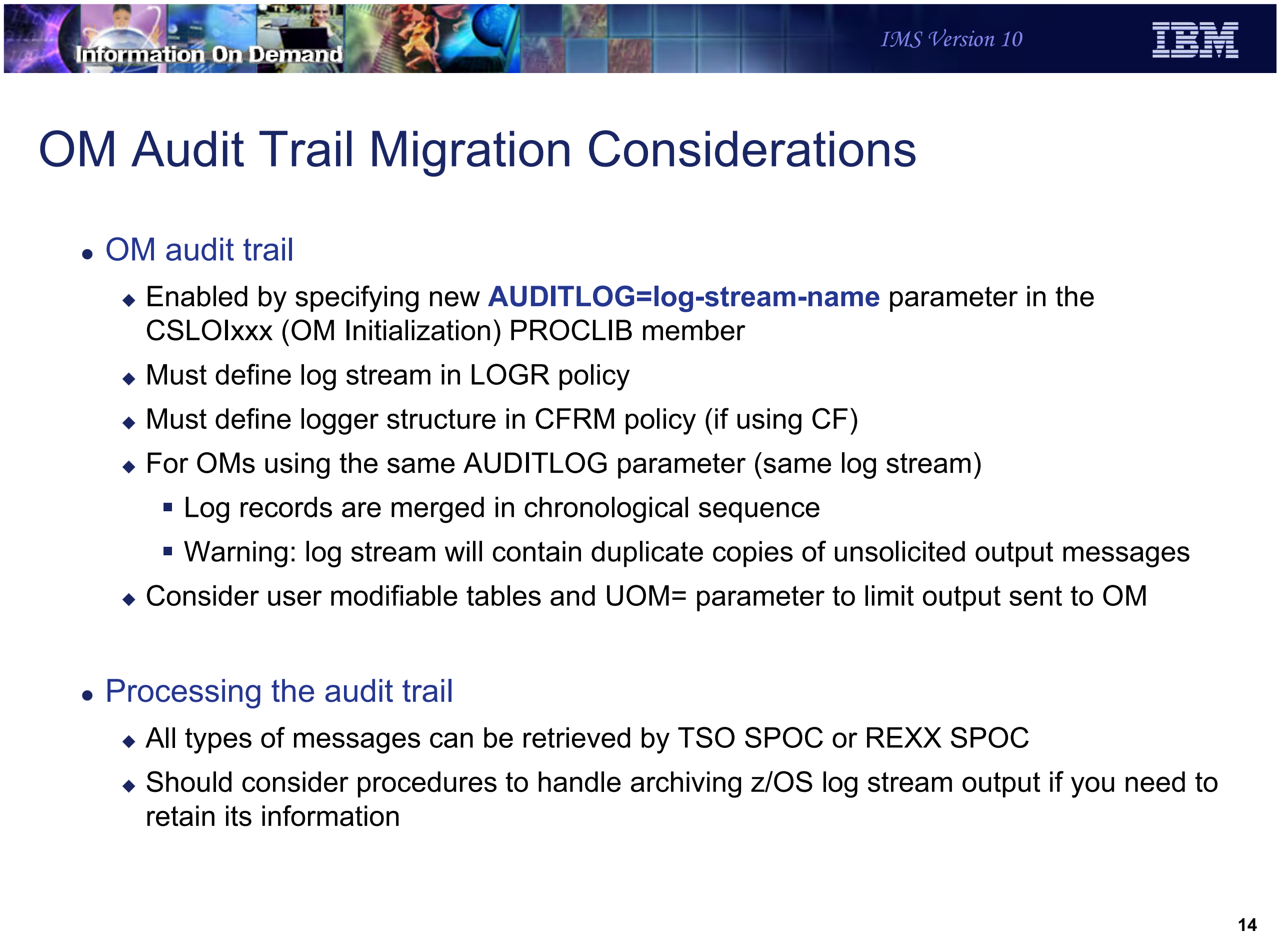
There are two ways to limit the unsolicited messages sent to OM.

IMS messages may be limited by using the UOM= parameter in the DFSDFxxx or DFSCGxxx PROCLIB member. UOM=MTO is the default. It specifies that the only unsolicited messages sent to OM from IMS are those that IMS sends to the MTO or system console. UOM=ALL specifies that all unsolicited messages from IMS are sent to OM. UOM=NONE specifies that no unsolicited messages are sent to OM from IMS.

There are optional user modifiable tables for IMS, CSL, and CQS which may be used to send or not send messages based on their message number. You may specify whether messages whose numbers are specified should be sent or not sent. You may also specify whether messages whose numbers are not specified in the table should be sent or not sent.

Of course, if messages are not sent to OM they will not appear in the audit trail.

You may want to limit the unsolicited output messages sent to OM for several reasons. You may only need some messages from to AOP processing. You may only need some for auditing purposes. You may not need any messages.


Information On Demand
IMS Version 10

OM Audit Trail Migration Considerations

- OM audit trail
  - Enabled by specifying new **AUDITLOG=log-stream-name** parameter in the CSLOIxxx (OM Initialization) PROCLIB member
  - Must define log stream in LOGR policy
  - Must define logger structure in CFRM policy (if using CF)
  - For OMs using the same AUDITLOG parameter (same log stream)
    - Log records are merged in chronological sequence
    - Warning: log stream will contain duplicate copies of unsolicited output messages
  - Consider user modifiable tables and UOM= parameter to limit output sent to OM

- Processing the audit trail
  - All types of messages can be retrieved by TSO SPOC or REXX SPOC
  - Should consider procedures to handle archiving z/OS log stream output if you need to retain its information

Figure 6-13. OM Audit Trail Migration Considerations CMA01.0

Notes:

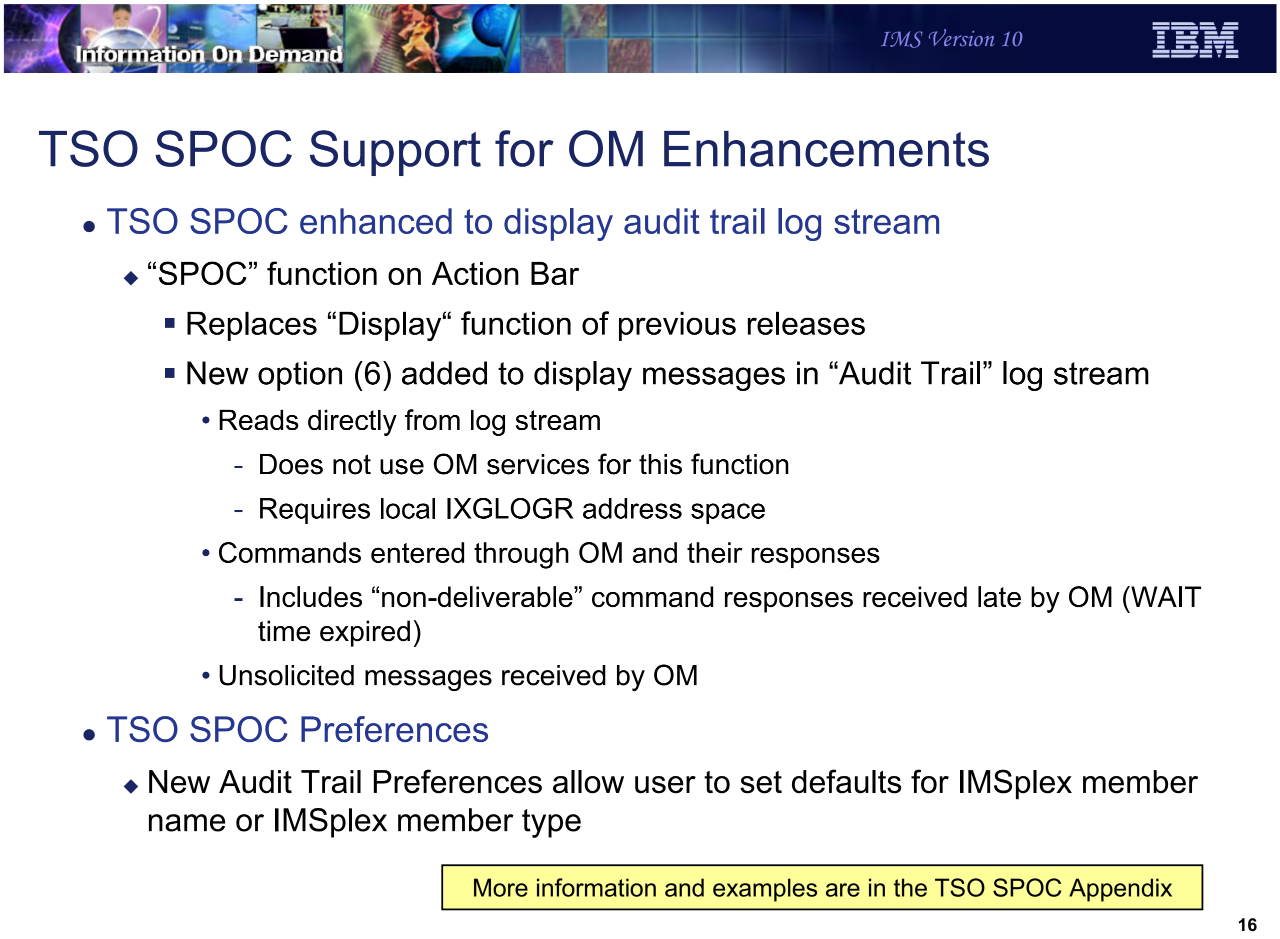
Information On Demand
IMS Version 10
IBM

TSO SPOC and REXX Support

Figure 6-14. [No title] CMA01.0

Notes:

This section addresses enhancements to the SPOC ISPF application and enhancements to REXX support for the OM interface.


Information On Demand
IMS Version 10
IBM

TSO SPOC Support for OM Enhancements

- TSO SPOC enhanced to display audit trail log stream
  - “SPOC” function on Action Bar
    - Replaces “Display“ function of previous releases
    - New option (6) added to display messages in “Audit Trail” log stream
      - Reads directly from log stream
        - Does not use OM services for this function
        - Requires local IXGLOGR address space
      - Commands entered through OM and their responses
        - Includes “non-deliverable” command responses received late by OM (WAIT time expired)
      - Unsolicited messages received by OM
- TSO SPOC Preferences
  - New Audit Trail Preferences allow user to set defaults for IMSplex member name or IMSplex member type

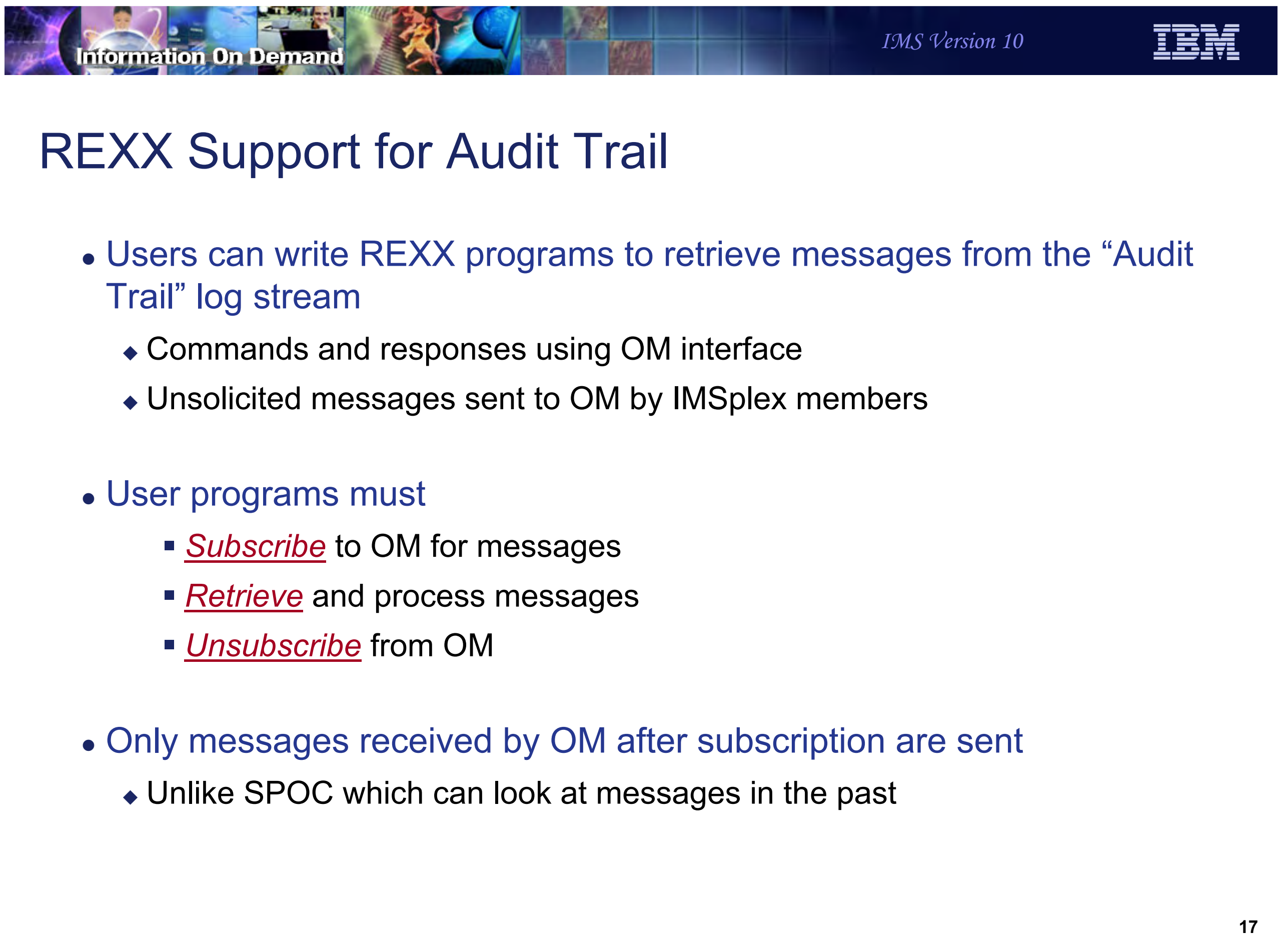
More information and examples are in the TSO SPOC Appendix

16

Figure 6-15. TSO SPOC Support for OM Enhancements CMA01.0

Notes:

The TSO SPOC program has been enhanced to support the new OM “audit trail” of command input/response through the OM and unsolicited messages sent by IMSplex members to OM. Note that the function of SPOC reads directly from the Audit Trail logstream. It does NOT retrieve any of these messages directly from OM. This allows the user to scroll back and forth through the entire logstream, going as far back in time as those log records still in the logstream. This does require that the SPOC be on an LPAR which has an ISGLOGR address space to which it can connect and request log records.


Information On Demand
IMS Version 10
IBM

REXX Support for Audit Trail

- Users can write REXX programs to retrieve messages from the “Audit Trail” log stream
  - Commands and responses using OM interface
  - Unsolicited messages sent to OM by IMSplex members
- User programs must
  - *Subscribe* to OM for messages
  - *Retrieve* and process messages
  - *Unsubscribe* from OM
- Only messages received by OM after subscription are sent
  - Unlike SPOC which can look at messages in the past

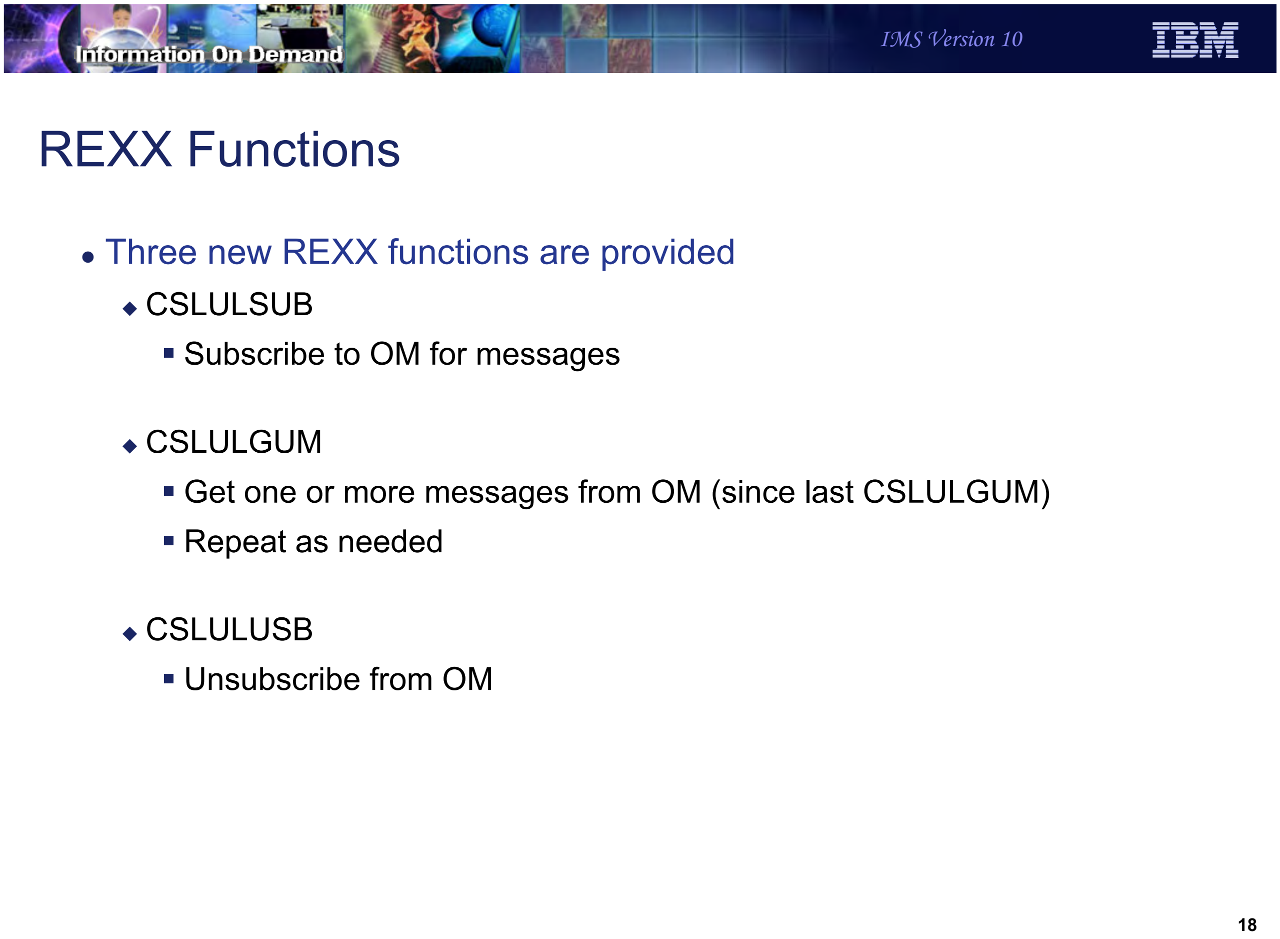
Figure 6-16. REXX Support for Audit Trail CMA01.0

Notes:

You can also write a REXX program to

- join the IMSplex
- subscribe to OM for unsolicited messages (actually, for any logstream message that arrives after you subscribe, including command input and response)
- retrieve the messages into a REXX stem variable
- unsubscribe from OM when you want to exit

Unlike the TSO SPOC application, this feature receives its messages from OM directly – not from the system logger. OM will only send messages received AFTER the program subscribes.


Information On Demand
IMS Version 10
IBM

REXX Functions

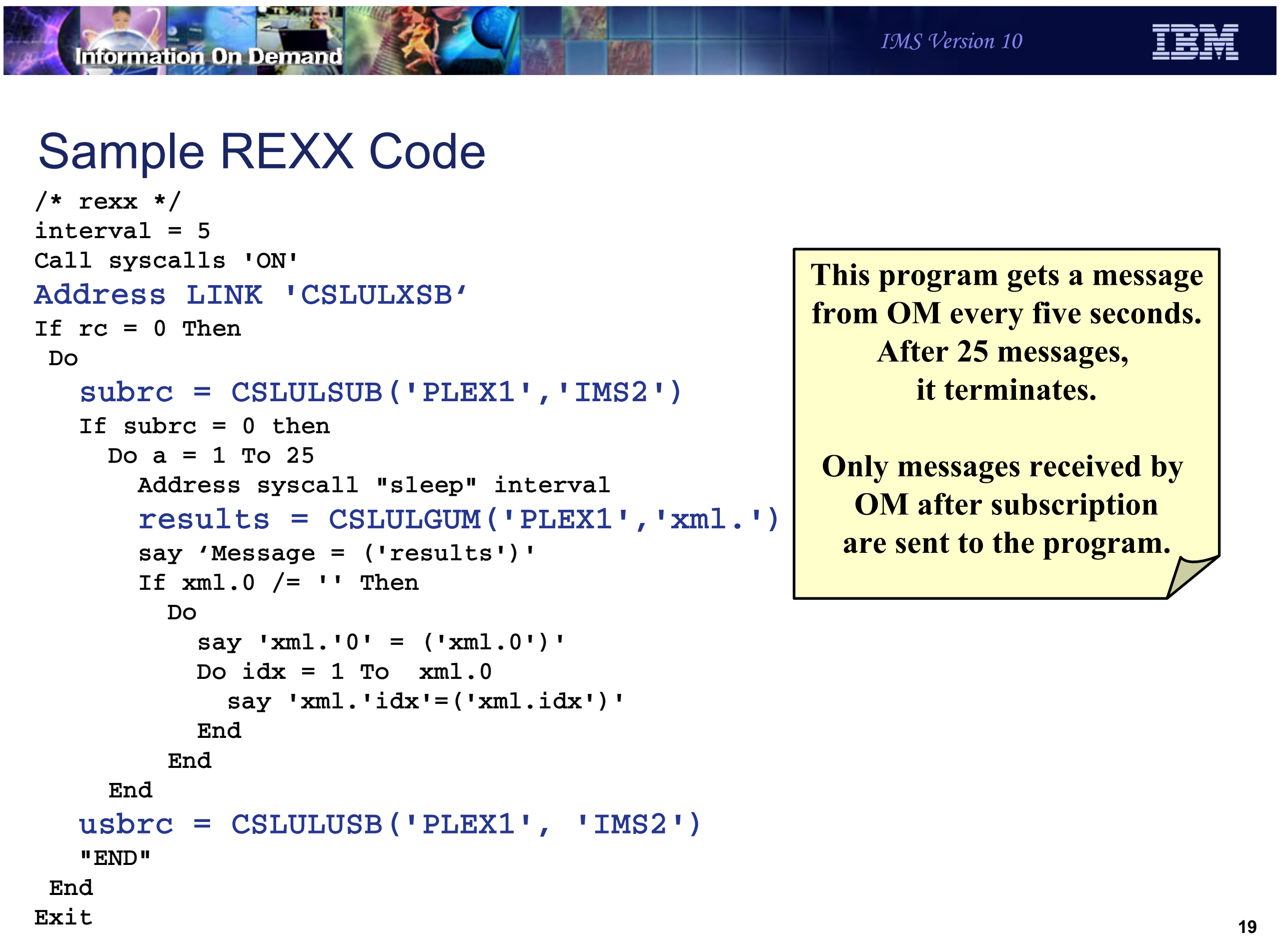
- Three new REXX functions are provided
  - CSLULSUB
    - Subscribe to OM for messages
  - CSLULGUM
    - Get one or more messages from OM (since last CSLULGUM)
    - Repeat as needed
  - CSLULUSB
    - Unsubscribe from OM

Figure 6-17. REXX Functions CMA01.0

Notes:

There are three new REXX functions provided by IMS to help you with these functions.

- CSLULSUB is a function used to subscribe to OM
- CSLULGUM retrieves messages from OM and puts them into a REXX stem variable
- CSLULUSB is used to unsubscribe to OM


Information On Demand
IMS Version 10

Sample REXX Code

```
/* rexx */
interval = 5
Call syscalls 'ON'
Address LINK 'CSLULXSB’
If rc = 0 Then
 Do
   subrc = CSLULSUB('PLEX1','IMS2')
   If subrc = 0 then
     Do a = 1 To 25
       Address syscall "sleep" interval
       results = CSLULGUM('PLEX1','xml.')
       say ‘Message = ('results')'
       If xml.0 /= '' Then
         Do
           say 'xml.'0' = ('xml.0')'
           Do idx = 1 To  xml.0
             say 'xml.'idx'=('xml.idx')'
           End
         End
     End
   usbrc = CSLULUSB('PLEX1', 'IMS2')
   "END"
 End
Exit
```

This program gets a message from OM every five seconds. After 25 messages, it terminates.

Only messages received by OM after subscription are sent to the program.

Figure 6-18. Sample REXX Code CMA01.0

Notes:

This is a sample of a REXX program which sets up the REXX SPOC environment, subscribes to OM for audit trail messages from IMS2 which is a member of CSLPLEX1.

You invoke the subscription service by coding **subrc = CSLULSUB** and its parameters. subrc will be set to the return code from this function. subrc=0 means it worked.

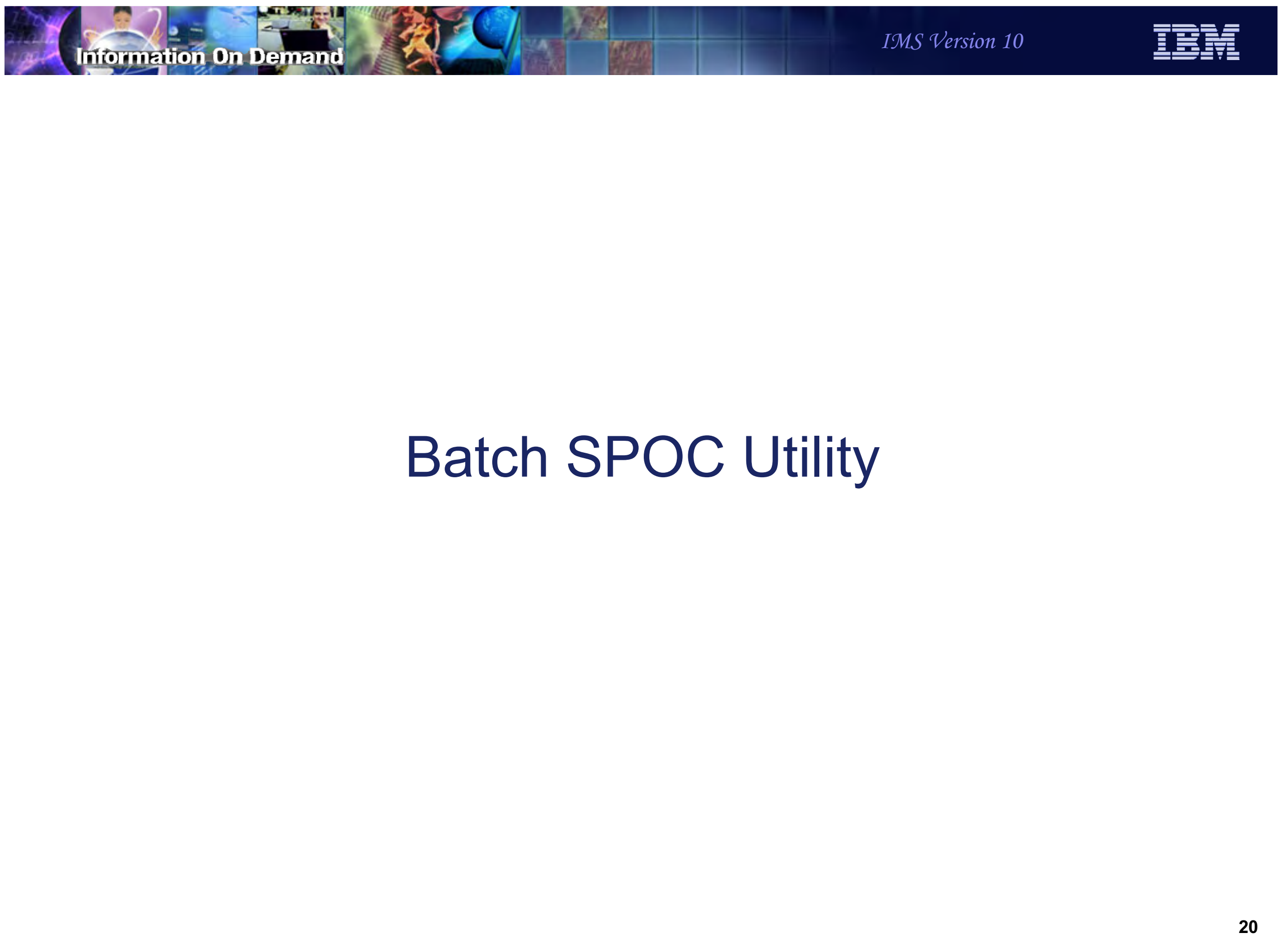
You retrieve the messages into the stem variable by coding - **results=CSLULGUM(plexname,stem-variable).** The stem variable in the example is xml. but you can use anything you like.

The results are returned in a stem variable named “xml.”

- xml.0 is the number of rows (lines) returned
- xml.1 is the first row
- xml.2 is the second row
- etc

It is up to the programmer to parse the responses and take action (if necessary).

Full details on coding these REXX functions is in the Systems Programming API Reference manual.



Information On Demand

IMS Version 10

IBM

Batch SPOC Utility

Figure 6-19. [No title] CMA01.0

Notes:

Batch SPOC Utility (CSLUSPOC)

- Provides a capability to submit IMS commands from a batch job
  - Uses the Operations Manager (OM) interface
  - Supports both Type-1 and Type-2 commands
- IMSplex environment defined in execution parameters
  - IMSplex name
  - Command routing
  - Wait time
- Commands defined in SYSIN file
  - Multiple commands allowed
  - Commands executed serially
- Output to SYSPRINT
  - Responses formatted to look like SPOC screen format

Figure 6-20. Batch SPOC Utility (CSLUSPOC) CMA01.0

Notes:

You can now invoke a batch SPOC application and submit IMS commands from a batch job using the OM interface. Like the online SPOC, both Type-1 and Type-2 commands are supported. Execution parameters include the IMSPLEX name, the routing (default routing is all IMSs), and how long you want OM to wait for IMS to respond before returning a negative response to the SPOC. Commands are coded in a SYSIN file with each command executed serially – that is, OM submits the first command and waits for a response before submitting the second command. The output goes to SYSPRINT and looks like a formatted SPOC screen.

Information On Demand
IMS Version 10

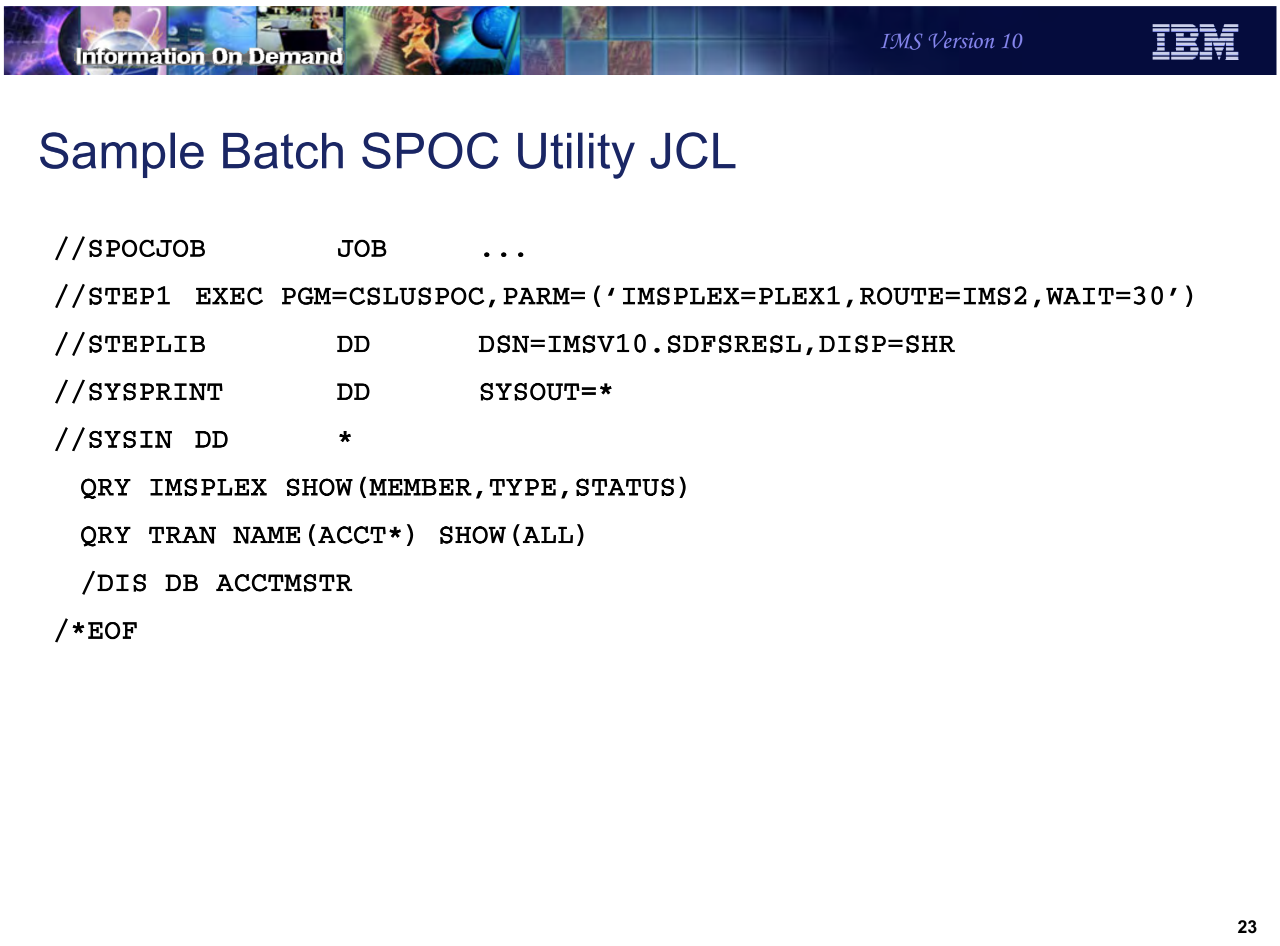
CSLUSPOC Parameters

- IMSPLEX= (required)
  - 5 character IMSplex name suffix
- ROUTE= (optional)
  - Defines routing for all commands in SYSIN file
  - Examples
    - ROUTE=IMS1
    - ROUTE=(IMS2,IMS4)
  - Default is all IMSs registered to OM
- WAIT= (optional)
  - Time OM waits for IMS to respond
  - Minutes and seconds (mmm:ss) or seconds (sssss)
  - WAIT=0
    - Submit next command immediately – don’t wait for response
  - Default is WAIT=5:00

Figure 6-21. CSLUSPOC Parameters CMA01.0

Notes:

- IMSPLEX name is required. (Note: you must have a SCI address space on the image you are running on, but IMS can be on any image in the sysplex.
- ROUTE defines the routing for the command. It is optional. If not coded, the default routing will be all IMSs.
- WAIT is also optional with a default of 5 minutes (probably way too long). Specifying WAIT=0 will cause the SPOC to submit commands immediately without waiting for a response to the previous command from OM. When you specify WAIT=0, OM will wait up to 5 minutes for an IMS to respond.


Information On Demand
IMS Version 10
IBM

Sample Batch SPOC Utility JCL

```
//SPOCJOB        JOB      ...
//STEP1  EXEC PGM=CSLUSPOC,PARM=('IMSPLEX=PLEX1,ROUTE=IMS2,WAIT=30')
//STEPLIB        DD       DSN=IMSV10.SDFSRESL,DISP=SHR
//SYSPRINT       DD       SYSOUT=*
//SYSIN  DD      *
  QRY IMSPLEX SHOW(MEMBER,TYPE,STATUS)
  QRY TRAN NAME(ACCT*) SHOW(ALL)
  /DIS DB ACCTMSTR
/*EOF
```

Figure 6-22. Sample Batch SPOC Utility JCL CMA01.0

Notes:

This is the JCL for the batch SPOC. The program name is CSLUSPOC. The input parameters are the IMSPLEX name, where you want the commands routed to, and how long you want OM to wait for a response from IMS before returning to the SPOC. The SPOC will then submit the next command in the SYSIN file.

The example shows the SYSIN to be inline, however, you could put the commands into a data set and identify the data set in the SYSIN DD statement. This would allow you to just update the contents of the file and resubmit the job rather than change the job input stream

If the continuation character is a plus (+) sign, then the next line is concatenated to this one without any blanks. This would be required when you are (for example) coding the SHOW parameters and they don't all fit on one line. A minus (-) sign will insert a blank before whatever is coded on the next line. For example, the next keyword should be separated by a blank from the previous one.

Information On Demand
IMS Version 10
IBM

//SYSIN File

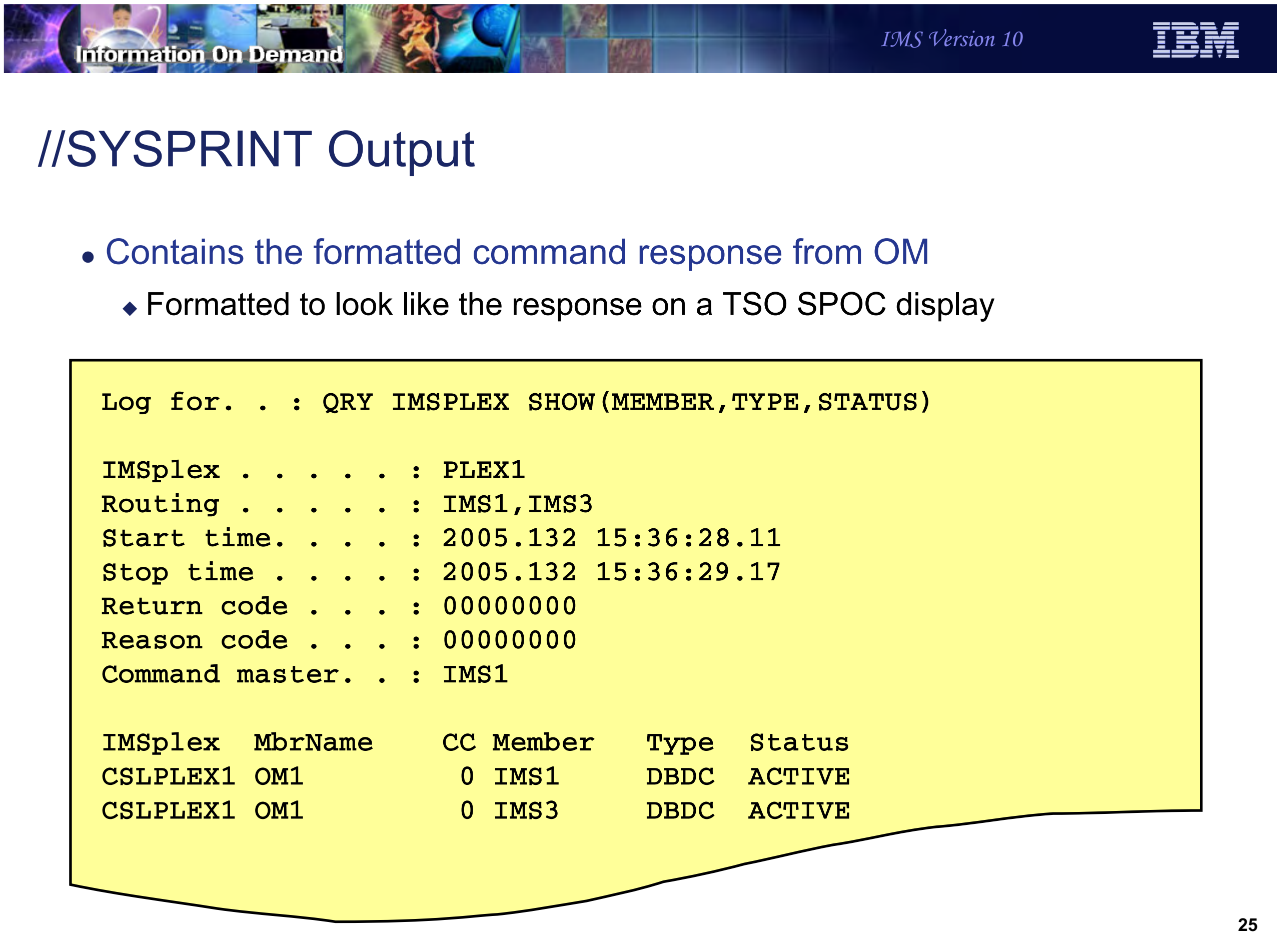
- Commands are read from a SYSIN file
  - Utility supports all commands supported by OM
  - Commands are executed serially
    - When OM responds to one command, the next is submitted
    - Exception: if WAIT=0, commands are submitted without waiting for response from OM

Figure 6-23. //SYSIN File CMA01.0

Notes:

The utility supports all commands supported by OM. They are executed serially, waiting for the response to one before submitting the next. If WAIT=0 is coded, commands are submitted without waiting for the response.

Command continuation characters allow commands to span multiple lines. A plus (+) sign means no blank between this line and the continuation line. A minus (-) sign means to insert a blank between lines.


Information On Demand
IMS Version 10
IBM

//SYSPRINT Output

- Contains the formatted command response from OM
  - Formatted to look like the response on a TSO SPOC display

```
Log for. . : QRY IMSPLEX SHOW(MEMBER,TYPE,STATUS)

IMSplex . . . . . : PLEX1
Routing . . . . . : IMS1,IMS3
Start time. . . . : 2005.132 15:36:28.11
Stop time . . . . : 2005.132 15:36:29.17
Return code . . . : 00000000
Reason code . . . : 00000000
Command master. . : IMS1

IMSplex  MbrName    CC Member   Type  Status
CSLPLEX1 OM1         0 IMS1     DBDC  ACTIVE
CSLPLEX1 OM1         0 IMS3     DBDC  ACTIVE
```

25

Figure 6-24. //SYSPRINT Output CMA01.0

Notes:

The output goes to SYSPRINT. If you print it, it will lock almost like the output to the same command submitted from the TSO SPOC (with a few things omitted, like function keys).

Information On Demand
IMS Version 10
IBM

//SYSPRINT Output

- If OM does not respond before WAIT interval, or if WAIT=0
  - Response is not written to SYSPRINT
  - Short summary information written

```
LOG FOR... : QRY IMSPLEX SHOW(MEMBER,TYPE,STATUS)

IMSplex ......:  PLEX1
Routing ......:  IMS1,IMS3
Start time ...:  2006.185 15:36:22.11
```

Figure 6-25. //SYSPRINT Output CMA01.0

Notes:

If OM does not respond within the wait interval time, but responds later, then that response does not go to sysprint. Instead, a short summary page is printed as shown. This is also true if you code WAIT=0.

Information On Demand
IMS Version 10

REXX XML Parser

Figure 6-26. [No title] CMA01.0

Notes:

This section addresses an XML parser that can be used with REXX programs to process command responses from IMS.

Prior to V10

- Responses to OM-submitted commands always encapsulated in XML
  - Commercial REXX XML parser does not exist
  - Makes interpreting command responses more difficult
    - Programs must look for XML tags
- Example of QRY TRAN command using the original function

```
...
“QRY TRAN NAME(CUS*)
results = CSLULGTS(‘qryinfo.’,cartid,’1:30’)"
...
```

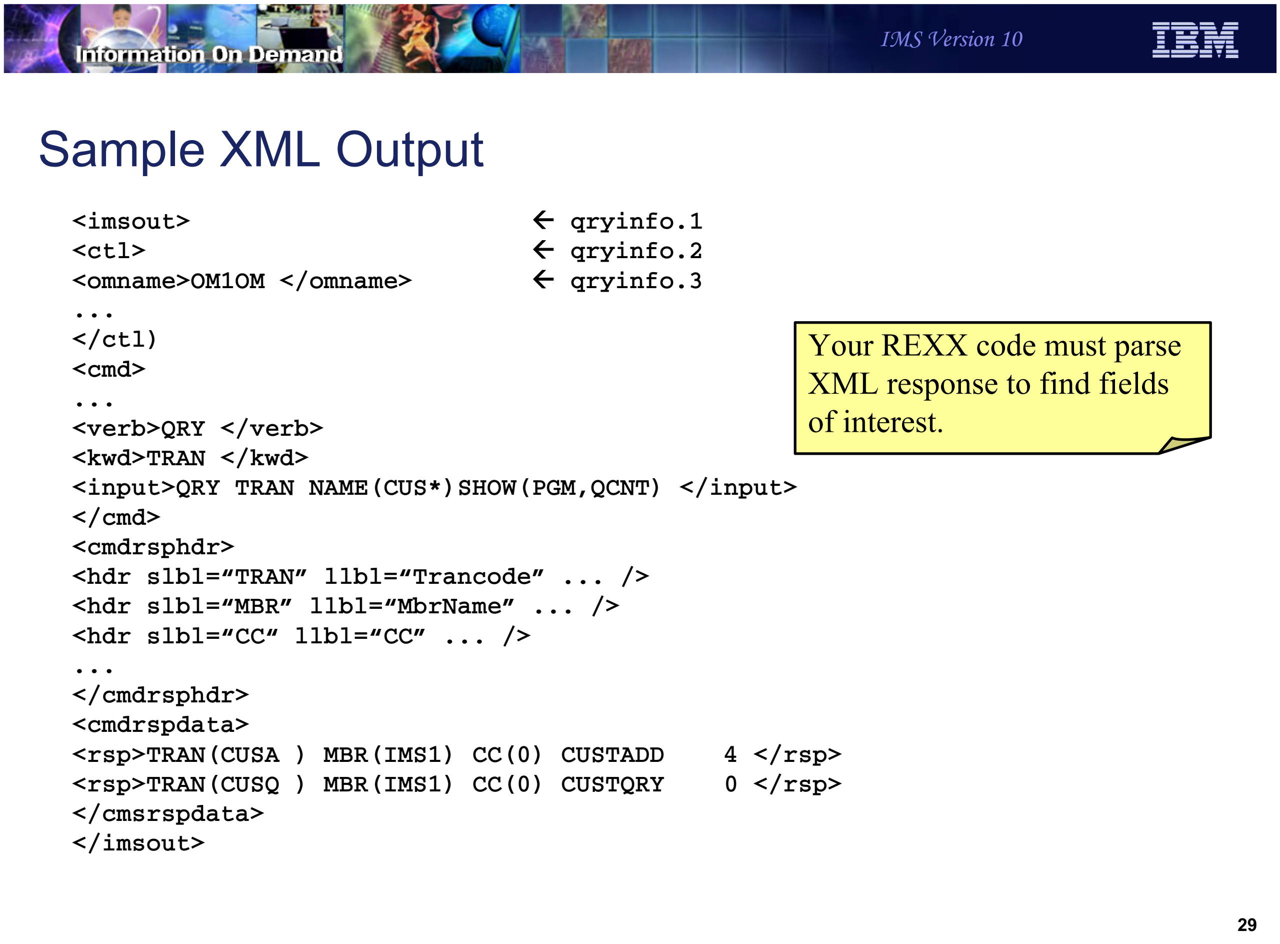
  - REXX stem variable is “qryinfo.”
    - qryinfo.0 = number of rows returned
    - qryinfo.1 = 1st row
    - qryinfo.2 = 2nd row
    - etc.

Figure 6-27. Prior to V10 CMA01.0

Notes:

IMS has provided support for REXX programs to join an IMSplex and submit commands to IMS using the OM interface. However, IMS always responds to these commands by encapsulating the response in XML. This made it difficult for the programmer to analyze and take any appropriate action.

The example shows a REXX function (CSLULGTS) used to retrieve the IMS response and put the “output lines” into a REXX stem variable. For example “qryinfo.1” would be the first line, “qqyinfo.2” the second line, etc. “qryinfo.0” is the number of lines (or rows) returned.



Sample XML Output

```
<imsout>                             ← qryinfo.1
<ctl>                                ← qryinfo.2
<omname>OM1OM </omname>              ← qryinfo.3
...
</ctl)
<cmd>
...
<verb>QRY </verb>
<kwd>TRAN </kwd>
<input>QRY TRAN NAME(CUS*)SHOW(PGM,QCNT) </input>
</cmd>
<cmdrsphdr>
<hdr slbl="TRAN" llbl="Trancode" ... />
<hdr slbl="MBR" llbl="MbrName" ... />
<hdr slbl="CC" llbl="CC" ... />
...
</cmdrsphdr>
<cmdrspdata>
<rsp>TRAN(CUSA ) MBR(IMS1) CC(0) CUSTADD    4 </rsp>
<rsp>TRAN(CUSQ ) MBR(IMS1) CC(0) CUSTQRY    0 </rsp>
</cmsrspdata>
</imsout>
```

Your REXX code must parse XML response to find fields of interest.

Figure 6-28. Sample XML Output CMA01.0

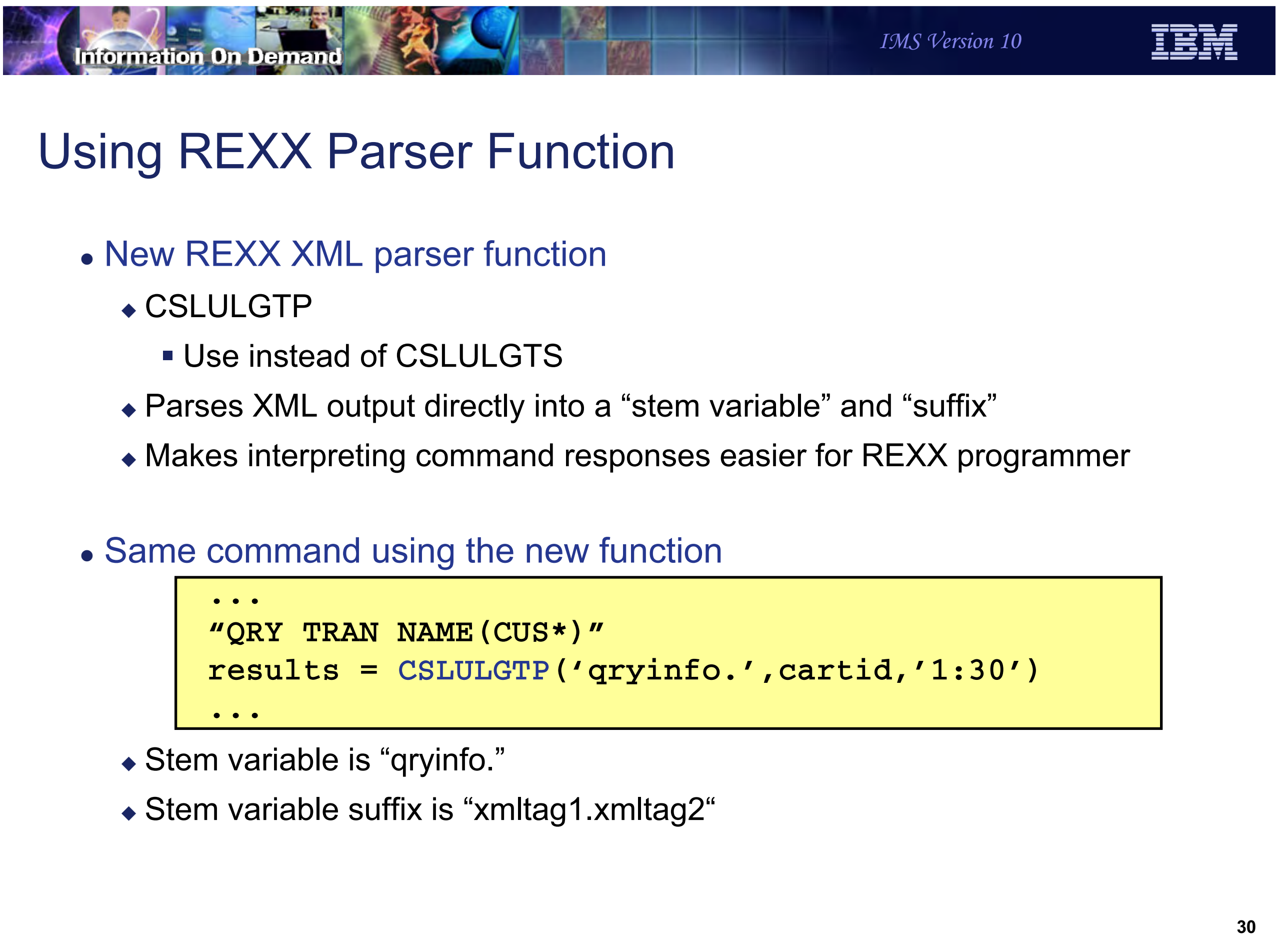
Notes:

This slide shows a sample response to a QRY TRAN command with the XML tags. It is difficult for the average programmer to write code to analyze this response.

- qryinfo.1 = <imsout>
- qryinfo.2 = <ctl>

Somewhere down the line (line n) is the actual response that you are interested in:

- qryinfo.n = <rsp>TRAN*CUSA) MBR(IMS1) CC(0) CUSTADD   4 </rsp>



Using REXX Parser Function

- New REXX XML parser function
  - CSLULGTP
    - Use instead of CSLULGTS
  - Parses XML output directly into a “stem variable” and “suffix”
  - Makes interpreting command responses easier for REXX programmer

- Same command using the new function

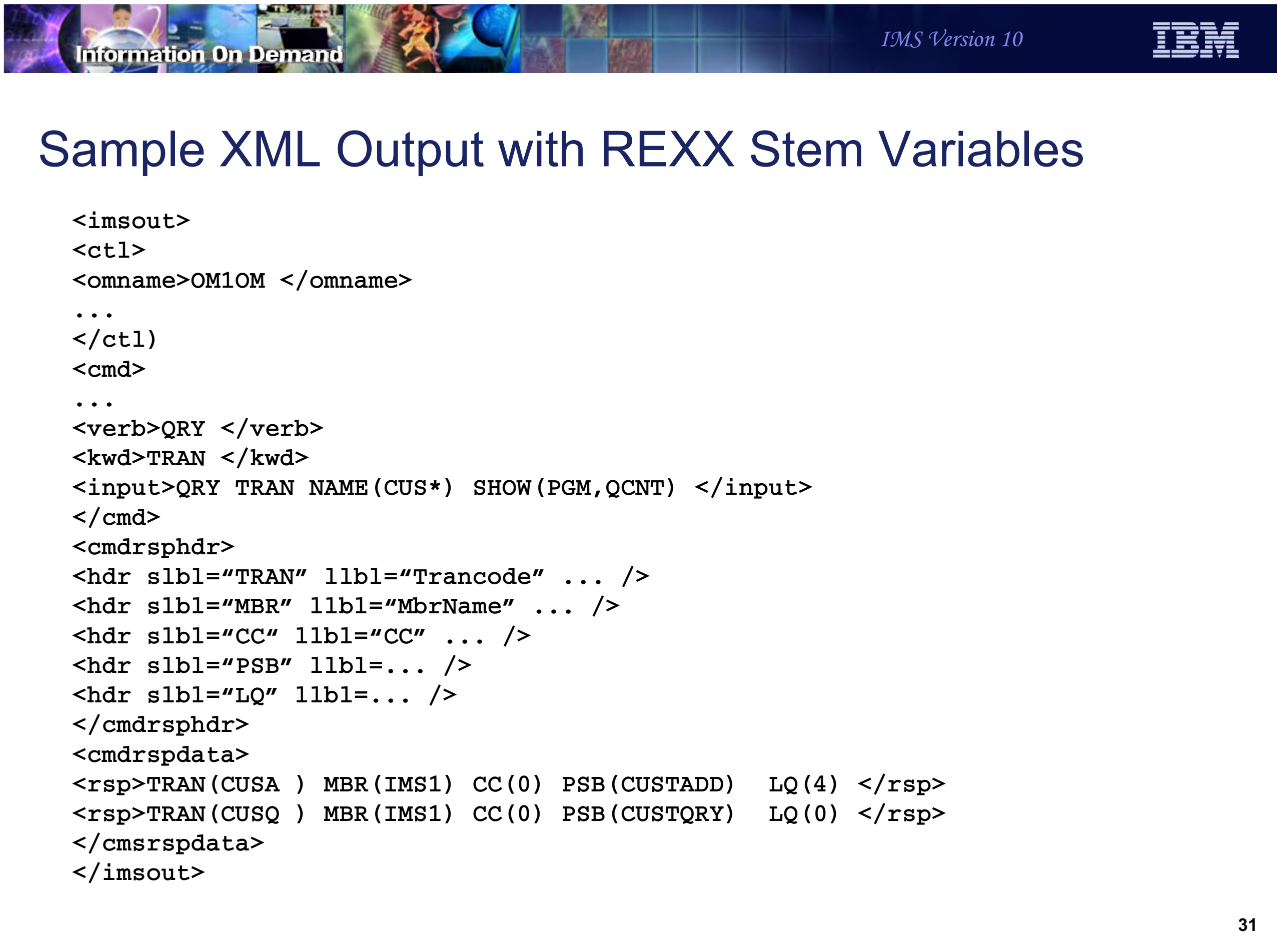
```
...
“QRY TRAN NAME(CUS*)”
results = CSLULGTP(‘qryinfo.’,cartid,’1:30’)
...
```

  - Stem variable is “qryinfo.”
  - Stem variable suffix is “xmltag1.xmltag2“

Figure 6-29. Using REXX Parser Function CMA01.0

Notes:

IMS now provides a REXX parser function that makes it much easier. The new function is CSLULGTP and has the same format as CSLULGTS. The first parameter is the stem variable (qryinfo). What is new is that the function assign values to stem variable suffixes which the programmer can use to find the pertinent information in the response. The suffix is always the xmltag1.xmltag2 where xmltag1 is a high level tag and xmltag2 is an imbedded tag within xmltag1. This will easier to understand with an example.


Information On Demand
IMS Version 10
IBM

Sample XML Output with REXX Stem Variables

```
<imsout>
<ctl>
<omname>OM1OM </omname>
...
</ctl)
<cmd>
...
<verb>QRY </verb>
<kwd>TRAN </kwd>
<input>QRY TRAN NAME(CUS*) SHOW(PGM,QCNT) </input>
</cmd>
<cmdrsphdr>
<hdr slbl=“TRAN” llbl=“Trancode” ... />
<hdr slbl=“MBR” llbl=“MbrName” ... />
<hdr slbl=“CC“ llbl=“CC” ... />
<hdr slbl=“PSB” llbl=... />
<hdr slbl=“LQ” llbl=... />
</cmdrsphdr>
<cmdrspdata>
<rsp>TRAN(CUSA ) MBR(IMS1) CC(0) PSB(CUSTADD)  LQ(4) </rsp>
<rsp>TRAN(CUSQ ) MBR(IMS1) CC(0) PSB(CUSTQRY)  LQ(0) </rsp>
</cmsrspdata>
</imsout>
```

31

Figure 6-30. Sample XML Output with REXX Stem Variables CMA01.0

Notes:

Here is that same XML output again. Note that there are xmltags imbedded within xmltags. For example, with the <cmd> ... </cmd> tags are several other tags - <verb>...</verb>, <kwd>...</kwd>, etc. The suffixes assigned by CSLULGTS are, for example:

- qryinfo.cmd.verb
- qryinfo.cmd.kwd
- qryinfo.cmd.input

Likewise

- qryinfo.ctl.omname is the name of the OM that forwarded the response from IMS

qryinfo is just an example of a stem variable name. You can choose anything you like as the name.

Information On Demand
IMS Version 10

Using REXX Parser Function

- Using previous example
  - qryinfo.ctl.omname = OM1OM
  - qryinfo.cmd.verb = QRY
  - qryinfo.cmd.kwd = TRAN
  - qryinfo.cmd.input = QRY TRAN NAME(CUS*)
- For repeating tags (e.g., the <rsp> tag represents 1 “row” from IMS)
  - qryinfo.rsp.0 = number or rows = 2
  - qryinfo.rsp.1 = 1st row = TRAN(CUSA) MBR(IMS1) CC(0) CUSTADD 4
  - qryinfo.rsp.2 = 2nd row = TRAN(CUSQ) MBR(IMS1) CC(0) CUSTQRY 0
- Within each row
  - qryinfo.rsp.1.0 = number of columns = 5
  - qryinfo.rsp.1.1 = 1st column = TRAN(CUSA)
  - qryinfo.rsp.1.5 = 5th column = 4

Figure 6-31. Using REXX Parser Function CMA01.0

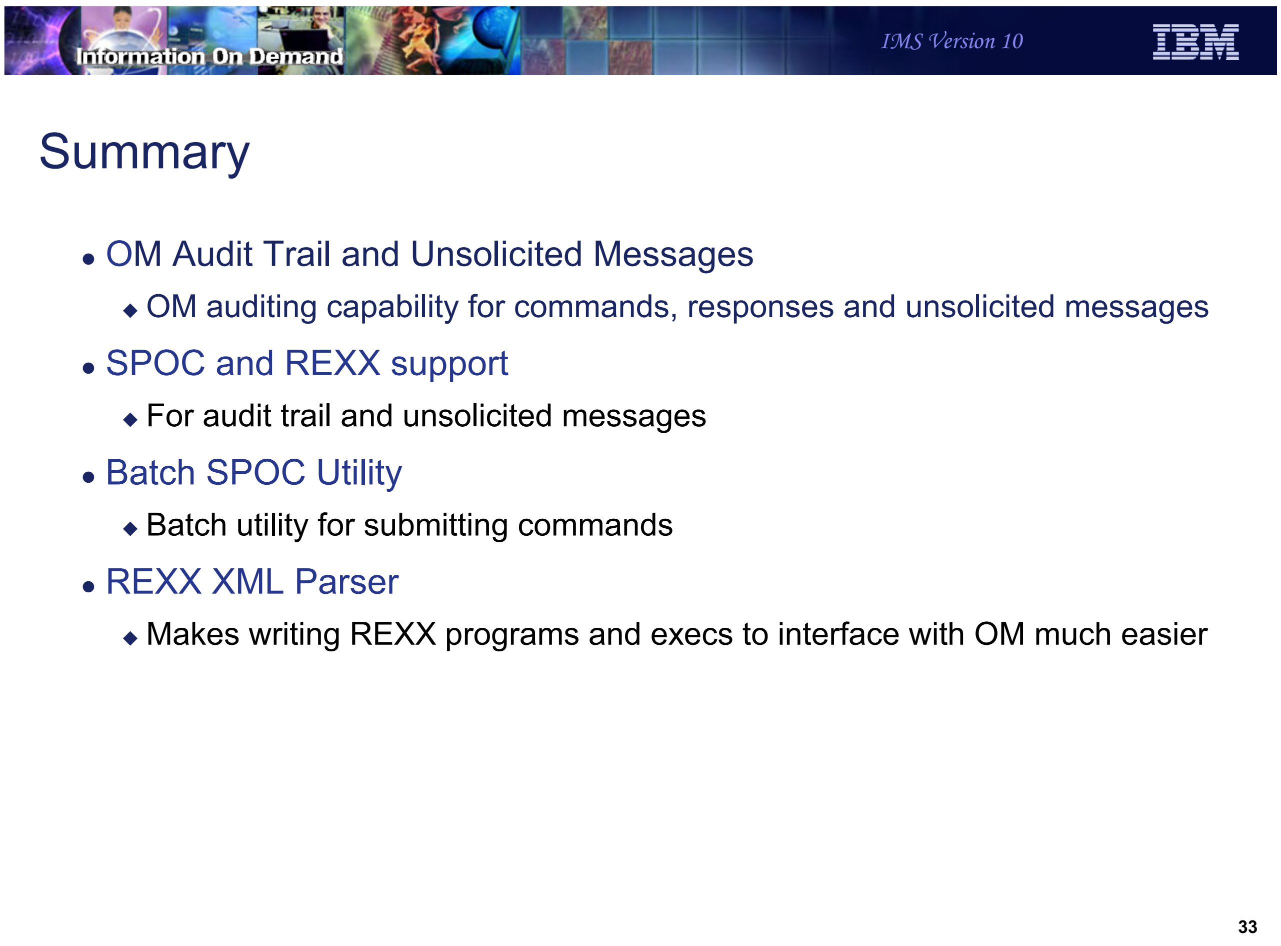
Notes:

Things get a little trickier when there are repeating tags. For example, the query in the previous example resulted in IMS returning several lines of response, each of which is imbedded within the <rsp> ... </rsp> tags. In this case:

- qryinfo.rsp.0 would be the number of rows
- qryinfo.rsp.1 would be the entire 1st row

If you want to parse each “column” in a “row”, then:

- qryinfo.rsp.1.0 is the number of columns in that row
- qryinfo.rsp.2.3 would be the 3rd column in row 2


Information On Demand
IMS Version 10
IBM

Summary

- OM Audit Trail and Unsolicited Messages
  - OM auditing capability for commands, responses and unsolicited messages
- SPOC and REXX support
  - For audit trail and unsolicited messages
- Batch SPOC Utility
  - Batch utility for submitting commands
- REXX XML Parser
  - Makes writing REXX programs and execs to interface with OM much easier

Figure 6-32. Summary CMA01.0

Notes:

Enhancements in SPOC make the use of CSL and OM much more attractive to even the single IMS user.

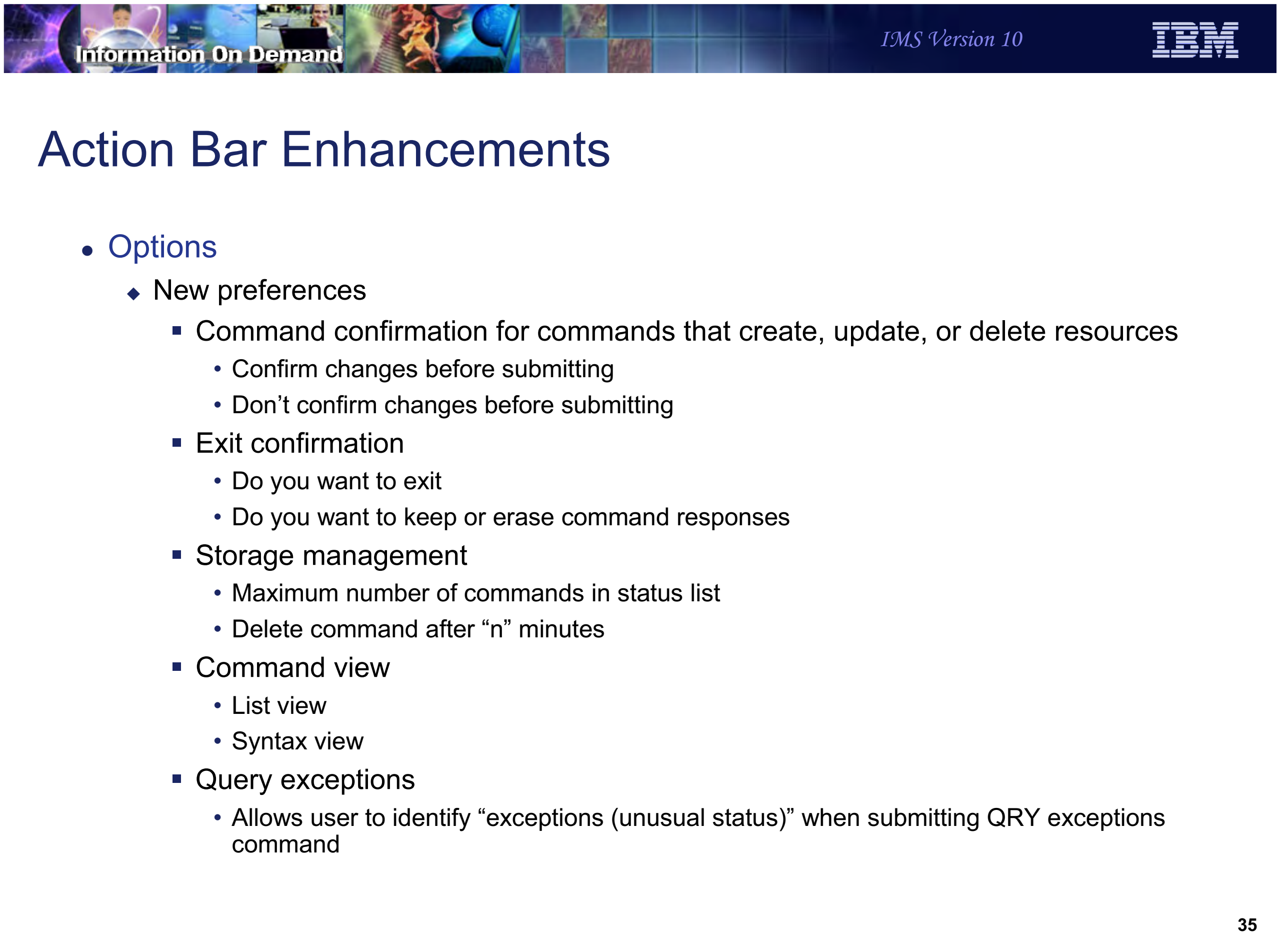
Enhancements in IMS support for REXX makes writing your own AO programs much more feasible.

TSO SPOC Appendix

Figure 6-33. [No title] CMA01.0

Notes:

This section addresses an XML parser that can be used with REXX programs to process command responses from IMS.


Information On Demand
IMS Version 10
IBM

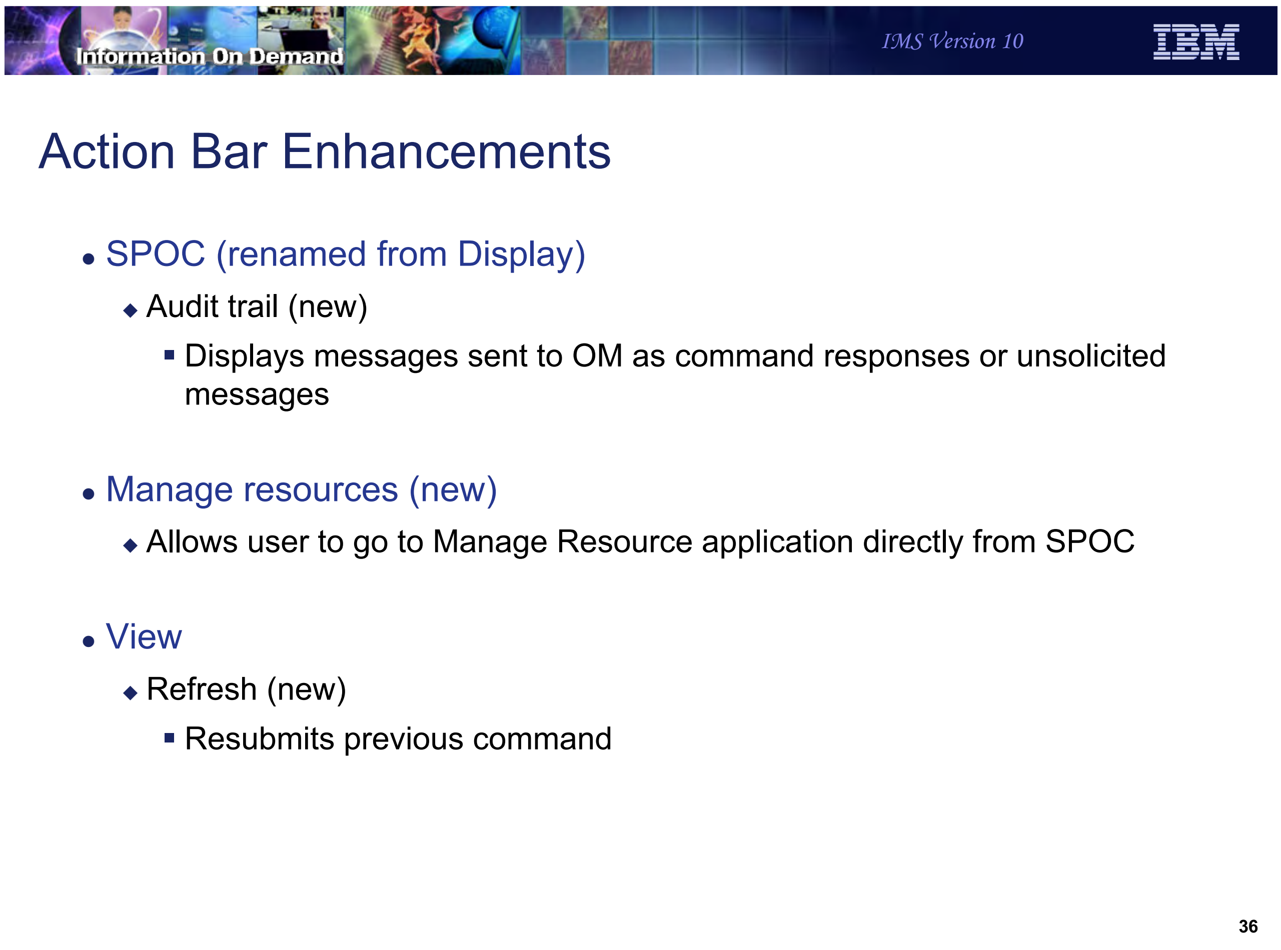
Action Bar Enhancements

- Options
  - New preferences
    - Command confirmation for commands that create, update, or delete resources
      - Confirm changes before submitting
      - Don’t confirm changes before submitting
    - Exit confirmation
      - Do you want to exit
      - Do you want to keep or erase command responses
    - Storage management
      - Maximum number of commands in status list
      - Delete command after “n” minutes
    - Command view
      - List view
      - Syntax view
    - Query exceptions
      - Allows user to identify “exceptions (unusual status)” when submitting QRY exceptions command

Figure 6-34. Action Bar Enhancements CMA01.0

Notes:

There are new “Preferences” that can be chosen. Command Confirmation, Exit Confirmation, Storage Management, Command View, and Query Exceptions. The impact of each of these will be shown in later slides.



Information On Demand
IMS Version 10
IBM

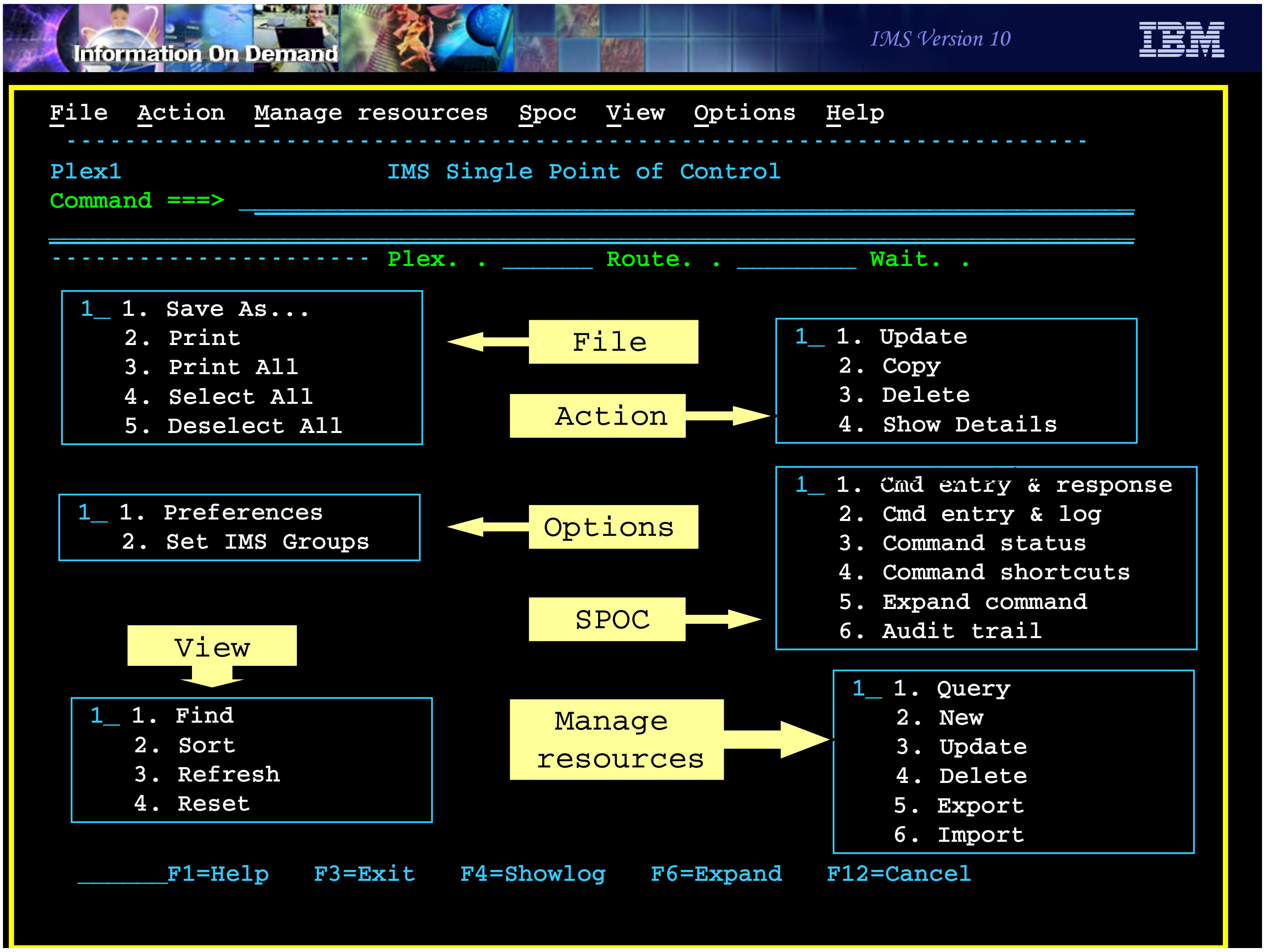
Action Bar Enhancements

- SPOC (renamed from Display)
  - Audit trail (new)
    - Displays messages sent to OM as command responses or unsolicited messages
- Manage resources (new)
  - Allows user to go to Manage Resource application directly from SPOC
- View
  - Refresh (new)
    - Resubmits previous command

Figure 6-35. Action Bar Enhancements CMA01.0

Notes:

- SPOC is renamed from DISPLAY of earlier releases of SPOC. There is one new option to display the Audit Trail created by OM for command input and responses and unsolicited messages.
- Manage resources is new. It invokes an ISPF application similar to SPOC which can be used to manage resources using DRD. This will also be discussed in detail later.
- View has a new option. “Refresh” resubmits the previous command.



```
File  Action  Manage resources  Spoc  View  Options  Help
 ----------------------------------------------------------------------
Plex1                      IMS Single Point of Control
Command ===> ____________________________________________________________

---------------------- Plex. . ______ Route. . ________ Wait. .
```

File:
```
1_ 1. Save As...
   2. Print
   3. Print All
   4. Select All
   5. Deselect All
```

Action:
```
1_ 1. Update
   2. Copy
   3. Delete
   4. Show Details
```

Options:
```
1_ 1. Preferences
   2. Set IMS Groups
```

SPOC:
```
1_ 1. Cmd entry & response
   2. Cmd entry & log
   3. Command status
   4. Command shortcuts
   5. Expand command
   6. Audit trail
```

View:
```
1_ 1. Find
   2. Sort
   3. Refresh
   4. Reset
```

Manage resources:
```
1_ 1. Query
   2. New
   3. Update
   4. Delete
   5. Export
   6. Import
```

```
 ______F1=Help   F3=Exit   F4=Showlog   F6=Expand   F12=Cancel
```

Figure 6-36. [No title] CMA01.0

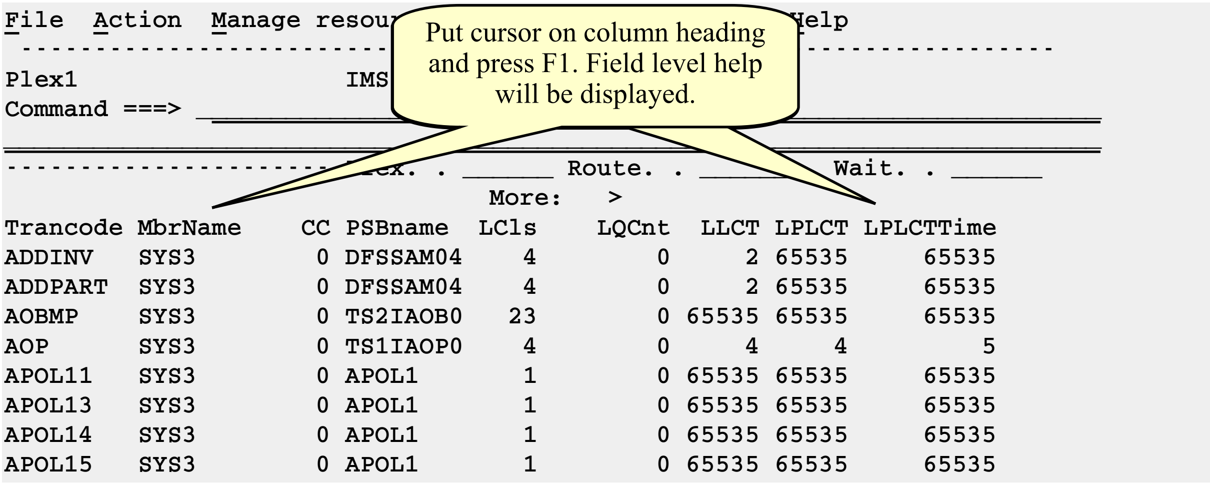
Notes:

This screen shot shows what each of these pull down menus from the Action Bar would look like.

Information On Demand
IMS Version 10
IBM

Improved Help

- Field level help added to QRY display screen
  - Put cursor on column heading and press F1



```
File  Action  Manage resou                                Help
 -------------------------                          ------------------
Plex1                   IMS
Command ===> _____________________________________________________________

==========================================================================
---------------------- ex. . ______ Route. . ______  Wait. . ______
                                More:    >
Trancode MbrName    CC PSBname  LCls    LQCnt   LLCT LPLCT LPLCTTime
ADDINV   SYS3        0 DFSSAM04    4        0      2 65535     65535
ADDPART  SYS3        0 DFSSAM04    4        0      2 65535     65535
AOBMP    SYS3        0 TS2IAOB0   23        0  65535 65535     65535
AOP      SYS3        0 TS1IAOP0    4        0      4     4         5
APOL11   SYS3        0 APOL1       1        0  65535 65535     65535
APOL13   SYS3        0 APOL1       1        0  65535 65535     65535
APOL14   SYS3        0 APOL1       1        0  65535 65535     65535
APOL15   SYS3        0 APOL1       1        0  65535 65535     65535
```

Put cursor on column heading and press F1. Field level help will be displayed.

Figure 6-37. Improved Help CMA01.0

Notes:

Field level help is now available for columns on the QRY result screen. Place the cursor on the column heading and press F1. A description of the meaning of that field will be displayed.

Support for OM Audit Trail

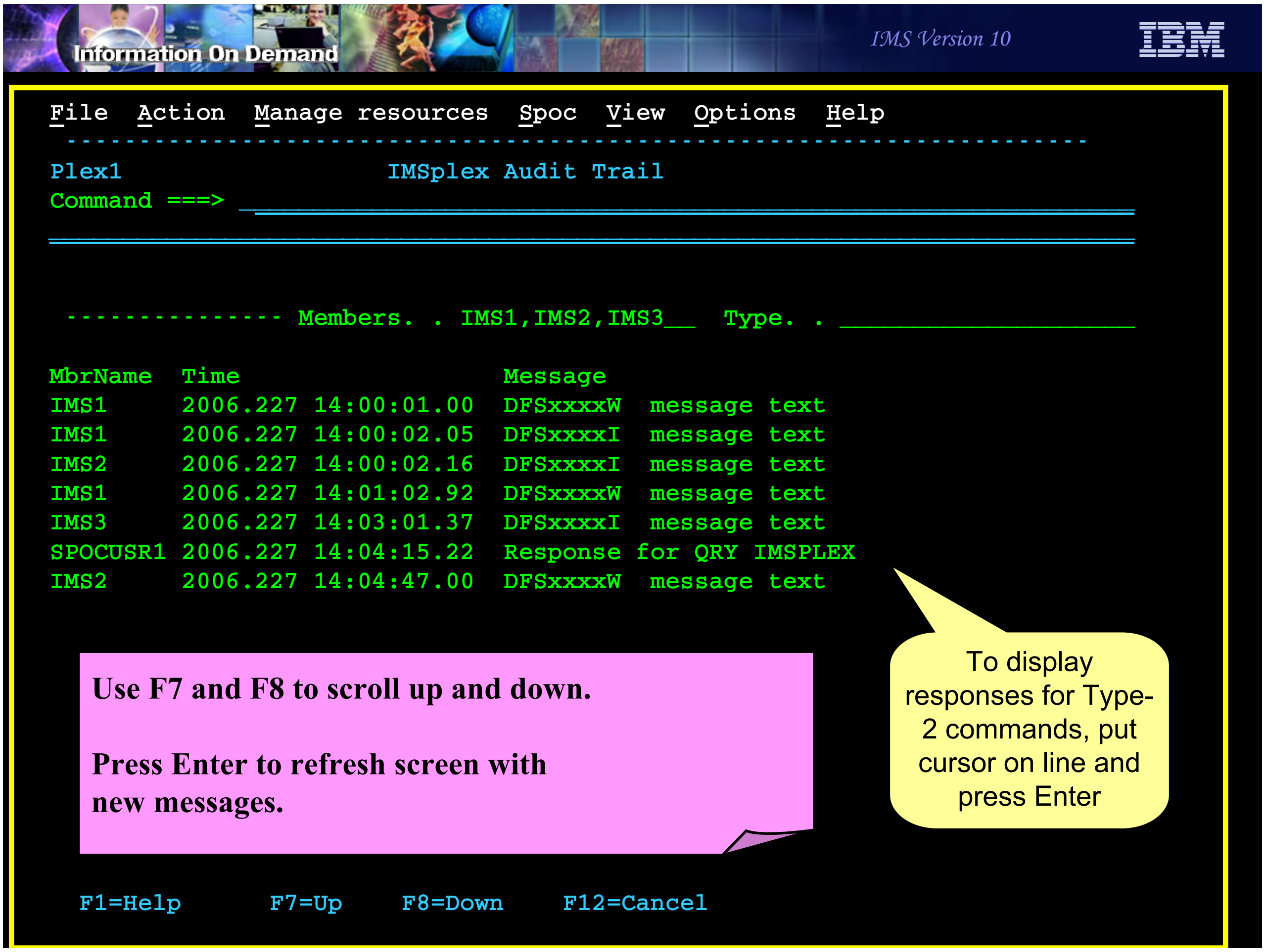
- Operations Manager provides support in V10 for an “audit trail” of
  - Commands and responses entered through OM interface
  - Unsolicited messages from IMSplex members

- SPOC provides support for displaying this audit trail
  - Preference screen
    - IMSplex member names or types
  - Display
    - Message written to “audit trail log stream” by OM
    - Read directly from log stream
  - Details earlier in this section of class

Figure 6-38. Support for OM Audit Trail CMA01.0

Notes:

Choosing Option 6 in the SPOC pull down menu invokes the SPOC Audit Trail display function.


Information On Demand
IMS Version 10
File Action Manage resources Spoc View Options Help
F1=Help F7=Up F8=Down F12=Cancel

```
Plex1                  IMSplex Audit Trail
Command ===> ____________________________________________________________

 --------------- Members. . IMS1,IMS2,IMS3__  Type. . ____________________

MbrName   Time                    Message
IMS1      2006.227 14:00:01.00   DFSxxxxW  message text
IMS1      2006.227 14:00:02.05   DFSxxxxI  message text
IMS2      2006.227 14:00:02.16   DFSxxxxI  message text
IMS1      2006.227 14:01:02.92   DFSxxxxW  message text
IMS3      2006.227 14:03:01.37   DFSxxxxI  message text
SPOCUSR1  2006.227 14:04:15.22   Response for QRY IMSPLEX
IMS2      2006.227 14:04:47.00   DFSxxxxW  message text
```

Use F7 and F8 to scroll up and down.

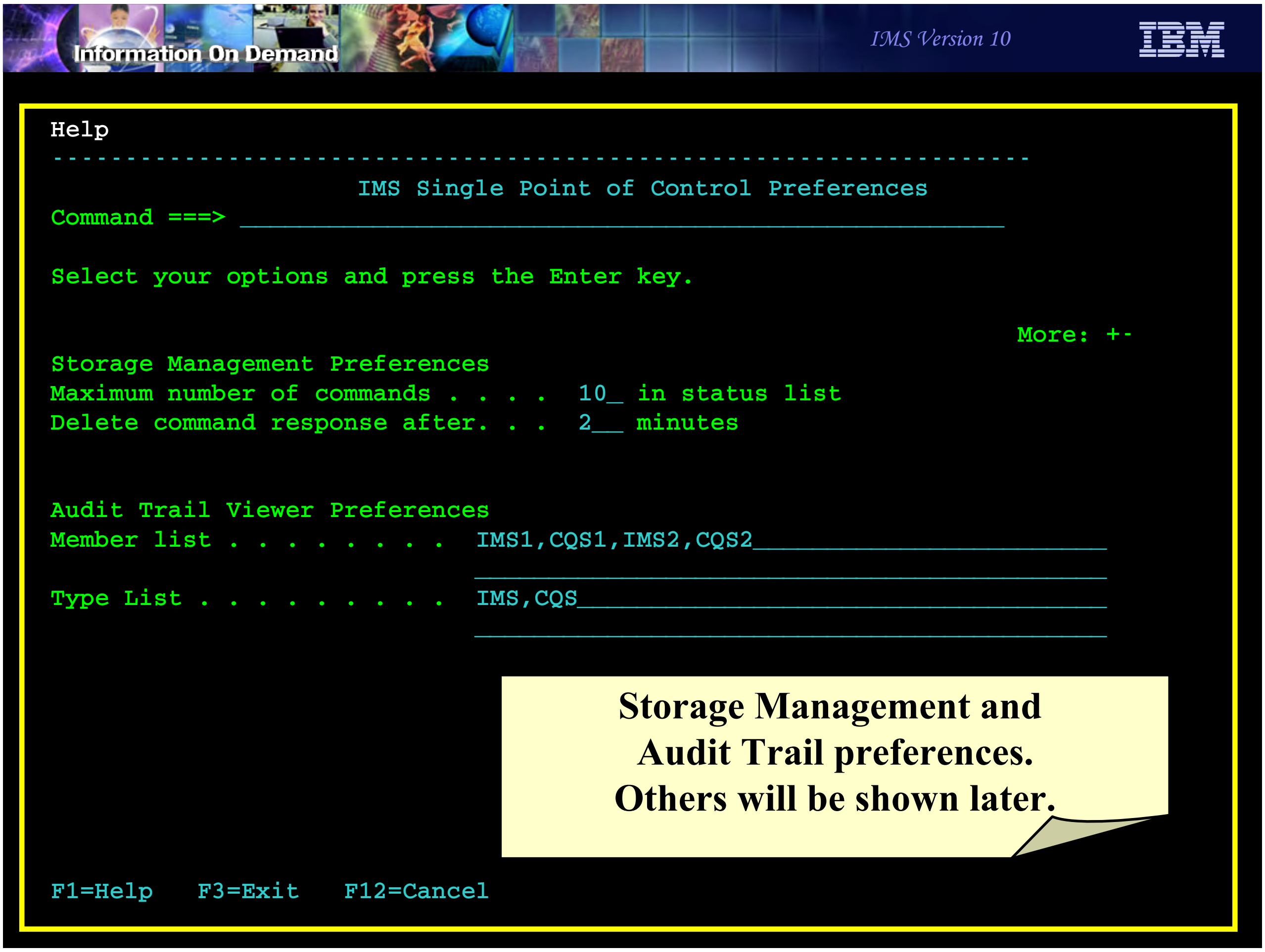
Press Enter to refresh screen with new messages.

To display responses for Type-2 commands, put cursor on line and press Enter

Figure 6-39. [No title] CMA01.0

Notes:

This is an example of a screen shot of an Audit Trail display. Note that Type-2 command responses may be seen by placing the cursor on the line and pressing Enter.



```
Help
------------------------------------------------------------------
                     IMS Single Point of Control Preferences
Command ===> ____________________________________________________

Select your options and press the Enter key.

                                                                 More: +-
Storage Management Preferences
Maximum number of commands . . . .  10_ in status list
Delete command response after. . .  2__ minutes


Audit Trail Viewer Preferences
Member list . . . . . . . . .  IMS1,CQS1,IMS2,CQS2________________________
                               ___________________________________________
Type List . . . . . . . . . .  IMS,CQS____________________________________
                               ___________________________________________








F1=Help   F3=Exit   F12=Cancel
```

Storage Management and
Audit Trail preferences.
Others will be shown later.

Figure 6-40. [No title] CMA01.0

Notes:

This screen shows the Storage Management Preferences

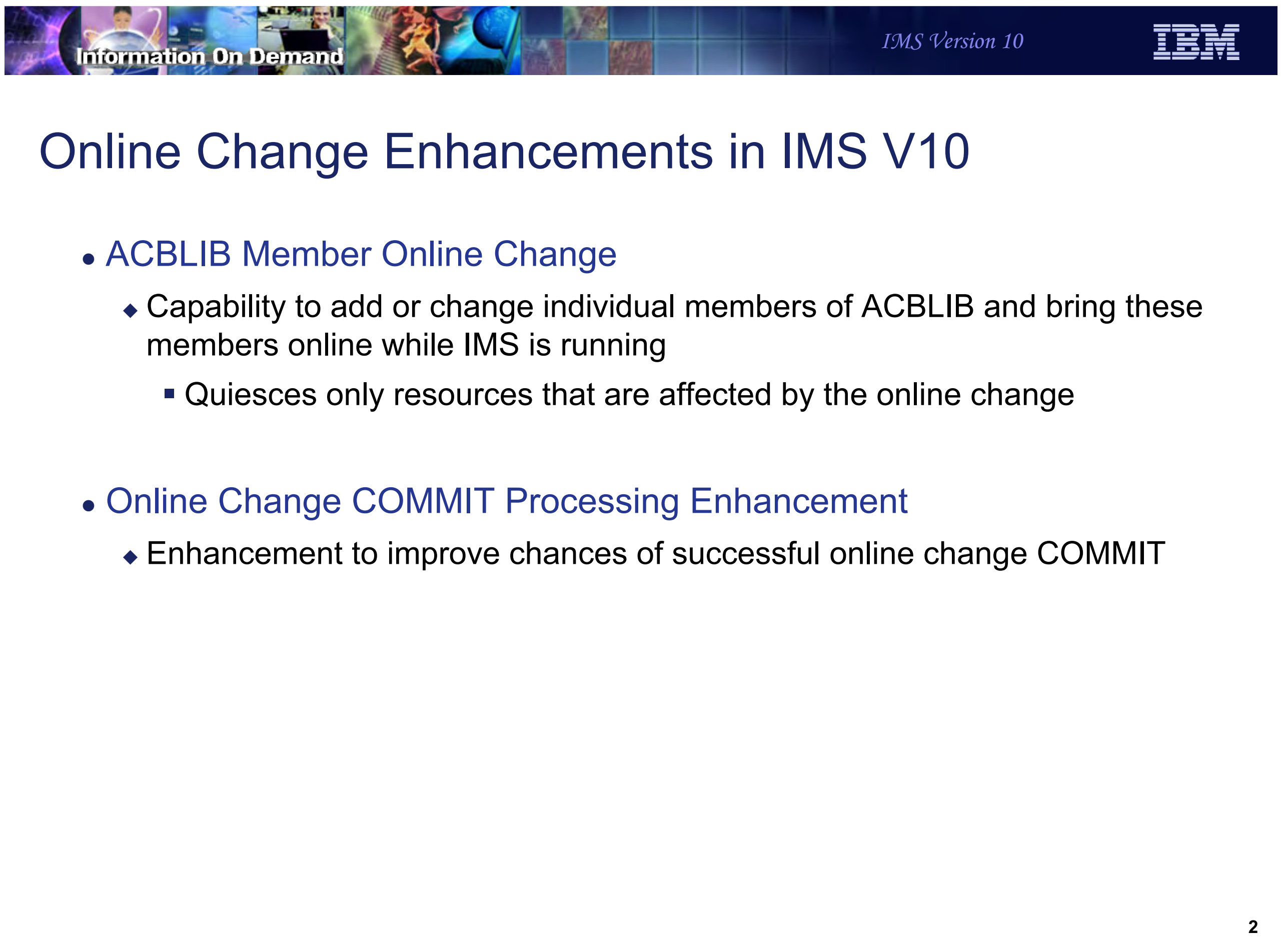
- What is the maximum number of commands and their responses to keep in the status list
- How long should a command/response stay in the status list before being deleted

Audit Trail Preferences

- Member list of messages to display
- Type list of messages to display

If both have values, the member list takes precedence

Unit 7. Online Change



Information On Demand
IMS Version 10
IBM

Online Change Enhancements in IMS V10

- ACBLIB Member Online Change
  - Capability to add or change individual members of ACBLIB and bring these members online while IMS is running
    - Quiesces only resources that are affected by the online change
- Online Change COMMIT Processing Enhancement
  - Enhancement to improve chances of successful online change COMMIT

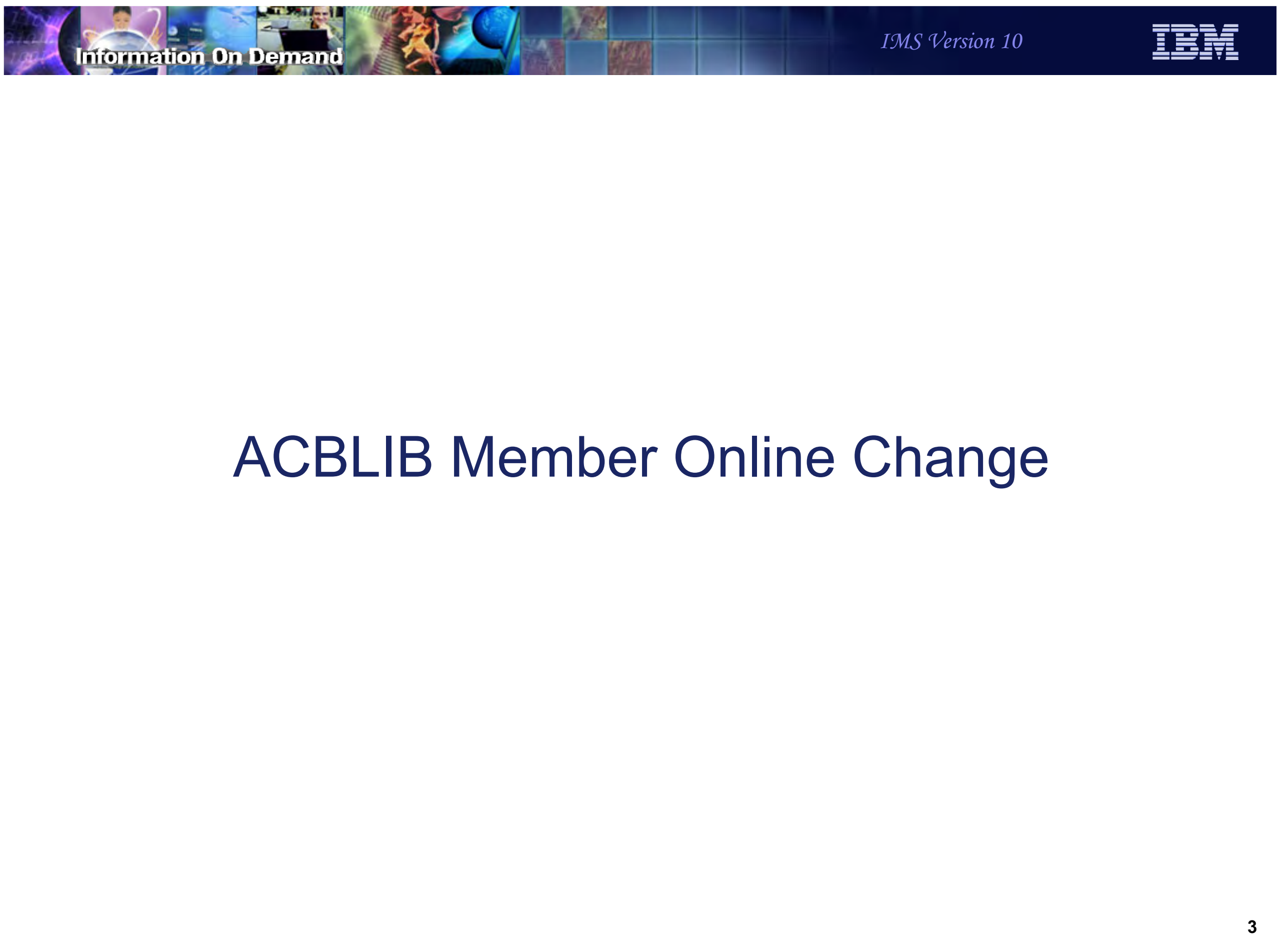
Figure 7-1. Online Change Enhancements in IMS V10 CMA01.0

Notes:

IMS Version 10 provides two enhancements for online change processing:

This first is ACBLIB Member Online Change. This is the capability to add or change individual members of ACBLIB without a switch of ACBLIBs. The new or changed members are brought online while IMS is running.

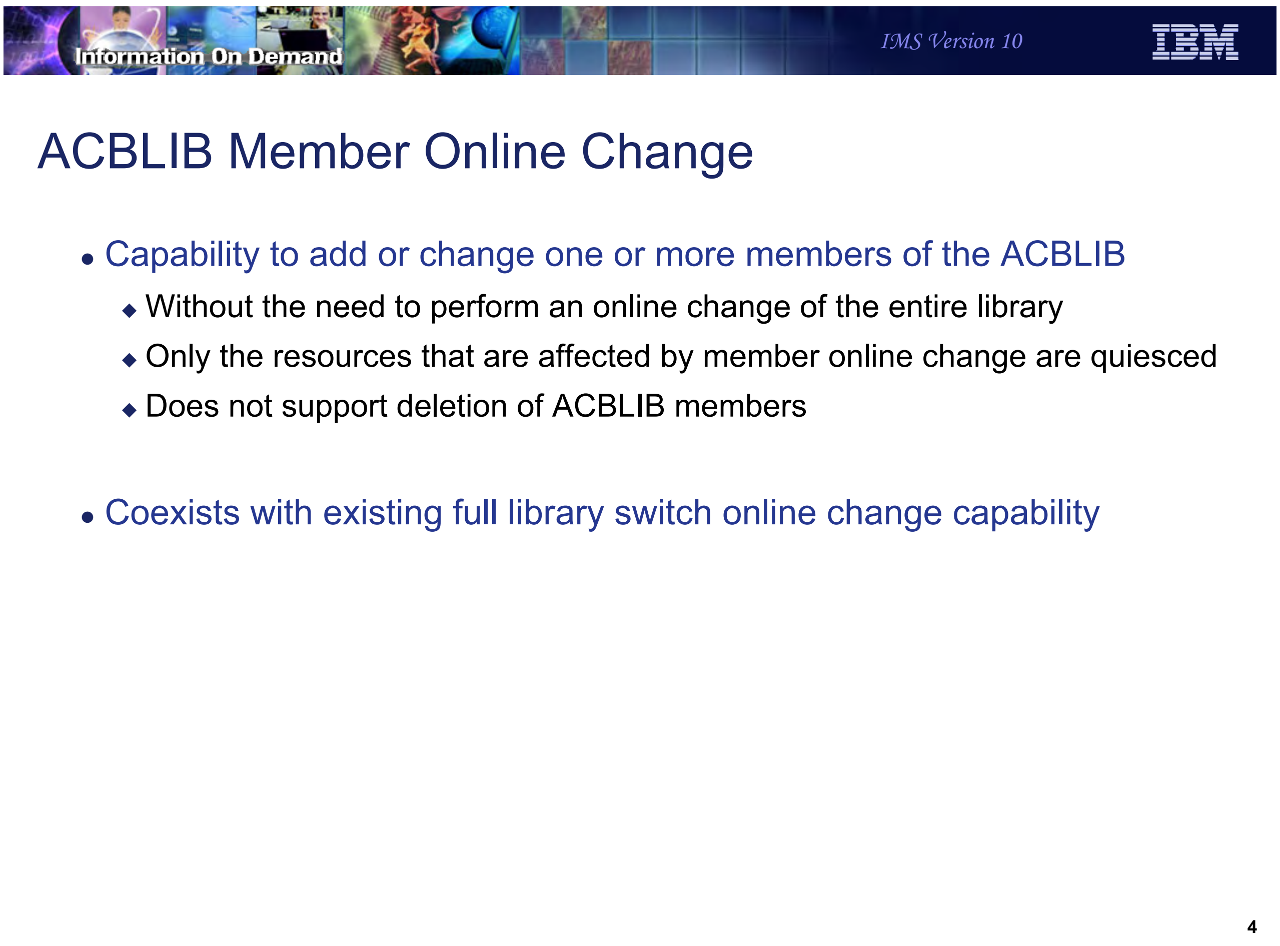
The second is a change in Online Change commit processing. This improves the chances for the success of commit processing by reducing the number of situations where commit will fail.


Information On Demand
IMS Version 10
IBM

ACBLIB Member Online Change

Figure 7-2. [No title] CMA01.0

Notes:


Information On Demand
IMS Version 10

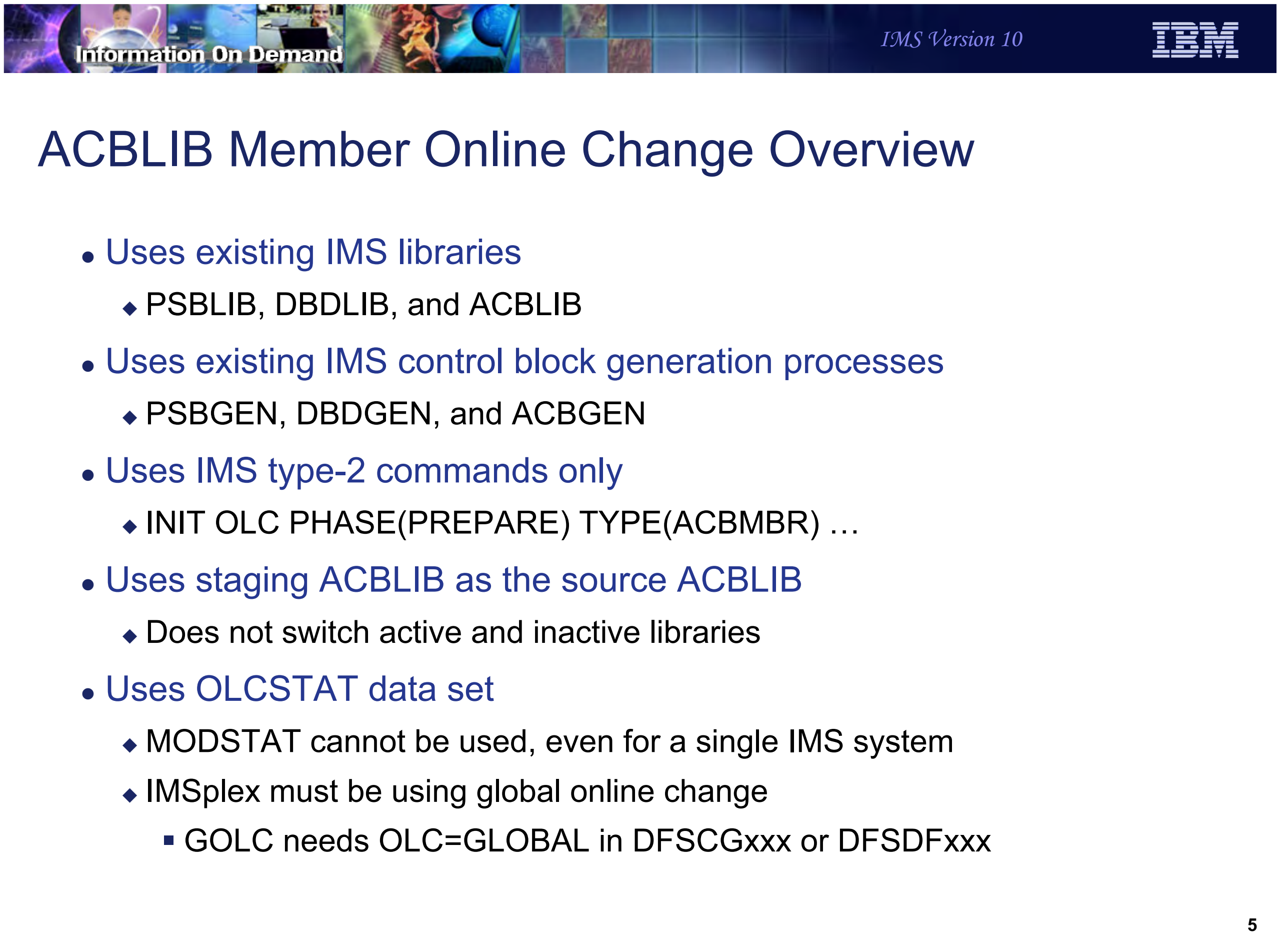
ACBLIB Member Online Change

- Capability to add or change one or more members of the ACBLIB
  - Without the need to perform an online change of the entire library
  - Only the resources that are affected by member online change are quiesced
  - Does not support deletion of ACBLIB members

- Coexists with existing full library switch online change capability

Figure 7-3. ACBLIB Member Online Change CMA01.0

Notes:



Information On Demand
IMS Version 10
IBM

ACBLIB Member Online Change Overview

- Uses existing IMS libraries
  - PSBLIB, DBDLIB, and ACBLIB
- Uses existing IMS control block generation processes
  - PSBGEN, DBDGEN, and ACBGEN
- Uses IMS type-2 commands only
  - INIT OLC PHASE(PREPARE) TYPE(ACBMBR) …
- Uses staging ACBLIB as the source ACBLIB
  - Does not switch active and inactive libraries
- Uses OLCSTAT data set
  - MODSTAT cannot be used, even for a single IMS system
  - IMSplex must be using global online change
    - GOLC needs OLC=GLOBAL in DFSCGxxx or DFSDFxxx

Figure 7-4. ACBLIB Member Online Change Overview CMA01.0

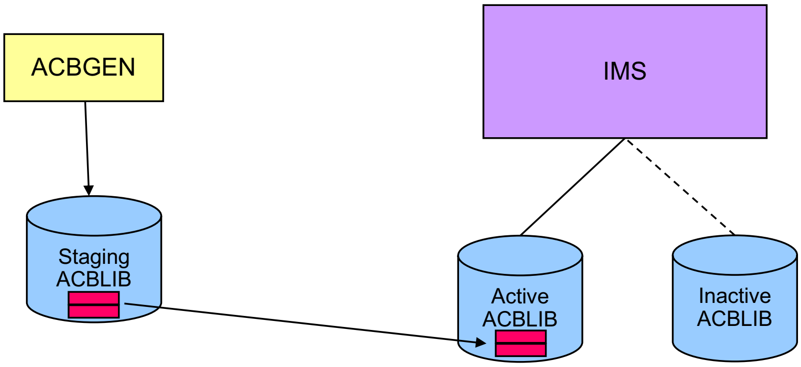
Notes:

TYPE(ACBMBR) on the INIT OLC PHASE(PREPARE) command indicates that this is an ACBLIB member online change. Since the INIT OLC command is used, an OLCSTAT data set and a CSL environment are required. OLC=GLOBAL must be specified either in the DFSCGxxx PROCLIB member or the DFSDFxxx PROCLIB member.

Information On Demand IMS Version 10 IBM

ACBLIB Member Online Change Overview

- Selected members of a staging library are copied to the active ACBLIB


ACBGEN
IMS
Staging ACBLIB
Active ACBLIB
Inactive ACBLIB

6

Figure 7-5. ACBLIB Member Online Change Overview CMA01.0

Notes:

This is a pictorial overview of the ACBLIB member online change process. First, an ACBGEN is done into a staging library. The online change process copies theselected members from the staging library to the active ACBLIB while the IMS system is running.

Step 1: Build updated member(s)

- Update PSB source and/or DBD source
- Perform PSBGEN/DBDGEN for affected members
- Perform ACBGEN into the staging ACBLIB library for affected members
  - BLDPSB=YES is required on BUILD DBD= statement
    - Rebuilds all PSBs referenced by DBDs in BUILD statement
    - New parameter in IMS Version 10
    - BLDPSB=YES is the default

```
BUILD DBD=(CUSTOMER,ORDER),BLDPSB=YES
```

Figure 7-6. Step 1: Build updated member(s) CMA01.0

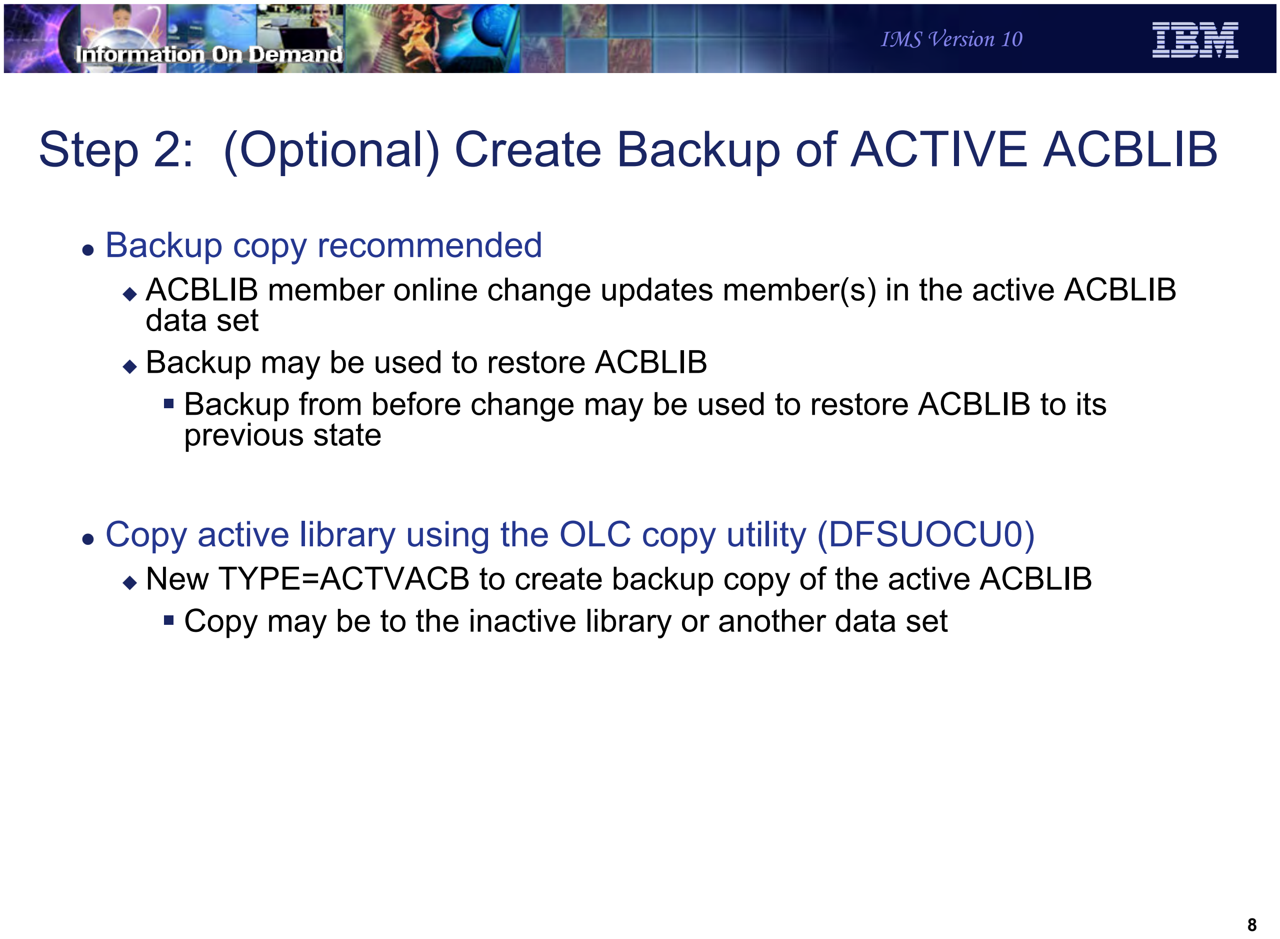
Notes:

The BLDPSB=YES option is new with IMS V10 and ACBLIB member online change. It indicates that ACBGEN rebuilds all PSBs that reference the changed DBD on the BUILD DBD=dbdname statement into the staging ACBLIB. Member online change requires that all PSBs be rebuilt (flag set in PDIR) or it will fail. For performance reasons, these relationships are found during the execution of ACBGEN in batch versus having to discover these relationships online when ACBLIB member online change is invoked. However, this means there may be additional processing done at ACBGEN time and the ACBGEN utility may run longer. BLDPSB=YES is the default.

In previous IMS versions, ACBGEN worked as if BLDPSB=NO was specified. BLDPSB=NO indicates that ACBGEN does not rebuild PSBs that reference the changed DBD if the changed DBD does not change the physical structure of the DBD. The DBD is rebuilt but PSBs are not necessarily rebuilt (ex. exit/randomizer change). You can still specify BLDPSB=NO if you are not planning to use ACBLIB member online change.

BLDPSB=NO is not the default, so this operates differently than in previous versions. It must be specified if you want this capability.

The ACBGEN utility does not clear out the staging ACBLIB. The ACBGEN utility only updates the members in the staging ACBLIB that are specified in the BUILD DBD and BUILD PSB statements.



Information On Demand

IMS Version 10

Step 2: (Optional) Create Backup of ACTIVE ACBLIB

- Backup copy recommended
  - ACBLIB member online change updates member(s) in the active ACBLIB data set
  - Backup may be used to restore ACBLIB
    - Backup from before change may be used to restore ACBLIB to its previous state
- Copy active library using the OLC copy utility (DFSUOCU0)
  - New TYPE=ACTVACB to create backup copy of the active ACBLIB
    - Copy may be to the inactive library or another data set

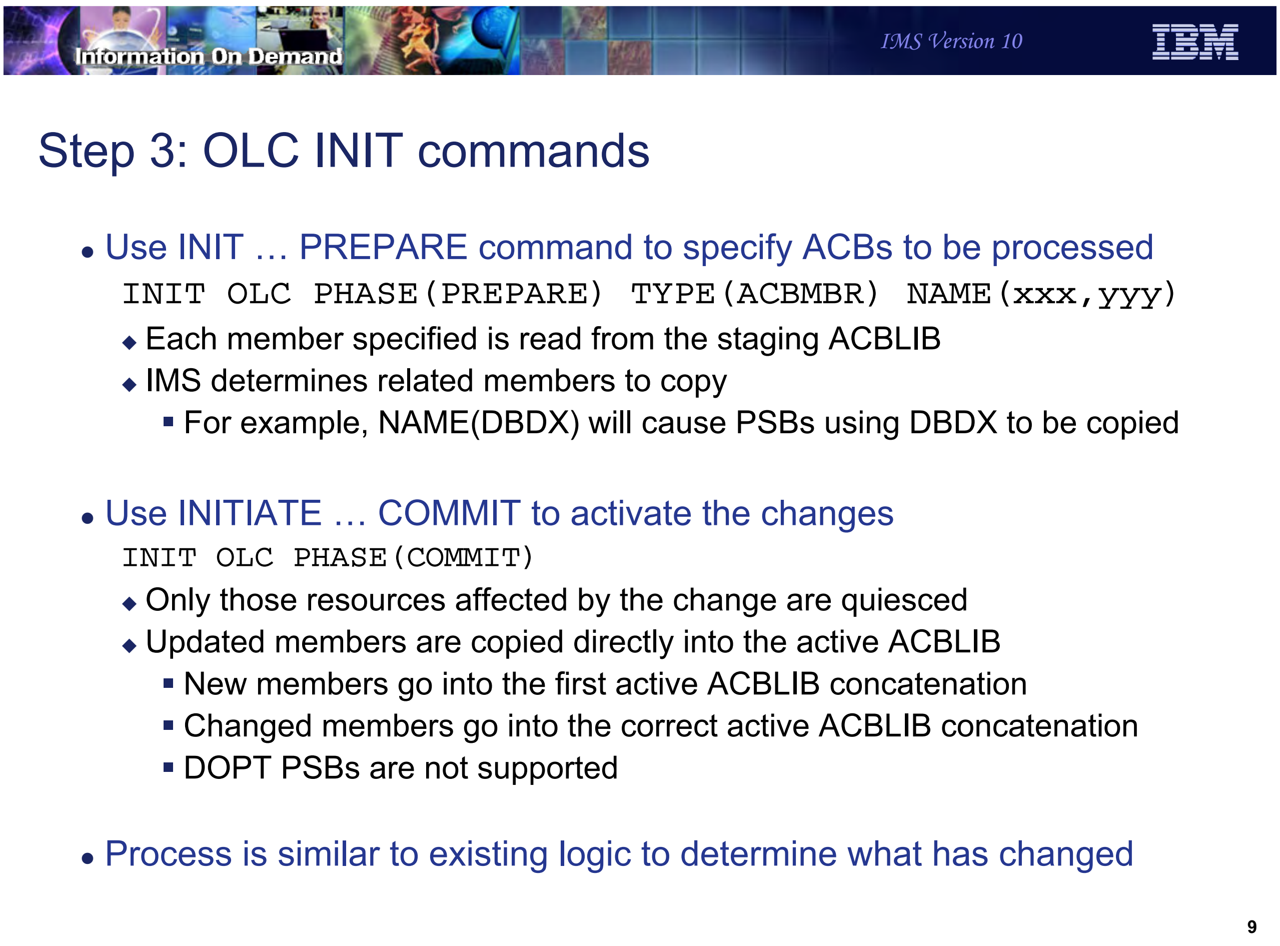
Figure 7-7. Step 2: (Optional) Create Backup of ACTIVE ACBLIB CMA01.0

Notes:

There is a new option on the OLC copy utility (DFSUOCU0) called TYPE=ACTVACB. This copies the active ACBLIB to another data set. This may be the inactive ACBLIB or another data set. The backup can be used in the event of a problem with ACBLIB member online change. If the copy is made to the inactive ACBLIB, a full library switch can be used to restore the ACBLIB to its previous state.

The ACTIVE ACBLIB is copied to the ACBLIB specified on the IMSACBO DD card in the OLCUTL procedure by specifying OUT=O. The ACTIVE ACBLIB is copied to the INACTIVE ACBLIB by specifying OUT=G.

If you have an IMSplex and you use DFSUOCU0 to copy the active ACBLIB to the inactive, your procedures will depend on your configuration. In an IMSplex where IMS subsystems share the active and inactive ACBLIBs, DFSUOCU0 needs to be executed on only one IMS in the IMSplex. In an IMSplex where IMS subsystems do not share ACBLIBs, DFSUOCU0 needs to be executed on every IMS in the IMSplex


Information On Demand
IMS Version 10
IBM

Step 3: OLC INIT commands

- Use INIT … PREPARE command to specify ACBs to be processed

```
INIT OLC PHASE(PREPARE) TYPE(ACBMBR) NAME(xxx,yyy)
```

  - Each member specified is read from the staging ACBLIB
  - IMS determines related members to copy
    - For example, NAME(DBDX) will cause PSBs using DBDX to be copied
- Use INITIATE … COMMIT to activate the changes

```
INIT OLC PHASE(COMMIT)
```

  - Only those resources affected by the change are quiesced
  - Updated members are copied directly into the active ACBLIB
    - New members go into the first active ACBLIB concatenation
    - Changed members go into the correct active ACBLIB concatenation
    - DOPT PSBs are not supported
- Process is similar to existing logic to determine what has changed

Figure 7-8. Step 3: OLC INIT commands CMA01.0

Notes:

TYPE(ACBMBR) is used to specify an ACBLIB member online change. It cannot be specified with TYPE(ACBLIB,FMTLIB,MODBLKS) in the same INIT OLC PHASE(PREPARE) command. TYPE(ACBLIB,FMTLIB,MODBLKS) and TYPE(ALL) are for full library switch online change and the inactive ACBLIB becomes the active ACBLIB. TYPE(ACBMBR) is for ACBLIB member online change. The active ACBLIB remains the active ACBLIB an the source ACBLIB is the staging ACBLIB.

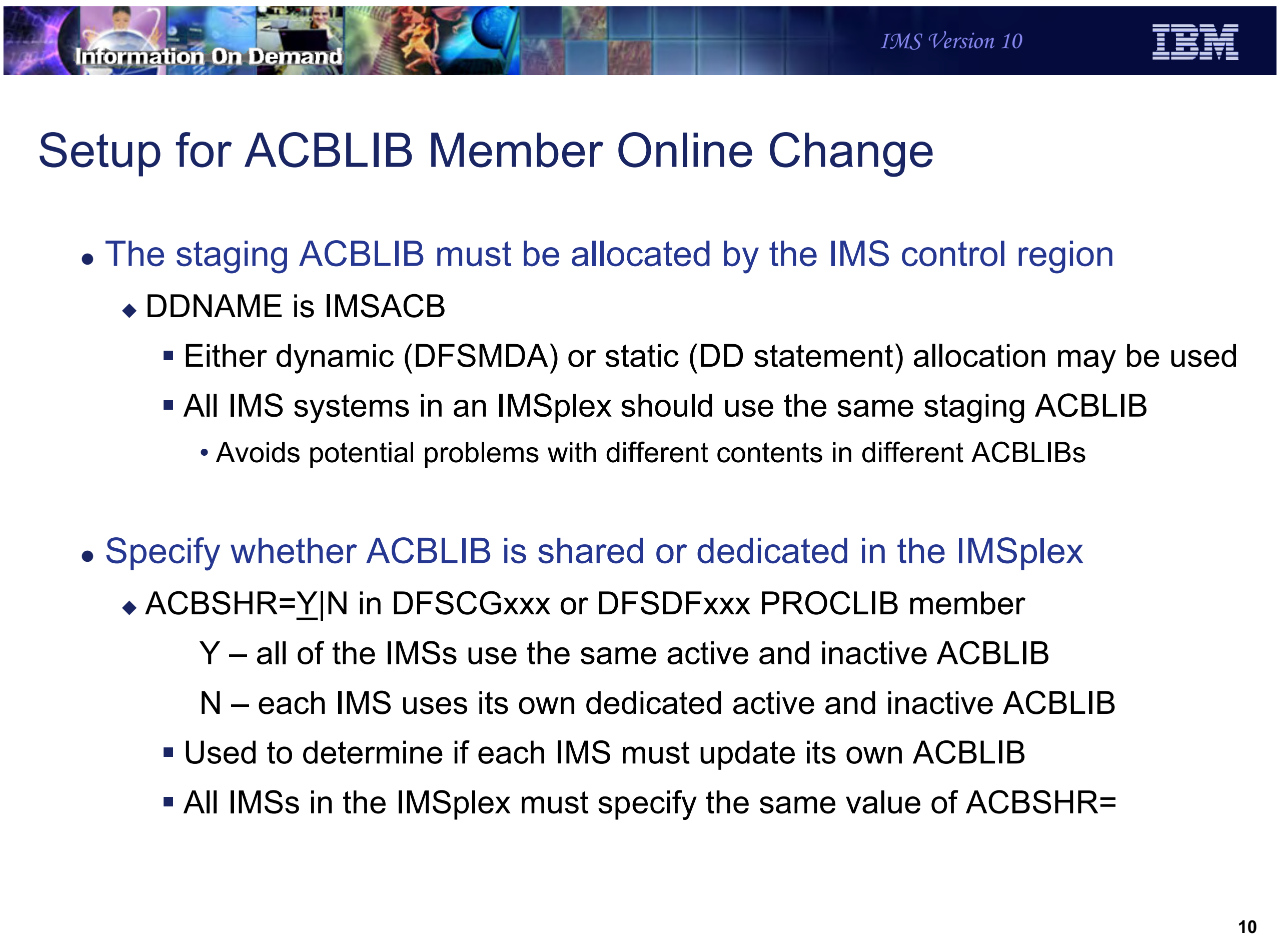
You need only specify the ACB member(s) with changes in the INIT OLC PHASE(PREPARE) command; IMS will find all the necessary members that are associated with the specified member(s). For example, there is an update to DBDX and an ACBGEN BUILD DBD=DBDX,BLDPSB=Y is done. DBDX is used by members PSBA, PSBB, and PSBC, so these PSBs are rebuilt by ACBGEN. The following command can be entered to bring member DBDX online, along with members PSBA, PSBB, and PSBC:

```
INIT OLC PHASE(PREPARE) TYPE (ACBMBR) NAME(DBDX)
```

Dynamic Option PSBs (DOPT) are not supported by ACBLIB member OLC. Therefore when ACBLIB member OLC processes a changed DBD that references a DOPT PSB, it does not copy the DOPT PSB from the staging ACBLIB to the active ACBLIB, though the changed DBD referenced by this DOPT PSB will be copied to the active ACBLIB. The INITIATE OLC PREPARE command output shows for the DOPT PSB entry a completion code indicating the DOPT PSB will not be copied to the active ACBLIB; COMMIT processing will still be successful if this occurs.

If a GPSB is specified in the INIT OLC PHASE(PREPARE) command, it will be copied to the active ACBLIB and its TTR address will not be refreshed. However, IMS only uses its own GPSB skeletal PSB and will not use the one in the ACBLIB.

If the staging ACBLIB is a subset of the active ACBLIB, the member OLC process reads only the directory entries for those ACB members in the staging ACBLIB. If the staging ACBLIB is a clone of the active ACBLIB, the member OLC process reads all the directory entries in the staging ACBLIB.



Setup for ACBLIB Member Online Change

- The staging ACBLIB must be allocated by the IMS control region
  - DDNAME is IMSACB
    - Either dynamic (DFSMDA) or static (DD statement) allocation may be used
    - All IMS systems in an IMSplex should use the same staging ACBLIB
      - Avoids potential problems with different contents in different ACBLIBs
- Specify whether ACBLIB is shared or dedicated in the IMSplex
  - ACBSHR=Y|N in DFSCGxxx or DFSDFxxx PROCLIB member

    Y – all of the IMSs use the same active and inactive ACBLIB

    N – each IMS uses its own dedicated active and inactive ACBLIB
    - Used to determine if each IMS must update its own ACBLIB
    - All IMSs in the IMSplex must specify the same value of ACBSHR=

Figure 7-9. Setup for ACBLIB Member Online Change CMA01.0

Notes:

Here is a sample DFSMDA member for dynamic allocation of the ACBLIB staging library for ACBLIB member online change.

```
DFSMDA   TYPE=INITIAL

DFSMDA   TYPE=IMSACB,DSNAME=STAGING.LIBRARY

DFSMDA   TYPE=FINAL
```

The staging library was not used in online change in previous IMS versions; therefore, it would not be in the IMS Control Region JCL. This new DFSMDA member of TYPE=IMSACB is created for the staging ACBLIB. With TYPE=IMSACB, users only need to specify the data set name for the staging ACBLIB. A dynamic allocation member with the name of IMSACB will be created for the staging ACBLIB. Because of this, no database MDA members should be created with a DDNAME=IMSACB. IMS dynamically allocates the staging ACBLIB with DISP=SHR.

If dynamic allocation fails, the OLC PREPARE process will fail and return a return/reason code and completion code indicating the error. The user will need to issue a TERMINATE

OLC command to abort the online change in progress, and then reissue the INIT OLC PHASE(PREPARE TYPE(ACBMBR) command.

If you don't set up a DFSMDA member for the staging library, you will need to include a DD statement for the staging library in your IMS control region procedure. If the DD statement exists, it will override any DFSMDA member. The following is an example of a DD statement:

```
//IMSACB  DD DSN=STAGING.LIBRARY, DISP=SHR
```

The staging ACBLIB is not updated by member online change. It may be shared or each IMS may have its own copy. If different IMS systems have different staging libraries, they must have the same contents.

The ACBSHR= parameter in the DFSCGxxx or DFSDFxxx PROCLIB member determines if all IMS systems use the same active and inactive ACBLIBs. If they use the same active ACBLIB, the new and changed members are only copied by one IMS during a member online change.

Setup for ACBLIB Member Online Change (cont.)

- Member OLC requires reinitialization of the OLCSTAT data set
  - Reinitialization with V10 DFSUOLC0 utility adds member OLC information
  - Do this after IMS V8 and V9 will no longer be used in the IMSplex
    - IMS V8 and V9 cannot use an IMS V10 level OLCSTAT
    - IMS V10 can use an IMS V8 or V9 level OLCSTAT
    - IMS V10 cannot use member online change with V8 or V9 level OLCSTAT
- Reinitialization process
  - Determine ACBLIB, FMTLIB, and MODBLKS suffixes, modify ID, and IMS systems using OLCSTAT
      - QUERY OLC LIBRARY(OLCSTAT) may be used to determine current values
  - Run DFSUOLC0 utility and specify values for suffixes, modify ID, and IMS systems
    - The utility may be run while IMS systems are up

Figure 7-10. Setup for ACBLIB Member Online Change (cont.) CMA01.0

Notes:

The OLCSTAT data set must be reinitialized before ACBLIB member online change may be used. When reinitialized by the IMS V10 DFSUOLC0 utility the data set is expanded to include ACBLIB member online change information. If there are IMS V8 and V9 systems using the OLCSTAT data set they require coexistence maintenance to tolerate a IMS V10 OLCSTAT data sets. This is provided by APARs PK23401 for IMS V8 and PK23402 for IMS V9. ACBLIB member online change cannot be used when IMS V8 or V9 systems are included in the IMSplex. The coexistence maintenance only provides the capability for IMS V8 and V9 systems to use OLCSTAT data sets which have been initialized at the IMS V10 level.

IMS V10 may use a OLCSTAT data set which was initialized for IMS V8 or V9, but it cannot use ACBLIB member online change with an IMS V8 or V9 level OLCSTAT data set.

Reinitialization of the OLCSTAT data set is done with the DFSUOLC0 utility. The IMS V10 will initialize the data set in the IMS V10 format by default. You must specify the correct suffixes for the ACBLIB, FMTLIB, and MODBLKS data sets as well as the correct modify

ID. If IMS systems are up you must also specify them to the utility. You may use the QUERY OLC LIBRARY(OLCSTAT) command to determine these values.

There is an option to use the IMS V10 DFSUOLC0 utility to initialize an OLCSTAT data set at the IMS V8 or V9 level. This is done with the VERSION=1 parameter which was added to the utility by APAR PK37127. VERSION=1. VESION=1 for the OLCSTAT is the IMS V8 and V9 version. VERSION=2 is the default for the DFSUOLC0 utility. VERSION=2 for the OLCSTAT is the IMS V10 version.

IMSplex Consideration

- When an IMS system in the IMSplex is not up when a change is made
  - If all IMS systems are sharing the active ACBLIB, this is not a problem
    - When IMS system is restarted it will use the updated ACBLIB
  - If each IMS system has its own ACBLIB, there is a potential problem
    - The system which was down did not get the updates
    - Procedure to handle this situation:
      - Specify INIT OPTION(FRCNRML) parameter for PHASE(PREPARE)
        - Allows a member online change to be processed if any IMS in OLCSTAT is shutdown normally
        - IMS that is down removed from OLCSTAT data set
      - Before IMS restart, copy updated active ACBLIB to ACBLIB which was not updated or update with ACBGEN
      - Message DFS3433W issued at restart

```
DFS3433W ACBLIB MEMBER OLD ID MISMATCH MOLCID=yyyydddhhmmss
```

Figure 7-11. IMSplex Consideration CMA01.0

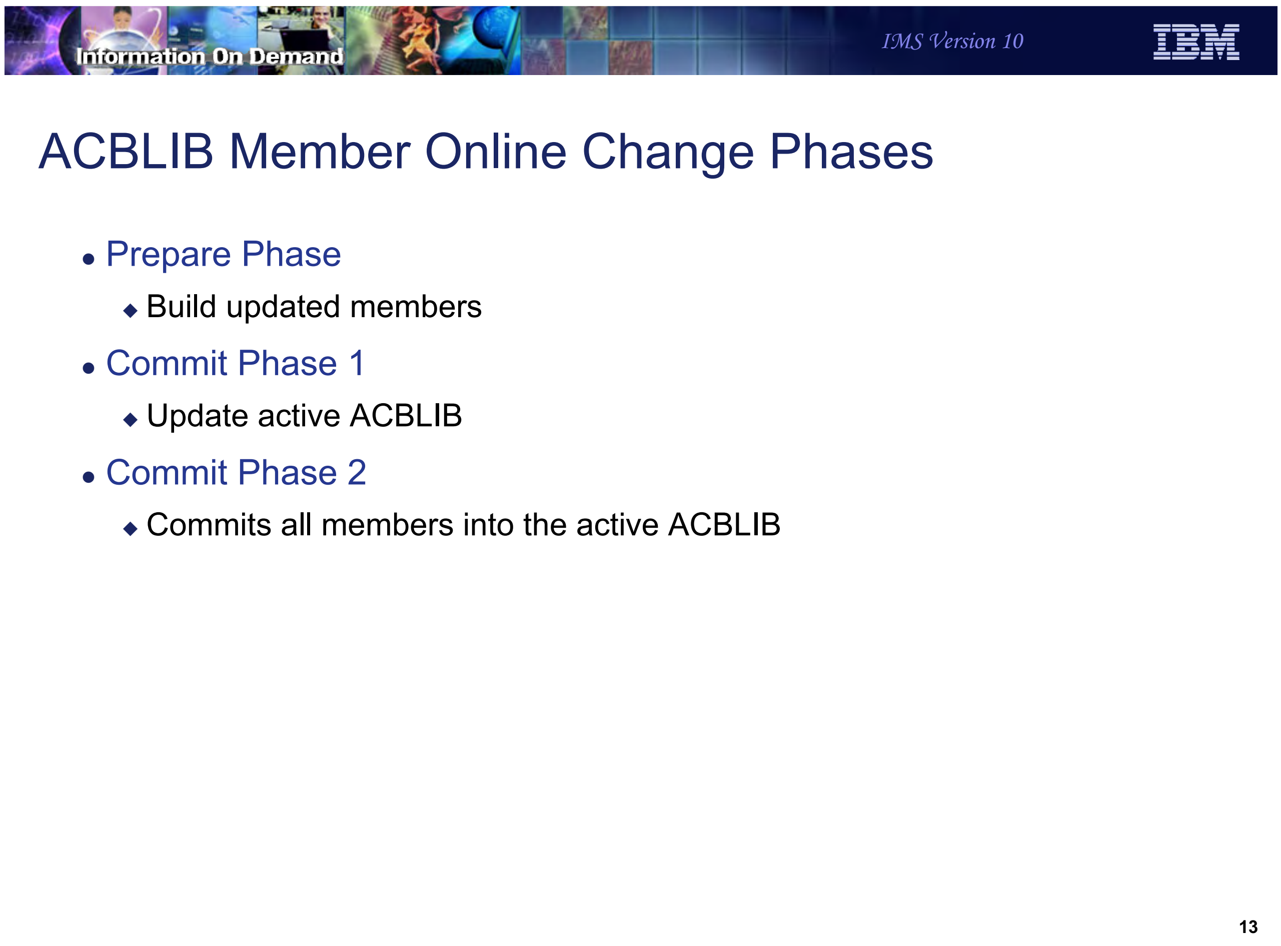
Notes:

The FRCNRML (Force Normal) option on the INIT OLC PHASE(PREPARE) command allows the user to do an online change even if one of the IMSs in the OLCSTAT data set is down normally. This does not apply to IMS systems which are down due to an abend.

When using the FRCNRML option and ACBSHR=Y (single, shared ACBLIB), there is no problem if an IMS system is down during the online change. Since there is only one ACBLIB, it will contain all member online changes that were processed while any IMS was down normally. The user does not have any additional ACBLIB processing to do.

When using the FRCNRML option and ACBSHR=N (separate, non-shared ACBLIBs), this member online change will be missing from the active ACBLIB when the terminated IMS is restarted. The user must ensure that the ACBLIB used by the system which was down during the online change is updated with the changed members. There are two possibilities. First, one can execute the ACBGEN utility to put those missing changes into the active ACBLIB before that IMS is restarted. Second, if the ACBLIBs of the IMS systems are clones, one can copy the active ACBLIB of another IMS system to the ACBLIB used by the IMS system which is down.

The FRCABND (Force Abend) option on the INIT OLC PHASE(PREPARE) command is not supported. This means that if an IMS system is down due to an abend, an ACBLIB member online change may not be done before it is emergency restarted.



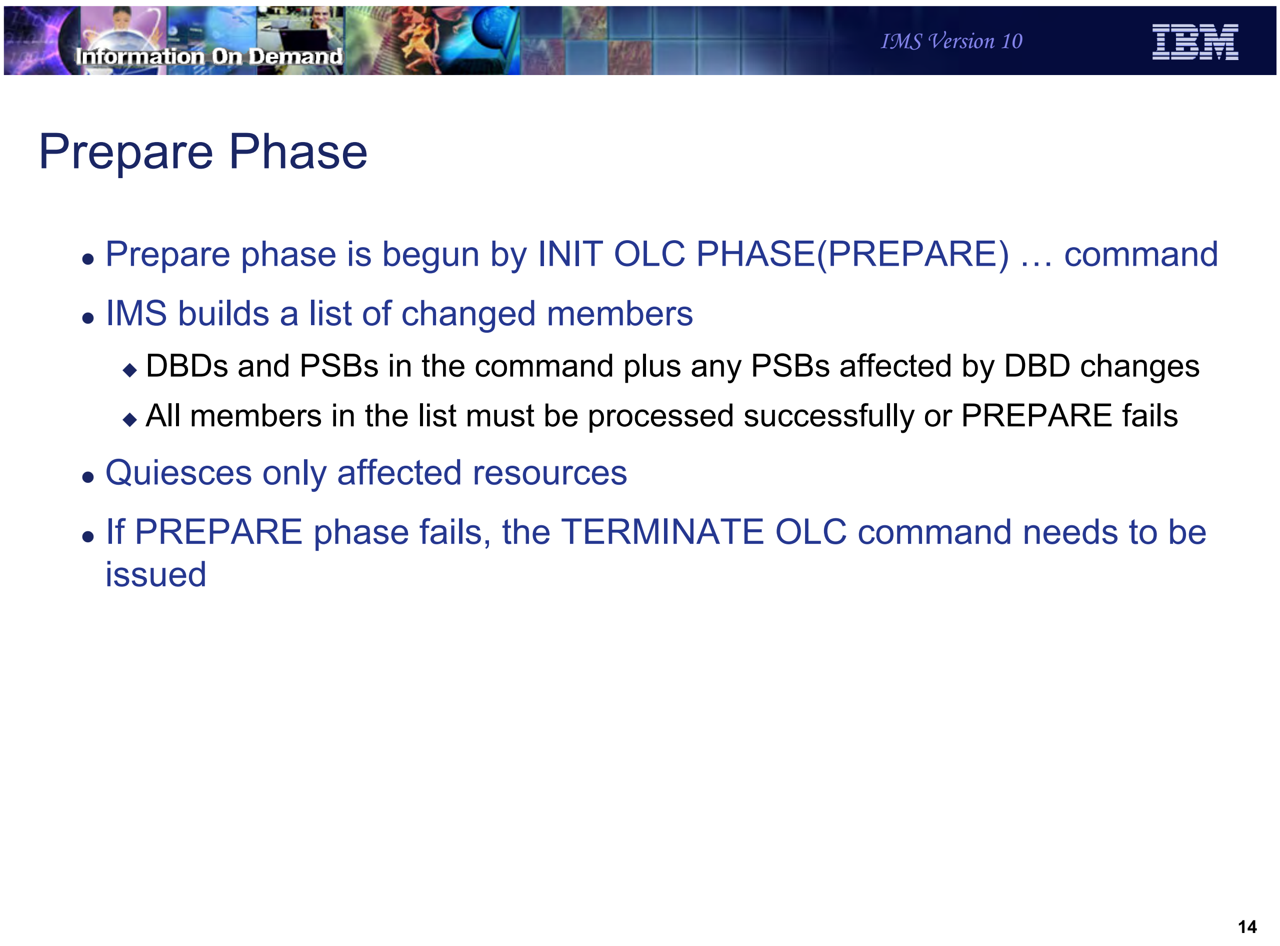
ACBLIB Member Online Change Phases

- Prepare Phase
  - Build updated members
- Commit Phase 1
  - Update active ACBLIB
- Commit Phase 2
  - Commits all members into the active ACBLIB

Figure 7-12. ACBLIB Member Online Change Phases CMA01.0

Notes:

There are three phases to the ACBLIB member online change. Details are described in the following charts.


Information On Demand
IMS Version 10

Prepare Phase

- Prepare phase is begun by INIT OLC PHASE(PREPARE) … command
- IMS builds a list of changed members
  - DBDs and PSBs in the command plus any PSBs affected by DBD changes
  - All members in the list must be processed successfully or PREPARE fails
- Quiesces only affected resources
- If PREPARE phase fails, the TERMINATE OLC command needs to be issued

Figure 7-13. Prepare Phase CMA01.0

Notes:

Commit Phase 1

- Commit phase is begun by INIT OLC PHASE(COMMIT) command
- Updated member(s) are written to the active ACBLIB with encrypted name(s)
  - Member name is translated to hex characters and written as new member to the active ACBLIB
  - Existing members are not updated
  - Ensures all new/updated members fit in the active ACBLIB
    - Prevents x37 abends from creating inconsistent members in the active ACBLIB
- If Commit Phase 1 fails, the TERMINATE OLC command needs to be issued

Figure 7-14. Commit Phase 1 CMA01.0

Notes:

Information On Demand IMS Version 10 IBM

Commit Phase 2

- Deletes old member(s)
- Renames encrypted member(s) to the actual name(s)
- Refreshes TTRs for changed members
- IMS now processes with new and/or changed members

Figure 7-15. Commit Phase 2 CMA01.0

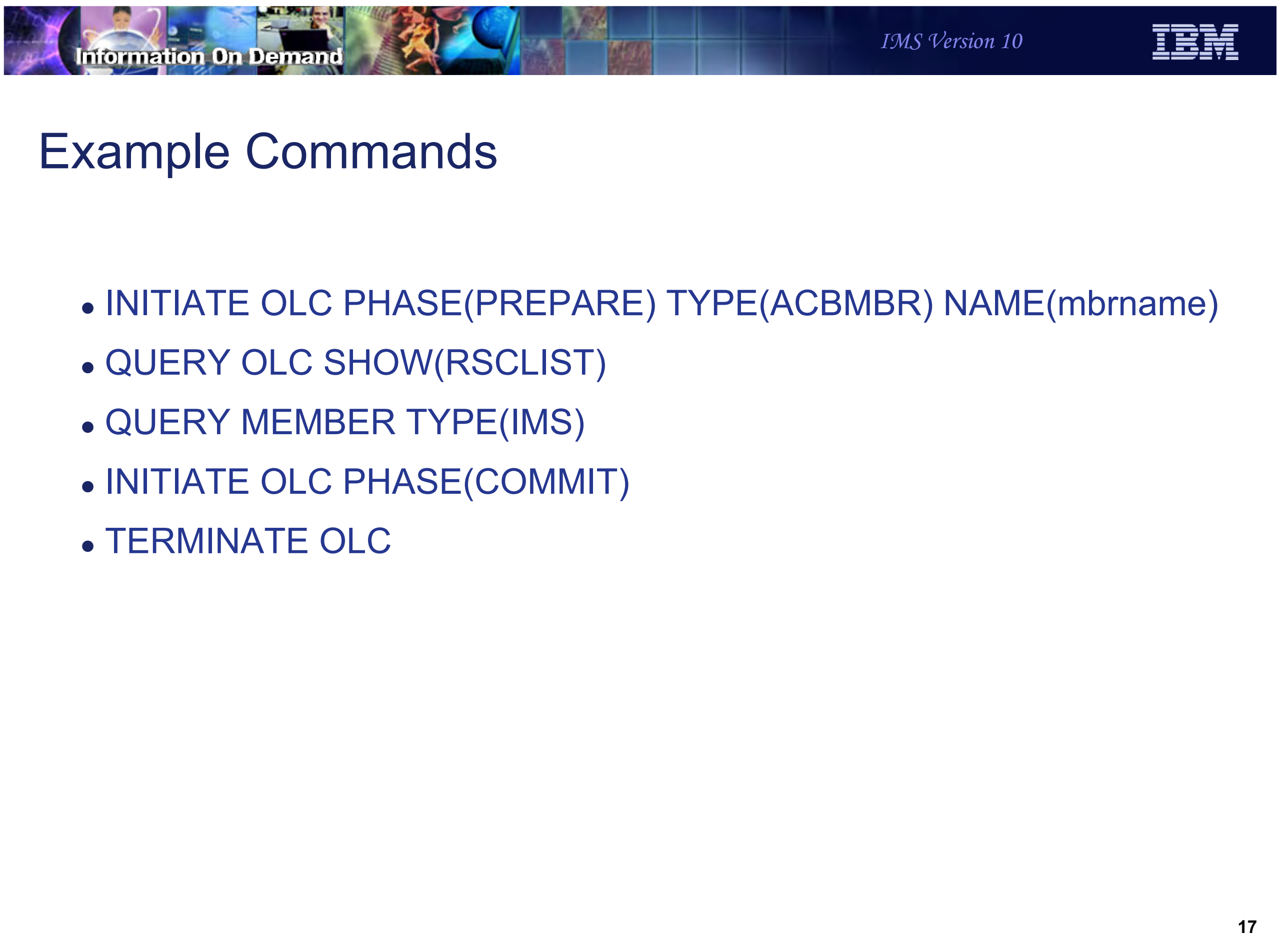
Notes:

ACBLIB member online change adds new and changed log records in IMS. The following table lists the six new log records that are created for ACBLIB Member Online Change and the x'4001' log record which is changed.

Log Record Type	**Conditions for writing log record**
X’7002’	Begin write Member OLC log record
X’7003’	Write complete Member OLC log record
X’7004’	Commit Member OLC log record
X’7005’	Commit Complete Member OLC log record
X’7006’	Restart Abort Member OLC log record
X’7010’	TERM OLC for Member OLC log record
X’4001’	Checkpoint record is updated to contain an MOLCID

These log records are processed by IMS emergency restart, by the XRF alternate, and by the FDBR tracker at log read time.

The X'4001' Checkpoint record is updated to contain the MOLCID (Member Online Change ID), a 12-byte UTC timestamp. This is used to compare with the MOLCID in the OLCSTAT at IMS warm restart to determine whether a member OLC occurred while an IMS system is down.


Information On Demand
IMS Version 10
IBM

Example Commands

- INITIATE OLC PHASE(PREPARE) TYPE(ACBMBR) NAME(mbrname)
- QUERY OLC SHOW(RSCLIST)
- QUERY MEMBER TYPE(IMS)
- INITIATE OLC PHASE(COMMIT)
- TERMINATE OLC

17

Figure 7-16. Example Commands CMA01.0

Notes:

INIT OLC PHASE(PREPARE) Command Example

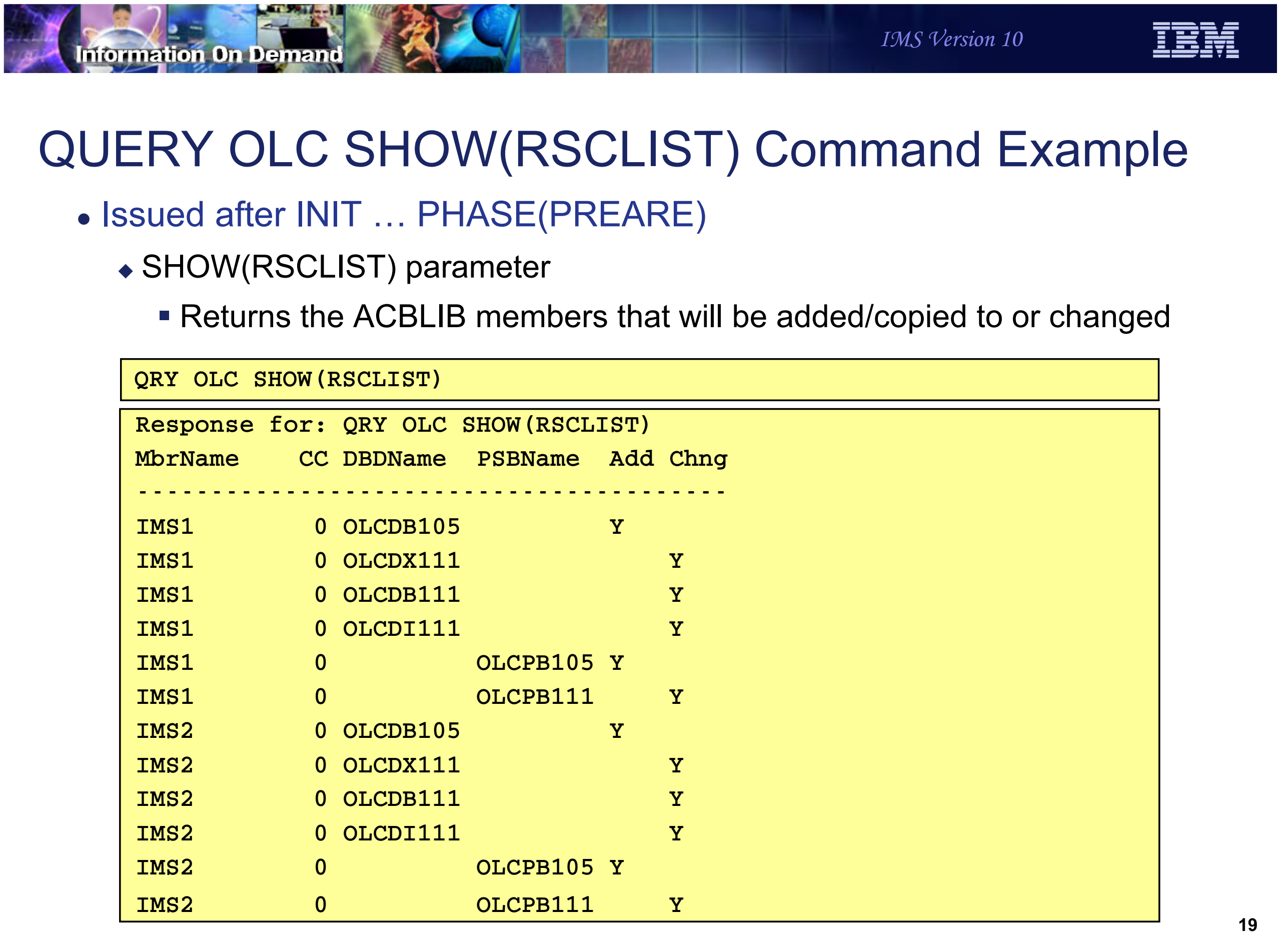
- Used to initiate a member online change

```
INIT OLC PHASE(PREPARE) TYPE(ACBMBR) NAME(OLCDB105,OLCDX111)
```

```
Response for: INIT OLC PHASE(PREPARE) TYPE(ACBMBR) NAME(OLCDB105,OLCDX111)
MbrName  Member      CC ACBSHR DBDName  PSBName  Add Chng
--------------------------------------------------------------------------
IMS2     IMS1         0 N
IMS2     IMS1         0       OLCDB105           Y
IMS2     IMS1         0       OLCDX111               Y
IMS2     IMS1         0       OLCDB111               Y
IMS2     IMS1         0       OLCDI111               Y
IMS2     IMS1         0                OLCPB105 Y
IMS2     IMS1         0                OLCPB111     Y
IMS2     IMS2         0 N
IMS2     IMS2         0       OLCDB105           Y
IMS2     IMS2         0       OLCDX111               Y
IMS2     IMS2         0       OLCDB111               Y
IMS2     IMS2         0       OLCDI111               Y
IMS2     IMS2         0                OLCPB105 Y
IMS2     IMS2         0                OLCPB111     Y
```

Figure 7-17. INIT OLC PHASE(PREPARE) Command Example

Notes:


Information On Demand
IMS Version 10
IBM

QUERY OLC SHOW(RSCLIST) Command Example

- Issued after INIT … PHASE(PREARE)
  - SHOW(RSCLIST) parameter
    - Returns the ACBLIB members that will be added/copied to or changed

```
QRY OLC SHOW(RSCLIST)
```

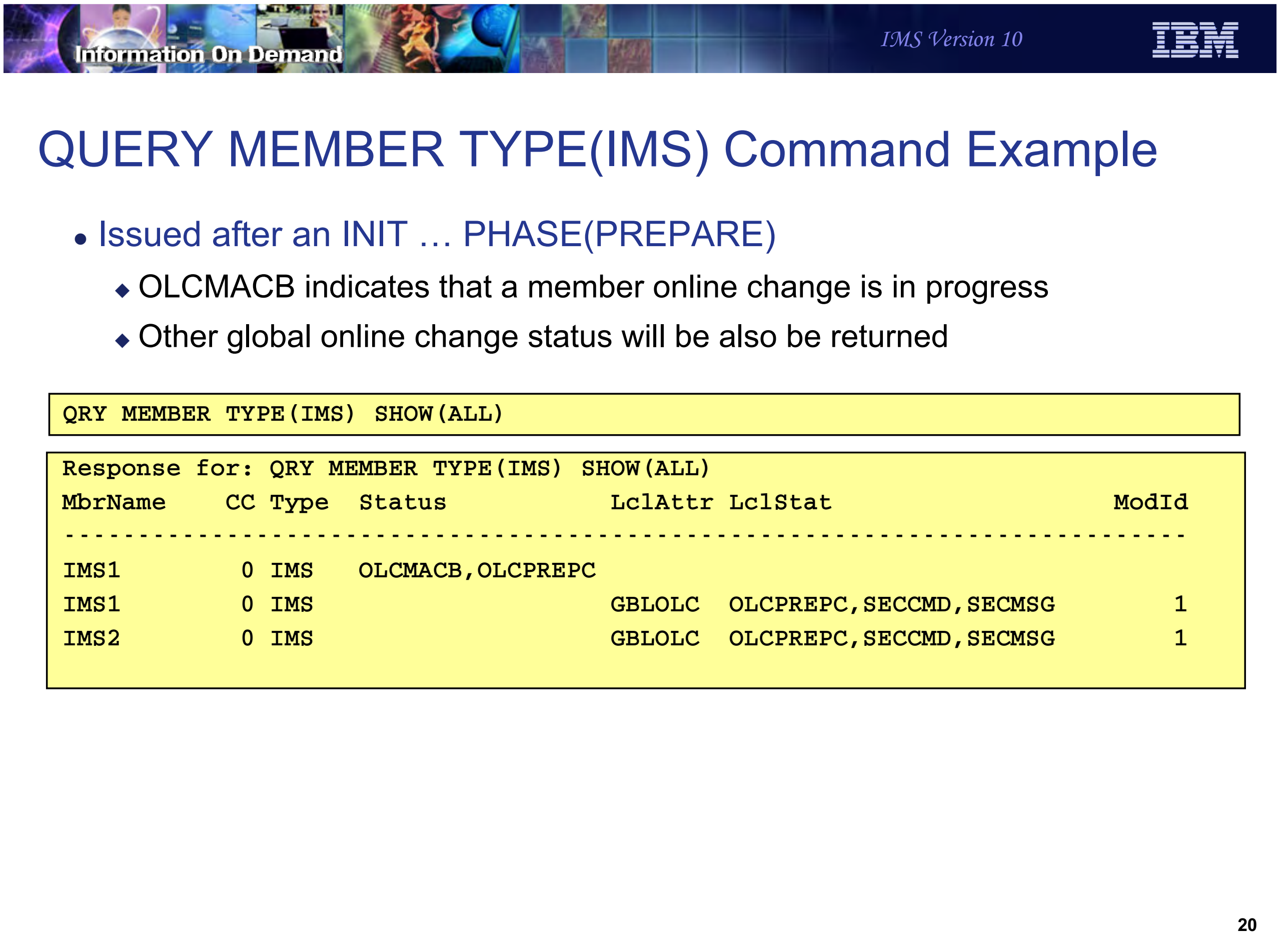
```
Response for: QRY OLC SHOW(RSCLIST)
MbrName    CC DBDName  PSBName  Add Chng
----------------------------------------
IMS1        0 OLCDB105          Y
IMS1        0 OLCDX111              Y
IMS1        0 OLCDB111              Y
IMS1        0 OLCDI111              Y
IMS1        0          OLCPB105 Y
IMS1        0          OLCPB111     Y
IMS2        0 OLCDB105          Y
IMS2        0 OLCDX111              Y
IMS2        0 OLCDB111              Y
IMS2        0 OLCDI111              Y
IMS2        0          OLCPB105 Y
IMS2        0          OLCPB111     Y
```

19

Figure 7-18. QUERY OLC SHOW(RSCLIST) Command Example CMA01.0

Notes:

The QUERY OLC SHOW(RSCLIST) command is valid only when a TYPE(ACBMBR) online change is in progress after an INIT OLC PHASE(PREPARE) command. It lists the DBDs and PSBs to be added or changed by the ACBLIB member online change.


Information On Demand
IMS Version 10
IBM

QUERY MEMBER TYPE(IMS) Command Example

- Issued after an INIT … PHASE(PREPARE)
  - OLCMACB indicates that a member online change is in progress
  - Other global online change status will be also be returned

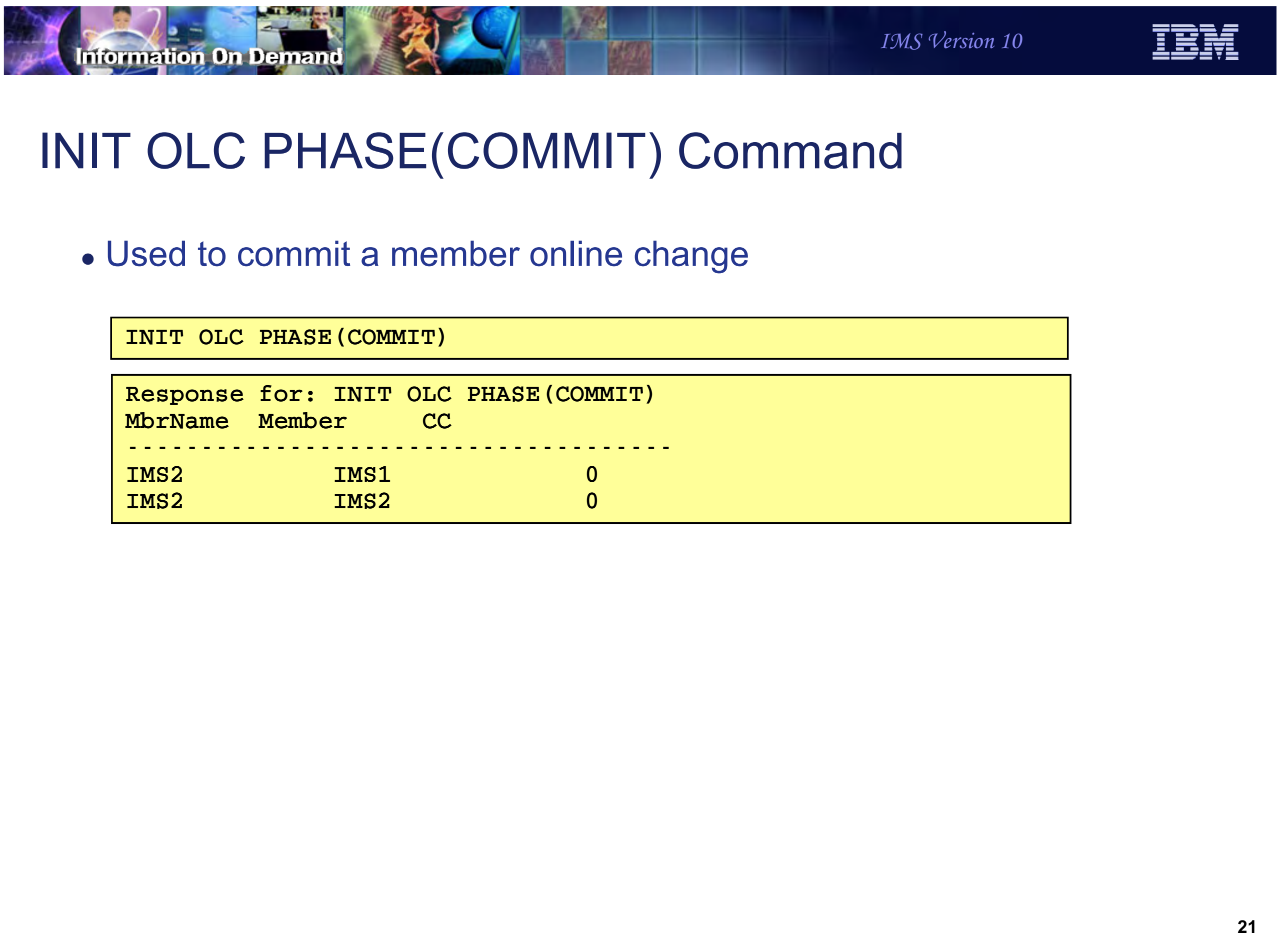
```
QRY MEMBER TYPE(IMS) SHOW(ALL)
```

```
Response for: QRY MEMBER TYPE(IMS) SHOW(ALL)
MbrName     CC Type  Status           LclAttr LclStat                    ModId
----------------------------------------------------------------------------
IMS1         0 IMS   OLCMACB,OLCPREPC
IMS1         0 IMS                    GBLOLC  OLCPREPC,SECCMD,SECMSG         1
IMS2         0 IMS                    GBLOLC  OLCPREPC,SECCMD,SECMSG         1
```

Figure 7-19. QUERY MEMBER TYPE(IMS) Command Example CMA01.0

Notes:

OLCMACB is a new status indicating a ACBLIB member online change is in progress.



Information On Demand
IMS Version 10
IBM

INIT OLC PHASE(COMMIT) Command

- Used to commit a member online change

```
INIT OLC PHASE(COMMIT)
```

```
Response for: INIT OLC PHASE(COMMIT)
MbrName  Member     CC
------------------------------------
IMS2          IMS1             0
IMS2          IMS2             0
```

Figure 7-20. INIT OLC PHASE(COMMIT) Command CMA01.0

Notes:

TERMINATE OLC Command

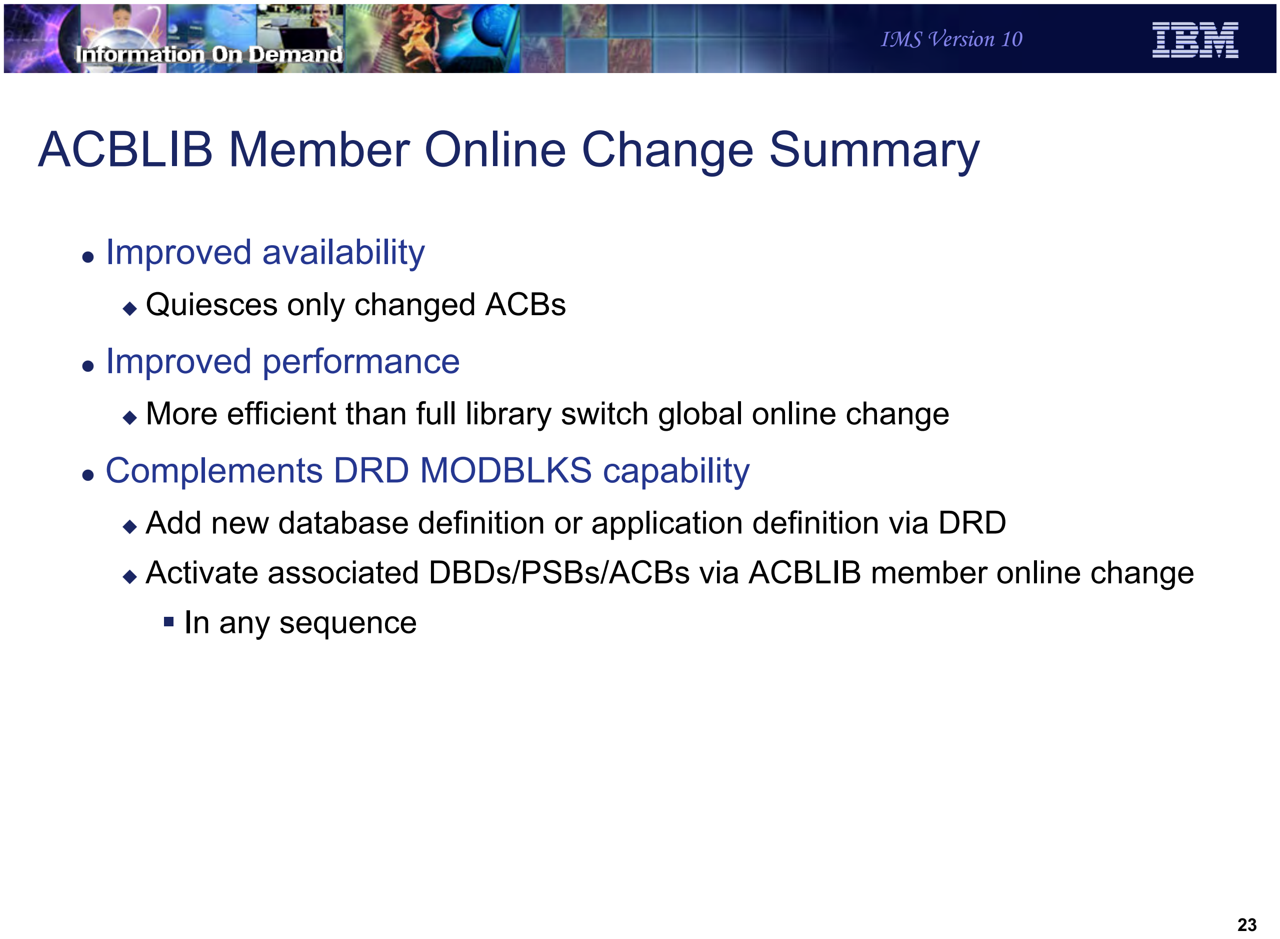
- Used to abort a member online change that is in progress
  - Can be issued
    - After an INIT OLC PHASE(PREPARE) TYPE(ACBMBR) error
    - After an INIT OLC PHASE(COMMIT) error

```
TERMINATE OLC
```

```
Response for: TERMINATE OLC
MbrName  Member     CC
----------------------------------
IMS2     IMS1        0
IMS2     IMS2        0
```

Figure 7-21. TERMINATE OLC Command CMA01.0

Notes:


Information On Demand
IMS Version 10
IBM

ACBLIB Member Online Change Summary

- Improved availability
  - Quiesces only changed ACBs
- Improved performance
  - More efficient than full library switch global online change
- Complements DRD MODBLKS capability
  - Add new database definition or application definition via DRD
  - Activate associated DBDs/PSBs/ACBs via ACBLIB member online change
    - In any sequence

23

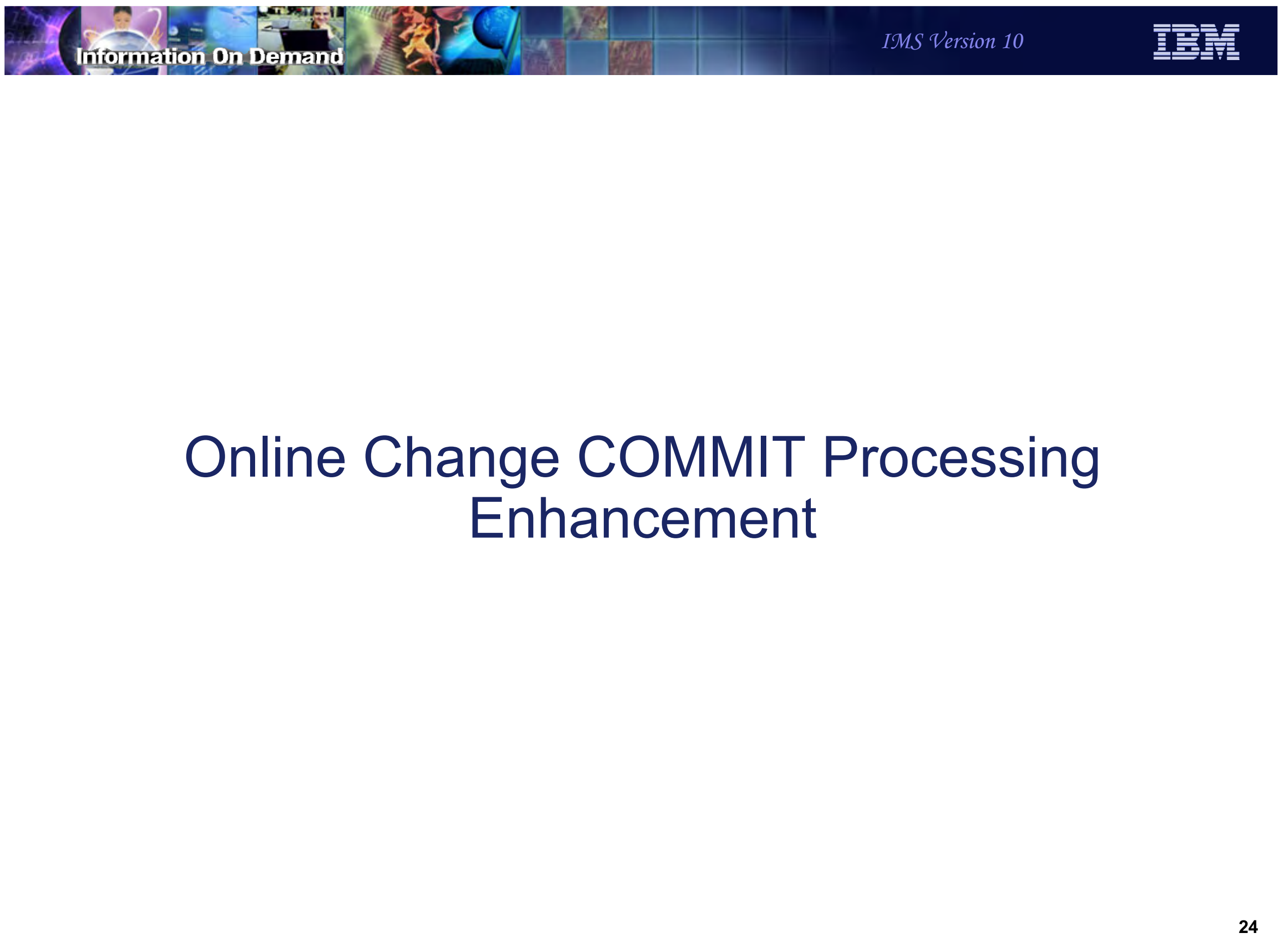
Figure 7-22. ACBLIB Member Online Change Summary CMA01.0

Notes:

ACBLIB member online change requires the use of OLCSTAT, not MODSTAT. It is invoked only with type-2 commands. This means that a CSL environments is required.

All IMSs in the IMSplex must be at IMS V10 to perform an ACBLIB member online change.

ACBLIB member online change supports XRF, FDBR, and DBCTL Warm Standby. It does not support RSR or MSDBs.



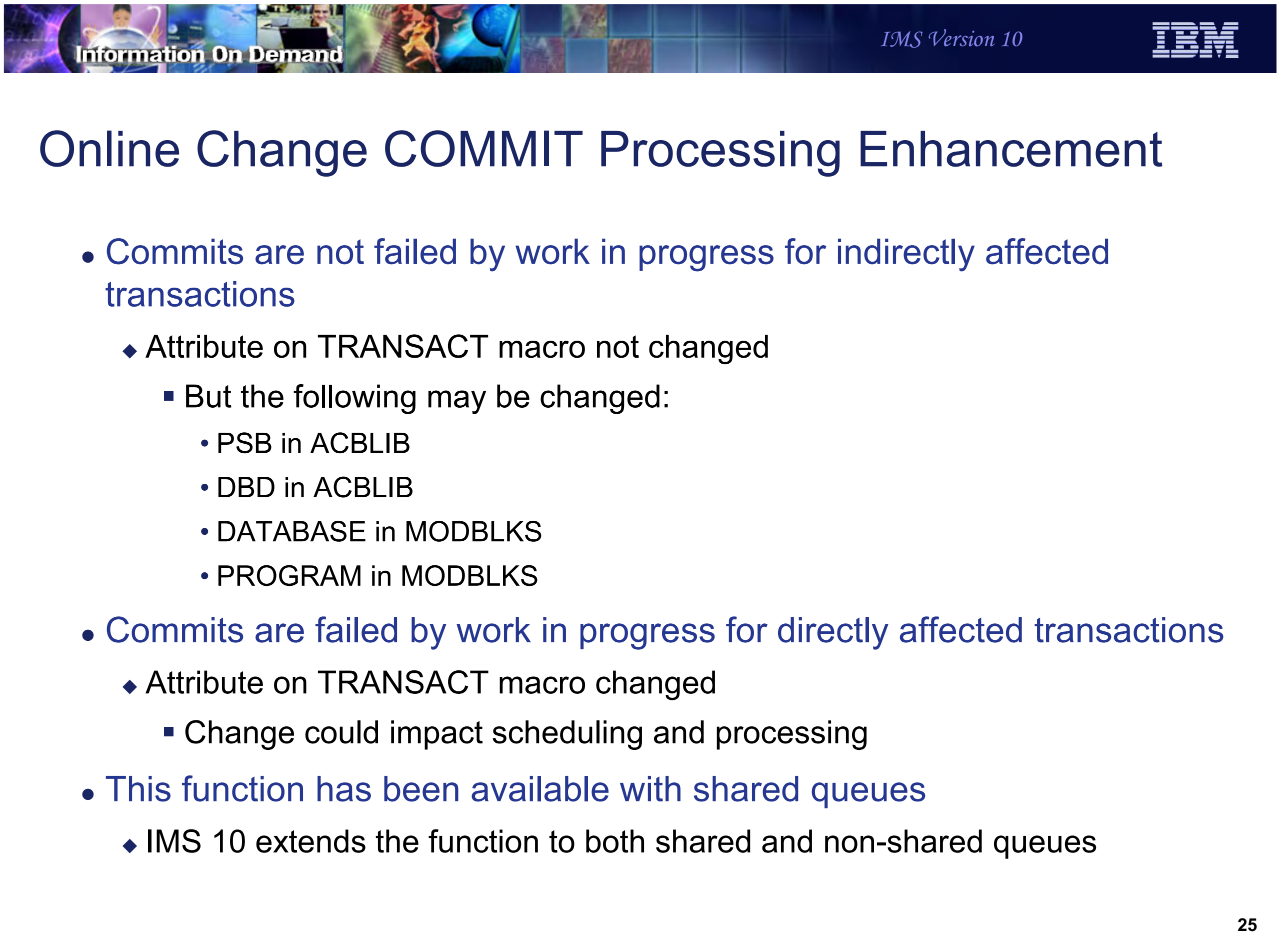
Information On Demand

IMS Version 10

Online Change COMMIT Processing Enhancement

Figure 7-23. [No title] CMA01.0

Notes:



Information On Demand

IMS Version 10

Online Change COMMIT Processing Enhancement

- Commits are not failed by work in progress for indirectly affected transactions
  - Attribute on TRANSACT macro not changed
    - But the following may be changed:
      - PSB in ACBLIB
      - DBD in ACBLIB
      - DATABASE in MODBLKS
      - PROGRAM in MODBLKS
- Commits are failed by work in progress for directly affected transactions
  - Attribute on TRANSACT macro changed
    - Change could impact scheduling and processing
- This function has been available with shared queues
  - IMS 10 extends the function to both shared and non-shared queues

25

Figure 7-24. Online Change COMMIT Processing Enhancement CMA01.0

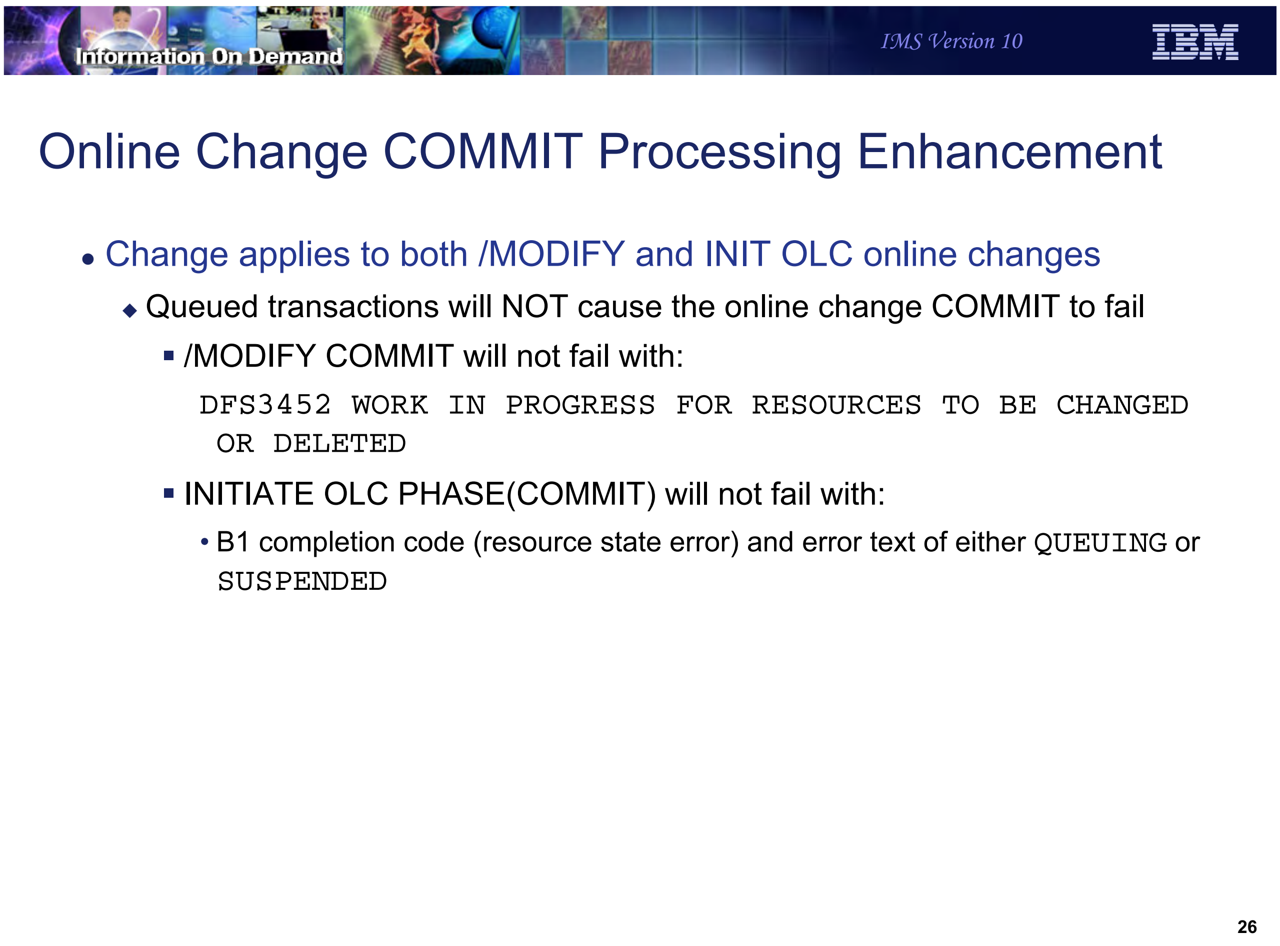
Notes:

In previous versions of IMS with a non-shared queues environment, online changes for indirectly affected transactions could cause commit processing to fail. Indirectly affected transactions are those for which the TRANSACT macro is not changed, but a PROGRAM macro or PSB in ACBLIB which the transaction uses is changed or a DATABASE macro or DBD in ACBLIB is changed for a database referenced by the transaction's program. In the previous releases if there were transactions on the queue for indirectly affected transactions the online change would fail commit processing. This changes in IMS V10. Only directly affected transactions will cause commit failures. Directly affected transactions are transactions that have changes to their definitions that could impact the scheduling and processing of queued or subsequently arriving transactions. Examples on the TRANSACT macro are AOI=, EDIT=, FPATH=, INQ=, and MSGTYPE=.

This applies to both local online change (/MODIFY command) and global online change (INIT OLC command).

This function has been available with shared queues with the following maintenance on previous IMS versions:

- PQ96043 for IMS V9
- PQ96402 for IMS V8


Information On Demand
IMS Version 10
IBM

Online Change COMMIT Processing Enhancement

- Change applies to both /MODIFY and INIT OLC online changes
  - Queued transactions will NOT cause the online change COMMIT to fail
    - /MODIFY COMMIT will not fail with:

```
DFS3452 WORK IN PROGRESS FOR RESOURCES TO BE CHANGED
 OR DELETED
```

    - INITIATE OLC PHASE(COMMIT) will not fail with:
      - B1 completion code (resource state error) and error text of either `QUEUING` or `SUSPENDED`

Figure 7-25. Online Change COMMIT Processing Enhancement CMA01.0

Notes:

Information On Demand

IMS Version 10

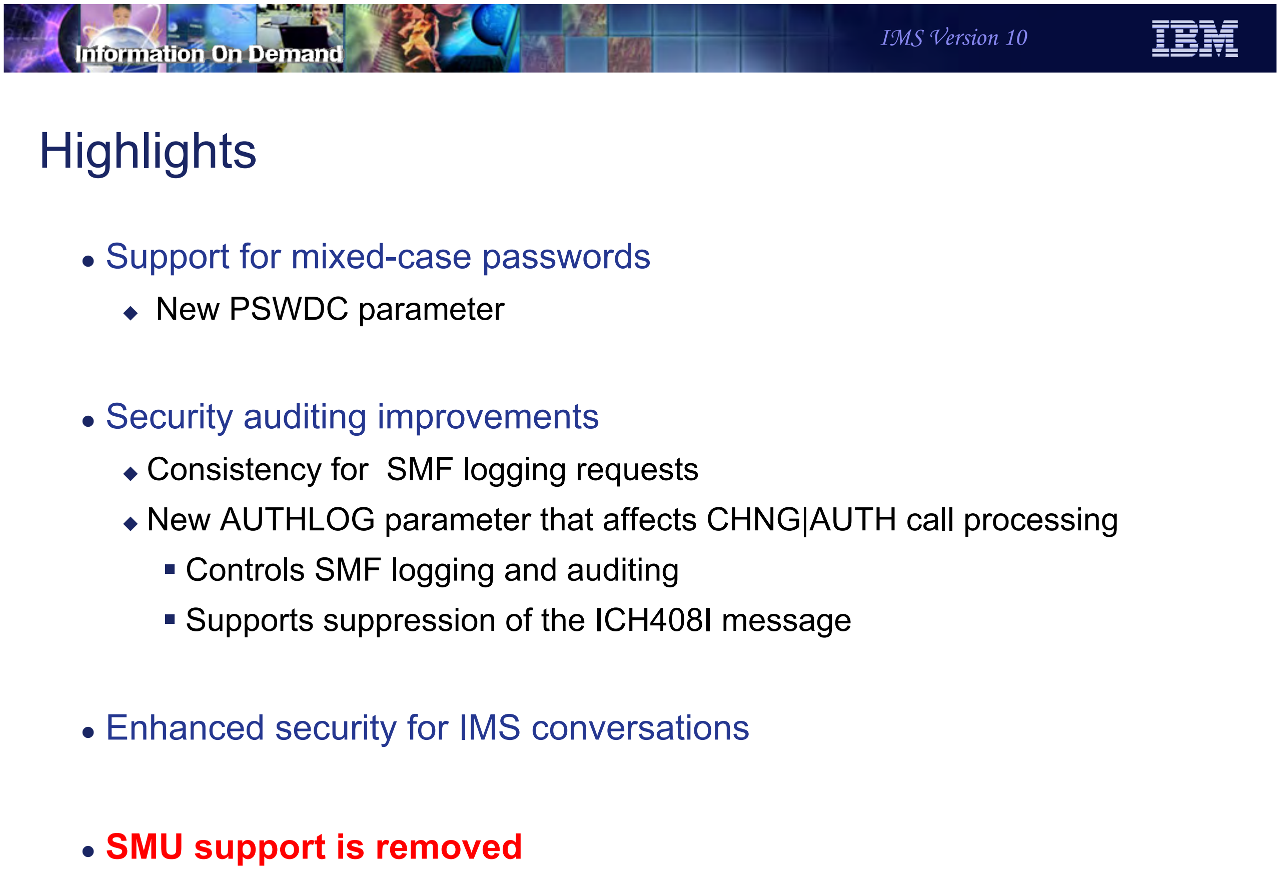
Online Change COMMIT Processing Enhancement Summary

- Improve chances of successful online change commit
- Happens automatically
  - No modifications to online change procedures needed
  - Applies to all online changes including ACBLIB member online change

Figure 7-26. Online Change COMMIT Processing Enhancement Summary CMA01.0

Notes:

Unit 8. Security



Information On Demand

IMS Version 10

Highlights

- Support for mixed-case passwords
  - New PSWDC parameter
- Security auditing improvements
  - Consistency for SMF logging requests
  - New AUTHLOG parameter that affects CHNG|AUTH call processing
    - Controls SMF logging and auditing
    - Supports suppression of the ICH408I message
- Enhanced security for IMS conversations
- **SMU support is removed**

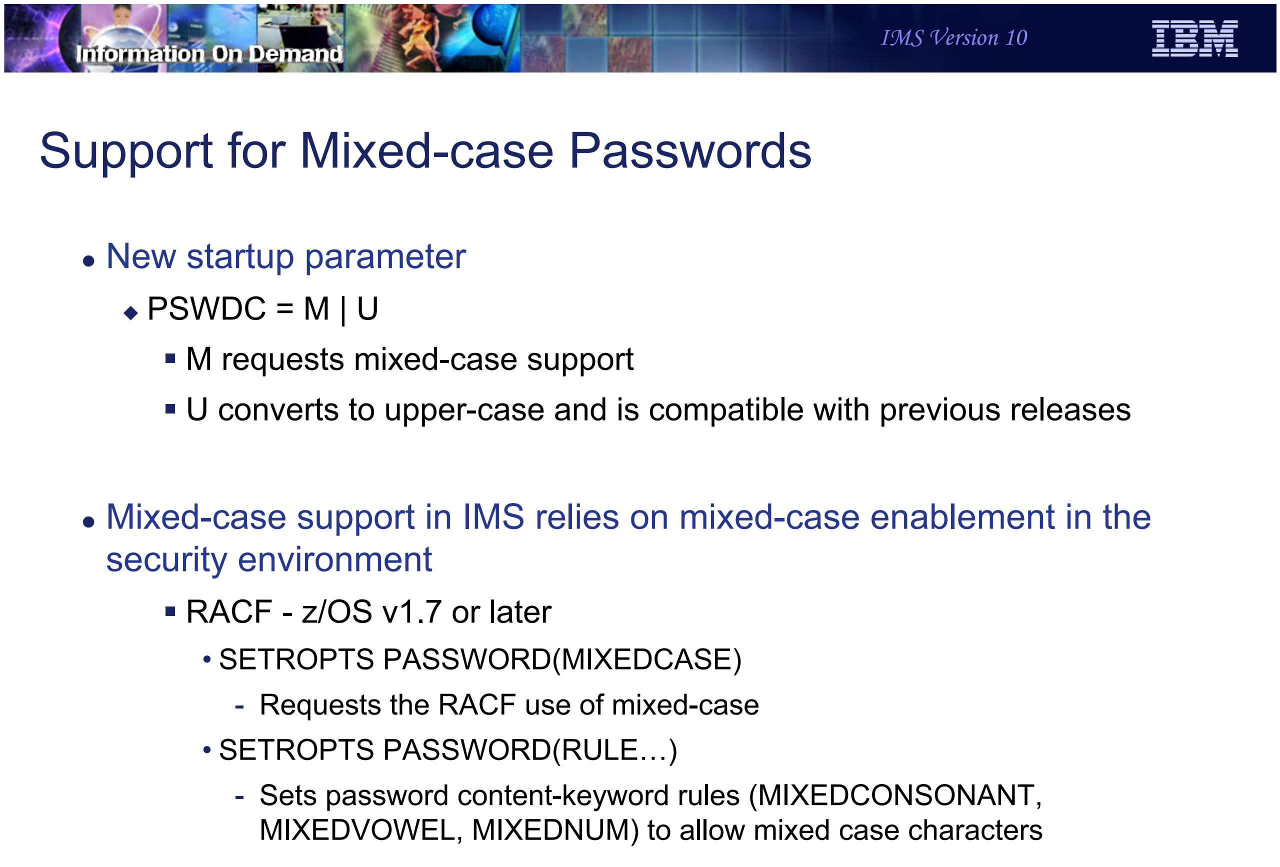
Figure 8-1. Highlights CMA01.0

Notes:

IMS V10 provides several enhancements in the security area:

- Support for mixed-case passwords through a new parameter, PSWDC.
- Greater consistency in how IMS requests SMF logging.
- A new AUTHLOG parameter that enhances auditing requests during DL/I CHNG|AUTH call requests even when DFSCTSE0 (Security Reverification Exit) exists.
- Enhanced security for IMS conversations when the terminal has been disconnected with an active conversation in progress.

Most importantly, IMS V10 removes SMU security and all the components associated with SMU. Migration from SMU to SAF/RACF (or equivalent product) is easiest if accomplished before the IMS V10 migration, otherwise the migration from SMU to SAF will need to coincide with the migration to IMS V10.


Information On Demand
IMS Version 10
IBM

Support for Mixed-case Passwords

- New startup parameter
  - PSWDC = M | U
    - M requests mixed-case support
    - U converts to upper-case and is compatible with previous releases

- Mixed-case support in IMS relies on mixed-case enablement in the security environment
    - RACF - z/OS v1.7 or later
      - SETROPTS PASSWORD(MIXEDCASE)
        - Requests the RACF use of mixed-case
      - SETROPTS PASSWORD(RULE…)
        - Sets password content-keyword rules (MIXEDCONSONANT, MIXEDVOWEL, MIXEDNUM) to allow mixed case characters

Figure 8-2. Support for Mixed-case Passwords CMA01.0

Notes:

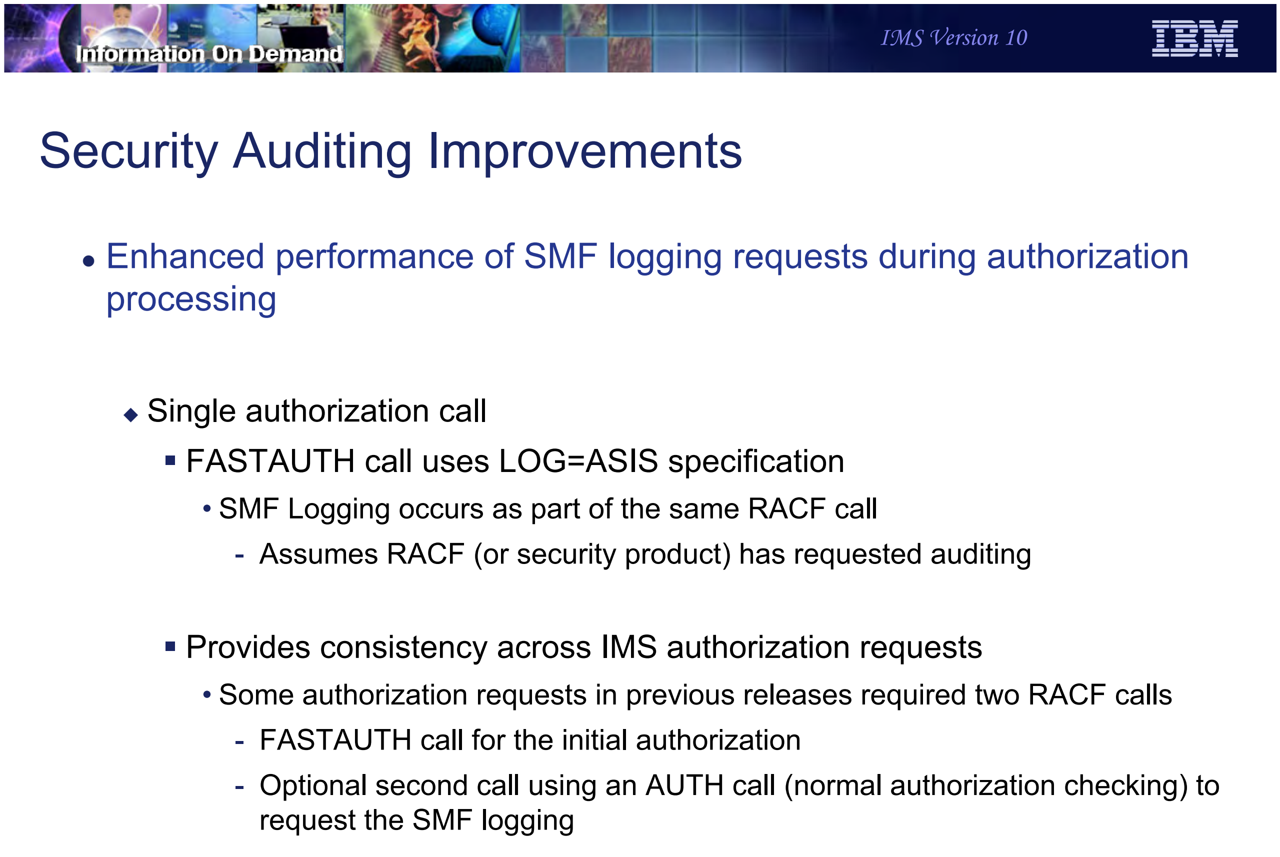
Support for mixed-case passwords is requested through a new startup parameter PSWDC. Setting the value to M (mixed-case) assumes that RACF has also been set up for mixed-case support.

For reference purposes, the applicable RACF commands include:

- SETROPTS PASSWORD(MIXEDCASE|NOMIXEDCASE). MIXEDCASE indicates that all applications on this system and those that share the RACF database support mixed-case passwords. The syntax rules must also be modified to allow mixed-case characters. When this option is activated, the RACF ALTUSER, ADDUSER, PASSWORD and RACLINK commands will no longer translate passwords to uppercase, nor will applications that request mixed-case password support such as IMS systems that specify PSWDC=M. This option is inactive by default.
- SETROPTS PASSWORD(RULE…) defines up to 8 syntax rules for new passwords and will be used to verify that the new password meets the criteria. The rules specifying mixed-case characters should only be set when the MIXEDCASE option is in effect. When the password rules are changed, there will be no immediate impact on the users.

The only time the change will have an effect is when the users change their passwords. The rules are only used to verify that a new password meets the criteria. Any existing passwords are just verified against what is entered and what is in the database. The users will not be asked to change their password just because the syntax rules have changed. If multiple rules are defined, a password that passes at least one rule is accepted.

- SETROPTS LIST can be used to display the current setting


Information On Demand
IMS Version 10
IBM

Security Auditing Improvements

- Enhanced performance of SMF logging requests during authorization processing
  - Single authorization call
    - FASTAUTH call uses LOG=ASIS specification
      - SMF Logging occurs as part of the same RACF call
        - Assumes RACF (or security product) has requested auditing
    - Provides consistency across IMS authorization requests
      - Some authorization requests in previous releases required two RACF calls
        - FASTAUTH call for the initial authorization
        - Optional second call using an AUTH call (normal authorization checking) to request the SMF logging

Figure 8-3. Support for Mixed-case Passwords … CMA01.0

Notes:

Since IMS provides several areas for a password to be entered, the new mixed-case support affects all these areas.

Be careful when enabling this capability. Users that were used to entering their passwords in any mode – lower, upper, mixed – may have their signon rejected if a specific case is expected.

Enhanced Security for IMS Conversations

- Ensures that on reconnecting to an active IMS conversation
  - After a system crash, session crash or normal termination
  - Only the userid authorized to an active conversation is allowed to access that conversation
  - Applicable to static terminals and ETO terminals where user structure is not unique to userid (e.g., LUname=USERname=LTERMname)
    - Previous releases - access to an active conversation is allowed for all userids associated with the active or any held conversations

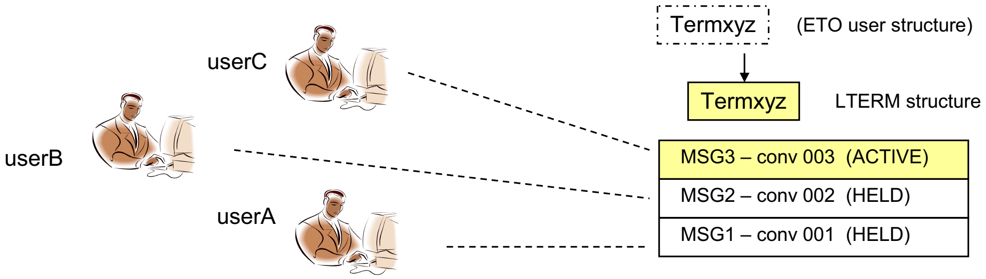


Figure 8-4. Security Auditing Improvements CMA01.0

Notes:

In previous releases, IMS authorization requests were inconsistent with respect to the auditing capability, i.e., SMF logging and writing the ICH408I message to the system console. Some SAF requests used the normal authorization request call (AUTH) which could determine whether the security product required auditing and requested the function as part of the call. Other requests used the fast authorization call (FASTAUTH) and, when auditing was needed, performed a second AUTH call; and yet other requests issued the FASTAUTH with the additional specification of LOG=ASIS to perform auditing, when required, as part of the fast authorization.

IMS V10 provides a consistent method of requesting auditing. IMS authorization requests have been changed, where needed, to use the FASTAUTH call with LOG=ASIS specification to honor SMF logging requests as part of the same SAF call thereby eliminating the optional second AUTH call that was previously required just to request SMF logging.

The RACF implementation of auditing and the way the installation requests this capability from RACF have not changed. RACF continues to perform auditing if the authorization check results in success (RC=0) or failure (RC=8), and determines that auditing is necessary based on the following conditions:

- the user's UAUDIT setting
- the AUDIT, GLOBALAUDIT, and WARNING options in effect for the resource
- the SETROPTS SECLABELAUDIT is in effect
- the AUDIT options in the resource SECLABEL profile
- the pre-processing or post-processing installation exit's indication of whether or not to do auditing.

SMU Support Removed

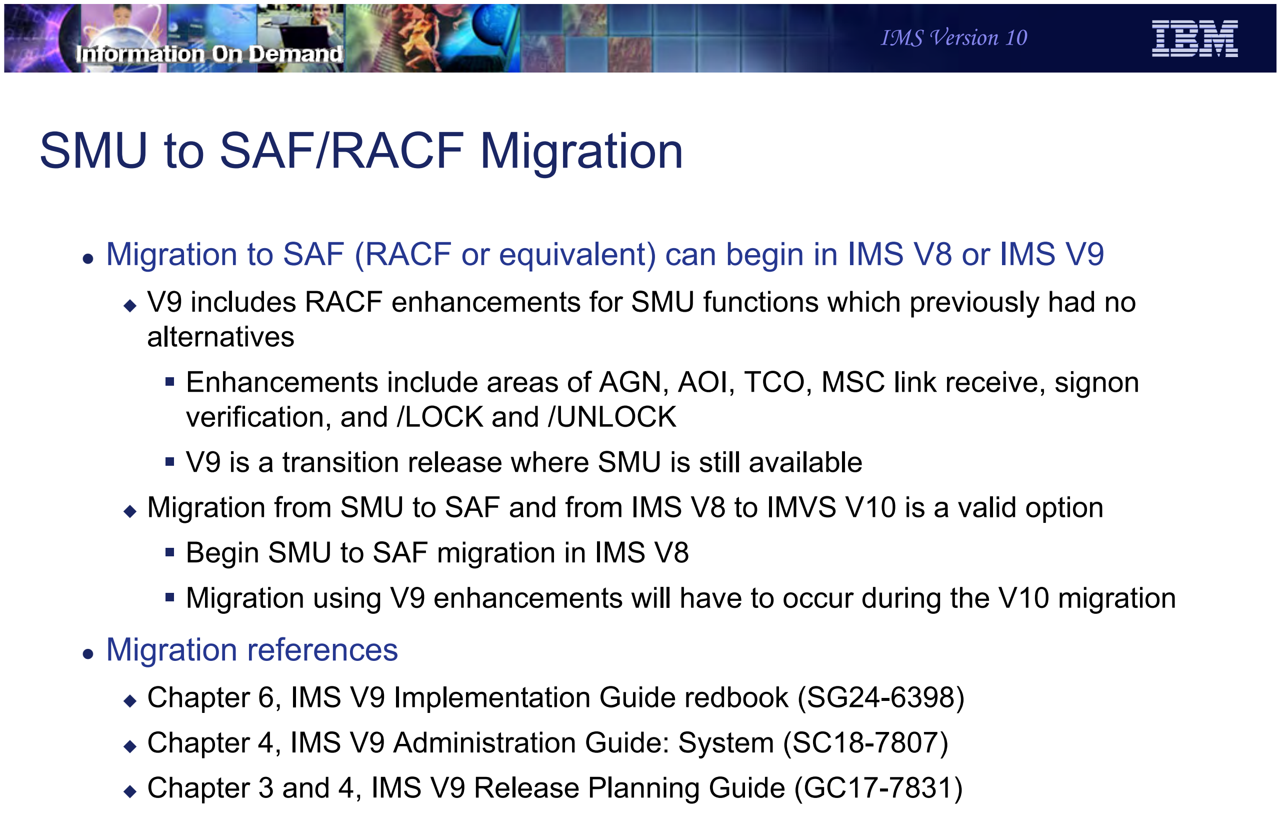
- IMS 10 removes SMU and SMU components
  - IMS-provided security
  - The Security Maintenance Utility
  - Application Group Name Exit Routine (DFSISIS0)
  - IMS.MATRIXx data sets
- Primary consideration
  - If migration from SMU to SAF/RACF has not already been done, migration to IMS 10 will also need to include migration from SMU to SAF/RACF

Figure 8-5. Security Auditing Improvements … CMA01.0

Notes:

A new DFSDCxxx startup parameter provides a system level specification on how IMS is to handle auditing for application programs that issue the AUTH calls.

- AUTHLOG=ALL (default) requests that SMF logging occur and RACF error message ICH408I be issued when appropriate
- AUTHLOG=NOMSG allows SMF logging to occur when appropriate but suppresses the RACF error message ICH408I
- AUTHLOG=NONE requests suppress of both SMF logging and RACF error message ICH408I


Information On Demand
IMS Version 10
IBM

SMU to SAF/RACF Migration

- Migration to SAF (RACF or equivalent) can begin in IMS V8 or IMS V9
  - V9 includes RACF enhancements for SMU functions which previously had no alternatives
    - Enhancements include areas of AGN, AOI, TCO, MSC link receive, signon verification, and /LOCK and /UNLOCK
    - V9 is a transition release where SMU is still available
  - Migration from SMU to SAF and from IMS V8 to IMVS V10 is a valid option
    - Begin SMU to SAF migration in IMS V8
    - Migration using V9 enhancements will have to occur during the V10 migration
- Migration references
  - Chapter 6, IMS V9 Implementation Guide redbook (SG24-6398)
  - Chapter 4, IMS V9 Administration Guide: System (SC18-7807)
  - Chapter 3 and 4, IMS V9 Release Planning Guide (GC17-7831)

Figure 8-6. Security Auditing Improvements … CMA01.0

Notes:

Another issue in prior releases was that IMS would not perform auditing if the IMS user exit DFSCTSE0 existed in the system. The documentation indicated that the user exit had to perform the auditing function.

In IMS V10, IMS provides new flags that allow the exit routine to request that IMS issue an authorization call in a form that will allow auditing to take place. To take advantage of this capability, the DFSCTSE0 exit routine will need to check a new flag (CTSEAUDR in CTSEFLG1) which indicates that the auditing function is needed, and when needed, set another new flag (CTSEDAUD in CTSEFLG2) to tell IMS to issue the authorization call and request auditing. The exit will only be able to set CTSEDAUD if CTSEAUDR is on.

Figure 8-7. Security Auditing Improvements … CMA01.0

Notes:

Security auditing in IMS V10 is based on:

- The installation’s specifications that requests auditing of resources, and
- The presence/absence of the IMS user exit DFSCTSE0 - if it exists, then the setting of the CTSEDAUD flag, and
- The specification of the AUTHLOG parameter requesting that a form of auditing be done

Figure 8-8. Enhanced Security for IMS Conversations CMA01.0

Notes:

IMS V10 enhances security for IMS conversational processing in the situation where a user attempts to signon from a terminal that is associated with an active conversation.

In prior releases, as long as a user signing on from the terminal was authorized to one of the conversations (the active one or any of the ones on hold), that user could gain access to the active conversation even if they were not authorized to the active conversation. This situation affected all static terminals as well as ETO terminals where the control block structure of the USER was not unique to a signing on userid. For ETO, the environment in question was one where all userids logging on from the terminal accessed the same control blocks, e.g., LUname=USERname=LTERMname.

Figure 8-9. Enhanced Security for IMS Conversations … CMA01.0

Notes:

The IMS V10 change ensures that a signon will be successful only if:

1. the userid attempting the signon is authorized to use the active conversation, or
2. no active conversations exist, i.e., all the conversations are held.

The impact to static and ETO terminals when an active conversation exists includes the following:

- Static: A user that is not authorized to the active conversation will fail signon. The user can issue /HOLD on the conversation and then reissue the /SIGN ON command. /RELEASE of the conversation to which the user is authorized can be entered to resume the appropriate interaction.
- ETO: Users that attempt a signon in environments that have generic naming conventions (e.g., LUname=USERname=LTERMname) must be authorized to the active conversation if one exists, otherwise the signon will fail. /HOLD is not allowed prior to a /SIGN ON.

Figure 8-10. Enhanced Security for IMS Conversations CMA01.0

Notes:

Terminology: - A *deferred conversational program switch* is one that responds to the terminal but causes the next input from the terminal to go to another conversational program. An *immediate program switch* passes the conversation directly to another conversational program. This capability addresses the deferred model.

When an application program inserts a deferred conversation program switch, IMS security mechanisms provide a method to validate authorization to the new transaction name when the SPA is inserted. The authorization check is done wherever the program runs using the security authorizations of that IMS system. IMS V10 addresses the concern that the program issuing the deferred conversation program switch could run on a back-end MSC or Shared Queues system where the authorizations might differ from the system where the switched-to transaction might actually run.

IMS V10 ensures that the user is authorized to access the switched-to transaction by adding an authorization check on the subsequent input from the end user. This check is made to validate that the user is authorized to access the switched-to transaction. If the terminal is disconnected during the conversation, and the conversation remains active, then the authorization check is made when the next signon is attempted.

Figure 8-11. SMU Support Removed CMA01.0

Notes:

IMS V10 no longer supports the Security Maintenance Utility (SMU) capability. As a result, SMU components have been removed including the SMU Utility, the AGN Exit Routine (DFSISIS0) and the MATRIX data sets. Note that the IMS.MATRIXx data sets have been deleted and removed from all IMS procedures and logic.

A primary consideration in this area includes migration. IMS systems that use SMU must migrate to the SAF interface at the time that the upgrate to IMS V10 occurs.

Figure 8-12. SMU Support Removed - Impact CMA01.0

Notes:

The impact of removing SMU affects the IMS environment in several ways.

- All SMU specifications in any of the system definition macros are ignored. These specifications were previously found in the COMM, IMSGEN and SECURITY macros.
- The Security Maintenance Utility no longer is available in IMS V10. Additionally, the Online Change Utility will no longer support the IMS.MATRIX data set. If the MATRIX dataset DD cards exist in the online change utility JCL, the utility will ignore those specific DD cards.

Figure 8-13. SMU Support Removed - Impact CMA01.0

Notes:

The changes to the startup parameters fall into several categories:

1. Parameter values requesting the use of SMU continue to be accepted in IMS V10 but are ignored. These include AGN (specification of an AGN name), AOI1=S (request to use SMU security for AOI Type 1 command authorization) and ISIS=<0|1|2>. The ISIS parameter is still supported for specification of combinations of SAF and user exit authorization. Only the three values of 0, 1, or 2 which were previously associated with SMU are now ignored.
2. Specifications that are no longer documented but are compatible with functionality in previous releases of IMS. Certain parameter values in previous releases allowed the use of SAF/RACF along with negating the loading of the signon security tables from the MATRIX data set. The specifications included: TRN=<E|X>, RCF=<B|R>, and SGN=<D|E|W. Although these values are no longer documented in the IMS V10 System Definition Guide because they request negation of loading tables from the MATRIX data set which is no longer supported, the values will be accepted for compatibility purposes and function in the same manner as the supported counterparts.

For example, TRN=E is equivalent to the supported value of TRN=F (TRN=E ? TRN=F). This is because TRN=E, in previous releases, requested option F plus the negation of loading the SMU signon tables. Likewise, TRN=X ? TRN=Y, RCF=B ? RCF=A, RCF=R ? RCF=S, SGN=D ?, SGN=D ? SGN=F, SGN=E ? SGN=G, SGN=W ? SGN=Y and SGN=X ? SGN=Z.

3. Specifications that have changed their meaning because the SMU capability no longer applies include some of the values that can be specified. The MSCSEC parameter previously included values that allowed SMU security to be used for non-directed routing requests. In IMS V10 the parameter values specify the use of SAF/RACF or no security.

Figure 8-14. SMU Support Removed – Impact … CMA01.0

Notes:

Certain commands such as /CHANGE PASSWORD and /DELETE PASSWORD|TERMINAL were strictly associated with SMU security. These commands are no longer supported in IMS V10 and, if entered, will result in error message DFS181.

SAF/RACF support for the resources associated in the /SET, /LOCK and /UNLOCK commands was introduced in IMS V9 along with the LOCKSEC parameter and additional resource classes for PSBS (IIMS/JIMS) and LTERMs (LIMS/MIMS). IMS V10 removes the SMU check and supports the SAF/RACF security check which includes two checks: both a validation that the userid of the signed on user is authorized to invoke the /SET, /LOCK, or /UNLOCK command; and a second check that userid is authorized against the resource being accessed – Transactions, LTERMS, Programs, Databases.

Password security is different when using SAF/RACF versus the previous capability with SMU. SAF/RACF uses the signed on user's password in a reverify capability whereas SMU used a global password as defined in the SMU tables. Note that the use of a password after the parameter defining the resource will continue to be supported for all keywords except PTERM and NODE. This restriction has not changed. The SAF/RACF check is accomplished using the RACF REVERIFY support. The REVERIFY support assumes that the RACF profile for the IMS resource is defined with the parameter "APPLDATA('REVERIFY')", then IMS (assuming RVFY=Y is specified as an IMS startup parameter) checks that the password is the same as the user's signon password. If the resource is defined to RACF but is not authorized for use, the command is rejected with message DFS3689W USE OF <TRANSACTION|LTERM|DB|PROG> resourcename BY <LOCK|UNLOCK> REJECTED.

Figure 8-15. SMU Support Removed – Impact … CMA01.0

Notes:

All SMU keywords are ignored when any of the restart commands are issued.

For Online Change requests, the PASSWORD, TERMINAL and TRANCMDS keywords do not apply to the IMS V10 system for either a /MODIFY or INIT OLC command. For INIT OLC, the keywords are still processed, and the appropriate flags in the MWA are set, so that in a sysplexed environment that consists of a mixture of V8, V9 and V10 systems, the keywords can be passed on to the IMS systems at the V8 or V9 level.

Figure 8-16. SMU Support Removed – Impact … CMA01.0

Notes:

The removal of the SMU support has also affected the documentation for certain messages and abend codes as well as specific application program status and return/reason codes. The documentation in the appropriate manuals has been changed to remove references to SMU-based security.

Figure 8-17. SMU Support Removed – Impact … CMA01.0

Notes:

IMS documentation has also changed to remove references to SMU security.

Figure 8-18. SMU to SAF/RACF Migration CMA01.0

Notes:

Both IMS V8 and V9 provide migration paths to the use of SAF/RACF security. IMS V9 provides migration capabilities that are not available with IMS V8.

Unit 9. Transaction Manager

Information On Demand

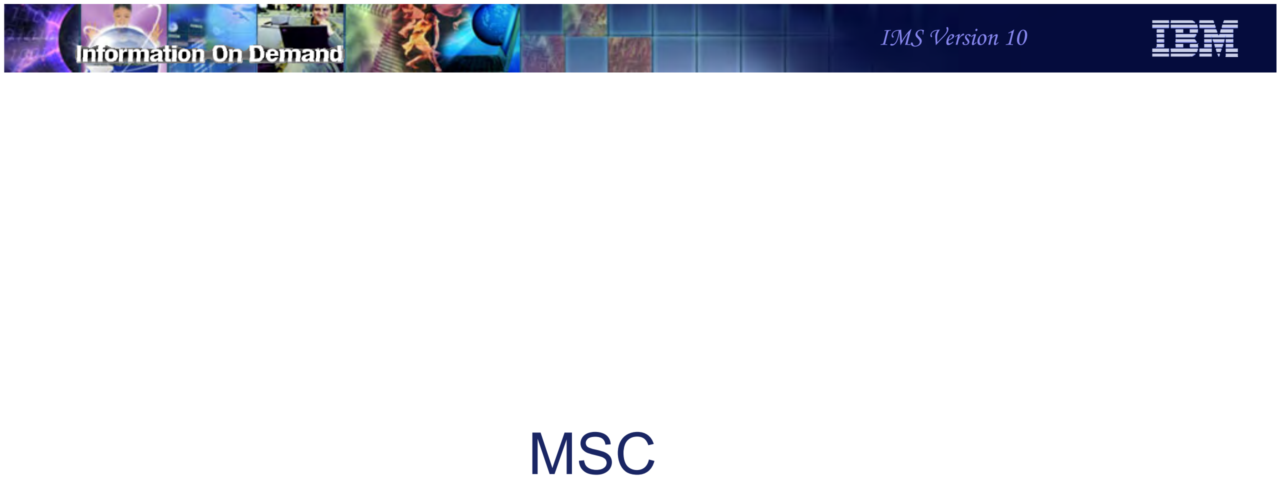
IMS Version 10

Transaction Manager

- MSC
- APPC
- Removal of BTAM Support

Figure 9-1. Transaction Manager CMA01.0

Notes:



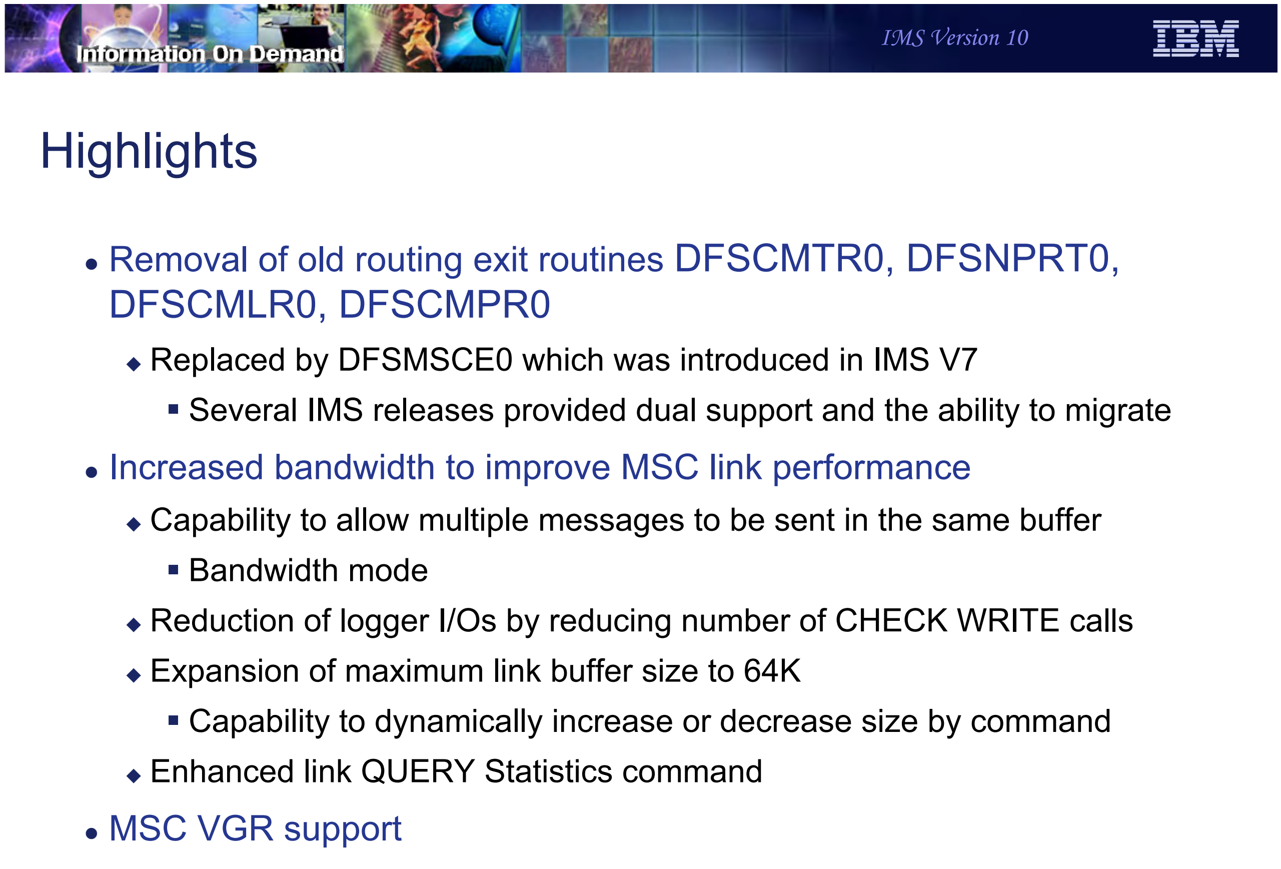
Information On Demand

IMS Version 10

MSC

Figure 9-2. MSC CMA01.0

Notes:


Information On Demand
IMS Version 10
IBM

Highlights

- Removal of old routing exit routines DFSCMTR0, DFSNPRT0, DFSCMLR0, DFSCMPR0
  - Replaced by DFSMSCE0 which was introduced in IMS V7
    - Several IMS releases provided dual support and the ability to migrate
- Increased bandwidth to improve MSC link performance
  - Capability to allow multiple messages to be sent in the same buffer
    - Bandwidth mode
  - Reduction of logger I/Os by reducing number of CHECK WRITE calls
  - Expansion of maximum link buffer size to 64K
    - Capability to dynamically increase or decrease size by command
  - Enhanced link QUERY Statistics command
- MSC VGR support

Figure 9-3. Highlights CMA01.0

Notes:

IMS V10 removes the routing exit routines that were replaced by the TM and MSC Message Routing and Control User Exit (DFSMSCE0) beginning in IMS V7. DFSMSCE0 consolidated and replaced the following exits: MSC Terminal Routing Exit (DFSCMTR0), MSC Input Message Routing Exit (DFSNPRT0), MSC Link Receive Routing Exit (DFSCMLR0), and the MSC Program Routing Exit (DFSCMPR0). Although the older exit routines were supported concurrently with DFSMSCE0 in V7, V8, and V9, they are now being removed in V10 and will no longer be supported or called. DFSMSCE0 must be used. Note: DFSMSCE0 can be used in both MSC and non-MSC environments, although not all routing options will apply to non-MSC systems.

To improve the performance and bandwidth requirements of high-volume MSC systems, many customers today use a large number of parallel MSC links between pairs of IMS systems. This scheme can complicate the operations of the IMS systems and adds to the complexity of the system configurations as well as load balancing schemes. IMS V10 introduces several enhancements to increase MSC bandwidth. These include:

- Blocking multiple messages and responses into a single buffer when sending messages across the MSC links. In previous releases, IMS only sent one message or response at a time.
- Reducing the logger I/O operations by reducing the number of CHECK WRITE calls.
- Expanding the maximum link buffer size to 64K (was 32K) so that more messages and responses can fit into a buffer. Additionally, the initial link buffer sizes set during the IMS system definition process can be increased or decreased and displayed dynamically with IMS commands.
- Providing a command to provide statistics on a link. A secondary issue associated with MSC bandwidth is the difficulty that IMS operators and system programmers can encounter when MSC link performance is inadequate due to a message backup or poor performance. Inadequate performance data in previous IMS releases often resulted in days or weeks of analyzing log records and traces to resolve the issue. In IMS V10, this issue is resolved with an enhanced link Query Statistics command.

Old MSC Routing Exits Removed

- DFSMSCE0 user exit replaces DFSCMTR0, DFSNPRT0, DFSCMLR0, DFSCMPR0
    - Old exits were still supported in V7, V8, V9
    - DFSMSCE0 contains all the routing capabilities supported by the old exits plus more capabilities
- System Definition – COMM macro
    - NOMSPEX/MSPEXIT and NOMSLEX/MSLEXIT parameters ignored
        - G092 warning message
    - Old exits are no longer called by IMS even if they are linked into the NUCLEUS or SDFSRESL

Figure 9-4. Old MSC Routing Exits Removed CMA01.0

Notes:

IMS V10 only supports the TM and MSC Message Routing and Control User Exit Routine (DFSMSCEO) for routing. There is no longer any support for the older routing exit routines DFSCMTR0, DFSNPRT0, DFSCMLR0, DFSCMPR0. DFSMSCE0 was introduced in IMS V7 to provide an opportunity to migrate over several releases.

The COMM macro specification during system definition, therefore, ignores the NOMSPEX/MSPEXIT and NOMSLEX/MSLEXIT parameters. Previous releases of IMS allowed specification of the parameters NOMSPEX/MSPEXIT to exclude/include the MSC Program routing exit (DFSCMPR0) and NOMSLEX/MSLEXIT to exclude/include the MSC Link Receive routing exit (DFSCMLR0). IMS no longer recognizes these routines even if they are included in the IMS resource library.

If the older exit routines still exist in the IMS release from which the migration is being done, then migration to DFSMSCE0 becomes a required action for the IMS V10 migration. The IMS Customization Guide documents a sample DFSMSCE0 exit along with the user edit parameter list macro (DFSMSCEP) which provides information on using and customizing the exit. To include the DFSMSCE0 replacement exit into the IMS system, it must be bound (linkedited) into IMS SDFSRESL (or a concatenated library).

Increased Bandwidth - Background

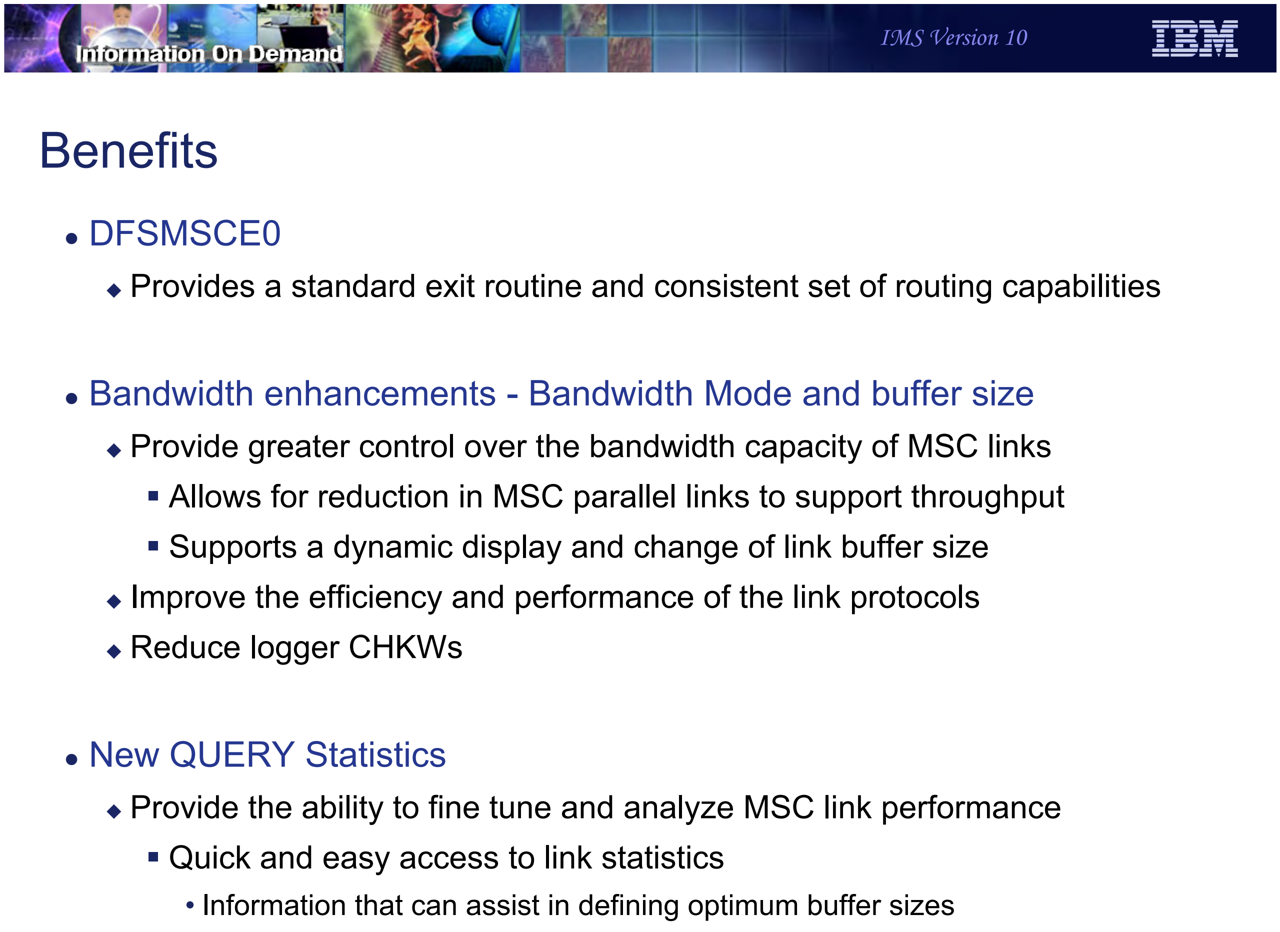
- Prior Releases
  - Input and output buffer to send messages on a physical link
    - Defined via BUFSIZE on MSPLINK macro at system definition
      - Applicable for the duration that the link is active
- Issues
  - Buffer size is fixed and may not account for increased traffic
  - Only one message or response can be sent per buffer
    - Even if the buffer is large enough to hold multiple messages/responses
  - The next message is not sent until the partner IMS responds that it has
    - Received, queued, and logged the message

Figure 9-5. Increased Bandwidth - Background CMA01.0

Notes:

To understand the changes for MSC bandwidth, a review of the issue is provided on this visual.

The definition of MSC physical links provides a specification for a BUFSIZE on the MSPLINK system definition macro. The buffer sizes specified are fixed at system definition. When the link is started, an input and output buffer is acquired and held for the duration of the link restart (i.e. link active), at the specified size to send and receive data (messages) across the link. Since each side (partner) of the link has a send and receive buffer, a message or response may be simultaneously sent each way. However, only one message or response is sent per buffer, even if the buffer is large enough to hold multiple messages/responses. Another message is not sent until the partner IMS responds that it has received, queued, and logged the message. For high-volume systems, the wait associated for a freed buffer can be unacceptable. To get around this issue, high-volume systems oftentimes are defined with a large number of parallel links to support the concurrent traffic from one IMS to another.



Information On Demand

IMS Version 10

IBM

Benefits

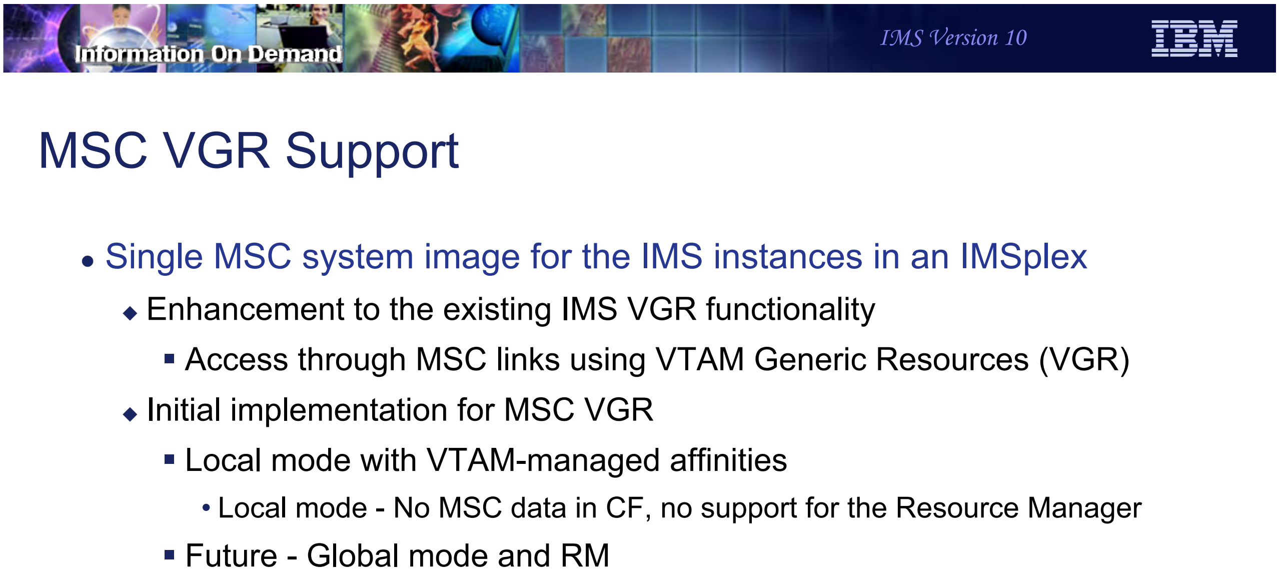
- DFSMSCE0
  - Provides a standard exit routine and consistent set of routing capabilities
- Bandwidth enhancements - Bandwidth Mode and buffer size
  - Provide greater control over the bandwidth capacity of MSC links
    - Allows for reduction in MSC parallel links to support throughput
    - Supports a dynamic display and change of link buffer size
  - Improve the efficiency and performance of the link protocols
  - Reduce logger CHKWs
- New QUERY Statistics
  - Provide the ability to fine tune and analyze MSC link performance
    - Quick and easy access to link statistics
      - Information that can assist in defining optimum buffer sizes

Figure 9-6. Increased Bandwidth… CMA01.0

Notes:

The MSC bandwidth mode in IMS V10 is a mechanism to determine whether or not to send multiple messages in one buffer. In non-bandwidth mode, MSC sends a maximum of one message or response per I/O operation (i.e. Send or write). In bandwidth mode, IMS attempts to maximize the capacity of a link by sending as many messages that are queued and ready to go, and responses that are owed for messages received, in the same buffer. By increasing the link buffer size, more and more messages and responses may be sent simultaneously.

Note that BANDWIDTH mode is not a system definition option. It can only be set ON with a command. The default is non-Bandwidth mode which allows MSC to function as in previous releases.


Information On Demand
IMS Version 10

MSC VGR Support

- Single MSC system image for the IMS instances in an IMSplex
  - Enhancement to the existing IMS VGR functionality
    - Access through MSC links using VTAM Generic Resources (VGR)
  - Initial implementation for MSC VGR
    - Local mode with VTAM-managed affinities
      - Local mode - No MSC data in CF, no support for the Resource Manager
    - Future - Global mode and RM

Figure 9-7. Increased Bandwidth… CMA01.0

Notes:

MSC bandwidth mode is only available on IMS 10 to IMS 10 connections.

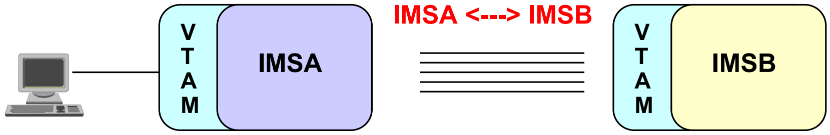
The bandwidth mode may be set ON/OFF by the IMS UPDATE command on a link by link basis for all the CTC, MTM, or VTAM links.

The specified mode must be the same on both sides of the link or the link restart will be rejected with a DFS3218 message. IMS V10 MSC links that are connected to a down level release of IMS (i.e. V8, V9) will not be able to set bandwidth mode on the V8/V9 side of the link so the link, in effect, will not be able to operate in bandwidth mode.

Information On Demand IMS Version 10 IBM

MSC VGR Support …

- MSC Environment - without VGR



- MSC Environment - with VGR

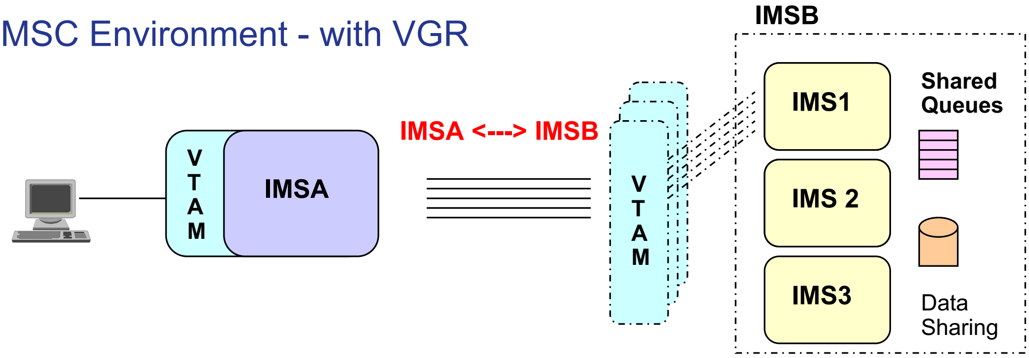


Figure 9-8. Increased Bandwidth… CMA01.0

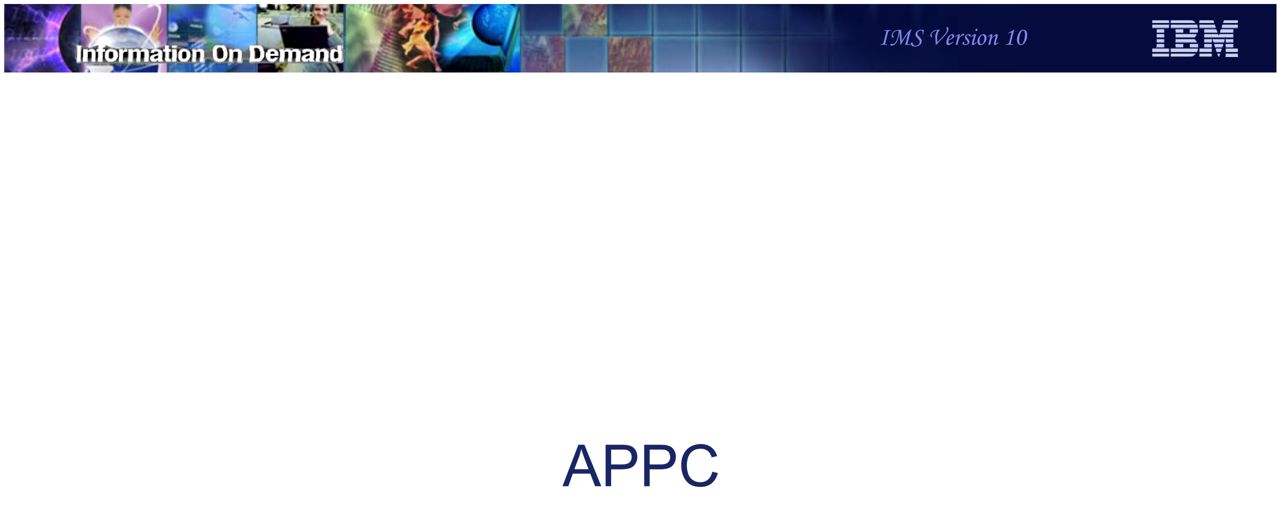
Notes:

Turning bandwidth mode ON can also impact logger I/O. As a general rule, there are 2 CHKWs in the path of a recoverable message that is sent across an MSC link, one on the send side when the last part of a message is sent, and one on the receive side when the received message is enqueued. There are no CHKWs issued for non-recoverable messages. Note, however that even though a message that is sent from a front-end IMS to a back-end system is non-recoverable (therefore no CHKWs), its response is always considered recoverable and will incur CHKWs. Therefore, recoverable remote transactions incur 4 CHKWs, two for the message going to the back-end and two for the response. Non-recoverable remote transactions incur 2 CHKWs, none for the non-recoverable message that is sent to the back-end, and two for the reply which IMS always sets to recoverable.

The choice of bandwidth mode, however, does impact the number of CHKWs that can be issued in an IMS V10 environment. Bandwidth mode potentially reduces the number of CHKWs issued because only one CHKW is written per send buffer regardless of how many messages are contained in the buffer. For example, if there are five messages in the buffer then only one CHKW is written rather than five. Note that on the receive side, the back-end

IMS continues to issue one CHKW per message. When the response messages are ready, the back-end IMS will attempt to buffer as many messages as can fit in a buffer and issue one CHKW for the buffer. The front-end IMS which is the receiving system for the responses will issue one CHKW per message.

Non-bandwidth mode provides the same processing as pre-IMS V10 systems. For example, if the front-end IMS has five messages to send, these messages are all sent in separate buffers and five CHKWs will occur, one for each message. On the receive side, IMS will again issue one CHKW per message. The same processing and number of CHKWs occur for the responses from the back-end IMS to the front-end system.



Information On Demand
IMS Version 10
IBM

APPC

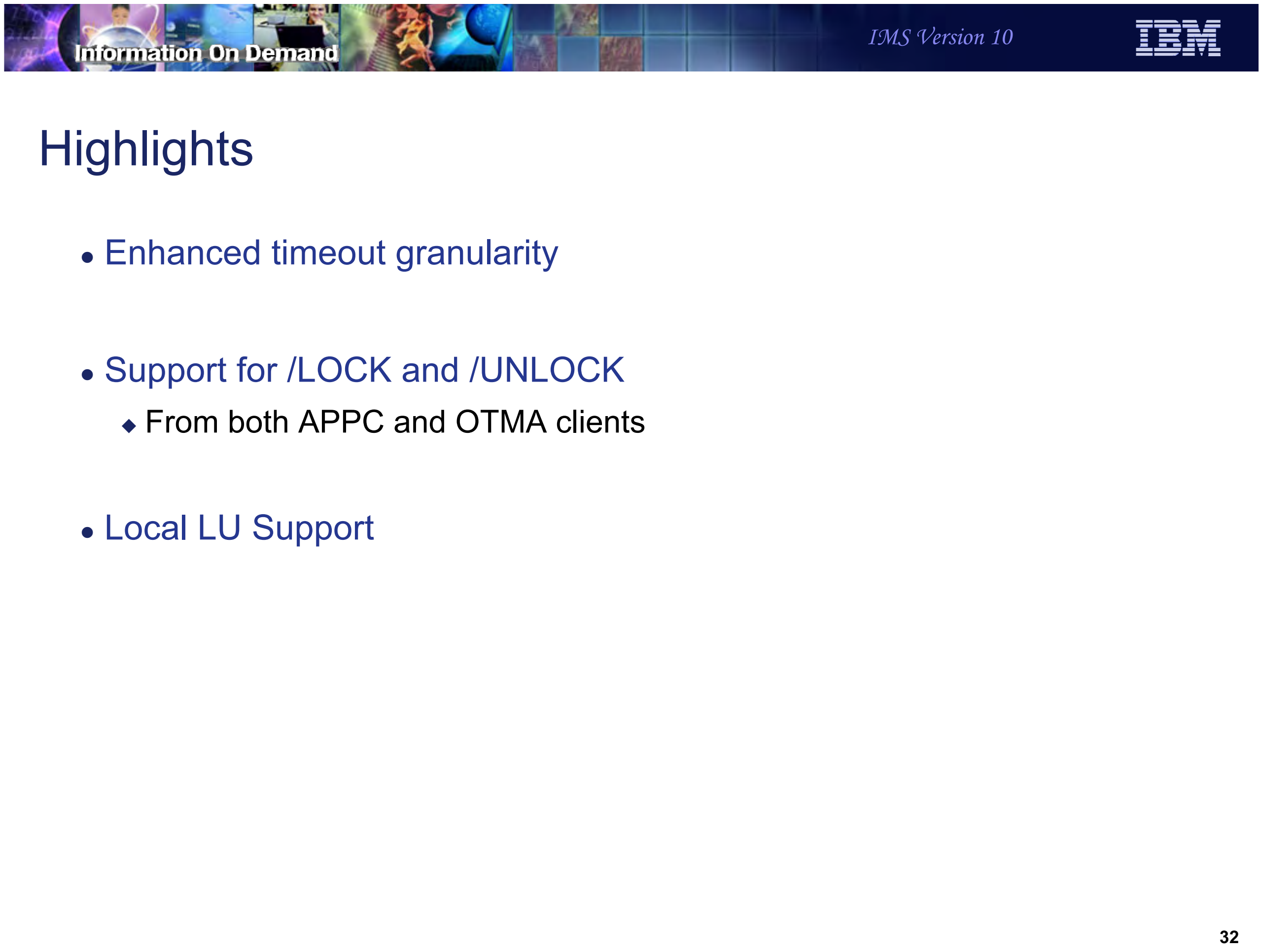
Figure 9-9. Increased Bandwidth CMA01.0

Notes:

IMS 10 increases the range of buffer size specifications. The new MSPLINK BUFSIZE range is from 1024 to 65536 and specifies the input and output buffer sizes for each logical link defined for a physical link. The range is standardized and applicable to CTC, MTM and VTAM links. Prior to version 10, the size for CTC and MTM links ranged from 160 to 32689 and from 208 to 30720 for VTAM links. Note that when bandwidth mode is used, a BUFSIZE of 1024 may be too small to take advantage of the ability to send multiple messages with one buffer especially if the messages include some of the extended headers that capabilities such as OTMA impose. A value of 4096 is more realistic.

Additionally, in prior releases IMS system definition added 78 bytes to the MSPLINK BUFSIZE for CTC and MTM links, and 288 bytes to VTAM links. IMS Initialization added another 28 bytes. This overhead has been removed in V10. IMS initializes the MSC buffers to the actual BUFSIZE specified. For V10 to be compatible with V9 and below, however, IMS will continue to add the overhead (i.e. 78+28 for CTC and MTM links, 288+28 for VTAM links), at link restart if the partner IMS is below a V10 level. This action insures that the buffer sizes are compatible. If the partner is at V10 or higher, the overhead is not added.

Also note that in prior IMS releases, VTAM required buffer sizes to reflect a formula of X times 2 to the power of Y, where X had to be a value of 8 through 15, and Y has to be a value from 3 to 13. This restricted the MSC VTAM buffer sizes to certain values that were documented in a VTAM table under the MSPLINK macro in the Installation Guide, Volume 2 manual. For V10, these restrictions are removed, but remain for compatibility when communicating to a V9 or earlier release of IMS**.** Any BUFSIZE from 1024 to 65536 is acceptable. The table in the manual has been updated to reflect the applicable sizes.



Information On Demand
IMS Version 10
IBM

Highlights

- Enhanced timeout granularity
- Support for /LOCK and /UNLOCK
  - From both APPC and OTMA clients
- Local LU Support

Figure 9-10. Increased Bandwidth - Implementation CMA01.0

Notes:

A label field has been added to the MSLINK macro to allow a name to be assigned to the logical link. The name, if provided, can be used for the new command capabilities that are provided to affect specific links. If not specified, IMS will generate the name DFSLxxxx where xxx is the link number. Link numbers, as in previous releases, continue to be assigned automatically by IMS based on the relative position of the MSLINK macro in the SYSGEN.

Enhanced Timeout Granularity

- Support for APPC/MVS timeout in seconds
  - Enhancement in z/OS V1R7
    - Prior releases provided the timeout capability in minutes
  - IMS enhancements to:
    - APPCIOT keyword in DFSDCxx member of IMS.PROCLIB
    - /CHANGE APPC TIMEOUT command

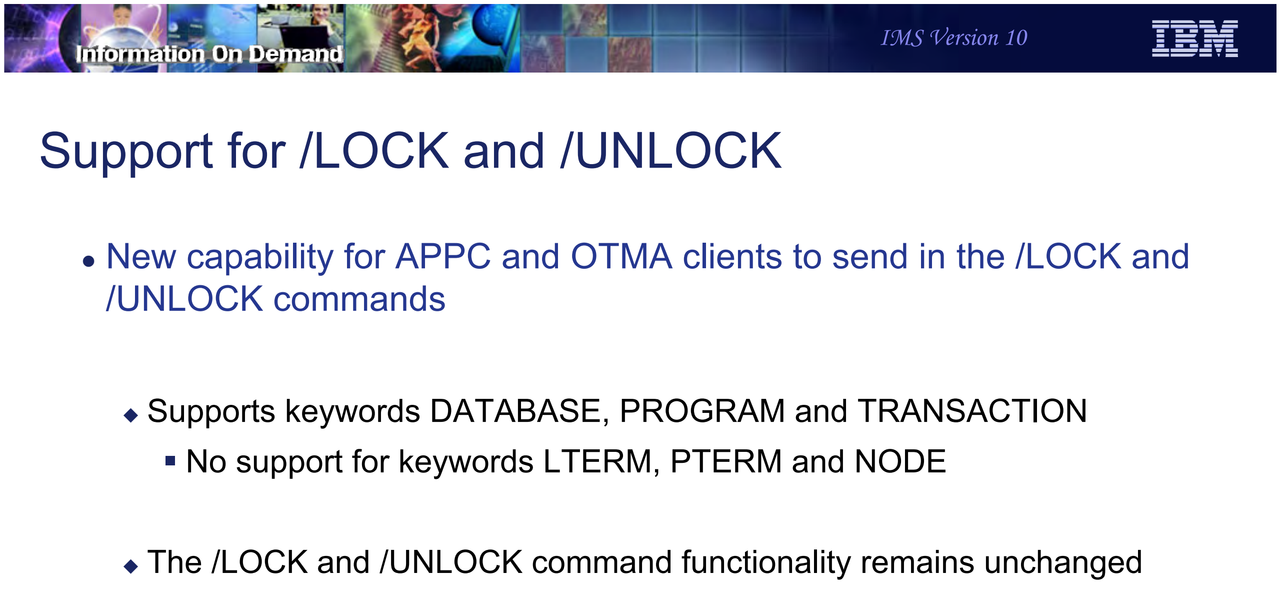
Figure 9-11. Increased Bandwidth - Commands CMA01.0

Notes:

To support the increased bandwidth enhancements, IMS V10 introduces several new command capabilities.

The update command has been enhanced to provide a mechanism to change the characteristics of the link associated with the name provided. The name is either the label value on the MSLINK definition or the default DFSLKxxx value. Possible actions that control the bandwidth capacity of the link include increasing and decreasing link buffer sizes, setting a higher limit for buffer size, and setting the bandwidth mode off and on. The command can be entered as either a Type2 (UPDATE) command which uses the Operations Manager (OM) or a Type1 (/UPD) which does not require OM. Any accepted changes are kept across a warm start.

If the characteristics of the link need to be changed, the link must first be stopped and idle. The UPDATE command, when issued, must also be issued on both IMS systems on either side of the link to reflect identical specifications for bandwidth mode and buffer sizes.



Information On Demand
IMS Version 10
IBM

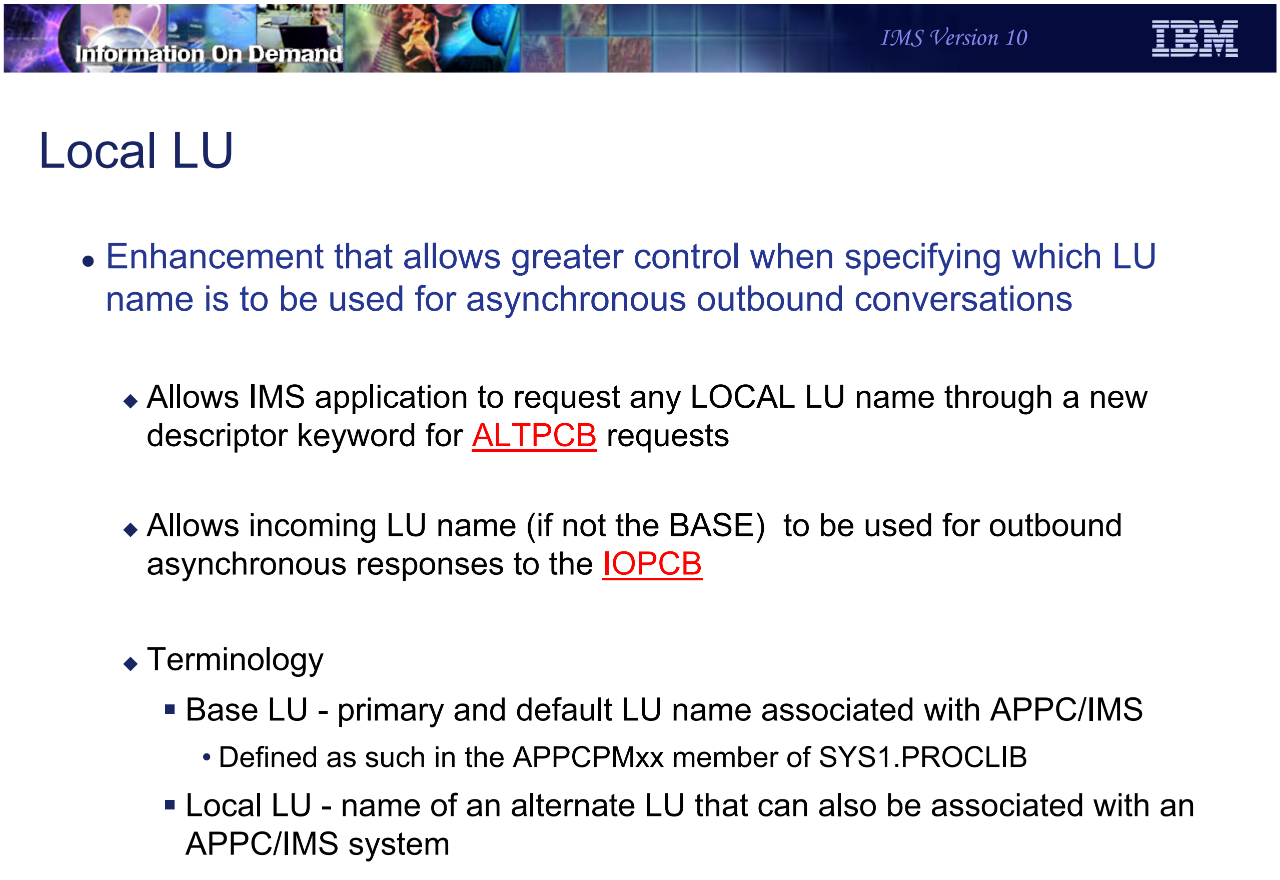
Support for /LOCK and /UNLOCK

- New capability for APPC and OTMA clients to send in the /LOCK and /UNLOCK commands
  - Supports keywords DATABASE, PROGRAM and TRANSACTION
    - No support for keywords LTERM, PTERM and NODE
  - The /LOCK and /UNLOCK command functionality remains unchanged

Figure 9-12. Increased Bandwidth - Commands … CMA01.0

Notes:

The /DISPLAY and QUERY commands show the links with their current buffer size and whether or not bandwidth Mode is on or off.


Information On Demand
IMS Version 10
IBM

Local LU

- Enhancement that allows greater control when specifying which LU name is to be used for asynchronous outbound conversations
  - Allows IMS application to request any LOCAL LU name through a new descriptor keyword for ALTPCB requests
  - Allows incoming LU name (if not the BASE) to be used for outbound asynchronous responses to the IOPCB
  - Terminology
    - Base LU - primary and default LU name associated with APPC/IMS
      - Defined as such in the APPCPMxx member of SYS1.PROCLIB
    - Local LU - name of an alternate LU that can also be associated with an APPC/IMS system

Figure 9-13. Statistics CMA01.0

Notes:

IMS V10 also provides a new capability to gather and show statistics on the MSC logical links.

A new log type x’4513’ is available to provide information on each logical link.

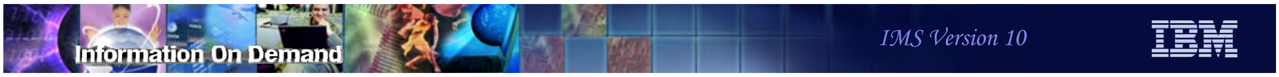
The QUERY command provides options to request statistics on the MSC links. This allows for quick and easy access to link performance. The information can be used to determine the efficiency of the link and assist in deciding on an optimum buffer size. Individual statistics for each MSLINK are collected in the DFSMSCWA workarea.

Note that link statistics are not displayed with the SHOW(ALL) keyword. SHOW(STATISTICS) must be used. This keyword also displays the statistics reset mode, RESET,CHKPT or NORESET,CHKPT.

There are three categories of statistics:

- General - such as, statistics start time, ITASK dispatch counts, ITASK processing times, and the rate and number of logger check writes

- Send - such as, messages sent, byte count sent, send message sizes, queue manager get counts and times, and send I/O times
- Receive - such as messages received, byte count received, receive message sizes, QMGR insert counts and times, and receive I/O times



Benefits

- Enhanced timeout capability
  - Releases resources more responsively
    - In less than one minute when needed
- Support for /LOCK and /UNLOCK
  - Enhances command capability for APPC clients such as the IMS Command Control Facility
    - Implemented for OTMA clients also
- Local LU capability
  - Provides greater control of the LU names that IMS uses for asynchronous outbound requests
  - Enhances the security environment

Figure 9-14. Statistics … CMA01.0

Notes:

The UPDATE MSLINK command can be invoked to reset the statistics counters and control how the resetting is to be done.

The first two forms of the command control the RESET mode:

- RESET,CHKPT mode causes the link statistics to be reset at each IMS checkpoint, after the statistics are logged. The recording interval, therefore, is from IMS checkpoint to checkpoint. This is the default mode and provides a reasonable interval for the statistics to be gathered. A longer interval might not be as useful in determining problems.
- NORESET,CHKPT mode does not reset the statistics at IMS checkpoint. To be reset, the operator must do so manually. The recording interval, therefore, is from IMS restart or when the last manual reset was issued until the time the command is issued. This mode is useful when running a benchmark or gathering statistics for a longer interval than between IMS checkpoints.

The third form of the command manually resets the link statistics, sets the start time to the current time, and begins a new recording interval.

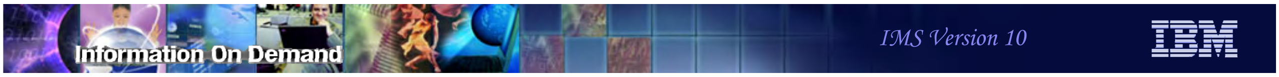
Information On Demand
IMS Version 10
IBM

Removal of BTAM Support

Figure 9-15. Statistics … CMA01.0

Notes:

The next two pages provide an overview of the statistics provided in the QUERY command. This visual shows the General Statistics.



Highlights

- IBM BTAM products were withdrawn from service several years ago
  - IMS continued to support BTAM through IMS V9
- IMS V10 removes BTAM support
  - Ignores all macro statements associated with the unsupported BTAM terminals during IMS system definition
    - Issues warning message

      G411 MACRO STATEMENT ASSOCIATED WITH AN UNSUPPORTED BTAM TERMINAL

      - A severity code of 2 is issued to allow system definition to continue
  - Devices such as Spool, Reader, Printer, Punch, Tape and Disk are not affected.

Figure 9-16. Statistics … CMA01.0

Notes:

This visual shows the type of information that can be analyzed by reviewing the Send and Receive statistics.

The Send statistics reference QGET calls. These calls are used to get messages off the queue for send processing.

The Receive statistics reference QPUT calls. These calls are used to put messages on the queue as part of receive processing.

Migration Considerations

- Affected macros that previously supported BTAM include:
  - CONFIG
  - CTLUNIT
  - DCLIST
  - IDLIST
  - LINE
  - LINEGRP
  - MSPLINK
  - NAME
  - POOL
  - STATION
  - SUBPOOL
  - TERMINAL
  - TYPE

Figure 9-17. Statistics - Report example CMA01.0

Notes:

When the QUERY MSLINK NAME(*name)* SHOW(STATISTICS) command is issued, all the statistics information on the link is displayed.

Figure 9-18. CMA01.0

Notes:

Several system definition messages have been changed to reflect the IMS V10 enhancements:

- G092 reflects the removal of the MSC exits on the COMM macro.
- G441 reflects the new BUFSIZE limits on the MSPLINK macro for V10 which must be in the range of 1024 to 65536 for all link types - CTC, MTM, and VTAM.
- G571 is issued if the specified MSLINK macro label field is not blank and is not a one-to-eight character alphanumeric name.

Additionally, Message DFS3218 has been expanded to support the bandwidth enhancements. When an MSC link is restarted, restart messages (referred to as a restart message and restart response message) are exchanged between the partner IMS systems. If the messages are invalid, unexpected, or contain incompatible information, a DFS3218 message will be issued on the IMS that detected the error, and the link restart will be aborted. Bandwidth support added more error conditions (e.g., Buffer sizes not equal, Bandwidth Mode not compatible on both sides). To more easily identify the error, a unique reason code was added for each error condition (for existing errors and new errors). The reason code identifies the module that detected the error and the error condition.

DFS3218 INVALID RESTART MESSAGE OR RESTART RESPONSE MESSAGE RECEIVED RSN=xxyy, LINK link#

Figure 9-19. Example - Bandwidth Mode Setup CMA01.0

Notes:

These next two visuals provide an example of MSC setup.

During system definition, three macros provide the mechanism to define the connection between 2 IMS systems. The MSPLINK macro has not changed but the range in values that can be specified for BUFSIZE reflect the new minimum and maximum limits. The MSLINK has changed to provide a label. In this example, the name is *LNK12V02* and is important because it will be used in the UPDATE command.

Once the system definition process is complete and IMS is initialized, the parameters on the MSLINK can be modified. Note that in this example, the UPD command identifies the specific MSLINK named *LNK12V02.* Two changes are made. The first is the setting of BANDWIDTH Mode to ON and the second is the increase of the BUFSIZE to 4096. Alternatively, the command can be issued as a Type 1 command using the /UPD form.

Figure 9-20. Example - Bandwidth Mode Setup … CMA01.0

Notes:

Once the changes to the link have been made, either the QRY command or the /DIS command can be used to display the link and ensure that the changes have actually been made.

Figure 9-21. Performance results CMA01.0

Notes:

This visual documents the results of a test comparing the use of Bandwidth Mode against a variety of buffer sizes. Note that the test results vary based on buffer size with a controlled message size and queue depth.

Figure 9-22. Migration Considerations CMA01.0

Notes:

When migrating to IMS V10, ensure that the older routing exit routines, if any, have been converted to DFSMSCE0. This migration can be accomplished in any of the supported previous releases (V7, V8, V9) prior to the actual IMS V10 migration. If the exit routines still exist while the V10 migration is in progress then the V10 migration tasks need to include the upgrade of the previous releases to DFSMSCE0.

A like-for-like migration from a previous release to IMS V10 allows MSC to be initialized in non-bandwidth mode. This mode is compatible with MSC operations in previous releases.

The ability to turn on bandwidth mode is provided via the UPD command and requires both partners of the link to be at an IMS V10 level. In this mode, the minimum size of the buffers should be 4096. To be more effective, the buffer sizes should be defined to accommodate multiple messages.

Figure 9-23. Migration Considerations … CMA01.0

Notes:

Note that the buffer size minimum and maximum ranges have changed in IMS V10. If the link is a VTAM link, the buffer size can now be any size defined in the range and no longer requires a size that fits into the previous formula of X times 2 to the power of Y. The MSPLINK macro description in the System Definition Guide still documents a table but only for compatibility purposes.

If the partner of the MSC link is a V8 or V9 IMS system then those definitions must be at a minimum of 1024.

Figure 9-24. Benefits CMA01.0

Notes:

The use of the TM and MSC Message Routing and Control User Exit (DFSMSCE0) routine consolidates the older message routing exits and provides a variety of enhanced options. By reducing the number of exit routines, a more consistent set of routing capabilities for all types of messages is provided along with a possible simplification of the coding and maintenance requirements.

Bandwidth is the rate at which data or messages can be sent and received across the MSC links. It is dependent on many factors such as: line speeds, processor speeds and other current activity going on in the system, I/O activity for logging and writing messages to the message queues, as well as the choice of protocols and buffer sizes. The bandwidth enhancements in IMS V10 address these many factors by improving the efficiency of MSC processing across the three link protocols - CTC, MTM and VTAM. Additionally, the new capability to view link statistics in the QUERY command can help determine how well the link is performing. The information can be used to optimize the buffer sizes based on actual traffic as messages flow through the system. This combination of being able to view the statistics and then dynamically change the link buffer sizes allows for more responsive control of the bandwidth capacity of MSC links.

Figure 9-25. MSC VGR Support CMA01.0

Notes:

IMS already supports VGR for other terminal types including ISC. This function adds the MSC environment as a local mode VGR with VTAM-managed affinities.

An affinity is a mapping of a node, and in the case of MSC all the parallel sessions associated with a link, to an IMS system.

Figure 9-26. MSC VGR Support … CMA01.0

Notes:

MSC VGR support provides the ability to grow the capacity and capabilities of the IMS MSC environment.

Figure 9-27. MSC VGR Support … CMA01.0

Notes:

The use of MSC VGR eases the requirements on the definitions that are needed. A remote IMS can use a single set of MSC link definitions to access any of the IMS systems in the VGR group. Likewise, IMS systems that are part of the local VGR group can clone definitions for MSC access to the single remote IMS.

Note that when VGR is used, all parallel sessions associated with the link are routed to the same IMS.

Figure 9-28. MSC VGR Support … CMA01.0

Notes:

MSC commands are available to terminate and restart the MSC links.

The information that MSC keeps about its sessions includes the link status, active or stopped, and the sequence numbers. If a planned outage is to occur, the parallel sessions on both sides of the link must be stopped and then restarted using the new link partner. To allow the information about sequence numbers to be cleared, the sessions need to be in cold or NRE status on both sides of the link.

On the other hand, if an IMS failure occurs where one IMS system terminates, the surviving IMS must also stop the parallel sessions. Note that if session status is changed from ERE to cold mode, messages could be duplicated or lost.

Figure 9-29. MSC VGR Support … CMA01.0

Notes:

If the links need to be re-established between the same two instances of IMS, then a way to accomplish this would be to first start the links on the instance of the IMS in the VGR group where the connection is desired. This action establishes affinity for all parallel sessions which remains in effect through subsequent restarts and until the last parallel session is terminated. Once the affinities are established, the links can also be started on the remote IMS system.

Figure 9-30. [No title] CMA01.0

Notes:

Figure 9-31. Highlights CMA01.0

Notes:

APPC enhancements include the following three areas:

- Greater level of timeout granularity with support for APPC/MVS timeout in seconds
- Ability for APPC and OTMA clients to issue the /LOCK and /UNLOCK commands
- Local LU Support

Figure 9-32. Enhanced Timeout Granularity CMA01.0

Notes:

The APPC/MVS capability for timeout has been supported by IMS since V6. In z/OS V1.R7, APPC/MVS enhanced its timeout support to specify a lower level of granularity. Prior z/OS releases supported timeout in minutes with the shortest value being one minute. Even one minute, however, can be too long a time when resources are held. With IMS V10 and z/OS V1R7, the timeout capability as provided in the first parameter of the APPCIOT keyword and in the /CHANGE APPC TIMEOUT command allow a value that can be specified in seconds.

Providing the ability to set timeout in seconds is particularly helpful when slow downs in the network occur or when APPC clients that are unable to respond in a timely manner cause IMS dependent regions to hang. Another benefit is one that could affect command processing when a slow or non-responding client impacts the IMS command task DFSCMTI0. For example, when an APPC node sends input for a transction that is stopped, IMS replies to this condition by sending a DFS065 message. This error message is sent under the IMS command task which waits for a response from the APPC node. When this node fails to respond, the task hangs until the wait is broken by the APPC/MVS timeout facility. This task, however, is quite vital in the IMS environment and plays a major role in handling most commands. When the condition just described happens, a slowdown in IMS throughput could result. An APPC timeout value of 1 minute, therefore, could be too long when suspending the command task in a production environment.

Figure 9-33. Enhanced Timeout Granularity - Implementation CMA01.0

Notes:

The APPCIOT keyword is an existing keyword in the DFSDCxxx member of IMS.PROCLIB. The first parameter of APPCIOT specifies the APPC/MVS time-out value expressed in minutes. This refers to an IMS wait for a requested APPC/MVS service to complete. IMS takes advantage of an APPC/MVS service to enforce the timeout with valid values between 0 and 1440 minutes. The second parameter which was introduced in IMS V9 provides an APPC/IMS timeout. This refers to an APPC/MVS wait for an IMS process to complete. For example, the timer begins when IMS receives a message from APPC/MVS and is reset when another APPC verb is issued, e.g., send of a reply, error message or deallocate. This second value is similar to putting a timeout on a queue to queue response. Like the first parameter, valid values are between 0 and 1440 minutes.

IMS V10 provides an enhancement for the first parameter which affects the APPC/MVS timeout specification. A value in this parameter can now be specified as (# of minutes : # of seconds). Note that the second parameter remains unchanged.

Similarly, the /CHANGE APPC TIMEOUT command has also been enhanced to provide a specification in minutes and seconds. Additionally, since the command can be issued during IMS execution, the value can be dynamically change as needed.

Figure 9-34. Support for /LOCK and /UNLOCK CMA01.0

Notes:

The restriction to sending in the /LOCK and /UNLOCK commands from APPC and OTMA clients has been lifted. The support allows the DATABASE, PROGRAM, and TRANSACTION keywords but restricts the use of LTERM, PTERM and NODE. Note that the commands themselves are not changed.

Figure 9-35. Support for /LOCK and /UNLOCK … CMA01.0

Notes:

This visual gives an example of an APPC client that specifies the /LOCK command as a TPNAME on the ALLOCATE request and the remainder of the command in the subsequent SEND_DATA verb. IMS responds with a DFS058 LOCK COMMAND COMPLETED which simply means that the command was accepted and processed by IMS. Note that the capability to send IMS commands from an APPC client is not new.

Figure 9-36. Support for /LOCK and /UNLOCK … CMA01.0

Notes:

Similarly, OTMA clients such as IMS Connect and MQ can also send in the command requests. Each OTMA client provides its own interface for remote applications. The example on this visual shows a request coming in from a remote application through IMS Connect.

Figure 9-37. Local LU CMA01.0

Notes:

The APPC/IMS support which was introduced in IMS V4 provided the ability to define both a primary VTAM APPC LU name to be associated with IMS as well as multiple secondary LU names. The primary name is defined as the BASE LU and the alternate names as LOCAL LUs.

A Base LU is specified as such in the APPCPMxx member of SYS1.PROCLIB which contains the APPC/MVS definitions. Local LUs must also be specified in the same member as being associated with APPC/IMS. Note that an LU must be unique to an IMS system in an IMSPlex.

IMS V10 enables IMS applications that use the alternate TP PCB (ALTPCB) for APPC asynchronous outbound requests to specify which LU name to use on the outbound ALLOCATE request through a new keyword in the DFS62DTx descriptor. Additionally, a new startup parameter APPCLLU provides the ability to specify whether or not the incoming LU name (if different from the Base LU) is to be used for any associated asynchronous IOPCB outbound conversations.

As a reminder, asynchronous APPC conversations are those that are sent without waiting for a reply. The verb set used is Allocate, Send_data, Deallocate. Synchronous conversations, on the other hand, are those that send a message and wait for a reply using the verb set Allocate, Send_data, Receive_and_wait, Deallocate. The Local LU support in this section applies only to asynchronous conversation requests.

Figure 9-38. Local LU … CMA01.0

Notes:

A new OUTBND keyword has been added to the DFS62DTx IMS.PROCLIB member. IMS applications that produce ALTPCB messages control where and how the message is sent through specifications in an LU62 descriptor that is identified with the same name as the ALTPCB destination. IMS uses the information to create the appropriate APPC requests. If the OUTBND keyword is present and the Local LU specified is valid for this IMS then the outbound ALLOCATE is sent using the Local LU name. The partner APPC application sees the same name when it ACCEPTs the conversation. If the OUTBND keyword is not available, then the default Base LU name is used. Note: The DLI API has not been changed to add the OUTBND keyword to the CHNG call's LU 6.2 options.

The LU 6.2 Edit Exit Routine (DFSLUEE0), if it exists, is always called for inbound and outbound conversations managed by IMS. Word 13 in the exit interface points to the Local LU name and can be examined as well as modified by the exit code.

Figure 9-39. Local LU … CMA01.0

Notes:

The example in this visual shows how specifying different Local LUs in the OUTBND parameter of two descriptors results in the remote partner seeing the different names. If the remote partner is IMS then the target application PGM1, in this example, sees a different LTERM name for each message.

Note that an APPC/IMS implicit environment is one where IMS functions as the LU 6.2 partner on behalf of the IMS application program. In this environment, IMS places the Partner LU name in the LTERM field of the IOPCB. A possible benefit of using the OUTBND capability is that the receiving IMS application can branch to different logic based on the LTERM name.

The remote partner can be any LU 6.2 partner.

Figure 9-40. Local LU … CMA01.0

Notes:

IMS V10 also provides a new startup parameter in the DFSDCxxx member of IMS.PROCLIB. APPCLLU affects asynchronous outbound messages that are inserted to the IOPCB. For this situation to occur, the message that the IMS application processed with a GU IOPCB had to originate in an APPC partner as an asynchronous inbound message.

Figure 9-41. Local LU … CMA01.0

Notes:

The two examples on this visual illustrate the difference between the APPCLLU specifications of N and Y.

In the first example, APPCLLU=N, the IOPCB asynchronous output reply is sent using the Base LU name associated with IMS2 regardless of which LU name was used for the inbound asynchronous conversation.

In the second example, APPCLLU=Y, IMS2 uses the Local LU name of IMSA which is the same name used on the inbound asynchronous request. This capability is of value when the remote application is designed to expect a response from a specific partner LU name.

Figure 9-42. Local LU … CMA01.0

Notes:

Some other enhancements provide support for the Local LU capability.

The /CHANGE DESC *descriptor* OUTBND *locallu* command modifies the descriptor to use the specified *locallu* name for any new outbound asynchronous conversations associated with the specified descriptor. Note that this impacts messages that use the descriptor after the command has been issued. Messages that are already on the queue but have not yet been sent are not affected by the change.

The /DISPLAY DESC command has also been enhanced to support the Local LU. If the OUTBND keyword exists in a descriptor then its value is displayed under the OUTBNDLU column. If the value for a descriptor displays as blanks then the Base LU will be used.

Figure 9-43. Local LU … CMA01.0

Notes:

Some areas to consider include the following:

Remote partners that receive an asynchronous conversation request from IMS have the ability to determine the LU name that IMS uses when sending a request. The Local LU capability provides the opportunity for different names to be provided.

Note that LU names are specific to IMS systems and cannot be shared within IMSplex members.

Any modifications that occur as a result of a /CHA command affect new messages that are created after the change.

Figure 9-44. Local LU Security CMA01.0

Notes:

When an inbound conversation in a secured environment is allocated with APPC/MVS, a security check is done using the target LU name for IMS (Base or Local). If authorization is granted, APPC/MVS passes the RACF object to IMS. The RACF object, similar to an ACEE, is used by the IMS security call to check if the userid is authorized to the transaction before scheduling the request. In the situation where the message was sent to IMS using a Local LU, the Local LU name is used. Once this is done, the RACF object is deleted. If RACF=FULL has been specified for APPC/IMS, the ACEE must again be built for the dependent region. Prior to IMS V10, the ACEE is always built using the Base LU. If a mismatch occurs because a secured message was sent in using a Local LU but the dependent region ACEE, using the Base LU, does not authorize the user, then the queued transaction will fail authorization. IMS V10 addresses this situation by using the applicable LU (Base or Local) that was used for the inbound message when building the dependent region ACEE.

Figure 9-45. Benefits CMA01.0

Notes:

The enhanced timeout granularity that provides values in the seconds range allows resources to be released more quickly when needed. The APPCIOT specification which is set at IMS initialization can be overriden dynamically by the /CHA command.

The support for /LOCK and /UNLOCK commands enhances the ability for tools such as the IMS Command Control Facility (CCF) which uses APPC to send commands to IMS.

The Local LU enhancement provides greater flexibility for the IMS environment to support multiple LU names. This is beneficial for application designs that rely on different partner names to trigger different events.

Figure 9-46. Removal of BTAM Support CMA01.0

Notes:

Figure 9-47. Highlights CMA01.0

Notes:

Although IBM withdrew marketing and service of BTAM products several years ago, IMS continued to support the BTAM macros through IMS V9. IMS V10 removes this support. The following list shows the device types affected:

BTAM Device Type	Comments or Other Specifications
1050	Switched Terminal
2740	Non-Station-Control
2740	Non-switched, model 1
2740	Non-switched, model 2
2740	Switched Terminal, model 1
2741	Non-switched
2741	Switched Terminal

2260	Local
2780	
3270	Remote, Non-switched
3270	Local
3270	Switched Terminal
3275	Switched Terminal
3741	Switched Terminal
SYSTEM/3	
SYSTEM/7	BSC, BSC and Contention
SYSTEM/7	Start/Stop, Start/Stop and Contention

Warning message G411 will be issued if the macro statement operand has an unsupported BTAM terminal specification during the IMS STAGE 1 system definition process. In addition, a severity code of 2 will be issued to allow system definition to continue. This warning message and severity code will be documented in the IMS V10 Messages and Codes manual.

Figure 9-48. An Example CMA01.0

Notes:

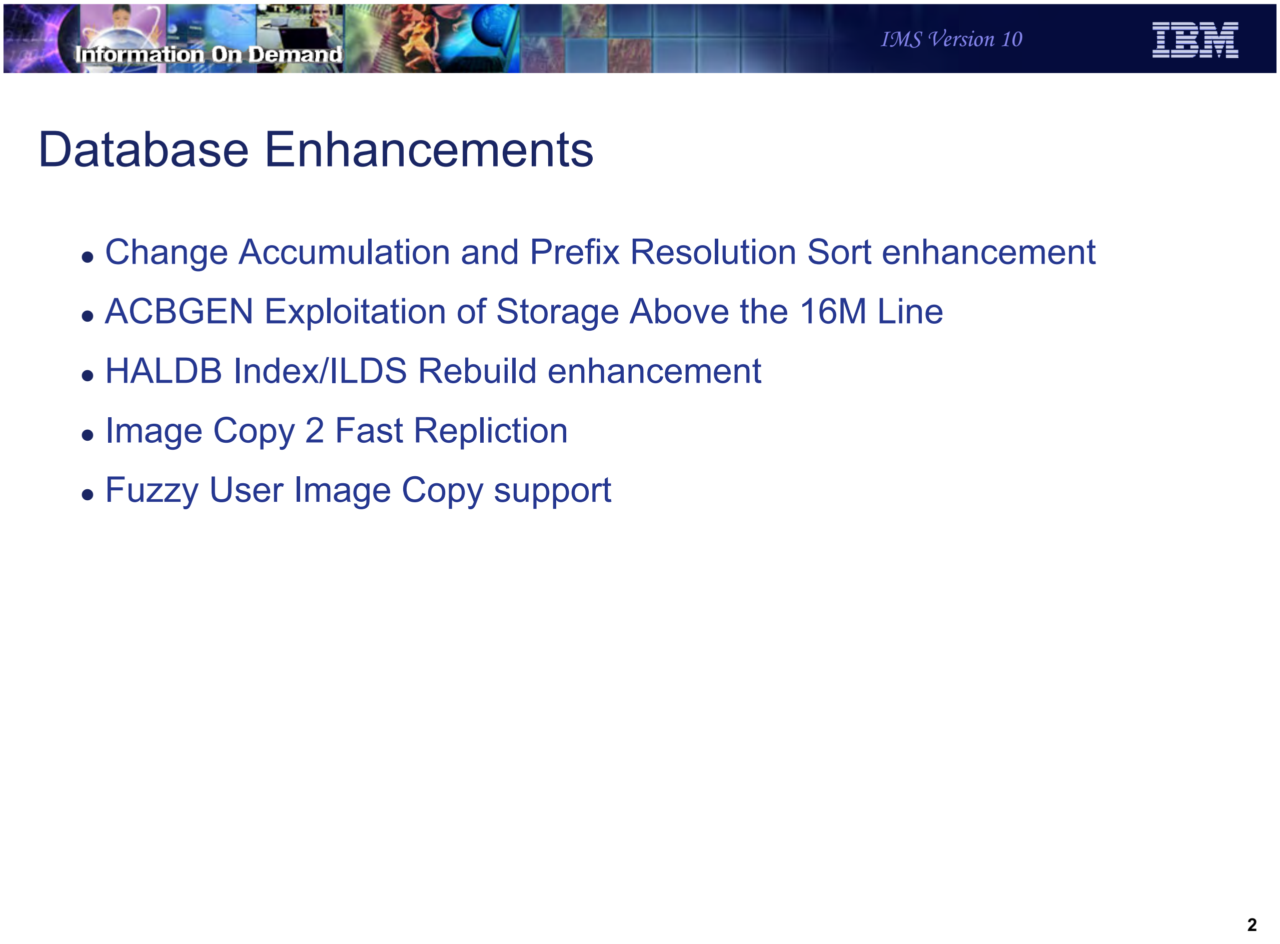
This example shows the warning message that is issued when attempting to run the system definition process specifying an unsupported BTAM device.

Figure 9-49. Migration Considerations CMA01.0

Notes:

This page lists the different IMS macros that previously supported BTAM specifications.

Unit 10.Database

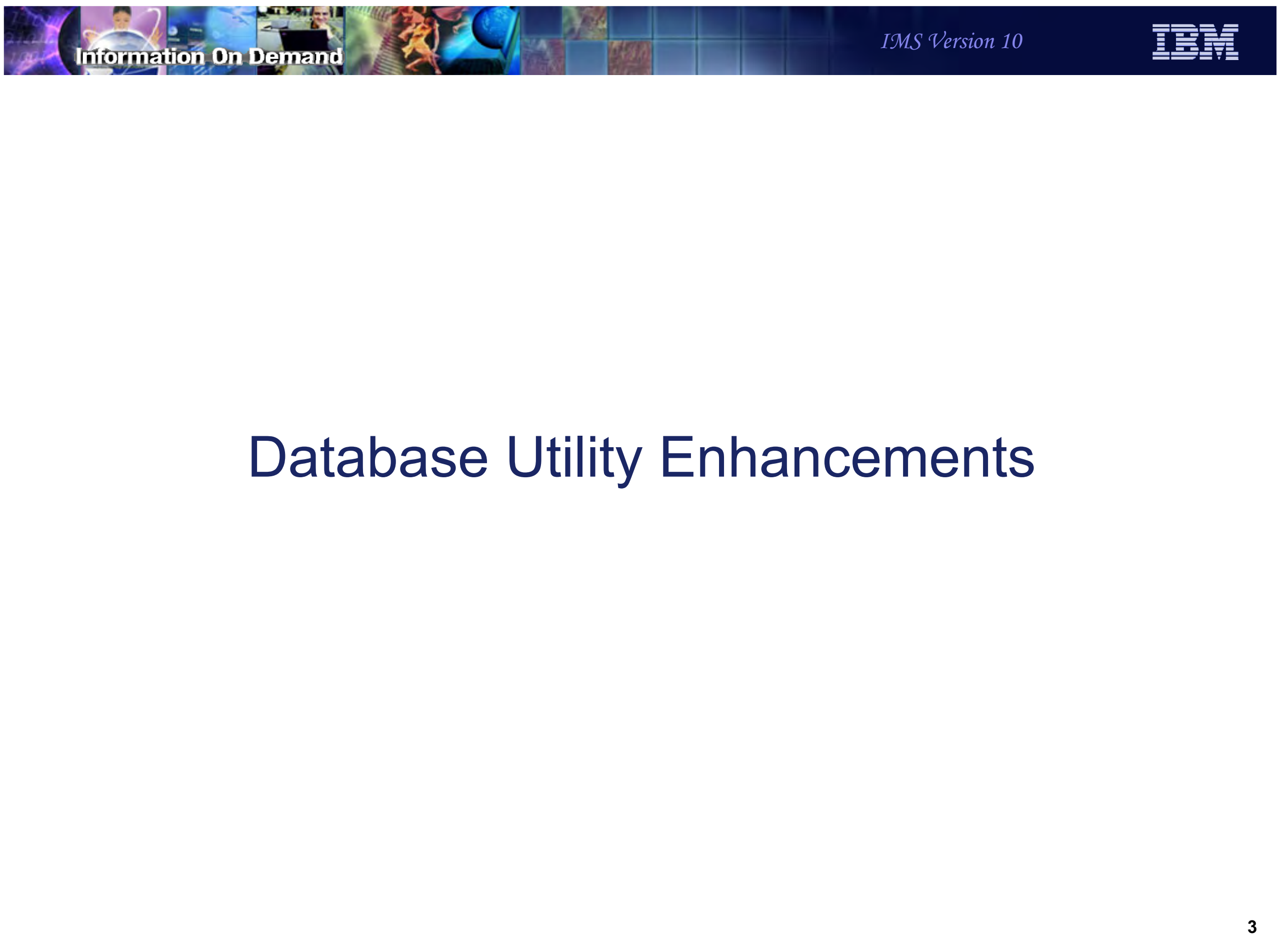


Database Enhancements

- Change Accumulation and Prefix Resolution Sort enhancement
- ACBGEN Exploitation of Storage Above the 16M Line
- HALDB Index/ILDS Rebuild enhancement
- Image Copy 2 Fast Repliction
- Fuzzy User Image Copy support

Figure 10-1. Database Enhancements CMA01.0

Notes:


Information On Demand
IMS Version 10
IBM

Database Utility Enhancements

3

Figure 10-2. [No title] CMA01.0

Notes:

Change Accum and Prefix Resolution Sort

- Default sort size changed from 200K to MAX
  - CORE= parameter in JCL for Change Accumulation and Database Prefix Resolution
    - CORE= specifies the storage used by sort
      - MAX allows the installation default value for sort to be used
    - CORE limits the number of sort work data sets used by OEM sort programs
      - UCBs for these data sets are below 16M line for some OEM sort programs
- Benefit
  - Allows more sort work data sets to be used with OEM sort programs which create UCBs below the 16MB line
    - This could shorten the elapsed execution time of the utilities

Figure 10-3. Change Accum and Prefix Resolution Sort CMA01.0

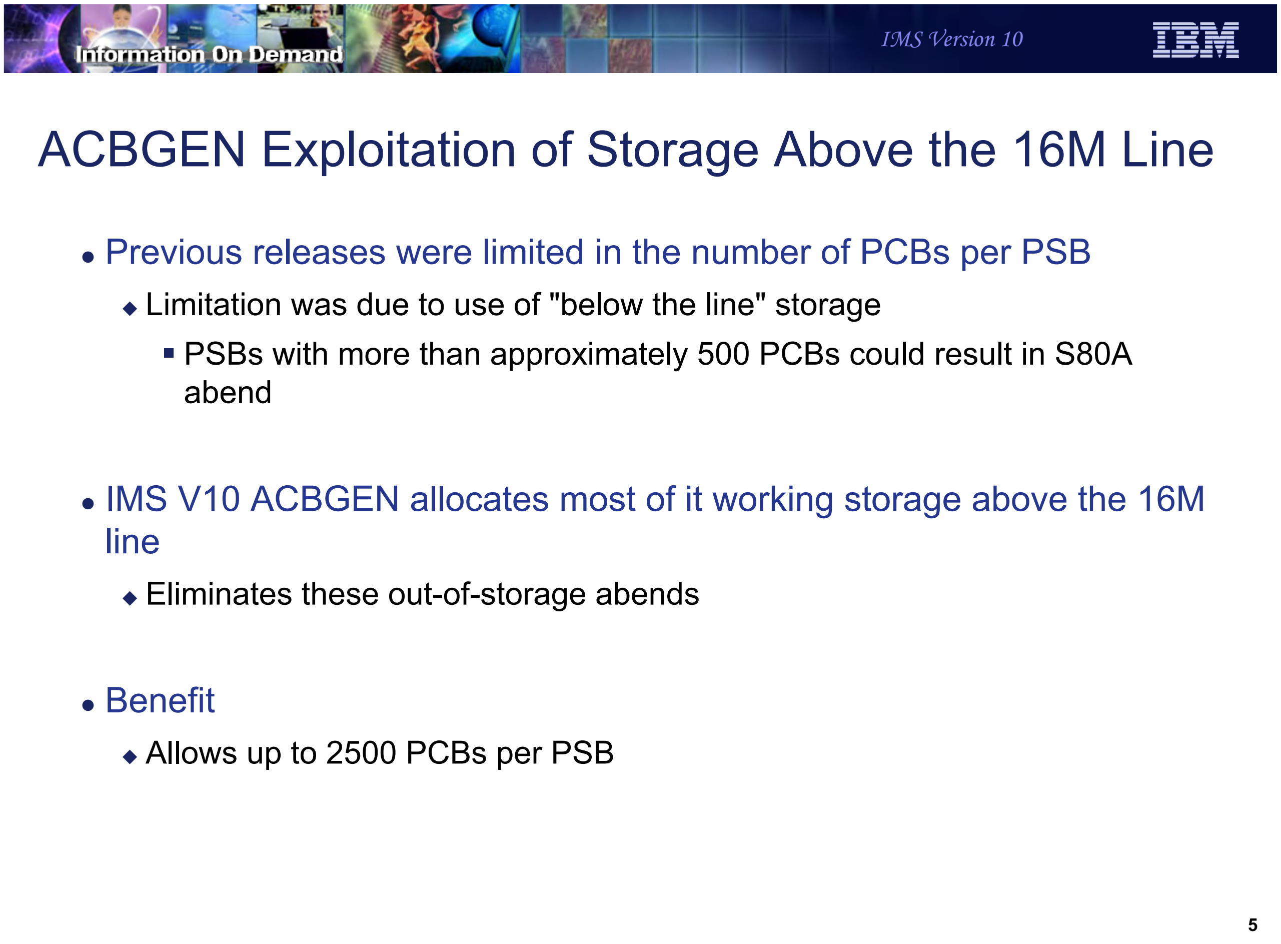
Notes:

The default for the CORE= parameter on the Database Change Accumulation and Database Prefix Resolution utilities is changed in IMS V10 from 200K to MAX. When MAX is used, the IMS utility does not limit the amount of space used by sort. It is limited by the default value specified for the sort product. The sort space is particularly important for some non-IBM sort programs which create UCBs for sort work data sets below the 16M line. The sort space may limit the number of these data sets that may be used. By increasing the default space, the number of sort work data sets may be increased. This may shorten the elapsed execution time for these utilities.

A new message has been added to the Database Change Accumulation and Database Prefix Resolution utilities. It is issued when the CORE= parameter is not specified. The message is:

```
DFS3007I SORT CORE SIZE WILL DEFAULT TO CORE=MAX
```

If an installation wants to limit the sort space to less than the system default, it should code the CORE= parameter for the desired value.


Information On Demand
IMS Version 10
IBM

ACBGEN Exploitation of Storage Above the 16M Line

- Previous releases were limited in the number of PCBs per PSB
    - Limitation was due to use of "below the line" storage
        - PSBs with more than approximately 500 PCBs could result in S80A abend
- IMS V10 ACBGEN allocates most of it working storage above the 16M line
    - Eliminates these out-of-storage abends
- Benefit
    - Allows up to 2500 PCBs per PSB

5

Figure 10-4. ACBGEN Exploitation of Storage Above the 16M Line CMA01.0

Notes:

ACBGEN has been modified to allocate most of its working storage above the 16M line. In previous releases, this working storage was allocated below the line. Very large PSBs could require more storage than was available below the line. Requests for more storage than was available resulted in S80A abends with RC=10 in previous releases. Since V10 allocates storage above the line, it is not limited to the available storage below the line. This should eliminate these abends. Very large PSBs, including those with up to 2500 PCBs, are handled by ACBGEN in V10.

The limitations of 2500 PCBs per PSB and 30,000 SENSEGs per PSB are enforced by the macros used in PSBGENs.

Information On Demand
IMS Version 10
IBM

HALDB ILDS Rebuild Enhancement

- New option for HALDB Index/ILDS Rebuild utility (DFSPREC0)
  - ILDS entries (ILEs) are written to ILDS sequentially in load mode
    - Avoids CI/CA splits during rebuild
    - Creates free space according to data set definition
  - ILDS entries are sorted in data spaces before they are written
  - Invoked with RECOVTYP=ILEF or BOTHF
    - Uses 4 data spaces of 2 GB each
  - Free space creation may result in larger ILDS
- Benefits
  - Faster execution of DFSPREC0 when rebuilding an ILDS
  - Free space may improve performance of subsequent reorganizations
    - Reduce CI/CA splits

Figure 10-5. HALDB ILDS Rebuild Enhancement CMA01.0

Notes:

The HALDB Index/ILDS Rebuild utility (DFSPREC0) is used to rebuild either a PHIDAM primary index, an ILDS, or both for a partition. This enhancement applies only to the rebuilding of the ILDS. When an ILDS is rebuilt, the utility reads the partition database data sets looking for segments which are targets of secondary indexes or logical relationships. An indirect list entry (ILE) is written to the ILDS for each target that is found. In previous releases, these writes are done when the target segments are found. This means that the writes are random. They must be done in update mode. The enhancement provides a new way of writing the ILEs. When using the enhancement, ILE entries are not immediately written to the ILDS. Instead, they are stored in data spaces, sorted in ILE key sequence, and then written. The writes are done in load mode. These writes are sequential. Load mode avoids CI/CA splits. These two characteristics can significantly reduce the elapsed time of the writes. Load mode also creates free space as defined in the IDCAM DEFINE for the ILDS.

The new option is invoked with new values for the RECOVTYP parameter. Previously, the valid parameter values were INDEX for rebuilding an PHIDAM index, ILE for rebuilding an ILDS, or BOTH for rebuilding both data sets. The new additional valid values are ILEF for rebuilding the ILDS sequentially or BOTHF for rebuilding both data sets and rebuilding the ILDS sequentially. The “F” in these values signifies that free space will be created.

Some installations limit the number of data spaces that address spaces may create. This is done with an z/OS IEFUSI exit. The default IEFUSI exit for z/OS allows the creation of the 4 data spaces used by DFSPREC0. If your installation has written its own exit, it may not allow 4 data spaces for most address spaces. In this case, you would need a modification to the exit to use the new option.

Since the ILDS that is rebuilt with the new option includes the defined free space, it may create a larger data set than was created in the past with this utility or by the HD Reload utility. The ILDS must be deleted and redefined (DELETE/DEFINE) before the execution of the utility. This is true with the old ILE and BOTH options and with the new ILEF and BOTHF options.

The use of the new option could significantly reduce the elapsed time required for rebuilding ILDSs. The creation of free space in the ILDS could improve the performance of subsequent HD Reload or Online Reorganization executions since the free space should reduce the need for CI/CA splits.

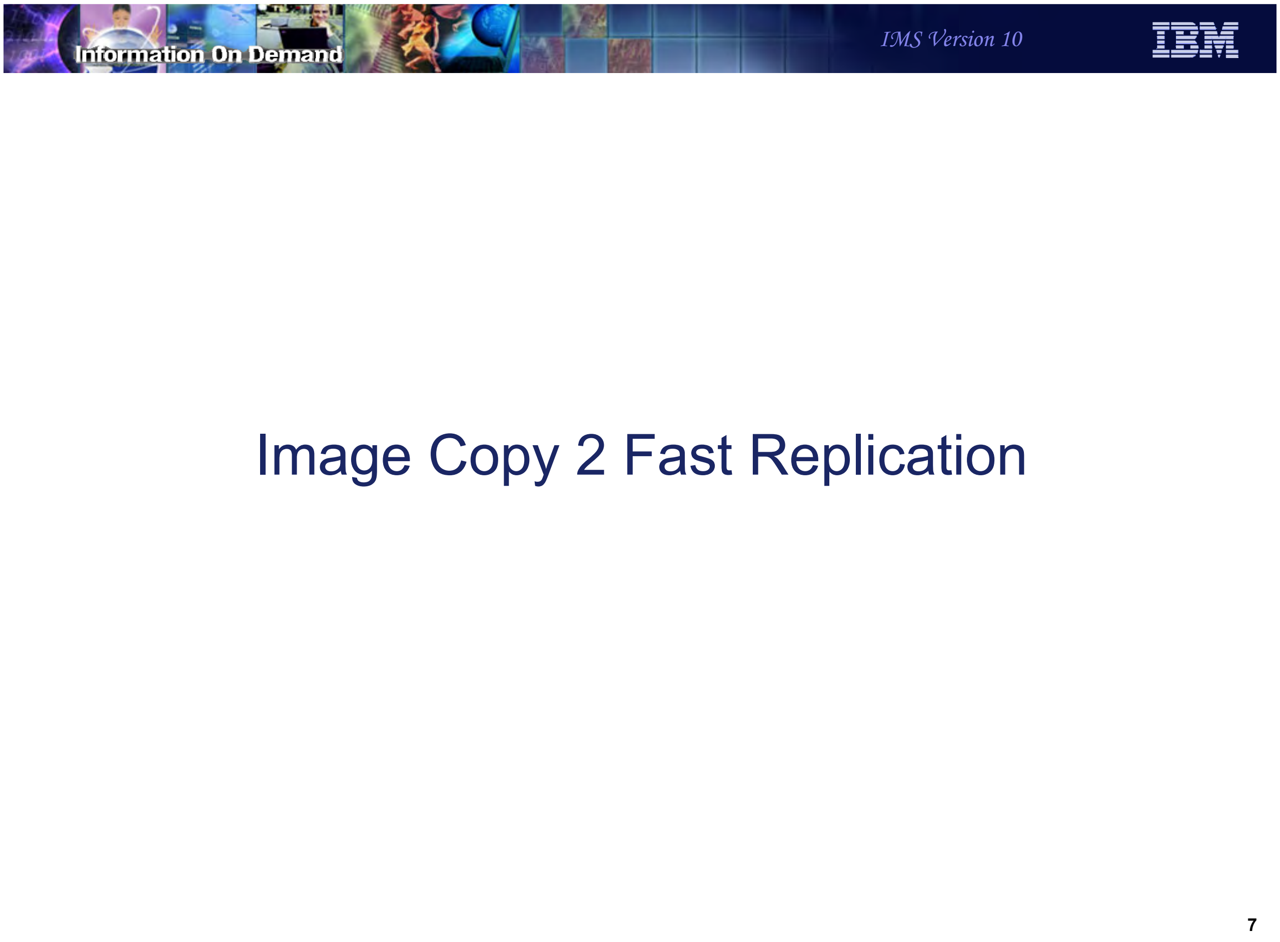
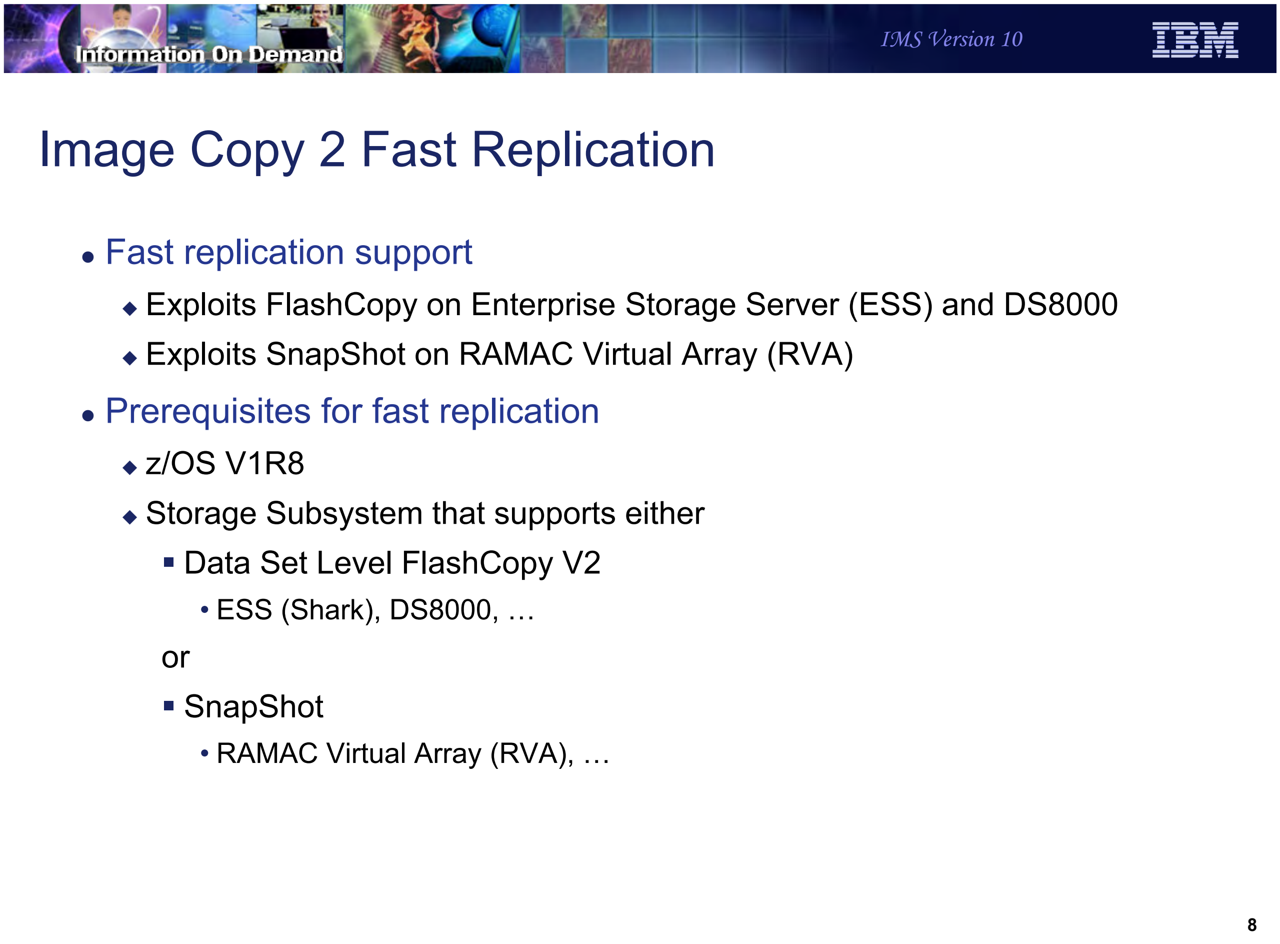

Information On Demand
IMS Version 10
IBM

Image Copy 2 Fast Replication

Figure 10-6. [No title] CMA01.0

Notes:



Information On Demand

IMS Version 10

Image Copy 2 Fast Replication

- Fast replication support
  - Exploits FlashCopy on Enterprise Storage Server (ESS) and DS8000
  - Exploits SnapShot on RAMAC Virtual Array (RVA)
- Prerequisites for fast replication
  - z/OS V1R8
  - Storage Subsystem that supports either
    - Data Set Level FlashCopy V2
      - ESS (Shark), DS8000, …

    or

    - SnapShot
      - RAMAC Virtual Array (RVA), …

Figure 10-7. Image Copy 2 Fast Replication CMA01.0

Notes:

IMS V10 adds fast replication support. Fast replication is a DFSMSdss function that exploits the FlashCopy capability of the Enterprise Storage Server (ESS or Shark) or TotalStorage DS8000 or the SnapShot capability of the IMS RAMAC Virtual Array (RVA). FlashCopy V2 must be installed on the ESS system. Fast replication also can be used with OEM storage devices which support the DFSMSdss Fast Replication interface.

This support is provided in the Image Copy 2 (DFSUDMT0) utility and the Database Recovery (DFSURDB0) utility. They use data set level Fast Replication functions of DFSMSdss which are only provided by z/OS V1R8.

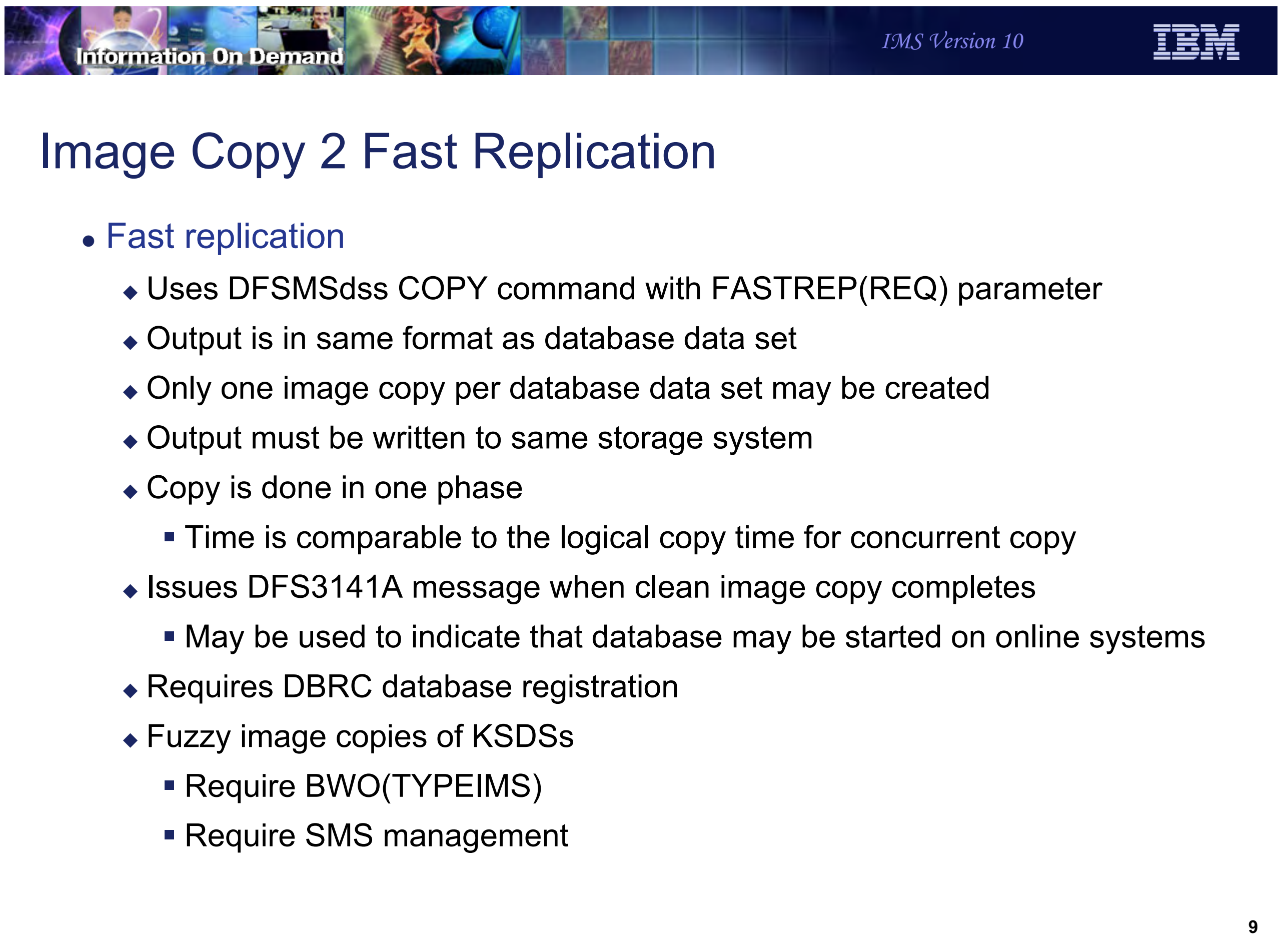


Image Copy 2 Fast Replication

- Fast replication
    - Uses DFSMSdss COPY command with FASTREP(REQ) parameter
    - Output is in same format as database data set
    - Only one image copy per database data set may be created
    - Output must be written to same storage system
    - Copy is done in one phase
        - Time is comparable to the logical copy time for concurrent copy
    - Issues DFS3141A message when clean image copy completes
        - May be used to indicate that database may be started on online systems
    - Requires DBRC database registration
    - Fuzzy image copies of KSDSs
        - Require BWO(TYPEIMS)
        - Require SMS management

Figure 10-8. Image Copy 2 Fast Replication CMA01.0

Notes:

Fast replication is invoked with the DFSMSdss COPY command with the FASTREP(REQ) parameter. The output or “image copy” is an exact copy of the database data set. It must be written to a volume in the same storage system. Only one output data set may be created for each input data set. Unlike the concurrent copy process, the fast replication copy process is not split into a logical copy phase and a physical copy phase. If you are creating a clean image copy, the database is unavailable for the duration of the copy process; however, the fast replication process is comparable in speed to the logical copy phase of the concurrent copy process.

Message DFS3141A is issued when a clean image copy completes. The DFS3141A message may be used to indicate that the database may be started on online systems. The DFS3141A lists the database, partition, area, or group that was copied. It is followed by DFS3141I messages for each data set that was copied.

Databases must be registered with DBRC. The IDCAMS DEFINE must specify BWO(TYPEIMS) for KSDSs if fuzzy image copies are taken. Fuzzy image copies also require that the database data set be managed by SMS. If a split occurs during the logical copy phase for concurrent copy or the initialization of copy for fast replication, the copy is retried up to nine more times if only one data set is being copied. If multiple data sets are being copied, retries are not attempted. A failure causes DFSMSdss to issue message ADR944E and IMS to issue message DFS3145A. The form of the ADR944E message when a fast replication copy fails due to a CI/CA split during the initialization is:

> ADR944E (*ttt*)-*mmmmm*(*yy*), DATA SET *dsname* IN CATALOG *catalog_name* ON VOLUME *volume_serial_number* CANNOT BE PROCESSED BECAUSE IT WAS UPDATED DURING THE FAST REPLICATION INITIALIZATION

The DFS3145A message is:

> DFS3145A ATTEMPT TO COPY KSDS CONCURRENT WITH UPDATE ACCESS FAILED

Image Copy 2 Fast Replication Invocation

- Image Copy 2 DBDS Control Statement
  - Used to specify a database data set to be copied
  - Position 62
    - ‘F’ invokes fast replication
  - Position 63
    - Blank indicates that DD statement is provided for output data set
      - Output DD name is in positions 22-29
      - DD statement for output must be in JCL
    - ‘H’ indicates that output data set is dynamically allocated

Figure 10-9. Image Copy 2 Fast Replication Invocation CMA01.0

Notes:

Fast replication copies are invoked by placing an ‘F’ in position 62 of the DBDS control statement.

If a blank is in position 63 a DD statement must be supplied for the image copy output data set. The data set may either be created prior to the Image Copy 2 step or may be created by the DD statement.

If an ‘H’ is in position 63 the image copy output data set is created dynamically by DFSMSdss. The attributes of the image copy data set are derived from the database data set. An HLQ control statement is required when an ‘H’ appears in position 63 of the DBDS control statement. It is shown on the following page.

Other positions on the DBDS control statement have the same meaning that they have without the fast replication option. For example, an ‘X’ in position 58 invokes a clean image copy and an ‘S’ in invokes a fuzzy image copy.

Fast replication produces only one image copy data set for a database data set. You cannot specify multiple output data sets. This means that positions 30 through 57 must be blank.

Unlike concurrent copies, fast replication does not have logical copy and physical copy phases. The fast replication process is comparable in speed to the logical copy phase of the concurrent copy process. Since there are not separate logical and physical copy phases, position 59 should be left blank. Position 59 is used for concurrent copies to indicate whether the database becomes available for updates after the logical phase completes or after the physical phase completes.

Position 60 should be blank for fast replication. It is used for the compression option with concurrent copies.

Image Copy 2 Fast Replication Invocation

- Image Copy 2 HLQ Control Statement
  - Used to specify the data set name of the image copy data set
  - Position 1
    - ‘H’ indicates this is an HLQ control statement
  - Position 3
    - ‘Y’ time-stamp trailer is appended to output data set name
    - ‘N’ time-stamp trailer is not appended to output data set name
  - Positions 4-29
    - High-level qualifier for output data set name
  - Positions 31-38
    - SMS storage class - optional
  - Positions 40-47
    - SMS management class - optional

Figure 10-10. Image Copy 2 Fast Replication Invocation CMA01.0

Notes:

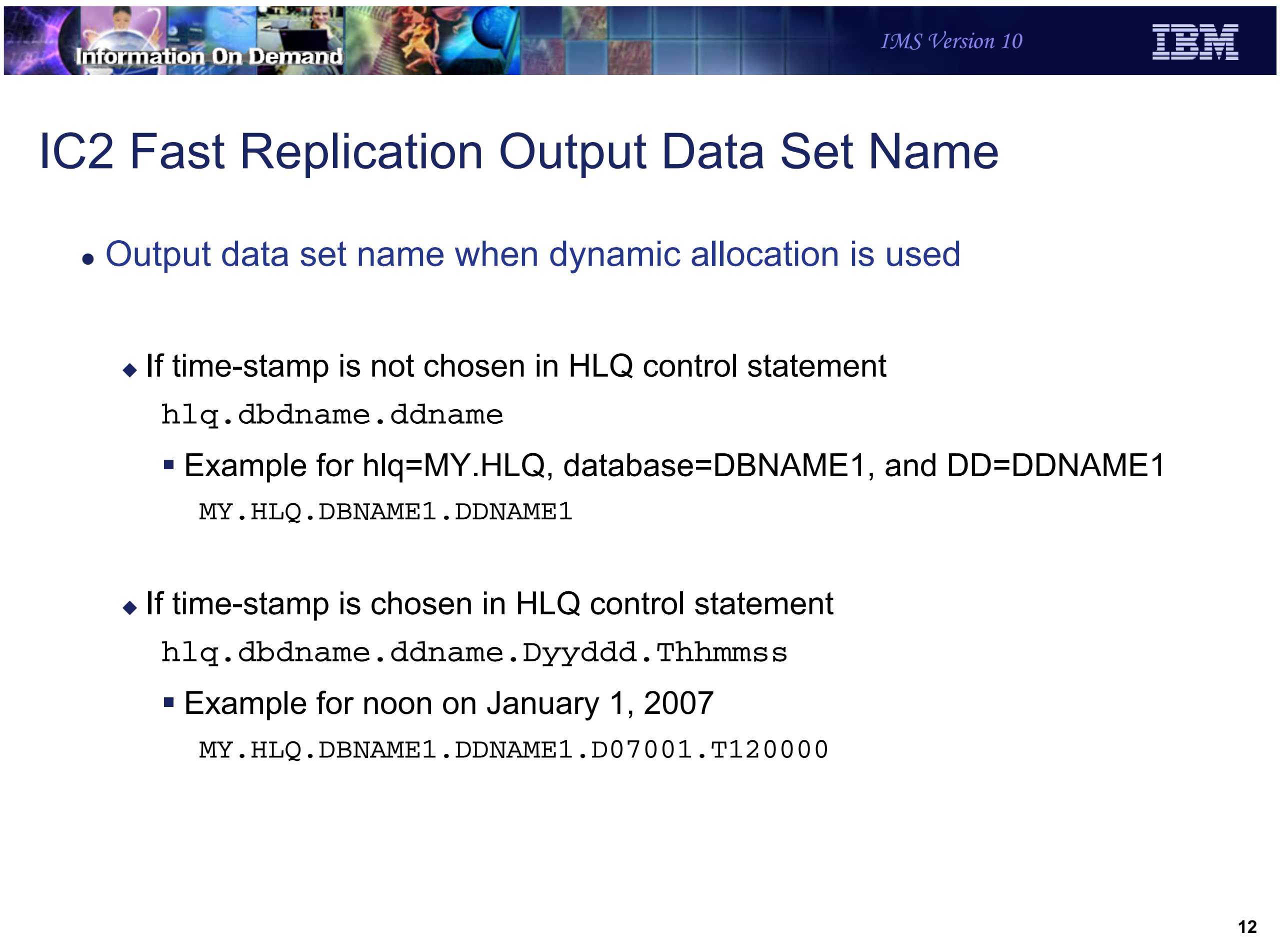
An ‘H’ in position 1 indicates that this is an HLQ (high level qualifier) control statement.

Position 3 indicates whether or not the time-stamp trailer is appended to the output data set name.

Positions 4-29 contain the high-level qualifier used in the output data set name.

Positions 31-38 may be used to specify the SMS storage class that is used by ACS routines.

Positions 40-47 may be used to specify the SMS management class that is used by ACS routines.


Information On Demand
IMS Version 10
IBM

IC2 Fast Replication Output Data Set Name

- Output data set name when dynamic allocation is used
  - If time-stamp is not chosen in HLQ control statement

    `hlq.dbdname.ddname`
    - Example for hlq=MY.HLQ, database=DBNAME1, and DD=DDNAME1

      `MY.HLQ.DBNAME1.DDNAME1`
  - If time-stamp is chosen in HLQ control statement

    `hlq.dbdname.ddname.Dyyddd.Thhmmss`
    - Example for noon on January 1, 2007

      `MY.HLQ.DBNAME1.DDNAME1.D07001.T120000`

12

Figure 10-11. IC2 Fast Replication Output Data Set Name CMA01.0

Notes:

This shows two examples of output data set names that are used when dynamic allocation is used for these data sets.

The first example is for a data set when the time-stamp trailer is not selected (position 3 on the HLQ control statement contains an 'N'). The high-level qualifier is MY.HLQ.

The second example is for a data set when the time-stamp trailer is selected (position 3 on the HLQ control statement contains a 'Y'). The high-level qualifier, database name, and DD name are the same as in the first example.

DBRC with Fast Replication

- New GENJCL.IC parameters
  - SMSONLC - generate IC2 JCL for fuzzy IC with fast replication
  - SMSOFFLC - generate IC2 JCL for clean IC with fast replication
  - HLQ(hlq) - specifies high-level qualifier for output data set name
  - DSNSUF|NODSNSUF - date/time stamp as suffix in output data set name
  - STORCLAS(name) - specifies Storage Class use by ACS routines for allocating image copy data set
  - MGMTCLAS(name) - specifies Management Class use by ACS routines for allocating image copy data set

Figure 10-12. DBRC with Fast Replication CMA01.0

Notes:

SMSONLC and SMSOFFLC are new image copy types recorded in image copy records in the RECONs. Other previously existing image copy types were SMSCIC (IC2 fuzzy IC using concurrent copy), SMSNOCIC (IC2 clean IC using concurrent copy), CIC (concurrent image copy using Image Copy utility), ONLINE (fuzzy image copy using the Online Image Copy utility), and BATCH (clean image copy using the Image Copy utility. High Performance Image Copy (HPIC) V4 also supports fast replication copies, but does not invoke IC2. APAR PK33115 adds the use of SMSONLC and SMSOFFLC by HPIC V4 when it uses fast replication. The HPIC V4 Recovery fund and Create Image Copy function support these IC types as input.

The GENJCL.IC command has new parameters.

The SMSONLC and SMSOFFLC parameters are used to generate fast replication Image Copy 2 utility JCL for either a fuzzy or clean image copy. SMSONLC and SMSOFFLC are also new image copy types recorded in image copy records in the RECONs. Other previously existing image copy types were SMSCIC (IC2 fuzzy IC using concurrent copy), SMSNOCIC (IC2 clean IC using concurrent copy), CIC (concurrent image copy using

Image Copy utility), ONLINE (fuzzy image copy using the Online Image Copy utility), and BATCH (clean image copy using the Image Copy utility. High Performance Image Copy (HPIC) V4 supports fast replication copies. APAR PK33115 adds the use of SMSONLC and SMSOFFLC by HPIC V4 when it uses fast replication. The HPIC V4 Recovery fund and Create Image Copy function support these IC types as input.

HLQ(hlq) is used to specify the high-level qualifier for the output data set name when the output data set is created dynamically. ‘hlq’ may be 1 to 26 characters. The value is placed in positions 31-56 of the DBDS control statement.

DSNSUF specifies that the date/time stamp suffix is to be added to the output data set name for dynamically created output data sets. NODSNSUF specifies that the date/time stamp suffix is not added. NODSNSUF is the default.

STORCLAS(name) and MGMTCLAS(name) are used to specify the storage class name and management class name to by used by the Access Control System (ACS) routines for allocating output data sets which are created dynamically.

Fast Replication with Database Recovery

- Database Recovery Utility reads Fast Replication image copies
  - No user specification required
    - Utility uses DBRC information to determine that IC is fast replication IC
  - Uses COPY with FASTREPLICATION(PREFERRED)
    - If Fast Replication copy is on the same storage system with target data set and storage system supports fast replication, fast replication is used to restore the database data set from the image copy
    - Else, traditional copy is used
  - DBDS DD statement is optional when using Fast Replication IC input
    - ‘D’ in position 64 of control statement indicates DD statement not used
    - DBRC information is used to allocate the data set
  - GENJCL.RECOV has added support for Fast Replication
    - DBDSDD|NODBDSDD parameters specify if DD statement is used

Figure 10-13. Fast Replication with Database Recovery CMA01.0

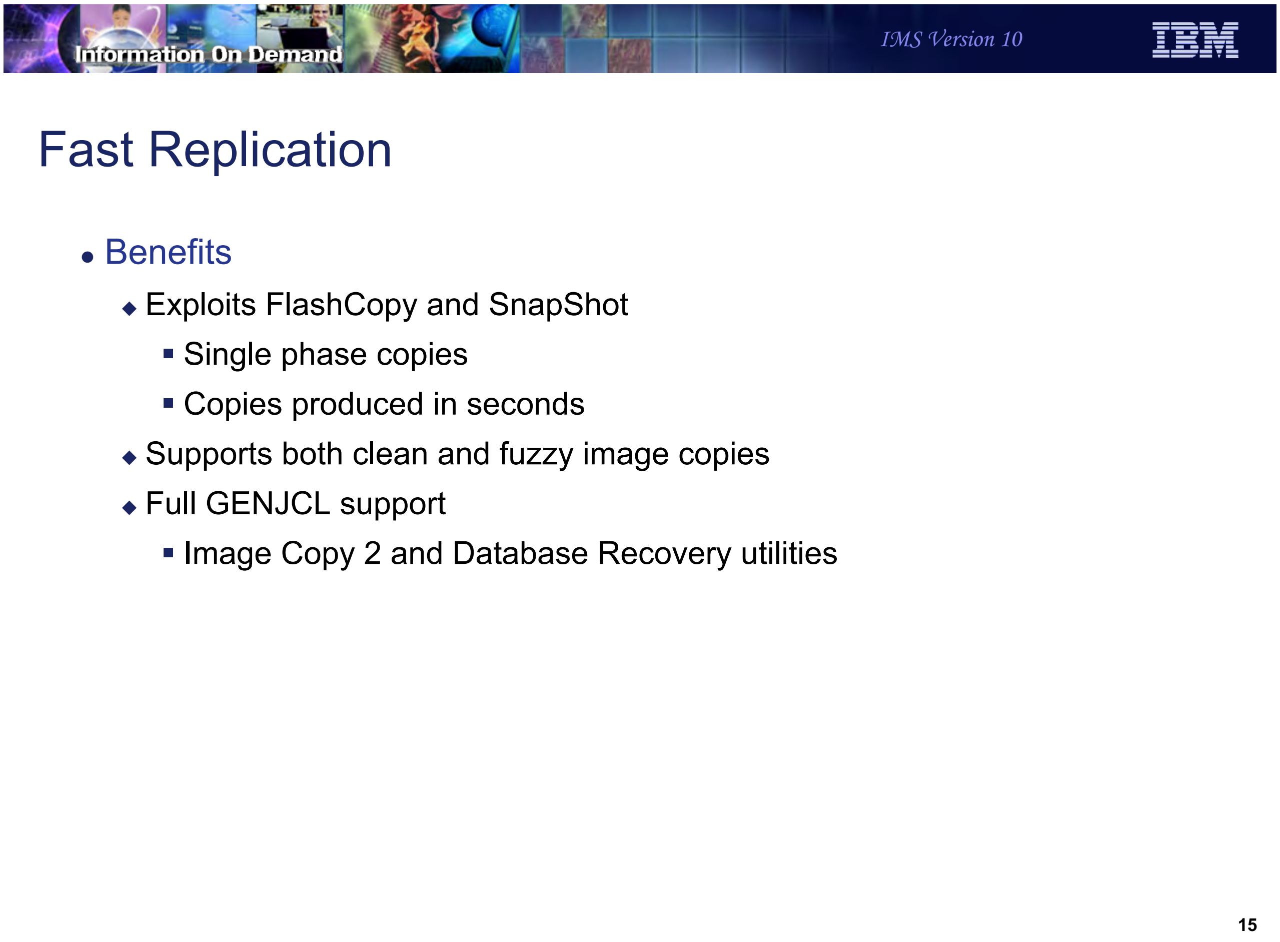
Notes:

The IMS Database Recovery utility (DFSURDB0) has been updated to read image copies created by using fast replication. In previous releases Database Recovery could read two kinds of image copies. Those in the original Image Copy format have header records followed by copies of the data. Those produced with concurrent copy were in DFSMSdss DUMP format. Fast replication image copies are exact copies of the database data sets, The Database Recovery utility uses DBRC information about the Image Copy to determine its format and how to read it.

Database Recovery invokes fast replication with the “preferred” option. This indicates that fast replication will be used if the image copy and the database data sets are on the same storage system and the storage system supports fast replication. If these conditions are not met, standard copy techniques are used to restore the image copy to the database data set. There are two reason primary reasons that the conditions for fast replication would not be met. First, the user may choose to recover the database data set to a different storage system. Second, the image copy data set may be been moved from its original storage system.

Without fast replication, the database data set must be specified on a DD statement. Dynamic allocation cannot be used. With fast replication this restriction is removed. The database data set may be dynamically allocated. To request dynamic allocation you must place a ‘D’ in position 64 of the Database Recovery utility control statement.

The DBRC GENJCL.RECOV command has been enhanced to support dynamic allocation of the database data set when using a fast replication image copy as input. This is done with the NODBDSDD keyword. The DBDSDD keyword is the default. It indicates that dynamic allocation will not be used and that a DD statement for the database data set will be generated by the GENJCL.RECOV command.



Information On Demand
IMS Version 10
IBM

Fast Replication

- Benefits
  - Exploits FlashCopy and SnapShot
    - Single phase copies
    - Copies produced in seconds
  - Supports both clean and fuzzy image copies
  - Full GENJCL support
    - Image Copy 2 and Database Recovery utilities

Figure 10-14. Fast Replication CMA01.0

Notes:

Information On Demand

IMS Version 10

Fuzzy User Image Copies

Figure 10-15. [No title] CMA01.0

Notes:

Information On Demand
IMS Version 10

Fuzzy User Image Copy Support

- DBRC support for fuzzy user image copies
  - Fuzzy image copies taken by utility or tool without a DBRC interface
    - Pack dump, DFSMSdss DUMP or COPY not invoked by IC2, etc.
  - Support:
    - NOTIFY.UIC can specify a fuzzy user image copy
      - CIC and STOPTIME(time) parameters are used
    - GENJCL.RECOV can be used to generate recovery from logs after fuzzy user image copy has been restored
      - USERIC(time) or LASTUIC parameters are used

Figure 10-16. Fuzzy User Image Copy Support CMA01.0

Notes:

IMS V10 has added support for fuzzy user image copies. User image copies are copies of database data sets that are done without a DBRC interface. Typically, these are done with utilities that are not IMS utilities. In previous releases IMS supported clean user image copies. IMS V10 adds support for image copies taken while the database data set is being updated.

The support has two parts.

First, the NOTIFY.UIC command has been extended to include a specification of a fuzzy image copy, sometimes called a concurrent image copy. The CIC keyword has been added to the NOTIFY.UIC command. It indicates that the user image copy is concurrent or fuzzy. When notifying DBRC of a fuzzy image copy you must specify the stop time of the image copy. This is in addition to the start time which is required for all image copies. The stop time is specified with the STOPTIME(time stamp) parameter.

Second, the GENJCL.RECOV command has been extended to generate the appropriate JCL to include the correct logs for recoveries using fuzzy user image copies. The USERIC(time stamp) parameter specifies the fuzzy user IC that has been restored.

LASTUIC is may be used instead of USERIC(time). It specifies that the last fuzzy user image copy recorded in the RECONs has been restored. When either of these parameters is used, GENJCL.RECOV selects the proper logs for input to the recovery. These are the logs that were created while the fuzzy user image copy was being taken and any created since that time. USERIC and LASTUIC cause the appropriate S and M control statements to be generated for the Database Recovery utility.

When recovering after restoring a fuzzy user image copy do not specify USEDBDS on the GENJCL.RECOV command. USEDBDS is only for use when a clean user image copy has been restored followed by a NOTIFY.RECOV identifying this clean user image copy.

Fuzzy User Image Copy Support

- DB Recovery utility (DFSURDB0) control statement
  - 'M' required in column 63 of the S control statement
    - Indicates that a fuzzy user image copy has been restored
    - Indicates that an M control statement follows
  - M control statement
    - M is column 1
    - Runtime of the user fuzzy image copy in columns 2-32

```
|...+....1....+....2....+....3....+....4....+....5....+....6...
//SYSIN    DD  *
S  DHVNTZ02 VNT0A001                                          M
M061531233296-0600
```

  - M control statement is created by GENJCL.RECOV with USERIC(time) or LASTUIC

Figure 10-17. Fuzzy User Image Copy Support CMA01.0

Notes:

The use of a fuzzy user image copy requires an M in column 63 of the S control statement used by the Database Recovery utility (DFSURDB0). The example shows that the database data set with DDNAME VNT0A001 in database DHVNTZ02 is being recovered. The M in column 63 indicates that a fuzzy user image copy has been restored. An M in column 63 requires that an M control statement also be supplied. The M control statement has an 'M' in column 1 and the runtime of the fuzzy user image copy in columns 2-32. The time is specified in standard time format. The example shows a compressed time of year 2006, day 153, hour 12, minute 33, second 29.6, with an offset of -6 hours. This offset is Central Standard Time in North America.

DBRC verifies that the specified user image copy is recorded in the RECONs.

When GENJCL.RECOV is used with a fuzzy user image copy, the proper M control statement is created. GENJCL.RECOV uses a fuzzy user image copy when either the USERIC(time) or LASTUIC parameter is specified.

Information On Demand

IMS Version 10

IBM

Procedures for Clean and Fuzzy User ICs

- Clean user IC
  1. Create clean user IC
  2. NOTIFY.UIC ... RUNTIME(time1)
- Fuzzy user IC
  1. Create fuzzy user IC
  2. NOTIFY.UIC … CIC RUNTIME(time1) STOPTIME(time2)
- Recovery using clean user IC
  1. Restore clean user IC
  2. NOTIFY.RECOV … RCVTIME
  3. GENJCL.RECOV … USEDBDS
- Recovery using fuzzy user IC
  1. Restore fuzzy user IC
  2. GENJCL.RECOV … USERIC(time1)

     or

  2. GENJCL.RECOV … LASTUIC

Figure 10-18. Procedures for Clean and Fuzzy User ICs CMA01.0

Notes:

This is a comparison of the procedures to use with clean and fuzzy user image copies.

After taking a fuzzy user image copy the NOTIFY.UIC must include both the CIC keyword and the STOPTIME parameter.

Surprisingly, the recovery with a fuzzy user image copy is bit simpler than the recovery with a clean user image copy. You do not have to issue the NOTIFY.RECOV command when using a fuzzy user image copy. The GENJCL.RECOV commands differ. USEDBDS is required after restoring a clean user image copy. The identification of the fuzzy user image copy is required after restoring a fuzzy user image copy. Either a specific copy may be specified with the USERIC(time1) parameter or the last copy may be specified by using LASTUIC. When recovering after restoring a fuzzy user image copy do not specify USEDBDS on the GENJCL.RECOV command. USEDBDS is only for use when a clean user image copy has been restored followed by a NOTIFY.RECOV identifying this clean user image copy.

The recovery processes shown here do not include setting the Recovery Needed flag with the CHANGE.DBDS … RECOV command. Some installations prefer to set the flag to protect themselves from unintentionally authorizing the database before the recovery is done.

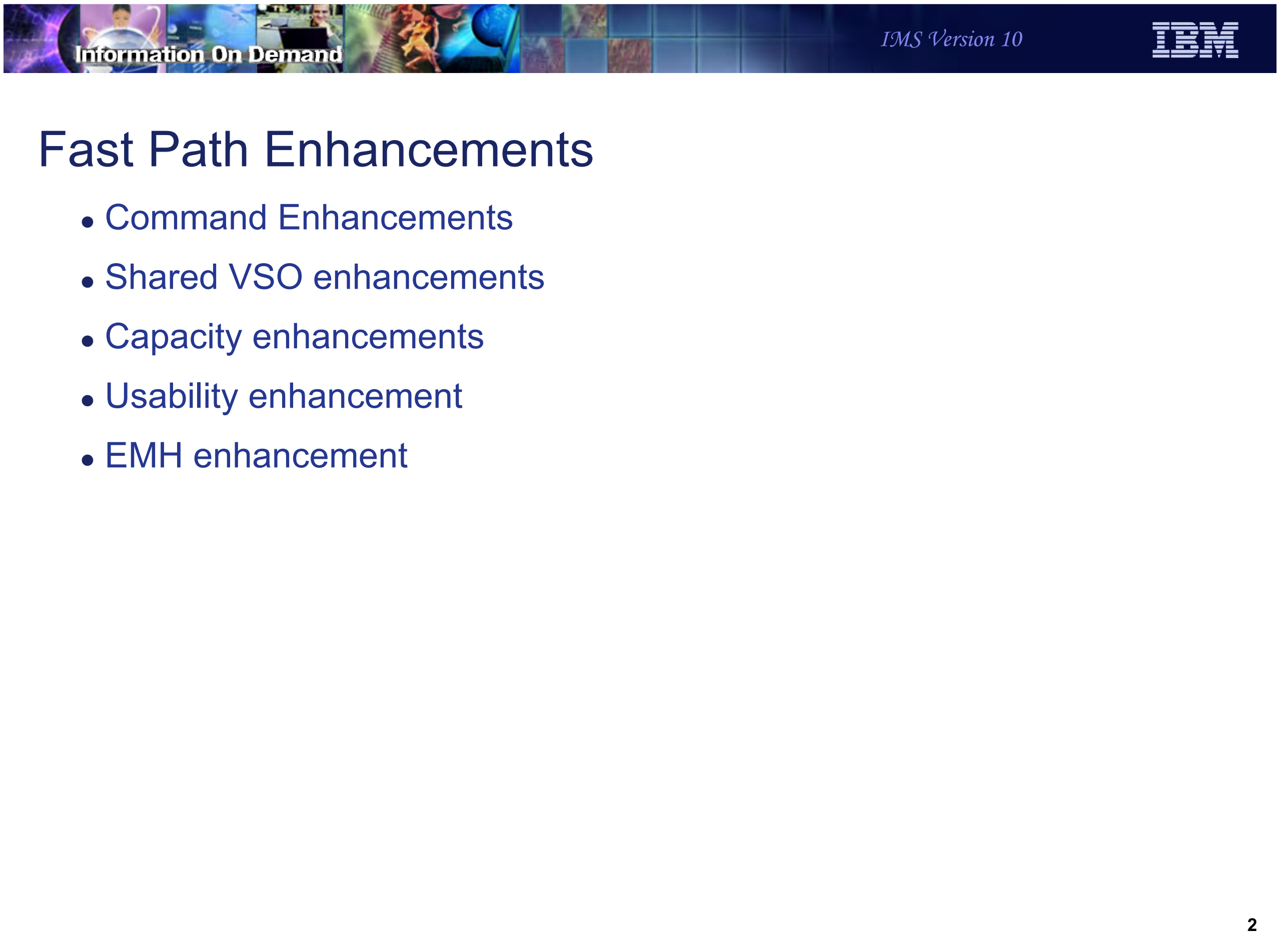
Fuzzy User Image Copies

- Benefits
  - Expanded supported for fuzzy image copies
    - Non-disruptive to online system
  - DBRC support for fuzzy user image copies
    - Registration of these copies
    - GENJCL support for selecting the appropriate logs

Figure 10-19. Fuzzy User Image Copies CMA01.0

Notes:

Unit 11.Fast Path

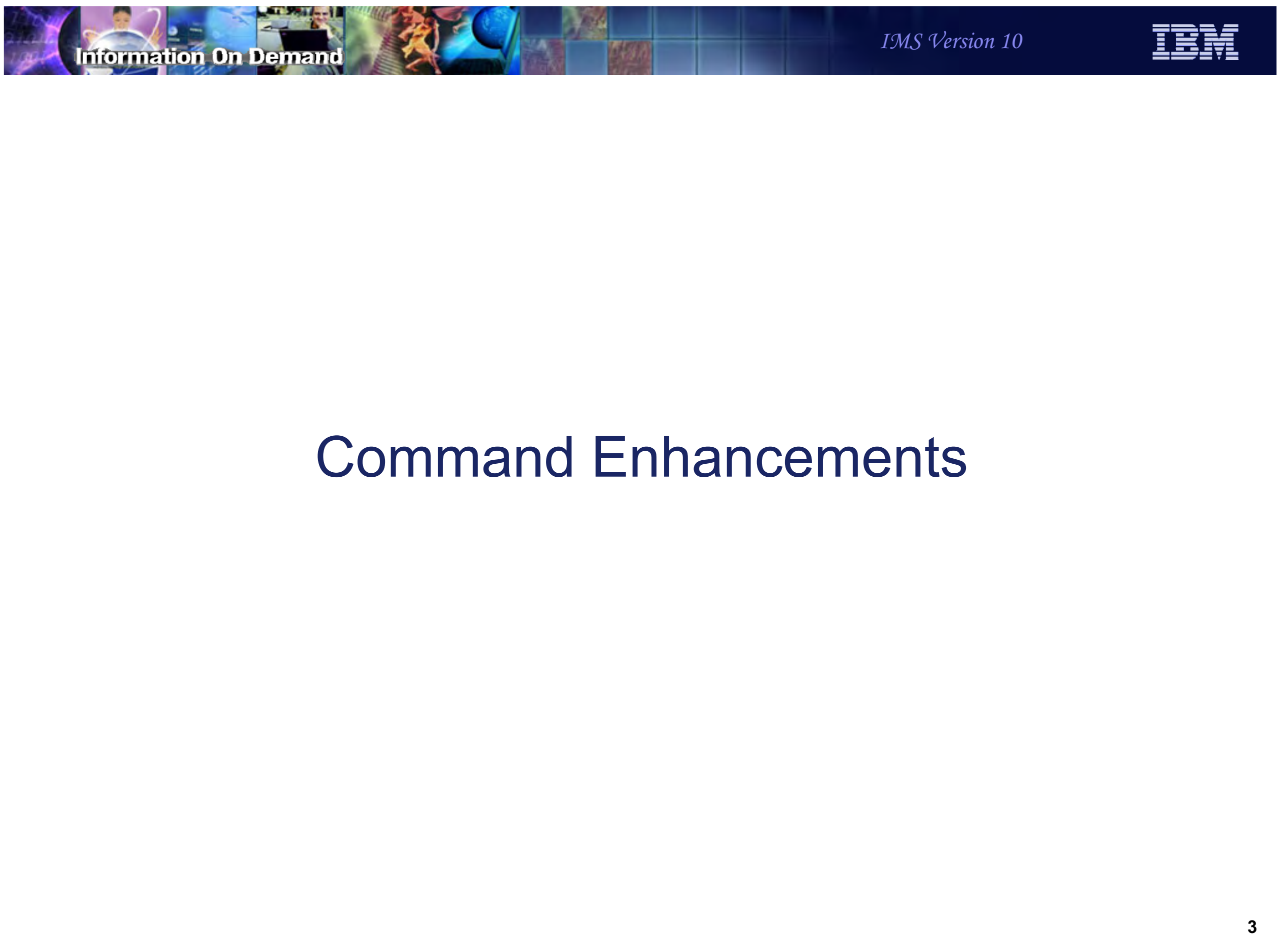

Information On Demand
IMS Version 10

Fast Path Enhancements

- Command Enhancements
- Shared VSO enhancements
- Capacity enhancements
- Usability enhancement
- EMH enhancement

Figure 11-1. Fast Path Enhancements CMA01.0

Notes:


Information On Demand
IMS Version 10
IBM

Command Enhancements

Figure 11-2. [No title] CMA01.0

Notes:

Information On Demand
IMS Version 10
IBM

Starting All Areas with UPDATE DB command

- Option to start all areas when starting a DEDB
  - UPDATE DB NAME(name) START(ACCESS) AREA(*)
    - AREA(*) starts all areas of the database
      - There is NOT a capability to specify an area with AREA(…) parameter
    - Example TSO SPOC command and response:

```
UPDATE DB NAME(DEBJN) START(ACCESS) AREA(*)


DBName          AreaName        MbrName         CC
DEDBJN          *               IMS1             0
```

- Benefit
  - Separate UPDATE AREA commands are not required for each area

Figure 11-3. Starting All Areas with UPDATE DB command CMA01.0

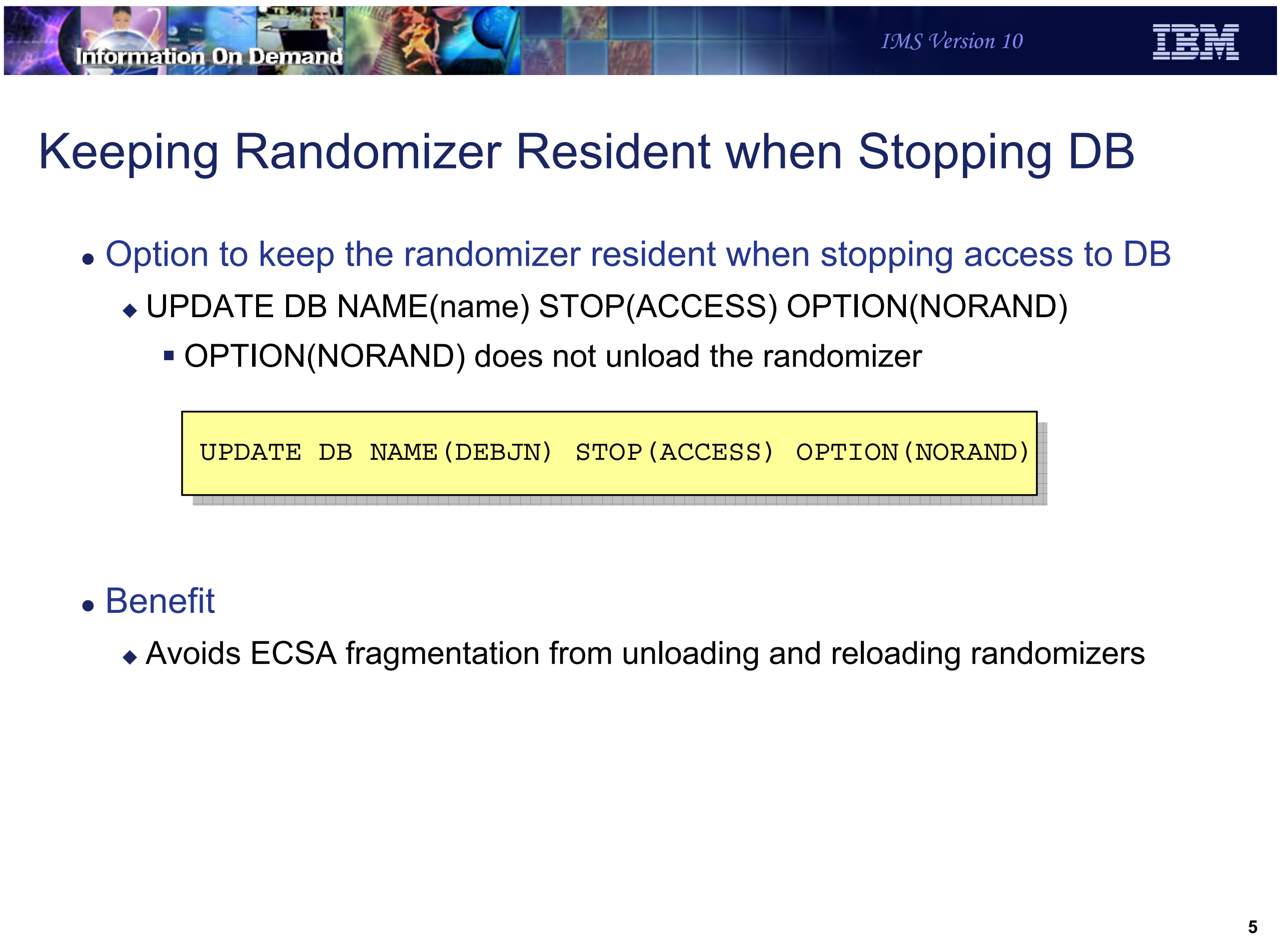
Notes:

This enhancements allows you to start all of the areas in a database when you also start the database. The only valid specification for the AREA parameter is AREA(*).

If an area cannot be started, a separate line is provided for it in the response. Here is an example where area ARJN20 was not started:

```
UPDATE DB NAME(DEBJN) START(ACCESS) AREA(*)
DBNameAreaNameMbrNameCC
DEDBJN*IMS18
DEDBJNARJN20IMS1A5
```

The 8 completion code for the database indicates that the command completed with errors for one or more areas of the database. The A5 completion code for area ARJN20 indicates that the DBRC ‘Prohibit Further Authorization’ flag was on for this area.


Information On Demand
IMS Version 10
IBM

Keeping Randomizer Resident when Stopping DB

- Option to keep the randomizer resident when stopping access to DB
  - UPDATE DB NAME(name) STOP(ACCESS) OPTION(NORAND)
    - OPTION(NORAND) does not unload the randomizer

```
UPDATE DB NAME(DEBJN) STOP(ACCESS) OPTION(NORAND)
```

- Benefit
  - Avoids ECSA fragmentation from unloading and reloading randomizers

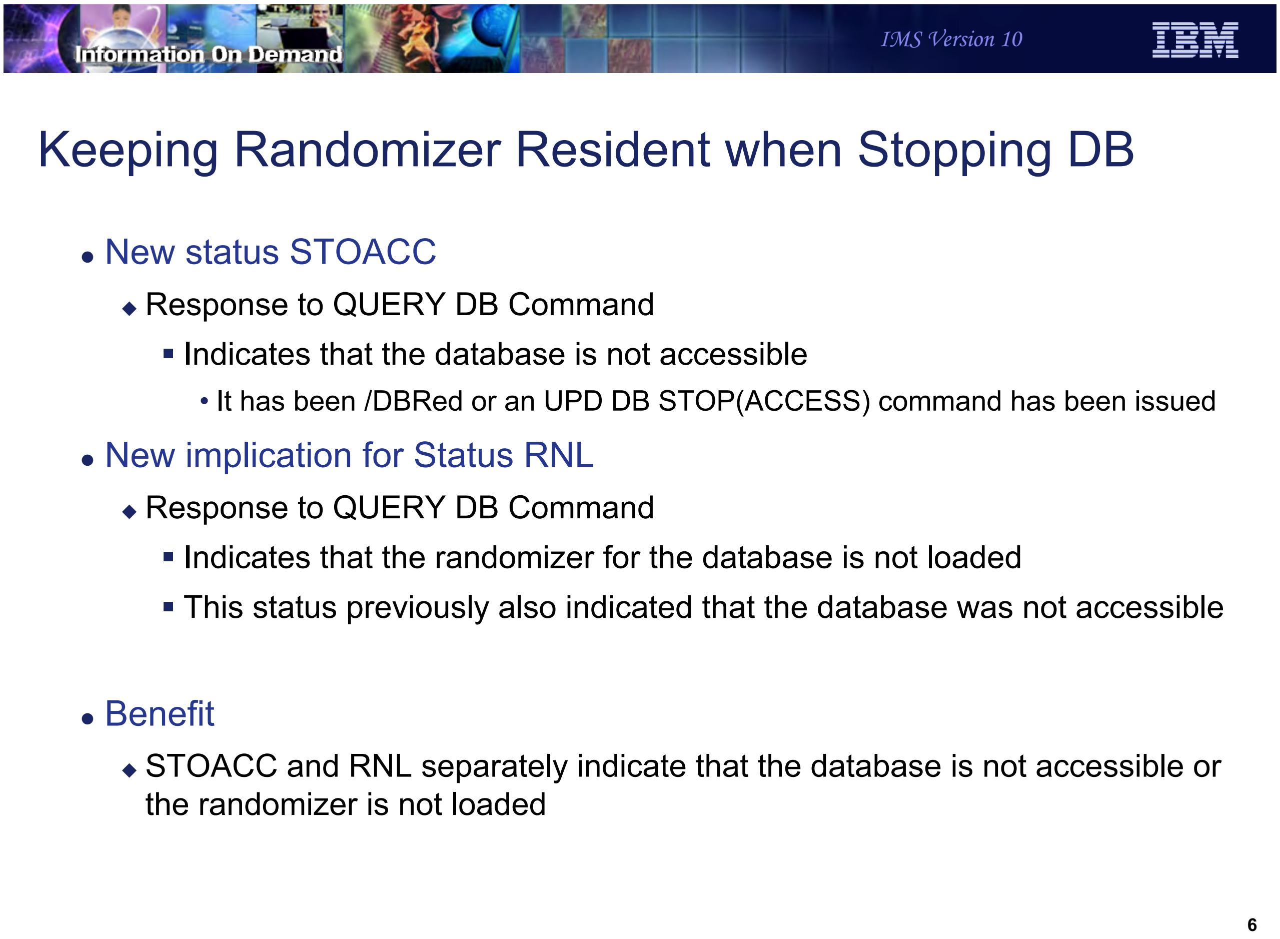
Figure 11-4. Keeping Randomizer Resident when Stopping DB CMA01.0

Notes:

In previous releases if you made a DEDB inaccessible, its randomizer was unloaded from memory if the randomizer was not being used by any other DEDB. DEDBs are made inaccessible with either a /DBR DB command or an UPDATE DB STOP(ACCESS) command.

IMS V10 adds an OPTION(NORAND) parameter to the UPDATE DB STOP(ACCESS) command. This causes IMS to keep the randomizer module in memory. This helps to avoid fragmentation of ECSA which can occur when modules are unloaded and reloaded.

The /DBR DB command does not have a parameter that is equivalent to OPTION(NORAND) on the UPD DB STOP(ACCESS) command.


Information On Demand
IMS Version 10
IBM

Keeping Randomizer Resident when Stopping DB

- New status STOACC
  - Response to QUERY DB Command
    - Indicates that the database is not accessible
      - It has been /DBRed or an UPD DB STOP(ACCESS) command has been issued
- New implication for Status RNL
  - Response to QUERY DB Command
    - Indicates that the randomizer for the database is not loaded
    - This status previously also indicated that the database was not accessible
- Benefit
  - STOACC and RNL separately indicate that the database is not accessible or the randomizer is not loaded

6

Figure 11-5. Keeping Randomizer Resident when Stopping DB CMA01.0

Notes:

In conjunction with the OPTION(NORAND) parameter on the UPDATE DB STOP(ACCESS) command, a new status is available with the QUERY DB command. In previous releases one could tell that a database was inaccessible because the RNL status was associated with the database. RNL indicated that the randomizer module was not loaded. The absence of the RNL status no longer indicates that the database is accessible. A new status is available for this. The STOACC status in the response to a QUERY DB command indicates that the database is inaccessible. That is, the database has been /DBRed or an equivalent UPDATE DB STOP(ACCESS) command has been processed for the database.

The RNL status is now used only to indicate that the randomizer module is not loaded. The STOACC status is used to indicate that the database is inaccessible. The STOACC status is not returned by the /DISPLAY DB command. It is only returned by QUERY DB commands.

After you issued the UPDATE DB NAME(DEBJN) STOP(ACCESS) command, you would expect that the response to a QRY DB NAME(DEBJN) SHOW(STATUS) command would include:

```
DBName   AreaName MbrName   CC   TYPE  LclStat
DEDBJN            IMS1      0    DEDB  NOTOPEN,STOACC,RNL
```

After you issued the UPDATE DB NAME(DEBJN) STOP(ACCESS) OPTION(NORAND) command, you would expect that the response to a QRY DB NAME(DEBJN) SHOW(STATUS) command would include:

```
DBName   AreaName MbrName   CC   TYPE  LclStat
DEDBJN            IMS1      0    DEDB  NOTOPEN,STOACC
```

RNL is not included in the second response.

Information On Demand
IMS Version 10

Shared VSO Enhancements

Figure 11-6. [No title] CMA01.0

Notes:

Shared VSO Private Buffer Pool Lookaside

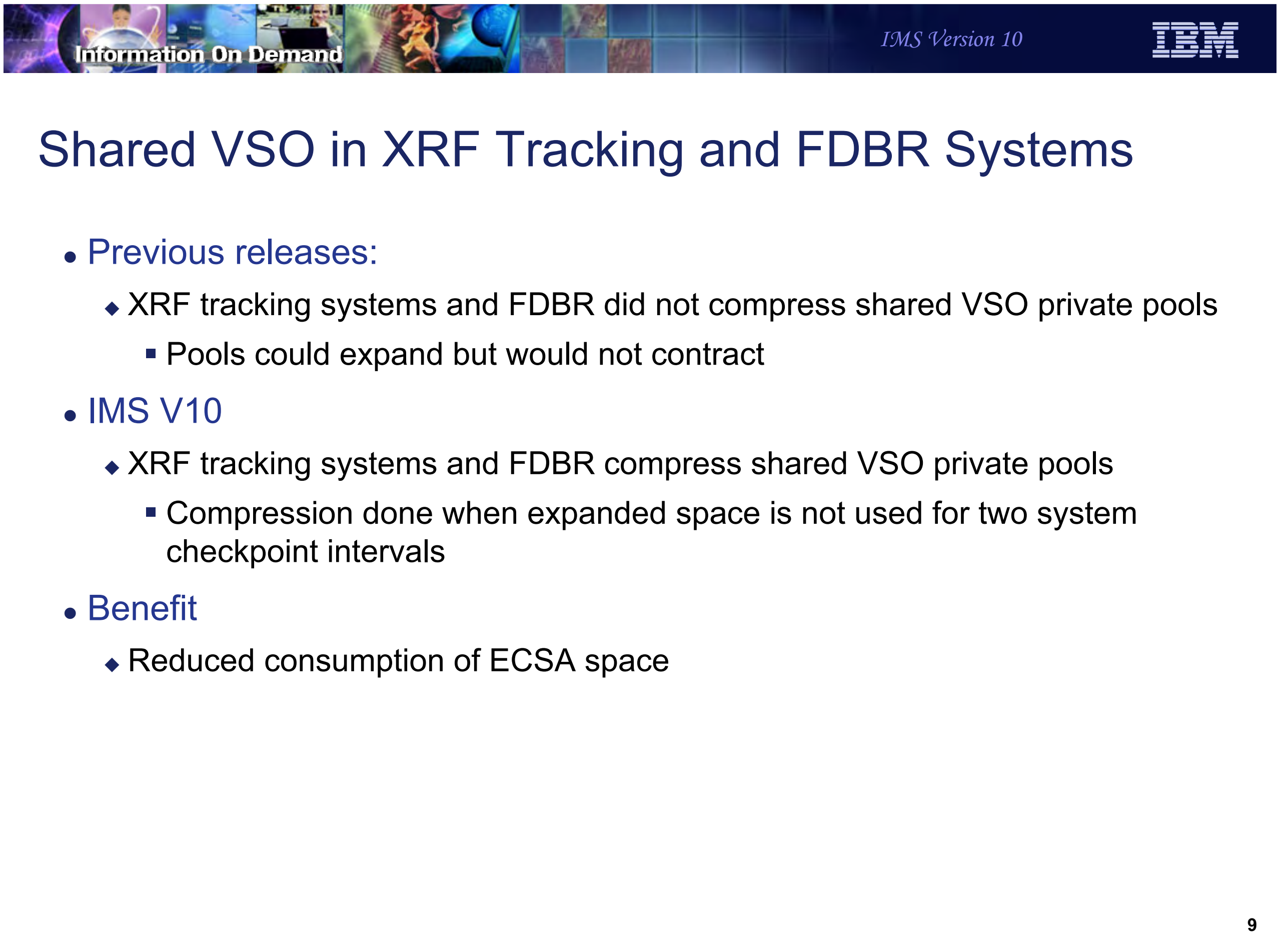
- Previous releases for single area structures:
  - Lookaside must be specified in two places:
    - DBRC INIT.DBDS or CHANGE.DBDS command
      - LKASID or NOLKASID parameter
    - DEDB statement in DFSVSMxx member
    - If specifications conflict, DBRC value is used
- IMS V10
  - Lookaside specification in the DEDB statement is not required
- Multiple area structures have always ignored DBRC specification
  - Specification in DEDBMAS statement continues to apply
- Benefit
  - No longer required to specify parameter in DEDB statement that is ignored

Figure 11-7. Shared VSO Private Buffer Pool Lookaside CMA01.0

Notes:

In previous releases shared VSO users who had single area structures specified whether or not lookaside buffering would be used for an area in two places. It was specified in DBRC with an INIT.DBDS or CHANGE.DBDS command and also in the DEDB statement in the DFSVSMxx member as part of the private pool definition. If the specifications conflicted, the DBRC value was used. Only the DBRC value mattered, nevertheless, some value had to be specified in the DEDB statement. The requirement to specify a lookaside value in the DEDB statement has been eliminated in V10. You do not have to specify any value there.

The specification of lookaside for multiple area structures has not changed in V10. Multiple area structures were introduced in IMS V9. IMS continues to ignore any lookaside specification in DBRC for areas using multiple area structures. Only the specification in the DEDBMAS statement of the DFSVSMxx member is used.



Information On Demand
IMS Version 10
IBM

Shared VSO in XRF Tracking and FDBR Systems

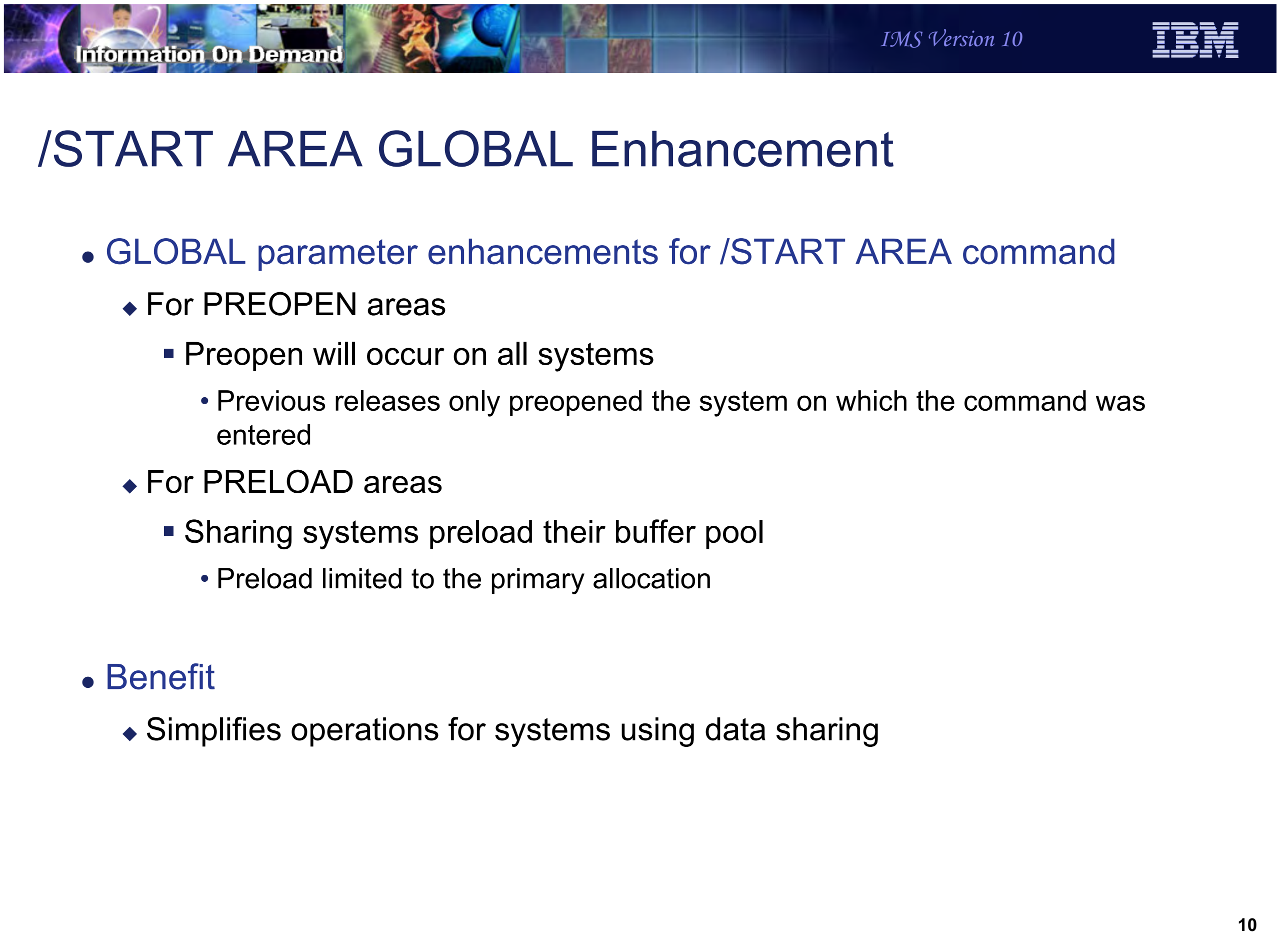
- Previous releases:
  - XRF tracking systems and FDBR did not compress shared VSO private pools
    - Pools could expand but would not contract
- IMS V10
  - XRF tracking systems and FDBR compress shared VSO private pools
    - Compression done when expanded space is not used for two system checkpoint intervals
- Benefit
  - Reduced consumption of ECSA space

9

Figure 11-8. Shared VSO in XRF Tracking and FDBR Systems CMA01.0

Notes:

In previous releases of IMS when shared VSO was used with XRF or FDBR, the XRF tracking system or FDBR system did not compress the shared VSO private pools. These pools can expand as more buffers are required for the areas in the pools. When the buffers are no longer needed, XRF active systems and non-XRF systems compress the pool by releasing buffers that were acquired for expansion. IMS V10 has enhanced XRF tracking systems and FDBR by compressing their pools just as non-XRF and active XRF systems compress pools. This will potentially reduce the use of ECSA by XRF tracking systems and FDBR for shared VSO users.


Information On Demand
IMS Version 10
IBM

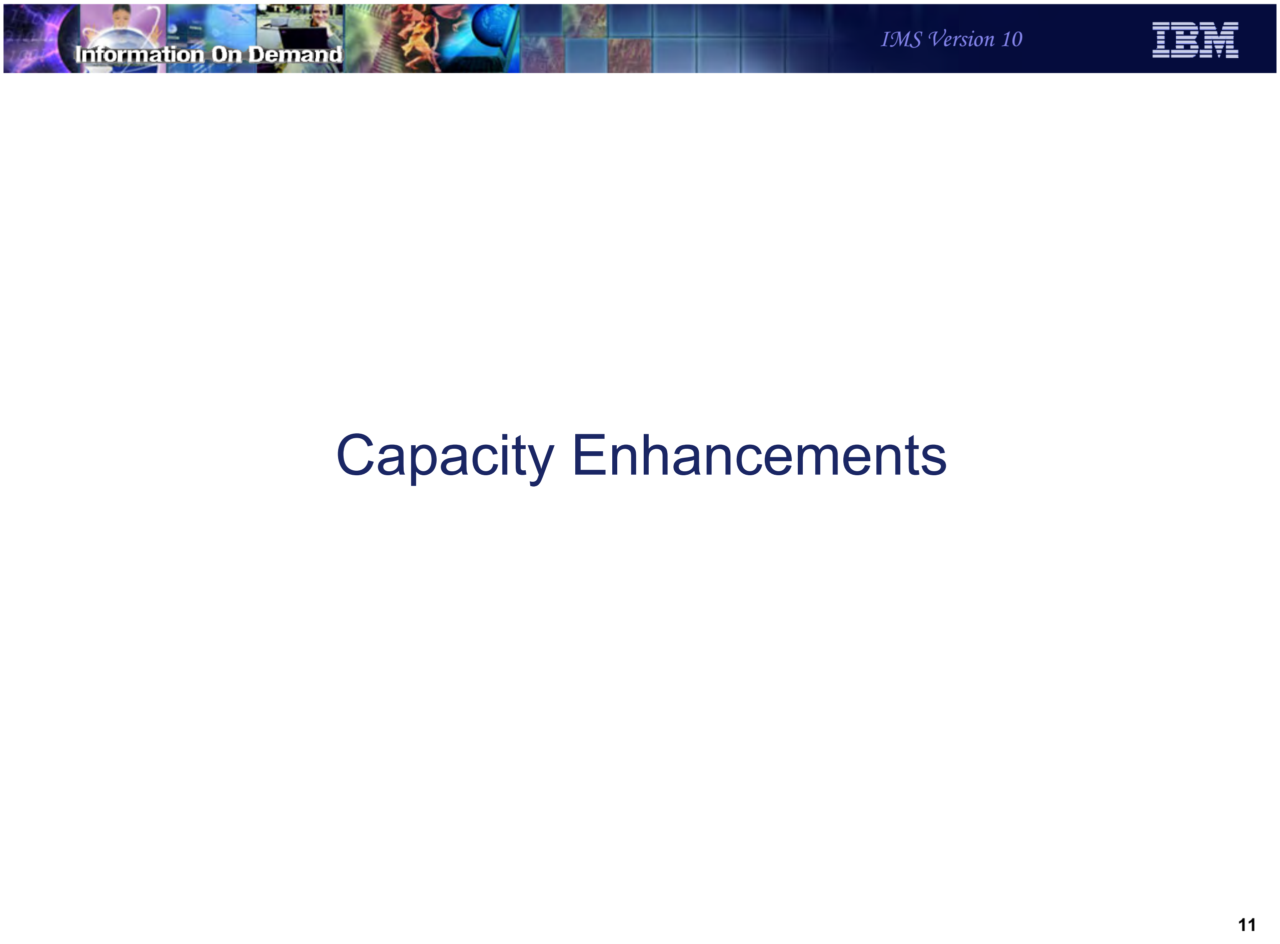
/START AREA GLOBAL Enhancement

- GLOBAL parameter enhancements for /START AREA command
  - For PREOPEN areas
    - Preopen will occur on all systems
      - Previous releases only preopened the system on which the command was entered
  - For PRELOAD areas
    - Sharing systems preload their buffer pool
      - Preload limited to the primary allocation

- Benefit
  - Simplifies operations for systems using data sharing

Figure 11-9. /START AREA GLOBAL Enhancement CMA01.0

Notes:

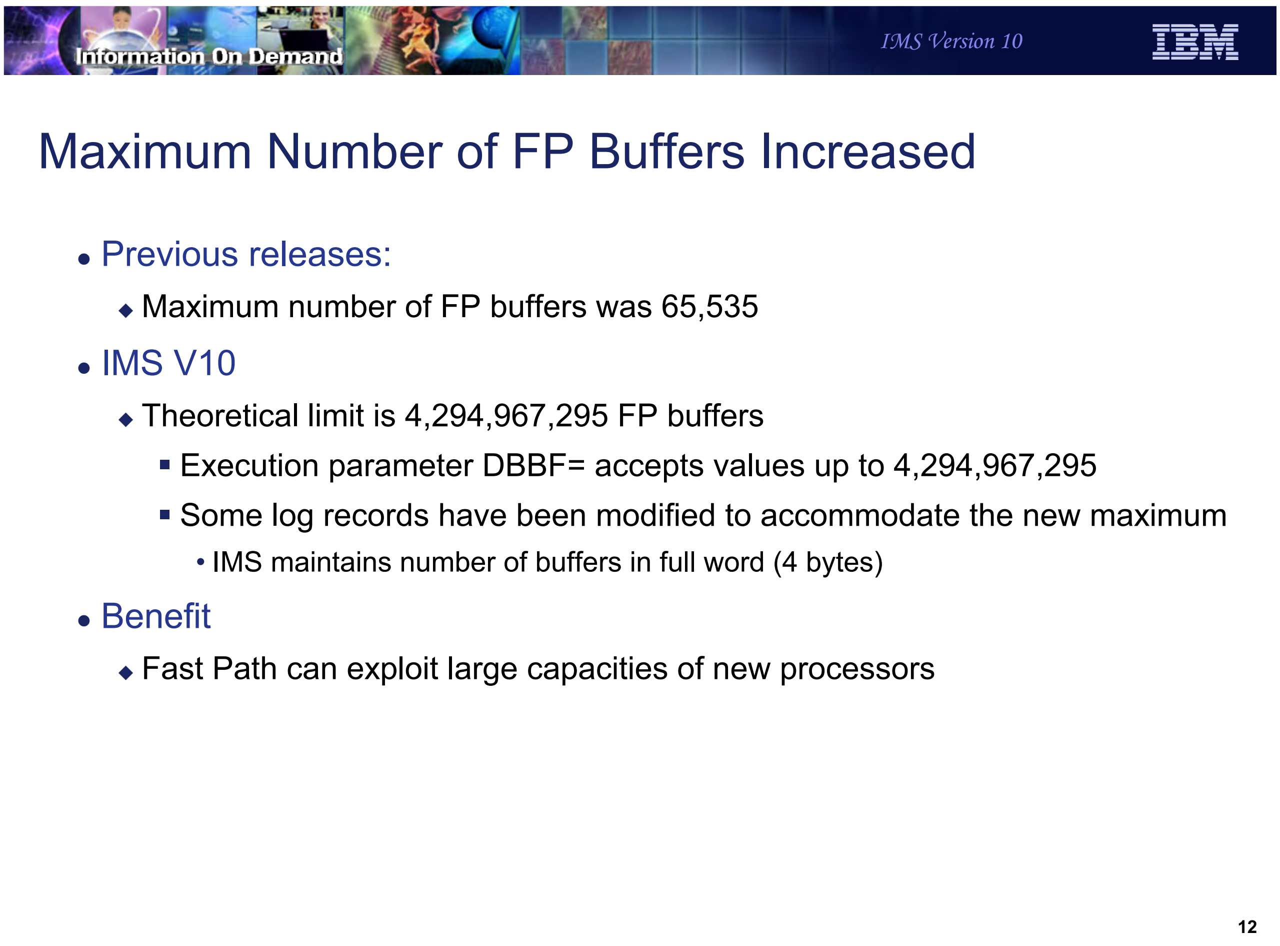


Information On Demand
IMS Version 10

Capacity Enhancements

Figure 11-10. [No title] CMA01.0

Notes:


Information On Demand
IMS Version 10
IBM

Maximum Number of FP Buffers Increased

- Previous releases:
  - Maximum number of FP buffers was 65,535
- IMS V10
  - Theoretical limit is 4,294,967,295 FP buffers
    - Execution parameter DBBF= accepts values up to 4,294,967,295
    - Some log records have been modified to accommodate the new maximum
      - IMS maintains number of buffers in full word (4 bytes)
- Benefit
  - Fast Path can exploit large capacities of new processors

12

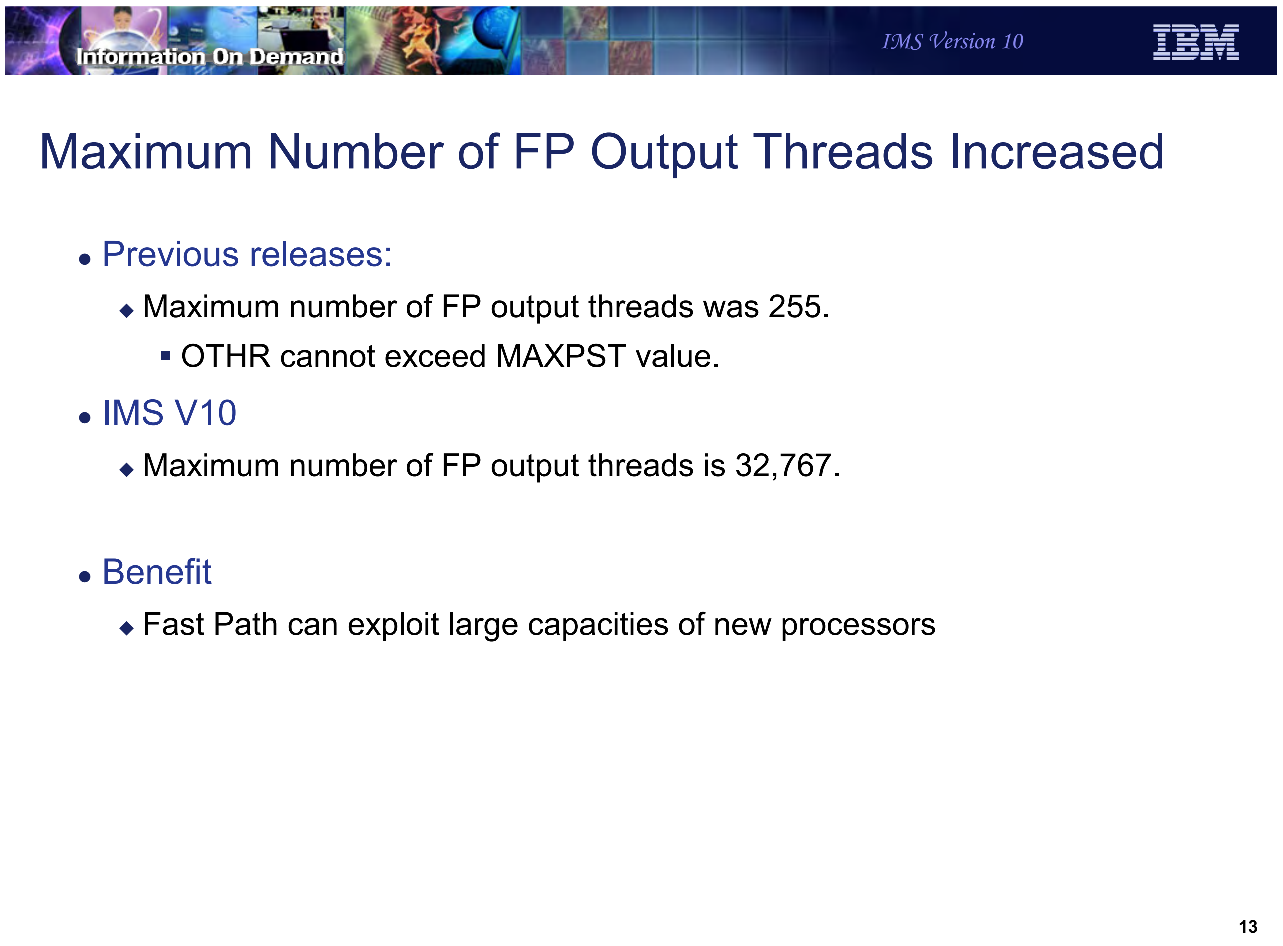
Figure 11-11. Maximum Number of FP Buffers Increased CMA01.0

Notes:

The maximum number of fast path buffers that can be used has been increased from 65,535 to a theoretical value of 4,294,967,295. This is a theoretical value since a system is unlikely to have the storage for so many buffers. Any value up to the theoretical limit may be specified on the DBBF= execution parameter for the online system. In previous releases you could specify the number of Fast Path buffers on the FPCTRL macro. This macro is not used in IMS V10. The DBBF= execution parameter must be used to specify these buffers in V10.

The log records that have been modified include the following that are mapped by the DBFLSRT macro: 5950, 5951, 5952, 5953, 5953, 5956, and 5957.

Other modified records are mapped by the DBFLGRSD macro. They include: 5955 and 5958.



Maximum Number of FP Output Threads Increased

- Previous releases:
  - Maximum number of FP output threads was 255.
    - OTHR cannot exceed MAXPST value.
- IMS V10
  - Maximum number of FP output threads is 32,767.
- Benefit
  - Fast Path can exploit large capacities of new processors

Figure 11-12. Maximum Number of FP Output Threads Increased CMA01.0

Notes:

In previous releases the maximum number of Fast Path output threads was the smaller of the MAXPST value or 255. In V10 the maximum is the 32,767.

This allows IMS V10 users to have many more output threads. This could be desirable as processor capacities increase.

There is another enhancement for Fast Path users in IMS Version 10. In previous releases most Fast Path control blocks and buffers were created in one module, DBFCONT0. This module resides in ECSA and it could be very large. IMS V10 places these control blocks and buffers in five modules. This allows installations to more efficiently use ECSA storage since smaller contiguous areas of ECSA may be used.

The major new modules and their contents are:

DBFCONT1: contains the ECNTs, MSDBs, and MSDB blocks

DBFCONT3: contains buffer headers (DHMRs) and buffers

DBFCONT4: contains DEDB blocks

DBFCONT5: contains output threads

DBFCONT6: contains BALGs (balancing groups)

Information On Demand
IMS Version 10
IBM

FDBR FPBUFF= Parameter Default

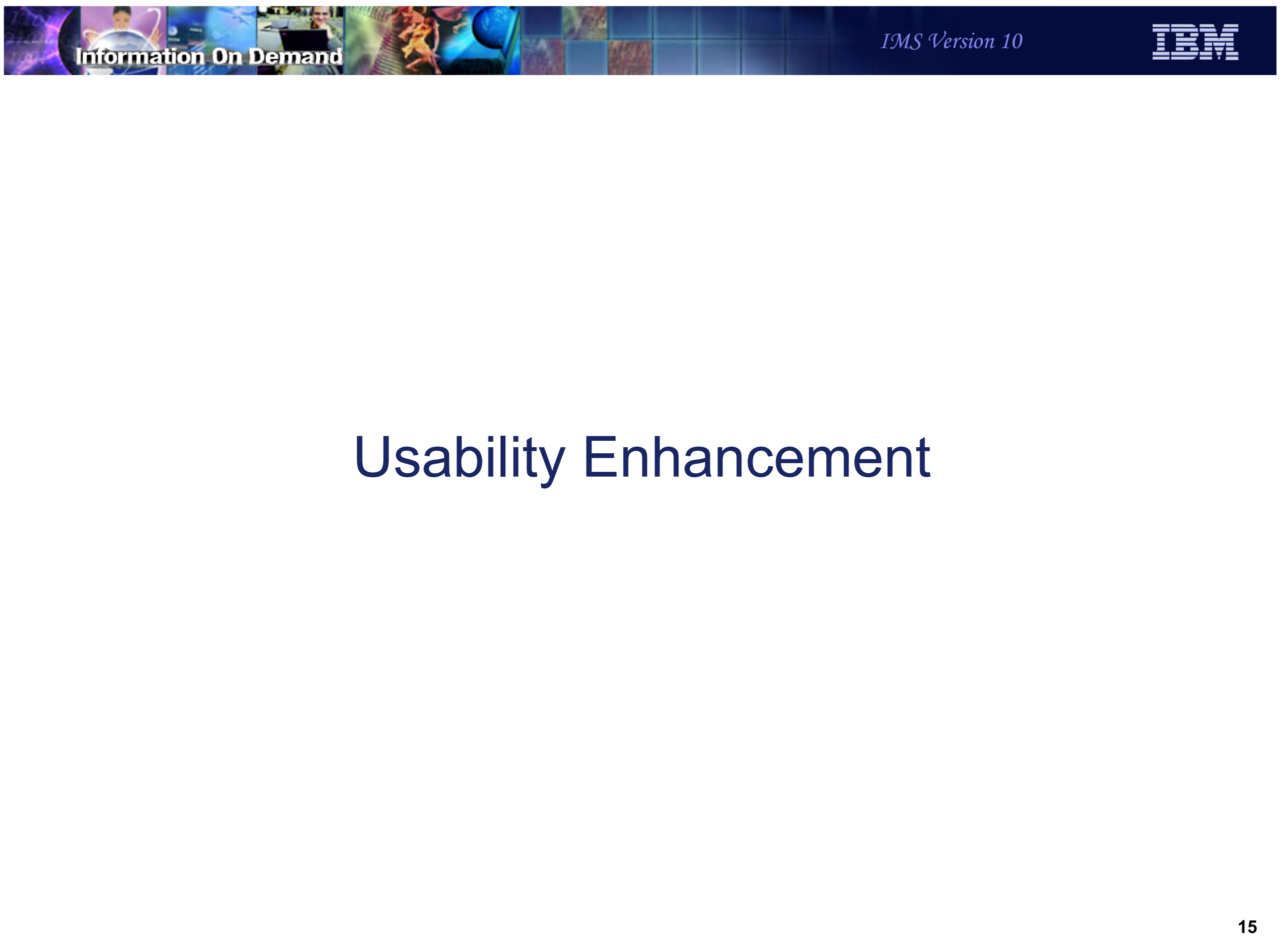
- FPBUFF= parameter in DFSFDRxx member controls location of FP buffers
    - Previous releases
        - FPBUFF=LOCAL placed DBFCONT0, HSSP, and HSRE buffers in extended private
            - DBFCONT0 includes database buffer pool (DBBF) and SVSO buffer pools
            - Recommended
        - Default was to place these blocks and buffers in ECSA
    - IMS V10
        - FPBUFF=LOCAL is the default
            - Places DBFCONTx modules and HSSP and HSRE buffers in extended private
            - Recommended
        - FPBUFF=ECSA places these modules and buffers in ECSA
            - Not recommended

Figure 11-13. FDBR FPBUFF= Parameter Default CMA01.0

Notes:

APARs PQ97745 for IMS V9 and PQ80264 and PQ97043 for IMS V8 added a new parameter in the DFSFDRxx PROCLIB member for FDBR. FPBUFF=LOCAL could be specified to place module DFSCONT0 and HSSP and HSRE buffers in extended private. There was no other valid value for FPBUFF=. If the parameter was not specified, the module and buffers were placed in ECSA.

IMS V10 changes the default. If the FPBUFF= parameter is not specified, DBFCONT1, DBFCONT3, DBFCONT4, DBFCONT5, DBFCONT6 and HSSP and HSRE buffers placed in extended private. A specification of FPBUFF=ECSA places these modules and buffers in ECSA. A specification of FPBUFF= LOCAL and allowing the parameter to default is recommended. There is no advantage to placing the modules and buffers in ECSA.


Information On Demand
IMS Version 10
IBM

Usability Enhancement

Figure 11-14. [No title] CMA01.0

Notes:

Information On Demand

DFSVSMxx Specification for QUITCI

- New DFSVSMxx parameter
  - SDEPQCI= S | D | A
    - S - Force QUITCI by Scan utility
    - D - Force QUITCI by Delete utility
    - B - Force QUITCI by both Scan and Delete utility
  - If SDEPCI= is not specified, utility control statement determines the use of QUITCI
    - No change from previous releases

- Benefit
  - Eliminates need to specify a QUITCI control statement on each Scan or Delete utility execution
  - May be used to enforce the use of QUITCI

Figure 11-15. DFSVSMxx Specification for QUITCI CMA01.0

Notes:

QUITCI is an optional parameter specified in the SYSIN data set for the DEDB SDEP Scan and Delete utilities. It is used in data sharing environments to cause other IMS systems to release their current SDEP CI and any preallocated SDEP CIs. Any partially filled SDEP CIs are written to the area data set. This simplifies the running of the Scan and Delete utilities by making it easy to ensure that they process all committed SDEPs without waiting on CIs to be filled.

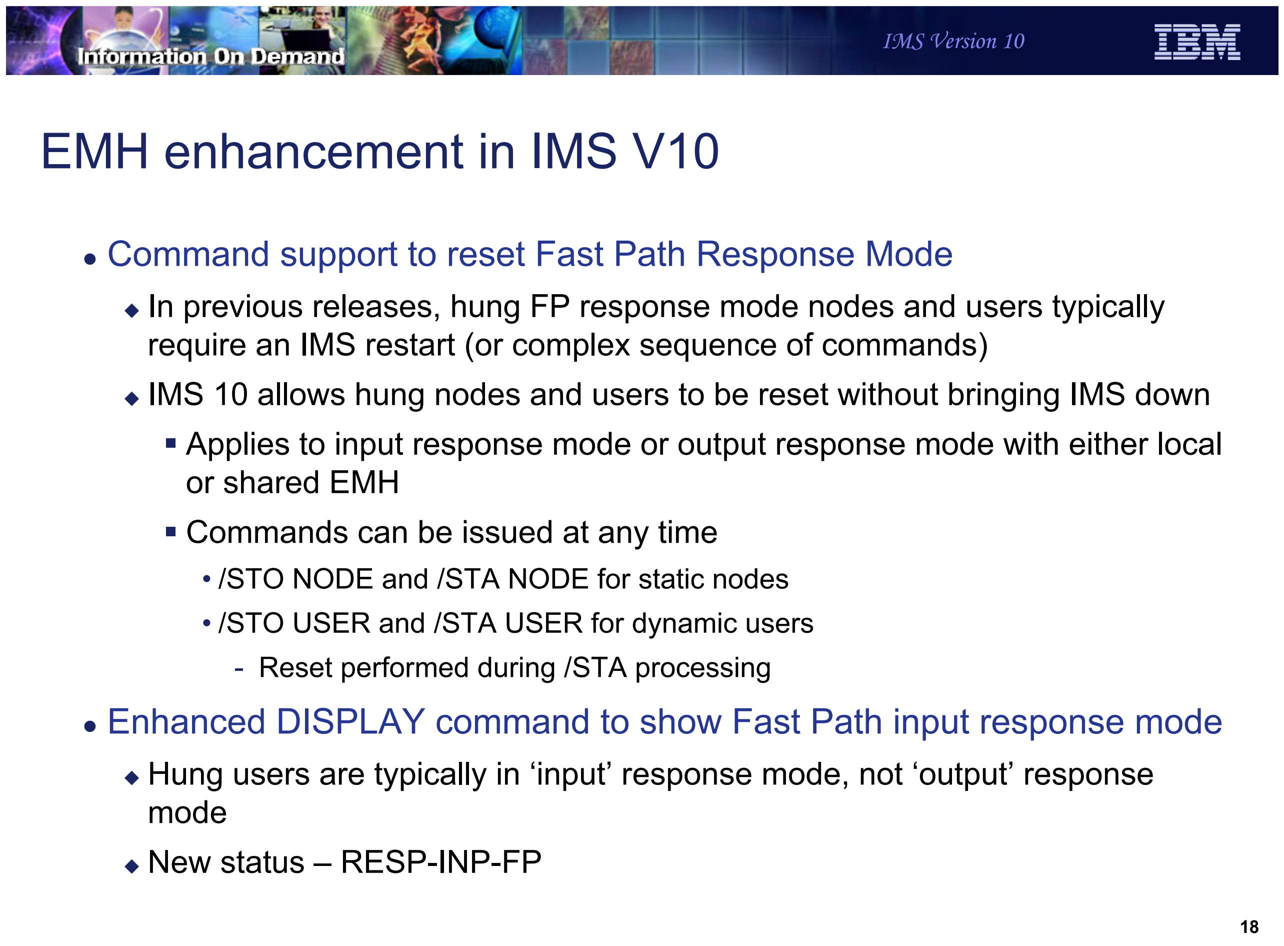
Some users want to ensure that QUITCI is used by all executions of the Scan and/or Delete utilities. The new SDEPQCI parameter in the DFSVSMxx member may be used enforce the use of QUITCI, even when it is not specified on the SYSIN input to the utilities.

Information On Demand
IMS Version 10
IBM

EMH Enhancement

Figure 11-16. [No title] CMA01.0

Notes:


Information On Demand
IMS Version 10
IBM

EMH enhancement in IMS V10

- Command support to reset Fast Path Response Mode
  - In previous releases, hung FP response mode nodes and users typically require an IMS restart (or complex sequence of commands)
  - IMS 10 allows hung nodes and users to be reset without bringing IMS down
    - Applies to input response mode or output response mode with either local or shared EMH
    - Commands can be issued at any time
      - /STO NODE and /STA NODE for static nodes
      - /STO USER and /STA USER for dynamic users
        - Reset performed during /STA processing
- Enhanced DISPLAY command to show Fast Path input response mode
  - Hung users are typically in ‘input’ response mode, not ‘output’ response mode
  - New status – RESP-INP-FP

Figure 11-17. EMH enhancement in IMS V10 CMA01.0

Notes:

This EMH enhancement addresses a rarely occurring situation; however, when it does occur, an IMS outage is likely needed to correct the problem.

Full function users have been able to reset response mode for hung terminals via the /RST, /STO, and /STA commands; this enhancement extends this support to Fast Path users. You cannot use the /RST command to reset response mode for Fast Path nodes/users as you can for full function users. This is because Fast Path requires a session to be in a quiesced state and /RST does not have this requirement.

RESP-INP-FP is a new status in IMS Version 10 that may be returned by the following /DIS commands

/DIS NODE, /DIS NODE RESPINP

/DIS USER, /DIS USER RESPINP

/DIS STATUS, /DIS STATUS NODE, /DIS STATUS USER

Input response mode is now indicated by two statuses, RESP-INP-FP for Fast Path, and RESP-INP for full function. If you have automation dependent on the RESP-INP status and use Fast Path EMH, your automation needs to be changed to handle both input response mode statuses.

Information On Demand IMS Version 10 IBM

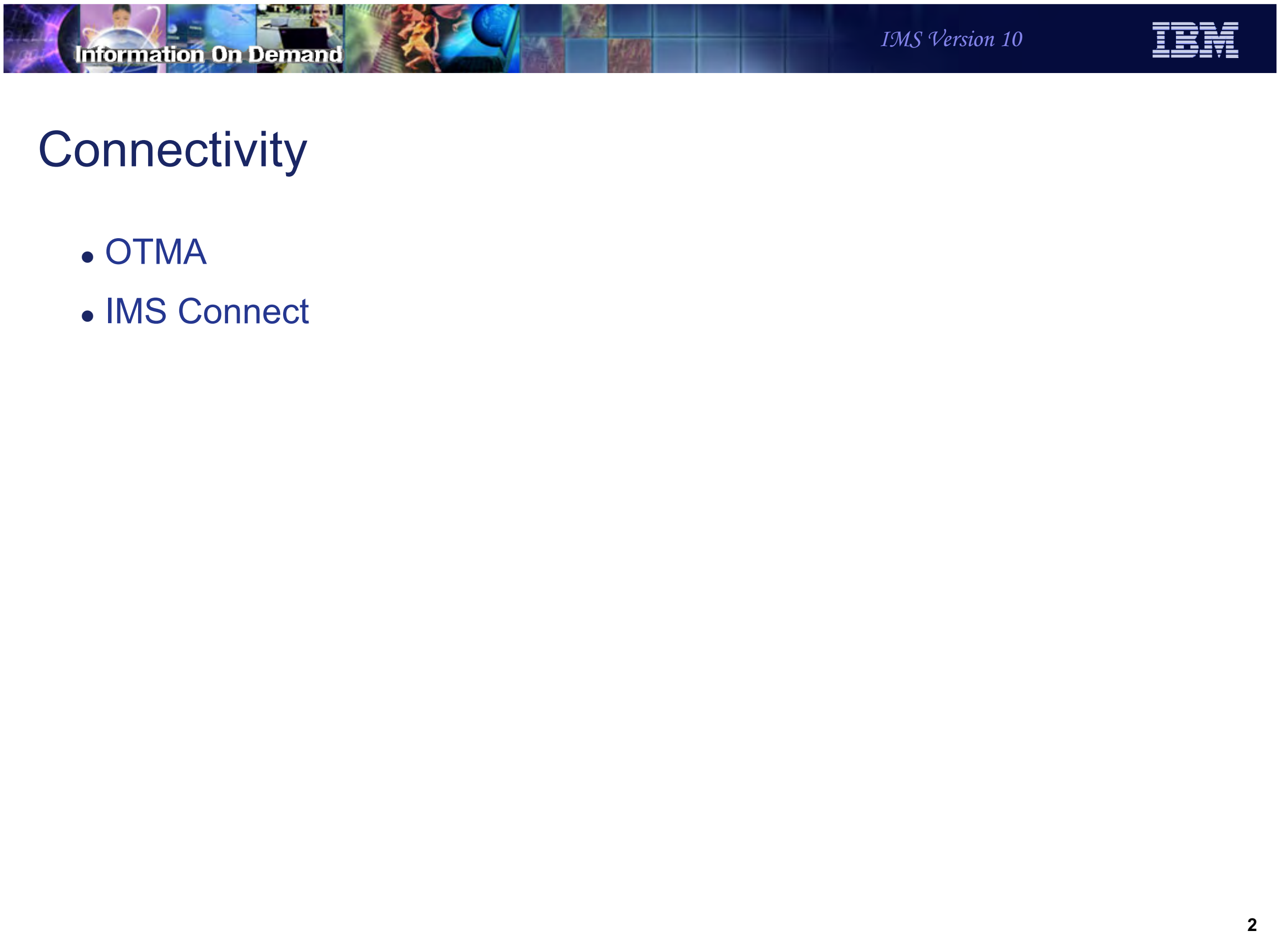
Fast Path Enhancements Summary

- Command Enhancements
  - Improved UPD DB and QRY DB commands
  - Improved use of ECSA
- Shared VSO enhancements
  - Simplified definition
  - Improved use of ECSA by FDBR and XRF tracking systems
- Capacity enhancements
  - Increased output thread and buffers
  - Improved use of ECSA
- Usability enhancement
  - Simple enforcement of QUITCI for SDEP Scan and Delete utilities
- EMH enhancement
  - Capability to reset Fast Path response mode

Figure 11-18. Fast Path Enhancements Summary CMA01.0

Notes:

Unit 12.Connectivity



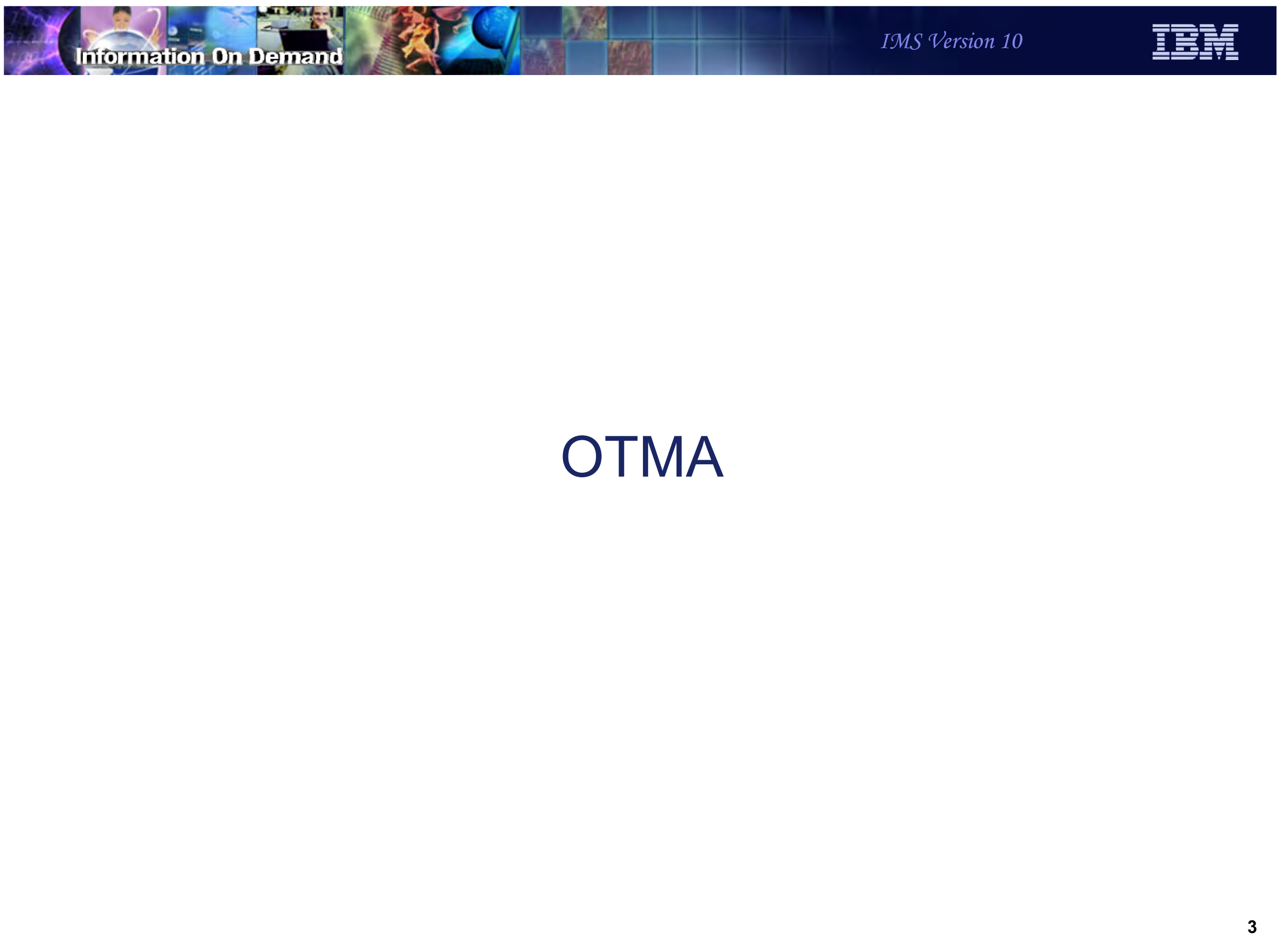
Information On Demand
IMS Version 10

Connectivity

- OTMA
- IMS Connect

Figure 12-1. Connectivity CMA01.0

Notes:



OTMA

Figure 12-2. [No title] CMA01.0

Notes:

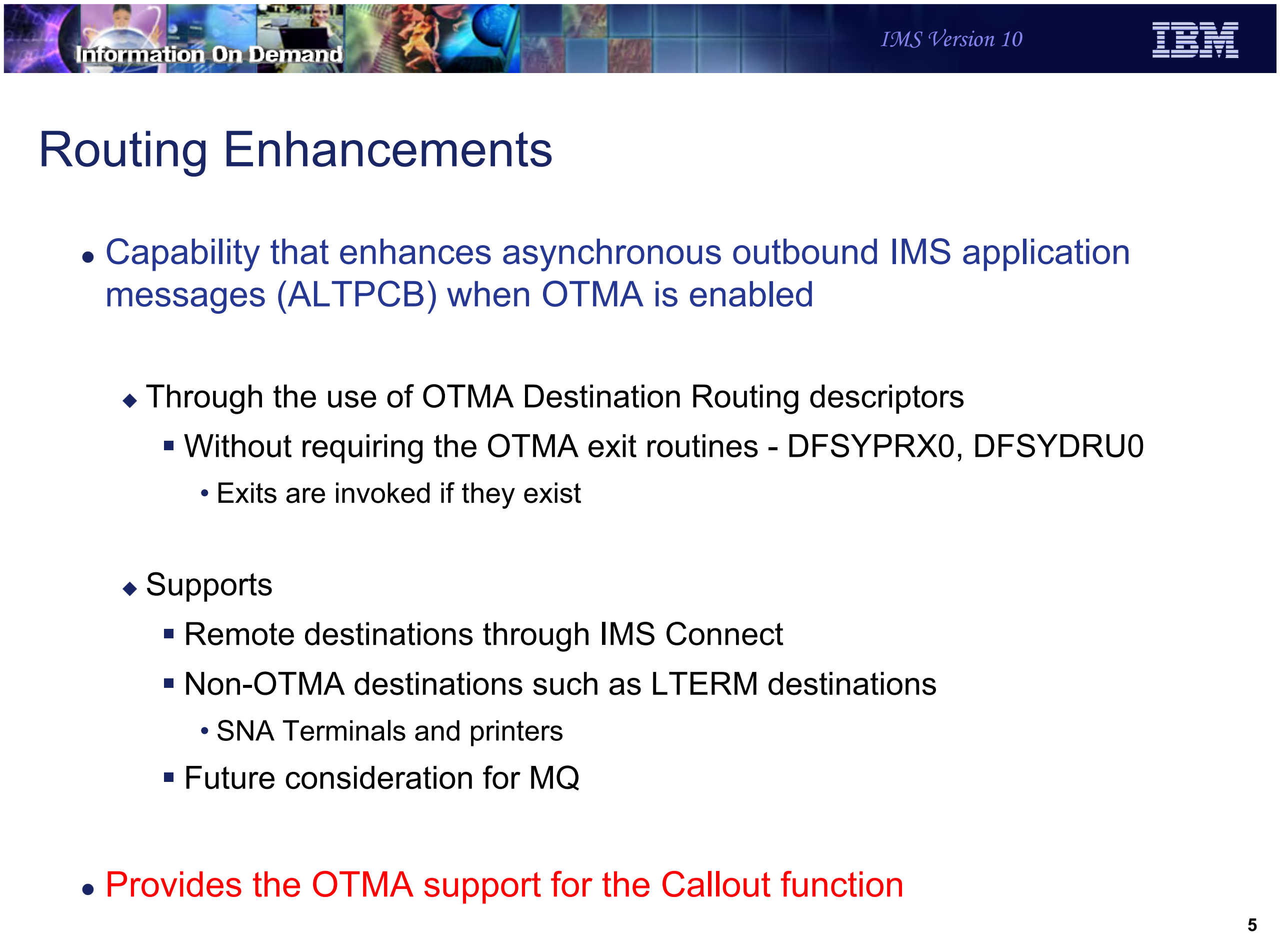
Highlights

- OTMA addresses high availability requirements through
  - Routing Enhancements
    - Destination Routing descriptors
    - Resume TPIPE security
  - Automatic flood detection and control of input messages
  - Time-out control
  - TPIPE storage clean-up
  - Member level security
  - Asynchronous message enhancements
  - Enhanced OTMA display information

Figure 12-3. Highlights CMA01.0

Notes:

OTMA enhancements address several high availability requirements. These include the areas listed on the visual.


Information On Demand
IMS Version 10
IBM

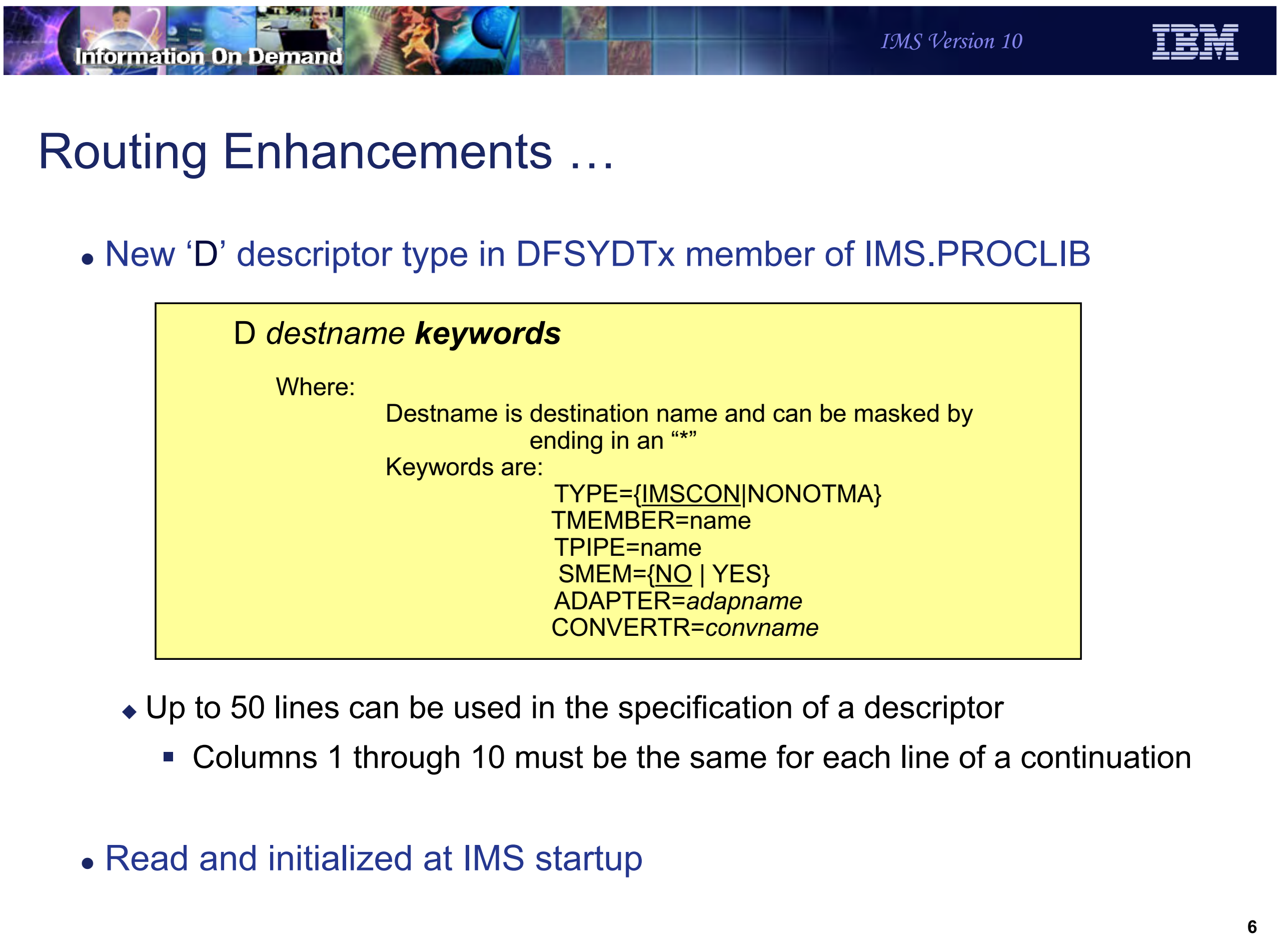
Routing Enhancements

- Capability that enhances asynchronous outbound IMS application messages (ALTPCB) when OTMA is enabled
  - Through the use of OTMA Destination Routing descriptors
    - Without requiring the OTMA exit routines - DFSYPRX0, DFSYDRU0
      - Exits are invoked if they exist
  - Supports
    - Remote destinations through IMS Connect
    - Non-OTMA destinations such as LTERM destinations
      - SNA Terminals and printers
    - Future consideration for MQ
- Provides the OTMA support for the Callout function

Figure 12-4. Routing Enhancements CMA01.0

Notes:

Prior to IMS V10, IMS systems that enabled OTMA and also produced ALTPCB outbound messages for external destinations required system programmers to code several assembler OTMA routing exits including DFSYPRX0 & DFSYDRU0. This requirement oftentimes inhibited or delayed the adoption of new connectivity implementations such as IMS Connect. IMS V10 introduces new OTMA Destination Routing Descriptors that can eliminate the requirement to code the OTMA exits by externalizing the definitions and specifications that the exits provide. Note, however, that if the exits exist, they will be called with the routing information provided by the descriptors already set. Additionally these Descriptors have the ability to route from OTMA to non-OTMA destinations such as SNA printers and terminals. Future support for MQ is under consideration.



Information On Demand
IMS Version 10
IBM

Routing Enhancements …

- New ‘D’ descriptor type in DFSYDTx member of IMS.PROCLIB

D *destname* ***keywords***

Where:

Destname is destination name and can be masked by ending in an “*”

Keywords are:

TYPE={IMSCON|NONOTMA}
TMEMBER=name
TPIPE=name
SMEM={NO | YES}
ADAPTER=*adapname*
CONVERTR=*convname*

- Up to 50 lines can be used in the specification of a descriptor
    - Columns 1 through 10 must be the same for each line of a continuation

- Read and initialized at IMS startup

Figure 12-5. Routing Enhancements … CMA01.0

Notes:

The new ‘D’ descriptor type for the DFSYDTx member of IMS.PROCLIB includes keywords as follows:

- TYPE= determines if output is destined for IMS Connect (IMSCON) or non-OTMA (NONOTMA). This is a required keyword.
- TMEMBER= 1 to 16 character client name. Required for TYPE=IMSCON. Ignored for TYPE=NONOTMA.
- TPIPE= 1 to 8 character TPIPE name. Optional for TYPE=IMSCON, defaults to Destination name. Ignored for NONOTMA.
- SMEM= indicates if this destination is a Super Member. Optional keyword for TYPE=ICON, defaults to SMEM=NO. Ignored for TYPE=NONOTMA. If “YES”, the name defined in the TMEMBER keyword becomes the Super Member name and can only be 4 characters.

- ADAPTER= 1 to 8 character name of the IMS Connect Adapter to be used for the message, e.g., one example is an adapter for XML transformation. Optional for TYPE=IMSCON and ignored for TYPE=NONOTMA.
- CONVERTR= 1 to 8 character name of the Converter to be used by the Adapter. Required for TYPE=IMSCON if ADAPTER is specified. Ignored for TYPE=NONOTMA.

Routing Enhancements …

- Example

```
M HWSICON1             DRU=DFSYDRU0  INPUT=5000 T/O=5

D OTMACL99 TYPE=IMSCON TMEMBER=HWS1 TPIPE=HWS1TP01
D OTMACL*  TYPE=IMSCON TMEMBER=HWS2

D PRNTR3A  TYPE=NONOTMA

D SOAPGW1  TYPE=IMSCON TMEMBER=HWS2 TPIPE=HWS2SOAP
D SOAPGW1  ADAPTER=XMLADPTR CONVERTR=XMLCNVTR

D SOAPGW*  TYPE=IMSCON TMEMBER=HWS3 TPIPE=HWS3SOAP
D SOAPGW*  ADAPTER=XMLADPTR CONVERTR=XMLCNVT3
```

Masked descriptor

continuation

Masked descriptor
Also note that for this specific example,
all destination matches for any destination
beginning with SOAPGW… will be
routed to the single TPIPE HWS3SOAP

Note: any descriptors that result in syntax errors are ignored

Figure 12-6. Routing Enhancements … CMA01.0

Notes:

Multiple OTMA descriptors can be defined in the same DFSYDTx member.

This example illustrates six descriptors:

- The first is a TMEMBER descriptor specifying the DRU exit for TMEMBER "HWSICON1".
- The second is a Destination Routing descriptor for destination OTMACL99 that will be routed to IMS Connect TMEMBER "HWS1" with TPIPE "HWS1TP01".
- The third is a Destination Routing descriptor for destinations matching the mask "OTMACL*". The messages will be routed to a TMEMBER of "HWS2" with a TPIPE of the destination matching the mask, e.g., OTMACL04 if the destination using this descriptor was OTMACL04. Note that this along with the second descriptor illustrate that more specific destinations must be coded ahead of generic ones.
- The fourth is a Destination Routing descriptor for destination "PRNTR3A" that will be routed to legacy IMS.

- The fifth is a Destination Routing descriptor for “SOAPGW1” that will be routed to IMS Connect TMEMBER “HWS2” with TPIPE “HWS2SOAP” and will result in XML translation.
- The sixth is a masked Destination Routing descriptor for destinations that begin with “SOAPGW*”. This last descriptor will not be used for SOAPGW1 because the previous descriptor already specifically addressed that destination name. Note that all the destination matches for this descriptor will be routed to the one specified TPIPE “HWS3SOAP”.

Information On Demand
IMS Version 10
IBM

Routing Enhancements …

- Usage
  - Application programmers
    - Ensure IMS application ALTPCB destination name matches the name of a Destination Routing descriptor
  - System programmers
    - Define the new ‘D’ descriptors in DFSYDTx
      - Code more specific descriptors before generic definitions
        - Searched and used in the order coded
  - Error message enhancement
  - DFS2385E SYNTAX ERROR FOR DESCRIPTOR = *descriptor errortext*
    - New error text for the ‘D’ descriptors
    - Also issued if a previous release of IMS attempts to read DFSYDTx with the new descriptors

Figure 12-7. Routing Enhancements … CMA01.0

Notes:

To correctly enable the use of the new callout descriptors, application programmers must code the destination of an asynchronous output message switch (ALTPCB) to match the destination name of a corresponding descriptor. IMS systems programmers or system administrators, on the other hand, will also need to create the sure descriptors in the DFSYDTx member of IMS.PROCLIB. Because descriptors can be masked (end in an “*”) and are searched in the order they are coded, more specific descriptors should appear before generic ones.

The DFS2385E error message has been enhanced to catch syntax errors associated with the new descriptor keywords. New error text include:

- DESTINATION NAME NOT GIVEN OR BEGINS AFTER COLUMN 3

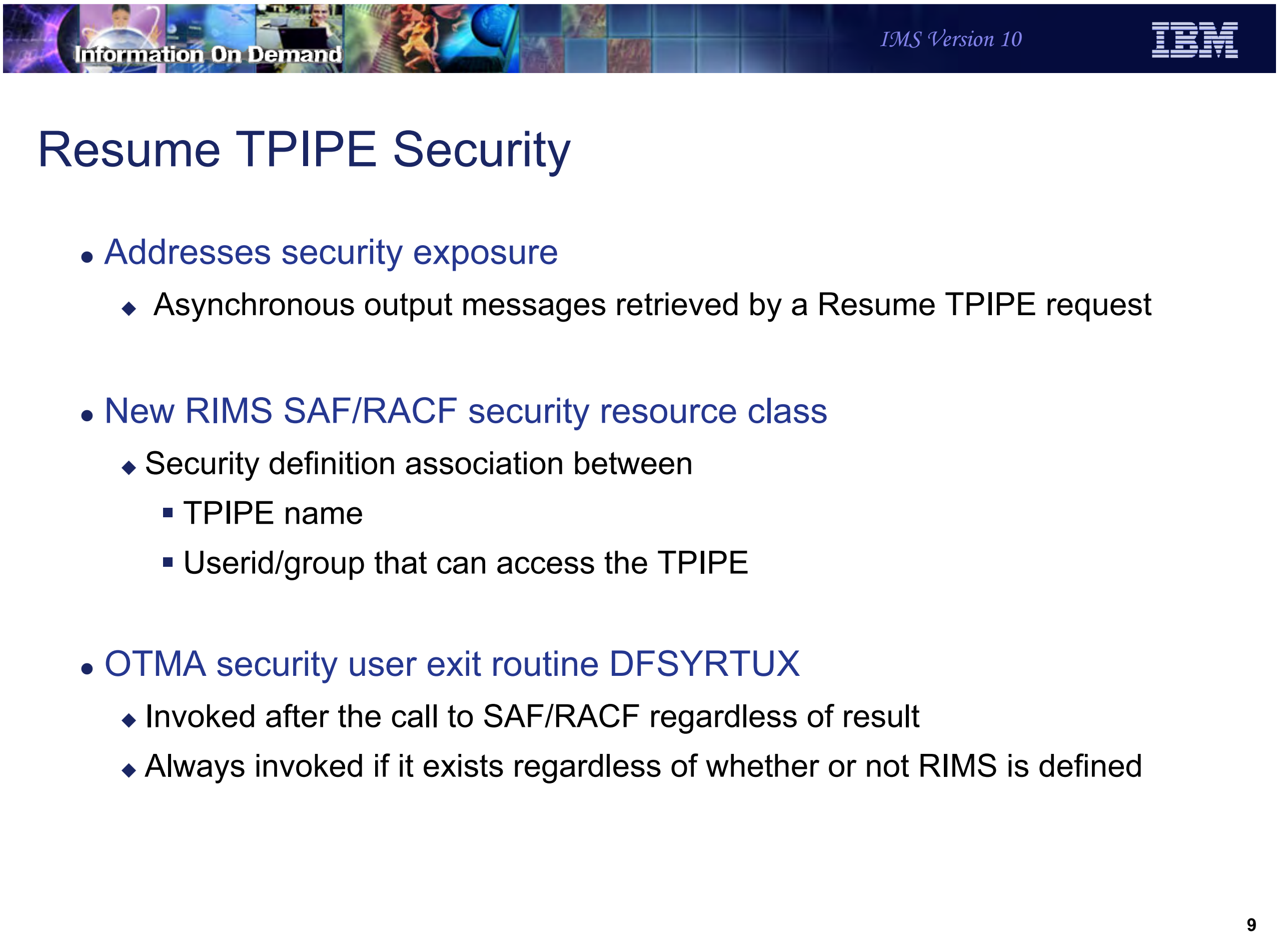
- DESTINATION NAME LONGER THAN 8 CHAR

- INVALID TYPE SPECIFIED

- DUPLICATE TYPE KEYWORD

- TMEMBER REQUIRED FOR TYPE=IMSCON

- TMEMBER MUST BE 1 TO 16 CHARACTERS LONG
- DUPLICATE TMEMBER KEYWORD
- TPIPE MUST BE 1 TO 8 CHARACTERS LONG
- DUPLICATE TPIPE KEYWORD
- SMEM MUST BE YES OR NO
- DUPLICATE SMEM KEYWORD
- ADAPTER MUST BE 1 TO 8 CHARACTERS LONG
- DUPLICATE ADAPTER KEYWORD
- CONVERTR MUST BE 1 TO 8 CHARACTERS LONG
- DUPLICATE CONVERTR KEYWORD
- CONVERTR REQUIRED WITH ADAPTER



Information On Demand
IMS Version 10
IBM

Resume TPIPE Security

- Addresses security exposure
  - Asynchronous output messages retrieved by a Resume TPIPE request
- New RIMS SAF/RACF security resource class
  - Security definition association between
    - TPIPE name
    - Userid/group that can access the TPIPE
- OTMA security user exit routine DFSYRTUX
  - Invoked after the call to SAF/RACF regardless of result
  - Always invoked if it exists regardless of whether or not RIMS is defined

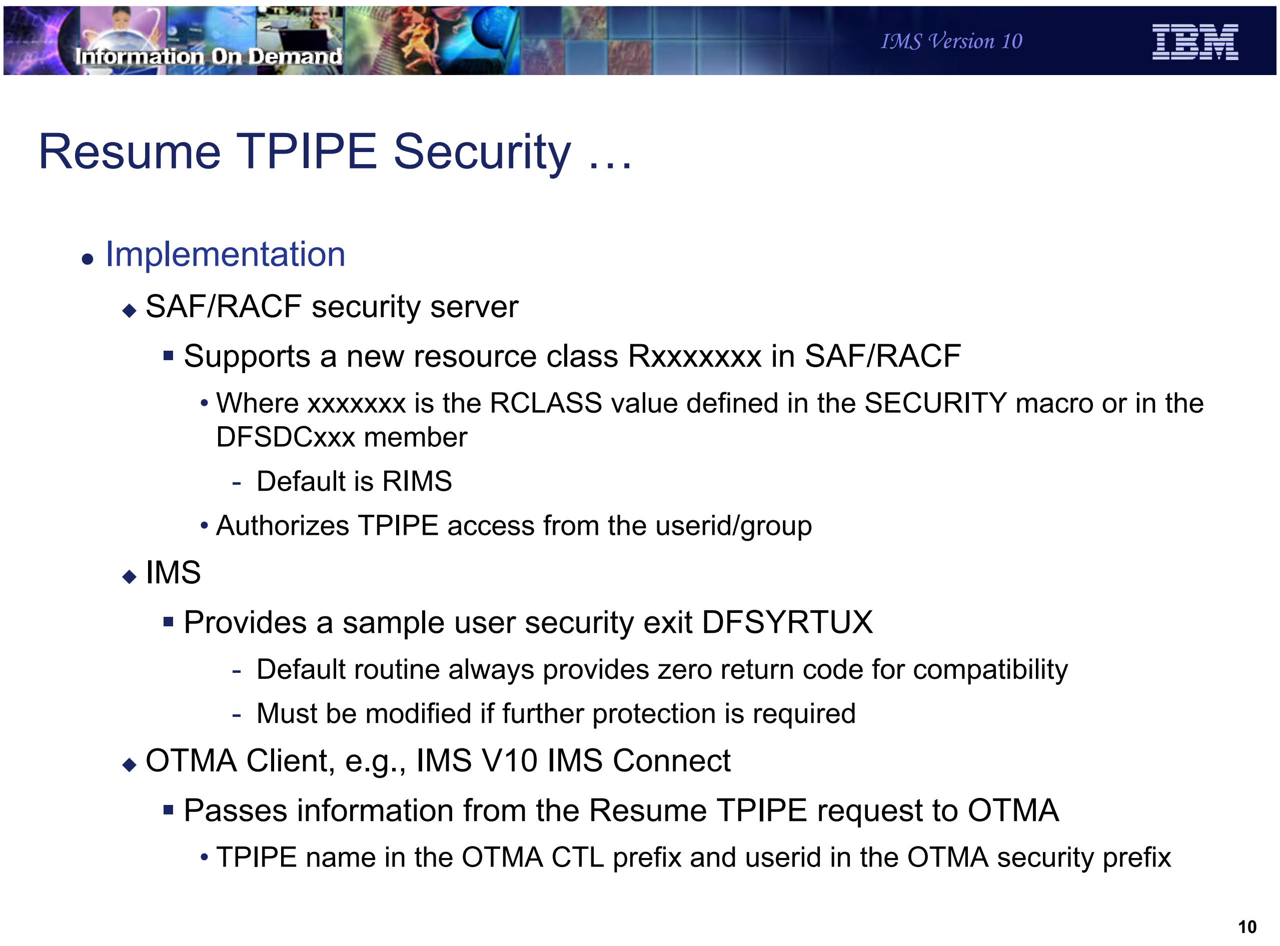
Figure 12-8. Resume TPIPE Security CMA01.0

Notes:

The next enhancement addresses security in the destination routing environment.

IMS transactions and commands that flow through OTMA from various clients are protected by current security classes, namely: TIMS and CIMS. The responses are guaranteed to be delivered to the client that initiated the transactions and commands. Output messages in the hold queue that are generated as a result of asynchronous processing, however, are not protected by any security class. When those messages are retrieved by an OTMA client using RESUME TPIPE, a security exposure can occur. The function provided by Resume TPIPE Security protects these output messages by establishing a security class named RIMS within RACF or any non-IBM security product. Within this class, the security definitions are associated with the TPIPE name along with the list of user IDs or group names under this TPIPE. The enhancement, therefore, allows IMS installations to optionally authorize the user ID, together with the TPIPE name that is contained in the Resume TPIPE command message before any of these messages are sent to a client.

OTMA also provides the DFSYRTUX security exit routine as an opportunity to overrule the SAF/RACF decision or to extend the security check to allow modifications as needed by the environment.


Information On Demand
IMS Version 10
IBM

Resume TPIPE Security …

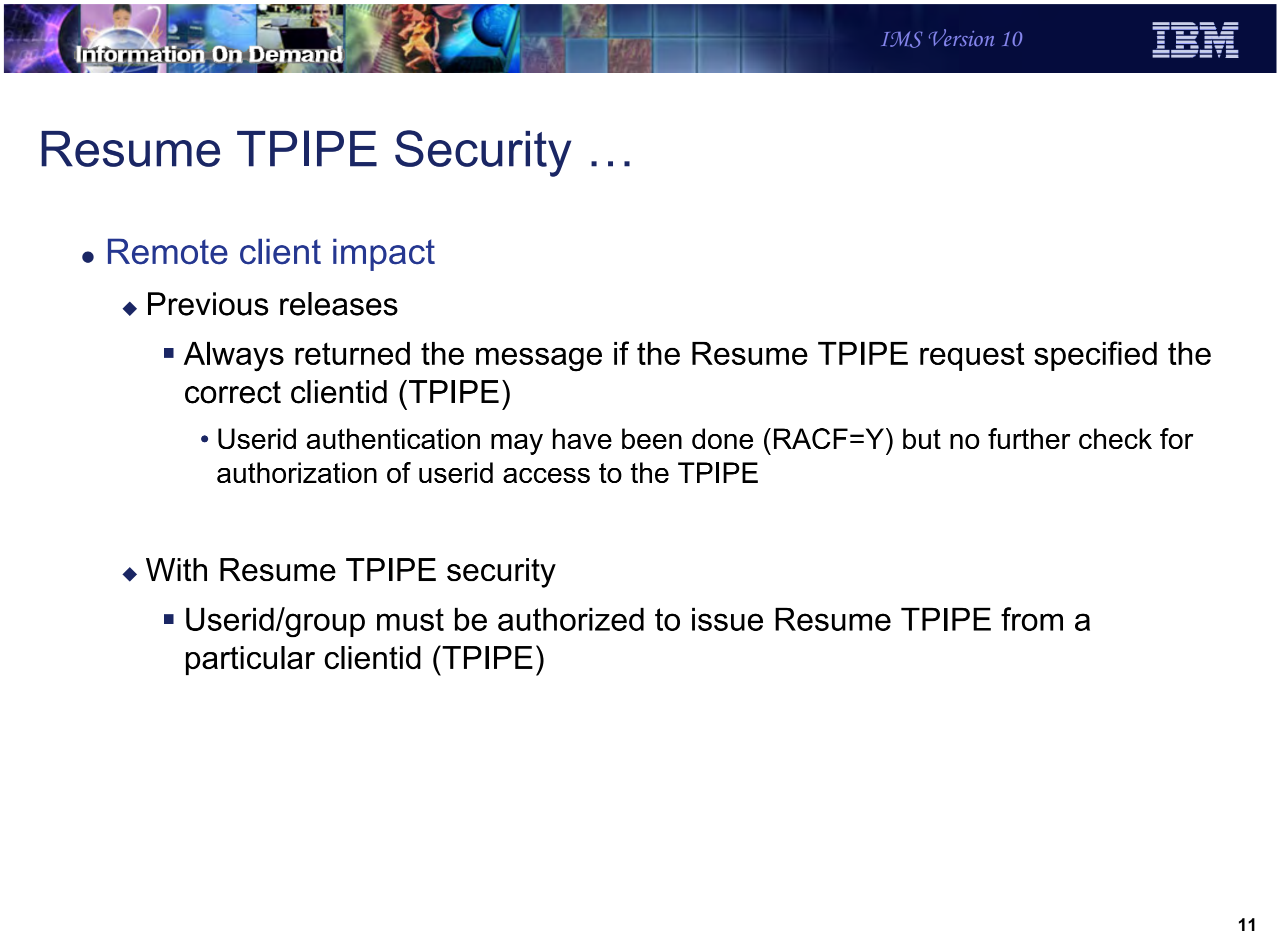
- Implementation
  - SAF/RACF security server
    - Supports a new resource class Rxxxxxxx in SAF/RACF
      - Where xxxxxxx is the RCLASS value defined in the SECURITY macro or in the DFSDCxxx member
        - Default is RIMS
      - Authorizes TPIPE access from the userid/group
  - IMS
    - Provides a sample user security exit DFSYRTUX
        - Default routine always provides zero return code for compatibility
        - Must be modified if further protection is required
  - OTMA Client, e.g., IMS V10 IMS Connect
    - Passes information from the Resume TPIPE request to OTMA
      - TPIPE name in the OTMA CTL prefix and userid in the OTMA security prefix

Figure 12-9. Resume TPIPE Security … CMA01.0

Notes:

To take advantage of Resume TPIPE Security, the following actions are required:

- Define a new resource class, TPIPE name, and user IDs in RACF (refer to SA22-7683 *Security Server RACF Security Administrator’s Guide*) or applicable security server for the environment. The resource class comprises the resource class type of “R” and the resource class name whose value is taken from the RCLASS parameter of the SECURITY macro or in the DFSDCxxx member of Proclib. If RCLASS is omitted, the resource class name defaults to “IMS.” The resulting class, then, is “Rxxxxxxx’ where ‘xxxxxxx’ is the value of RCLASS or “RIMS” as the default.
- If needed, code, assemble, and bind the user exit in a library that is concatenated with IMS SDFSRESL under DD name STEPLIB or JOBLIB.
- Ensure that the OTMA client, e.g., IMS Connect, is at the correct IMS V10 level to pass in the TPIPE name in the OTMA CTL prefix and the userid/group in the security prefix header as part of the message for the Resume TPIPE command.


Information On Demand
IMS Version 10
IBM

Resume TPIPE Security …

- Remote client impact
  - Previous releases
    - Always returned the message if the Resume TPIPE request specified the correct clientid (TPIPE)
      - Userid authentication may have been done (RACF=Y) but no further check for authorization of userid access to the TPIPE
  - With Resume TPIPE security
    - Userid/group must be authorized to issue Resume TPIPE from a particular clientid (TPIPE)

Figure 12-10. Resume TPIPE Security … CMA01.0

Notes:

The implementation of Resume TPIPE Security could possibly impact existing clients. In previous releases, a userid that was provided in the Resume TPIPE request was authenticated if IMS Connect was configured with security enabled. As long as the Resume TPIPE request, however, passed in the correct TPIPE name (clientid) then the associated asynchronous output messages could be retrieved.

In IMS V10, if the Rxxxxxxx | RIMS resource class is defined then security violations can occur where they previously did not. When retrieving the asynchronous output from IMS, the client signals IMS with a Resume TPIPE command and an IRM timeout value. This timeout value expires if there are no messages received by the client. Additionally, with the new support, a security check can result in success or failure. If the security validation is a success, normal processing takes place. On the other hand, if a security violation occurs, a new NAK message is sent to the client.

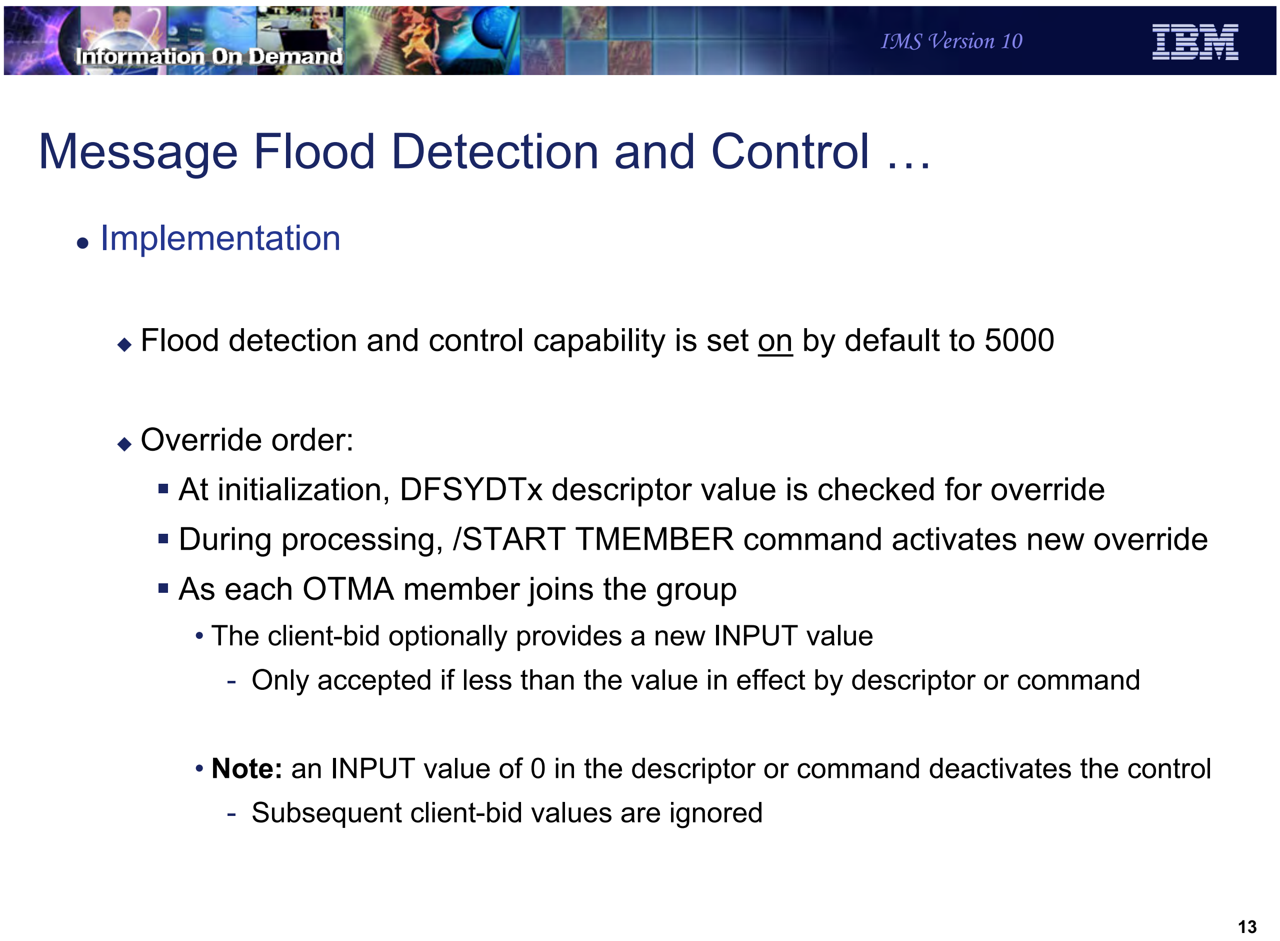
Message Flood Detection and Control

- Capability that automatically monitors the growth of active input messages
  - Sets a default max threshold of 5000 active input messages
  - If the IMS system has more than 5000 unscheduled, unprocessed, or orphaned input messages from an OTMA member
    - Any new input message from the same member is rejected
- Prevents possible S40D IMS Abends due to large number of OTMA control blocks associated with the queued requests
- Also delivered in IMS V8 and IMS V9:
  - V8: PK04461, V9: PK04463

Figure 12-11. Message Flood Detection and Control CMA01.0

Notes:

The next enhancement is the Message Flood Detection and Control capability. This function provides a mechanism to automatically monitor the growth of active input messages through OTMA and the control blocks associated with these requests. Specifically, when an OTMA member or client sends a transaction to IMS, OTMA internally creates a control block called the TIB (Transaction Instance Block) to track each active input message. For a send-then-commit (CM1) message, the control block is used for input and output processing after which the storage is freed or reused. For a commit-then-send (CM0) message, the control block is only used for input processing. If, however, several thousand OTMA input transactions are received and waiting to be processed, thousands of control blocks representing the requests could fill up LSQA storage below the line and possibly cause the IMS system to fail with an S40D abend. To prevent this type of OTMA message flood condition, OTMA supports the suppression or control of the input messages for OTMA based on a maximum value for the number of TIBs allowed for an OTMA member in the system.


Information On Demand
IMS Version 10

Message Flood Detection and Control …

- Implementation
  - Flood detection and control capability is set on by default to 5000
  - Override order:
    - At initialization, DFSYDTx descriptor value is checked for override
    - During processing, /START TMEMBER command activates new override
    - As each OTMA member joins the group
      - The client-bid optionally provides a new INPUT value
        - Only accepted if less than the value in effect by descriptor or command
      - **Note:** an INPUT value of 0 in the descriptor or command deactivates the control
        - Subsequent client-bid values are ignored

13

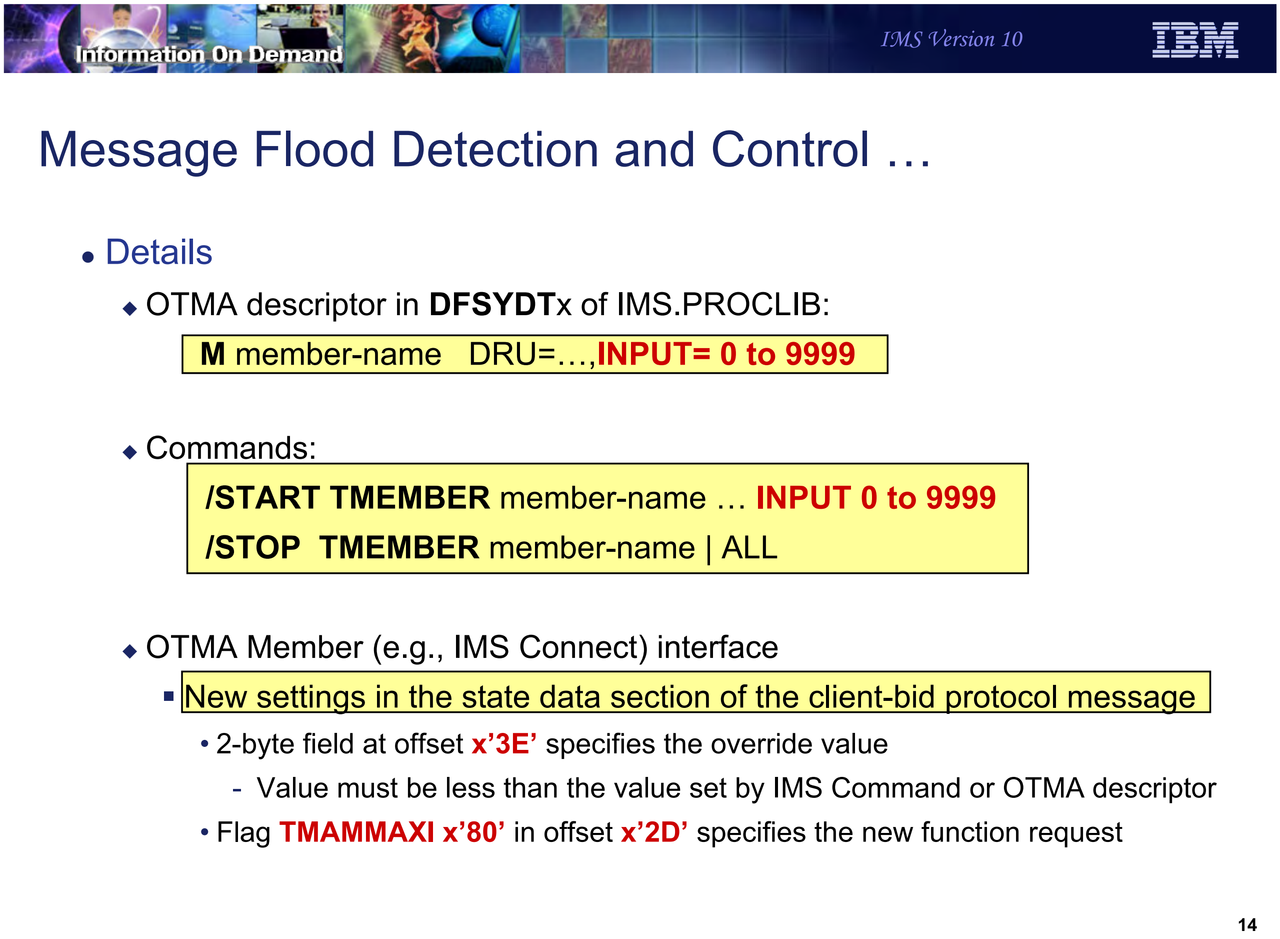
Figure 12-12. Message Flood Detection and Control … CMA01.0

Notes:

By default, the message flood detection and control capability is always on and the maximum threshold value set to 5000. To override this default, several choices are available:

- The OTMA descriptor DFSYDTx in the IMS.PROCLIB library can provide a value which IMS detects at initialization. It is not, however, until the TMEMBER associated with the descriptor is actually started that IMS implements the override value.
- At any time, an operator can issue the /START TMEMBER command that not only starts the member but also provides an override flood detection value that supersedes anything provided in the descriptor.
- As a member joins the group, the client-bid protocol message can also provide an override. This value is honored only if it is less than the value that is already in effect based on the overrides provided by the descriptor or /START TMEMBER command.

Note that the deactivation of the input message flood control capability can be requested by either the DFSYDTx descriptor or the /START TMEMBER command by specifying an INPUT value of 0. A client-bid message cannot override this specification.



Information On Demand
IMS Version 10
IBM

Message Flood Detection and Control …

- Details
  - OTMA descriptor in **DFSYDT**x of IMS.PROCLIB:

```
M member-name   DRU=…,INPUT= 0 to 9999
```

  - Commands:

```
/START TMEMBER member-name … INPUT 0 to 9999
/STOP  TMEMBER member-name | ALL
```

  - OTMA Member (e.g., IMS Connect) interface
    - New settings in the state data section of the client-bid protocol message
      - 2-byte field at offset **x'3E'** specifies the override value
        - Value must be less than the value set by IMS Command or OTMA descriptor
      - Flag **TMAMMAXI x'80'** in offset **x'2D'** specifies the new function request

14

Figure 12-13. Message Flood Detection and Control … CMA01.0

Notes:

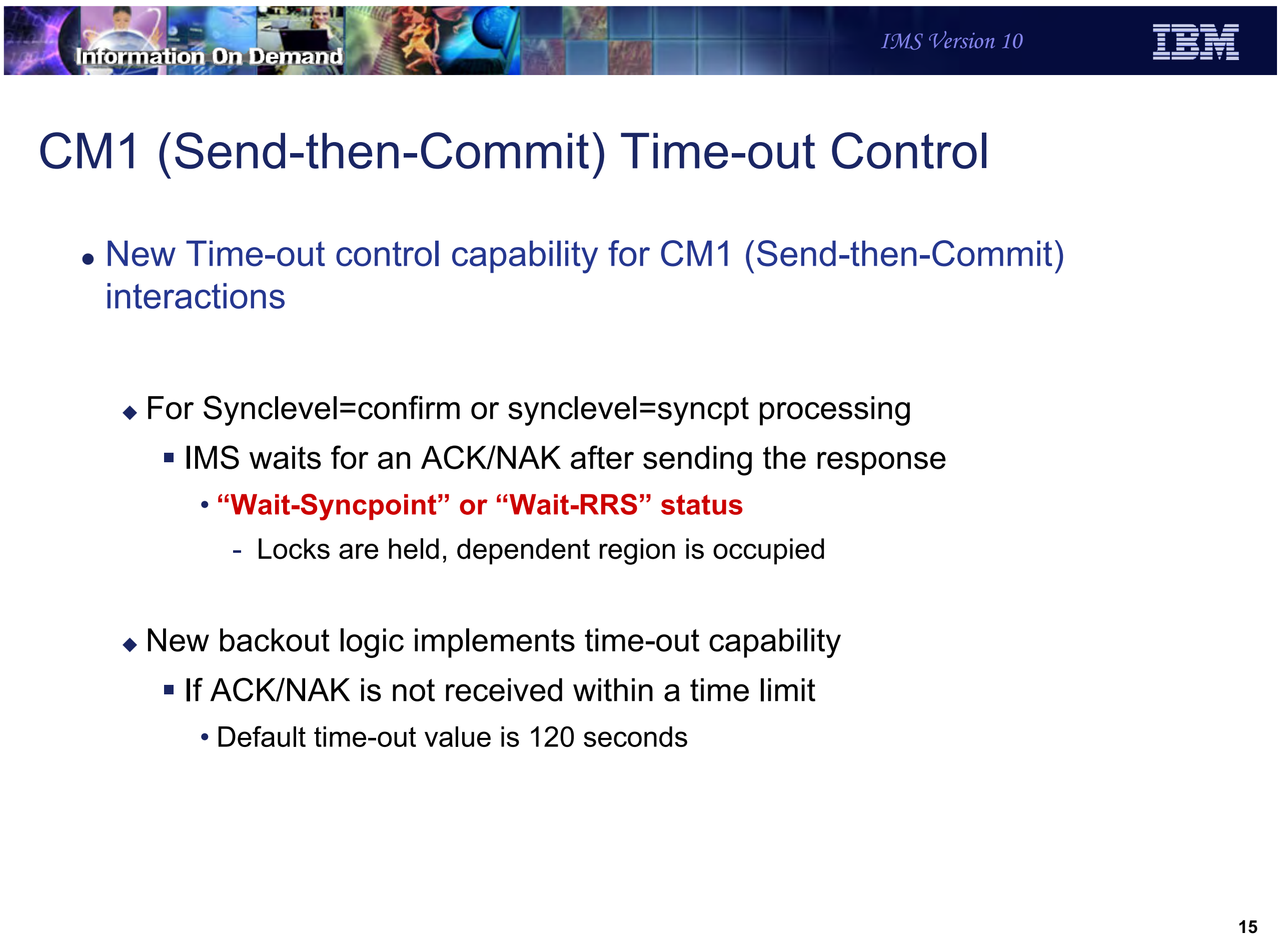
The specifics of the implementation are as follows:

- A new INPUT parameter in the OTMA descriptor DFSYDTx in IMS.PROCLIB. The parameter allows values from 0 to 9999. If the value is set to 0 then the capability for the message flood detection is disabled. Values between 1 and 200 are set to 200 and anything over 9999 is set to 9999.
- The /STOP TMEMBER member-name | ALL command which suppresses new input transactions or commands from a specific OTMA member or all OTMA members. This command does not affect the rest of communications between the stopped member and IMS. That means the following operations can still be performed for a stopped member: the client and server XCF connection remains unchanged; all of the running transactions currently scheduled in the IMS can still be processed, and the responses can be delivered; IMS conversational transactions can continue processing the existing conversation until it ends; all of the OTMA protocol commands including ACK/NACK can still be processed by OTMA. Note that after this command has been issued, a

STO-INPUT status will be displayed on a subsequent /DISPLAY TMEMBER command for the appropriate OTMA members.

- The /START TMEMBER INPUT command with the same range and meaning of the values 0 to 9999.
- New fields in the client-bid protocol interface between the OTMA member and OTMA. For the client-bid protocol message, a new 2-byte field in the offset x’3E’ of the state data can specify the maximum number of active input message for the member. The value can be specified between 0 and 9999. If the value is 0, OTMA will take the default of 5000 for the maximum active input message limit (note that specifying 0 does not turn the capability off). If the value is between 1 and 200, it will be treated as 200. If it is over 9999, it will be set to 9999. A further restriction is that the specified request must be less than any value previously set by a descriptor or command. If the client-bid tries to send in a higher value, it will be reset to the value established by the descriptor or command. Additionally, a new flag TMAMMAXI x’80’ in the offset x’2D’ of the state data needs to be set to inform the OTMA server that the maximum active input message checking function is requested. During processing, if OTMA detects that the active input message limit has been reached, a new NAK code of x’30’ will be sent the OTMA member.

OTMA monitors the growth of the active input messages from members. A warning message DFS1988W is sent to the console to indicate that the input message buildup is approaching the maximum limit. The message will be sent starting at 80% of the message limit and every 5% thereafter. When the maximum limit is reached, an error message DFS1989E is sent to the console and any subsequent OTMA input messages are rejected with a new OTMA sense code x’30’. Once the situation has been alleviated, DFS0767I is issued.


Information On Demand
IMS Version 10

CM1 (Send-then-Commit) Time-out Control

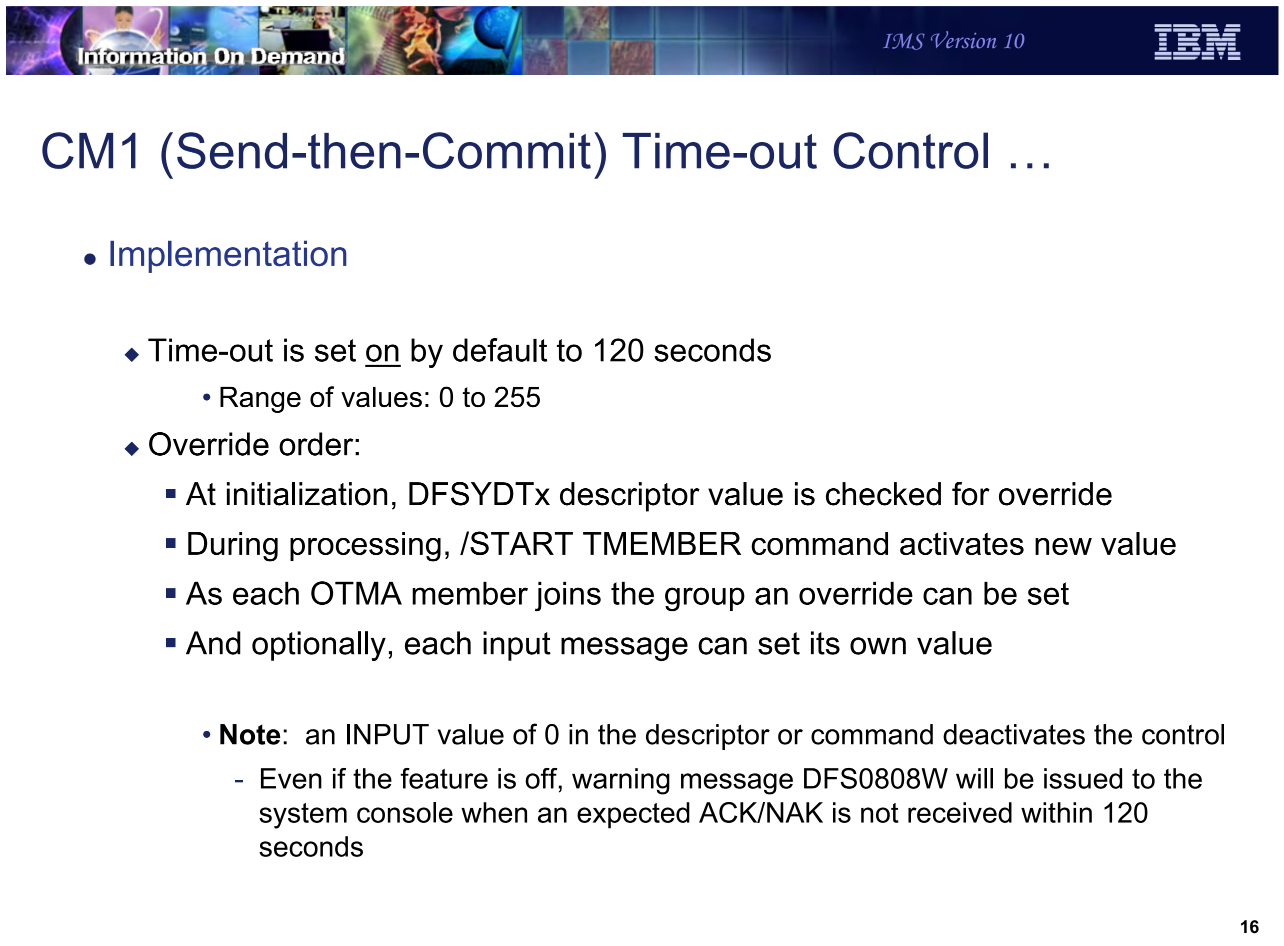
- New Time-out control capability for CM1 (Send-then-Commit) interactions
    - For Synclevel=confirm or synclevel=syncpt processing
        - IMS waits for an ACK/NAK after sending the response
            - **“Wait-Syncpoint” or “Wait-RRS” status**
                - Locks are held, dependent region is occupied
    - New backout logic implements time-out capability
        - If ACK/NAK is not received within a time limit
            - Default time-out value is 120 seconds

15

Figure 12-14. CM1 (Send-then-Commit) Time-out Control CMA01.0

Notes:

The next capability that OTMA provides is a time-out control option that is applicable for CM1 message processing. For an OTMA send-then-commit (CM1) response message with synclevel=confirm or synclevel=syncpt, IMS expects an ACK/NAK from the OTMA client. Due to the possibility of a client programming error or a network failure or delay, the expected ACK/NAK may not be received by IMS. A missing or delayed ACK/NAK results in a “wait-syncpoint” condition for the IMS dependent region that processed the OTMA transaction. To resolve this situation, OTMA has been enhanced to detect this “wait-syncpoint” condition and take an appropriate time-out action. The default time-out value is 120 seconds.



CM1 (Send-then-Commit) Time-out Control …

- Implementation
  - Time-out is set on by default to 120 seconds
    - Range of values: 0 to 255
  - Override order:
    - At initialization, DFSYDTx descriptor value is checked for override
    - During processing, /START TMEMBER command activates new value
    - As each OTMA member joins the group an override can be set
    - And optionally, each input message can set its own value
    - **Note**: an INPUT value of 0 in the descriptor or command deactivates the control
      - Even if the feature is off, warning message DFS0808W will be issued to the system console when an expected ACK/NAK is not received within 120 seconds

Figure 12-15. CM1 (Send-then-Commit) Time-out Control … CMA01.0

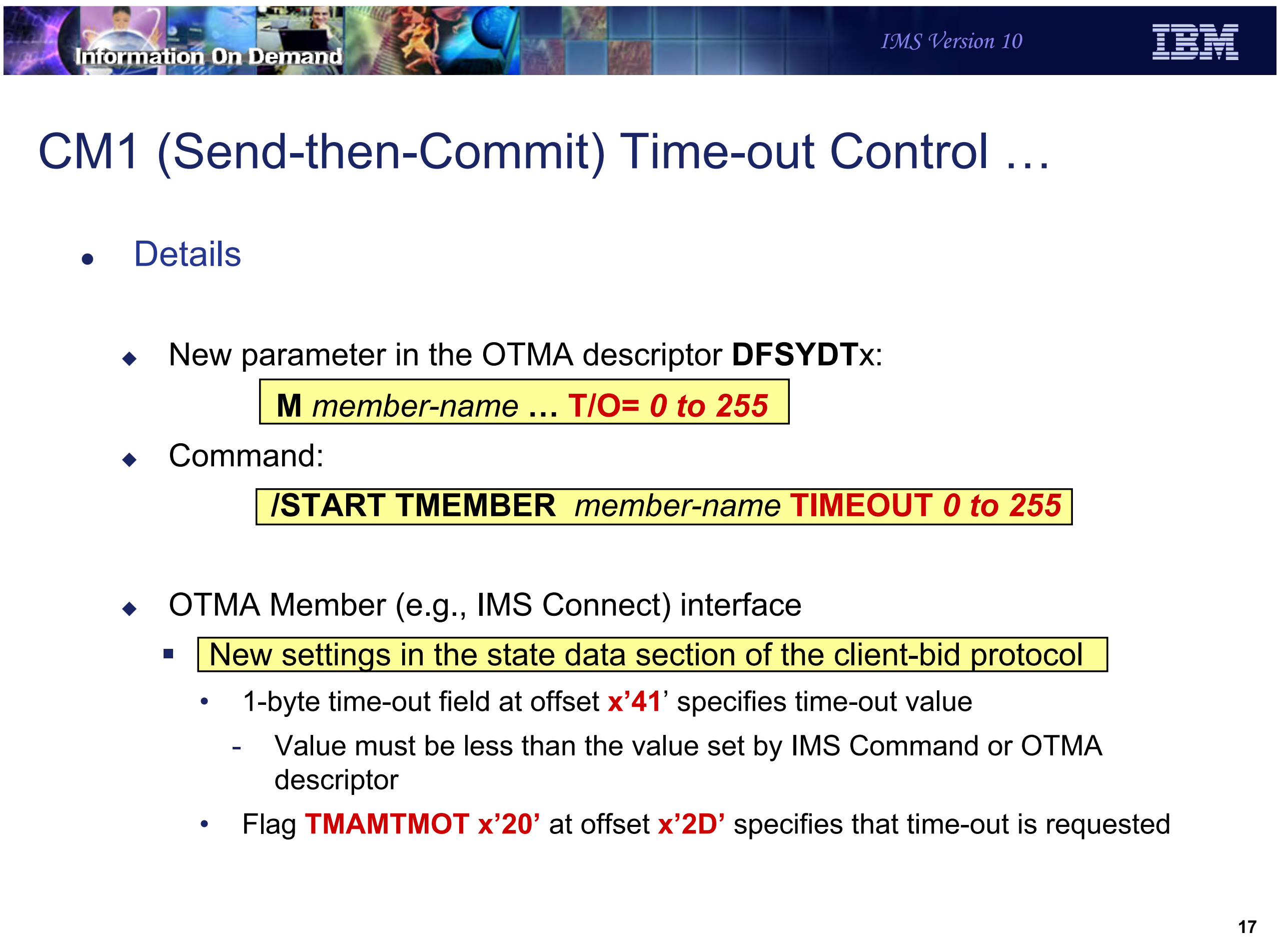
Notes:

The default 120 second timeout value for CM1 (send-then-commit) messages can be overridden in several ways:

- During IMS initialization with a new parameter in the OTMA descriptor member DFSYDTx. It is not until the member is actually started that the override value is honored.
- At any time with a new TIMEOUT specification in the /START TMEMBER command.
- By OTMA member request either when the member joins the group using the client-bid protocol message or, at a lower level of granularity, whenever a CM1 (send-then-commit message) flows into OTMA.

If none of the above methods is used to set the time-out value, the OTMA default of 120 seconds is used to determine when to perform the time-out action.

If needed, the /START TMEMBER TIMEOUT command and the OTMA descriptor can deactivate the OTMA time-out function by specifying a timeout value of 0. Once the function is deactivated, OTMA will not perform the time-out action. However, OTMA will still detect a long-waiting dependent region for a missing ACK or NAK and issue the following warning message: DFS0808W IMS REGION region-id IN BACK-END IMS aaaa HAS BEEN IN [WAIT-SYNCPOINT] or [WAIT-RRS] FOR otma-membername/tpipename FOR xx seconds.


Information On Demand
IMS Version 10
IBM

CM1 (Send-then-Commit) Time-out Control …

- Details
  - New parameter in the OTMA descriptor **DFSYDT**x:
    - **M** *member-name* **… T/O=** ***0 to 255***
  - Command:
    - **/START TMEMBER** *member-name* **TIMEOUT** ***0 to 255***
  - OTMA Member (e.g., IMS Connect) interface
    - New settings in the state data section of the client-bid protocol
      - 1-byte time-out field at offset **x'41'** specifies time-out value
        - Value must be less than the value set by IMS Command or OTMA descriptor
      - Flag **TMAMTMOT x'20'** at offset **x'2D'** specifies that time-out is requested

17

Figure 12-16. CM1 (Send-then-Commit) Time-out Control … CMA01.0

Notes:

The details of the implementation are as follows:

- A new T/O parameter in the OTMA descriptor member DFSYDTx in IMS.PROCLIB. The new parameter defines the time-out value in minutes for OTMA send-then-commit response messages. The value specified can be between 0 and 255 seconds. If the value is 0, OTMA will deactivate the time-out function. If it is over 255, it will be set to 120 which is the default.
- A new TIMEOUT specification in the /START TMEMBER command.
- New specification in the OTMA member request either when the member joins the group using the client-bid protocol message or, at a lower level of granularity, whenever a CM1 (send-then-commit message) flows into OTMA.

If a client-bid protocol message is used to specify the time-out value, the criteria of choosing the time-out value is provided through new flag specifications. Note that the client-bid cannot override the time-out specified by an OTMA descriptor or command. If the client-bid time-out value is equal to or greater than the current time-out value set by descriptor or command, OTMA will ignore the time-out value in the client-bid message. If the client-bid time-out value is less than the current time-out value set by the command or descriptor, the value from the client-bid will be used for the time-out action for this member.

CM1 (Send-then-Commit) Time-out Control …

- Details …
  - OTMA Member…
    - On an individual message level
      - 1-byte reserved field at offset **x’1E’** of the message control information prefix
        - Specifies time-out value for the input transaction
      - New flag **TMAMTTMO, x’08’** in byte 5 of the state data section
        - Specifies that OTMA can take the message level time-out value specified in the control data

Figure 12-17. CM1 (Send-then-Commit) Time-out Control … CMA01.0

Notes:

On an individual message basis, additional flags have been provided for an even lower level of time-out specification. This capability supports a message time-out value which can be different from the time-out value set for the entire member. Note, however, that this value follows similar restrictions in that it cannot override the member time-out value set by an OTMA descriptor or IMS command.

When an OTMA time-out occurs, OTMA will first back-out the transaction in order to get out of the wait-syncpoint or wait-RRS condition for any missing ACK/NACK region. Subsequently, an OTMA CM1 deallocation message will be sent to the member with the existing ABORT flag and the new “time-out” flag. The IMS system console operator will also receive a DFS0809E message “IMS REGION region-id IN BACK-END IMS aaaaaaaa HAS TIMED OUT FOR otma-membername/tpipename FOR xx MINUTES”. When OTMA takes the time-out action, byte 3 TMAMCCCI, of the OTMA commit-confirmation flag in the message control data prefix is set to TMAMCTMO, X’08’, to indicate that the transaction was aborted due to the OTMA time-out condition.

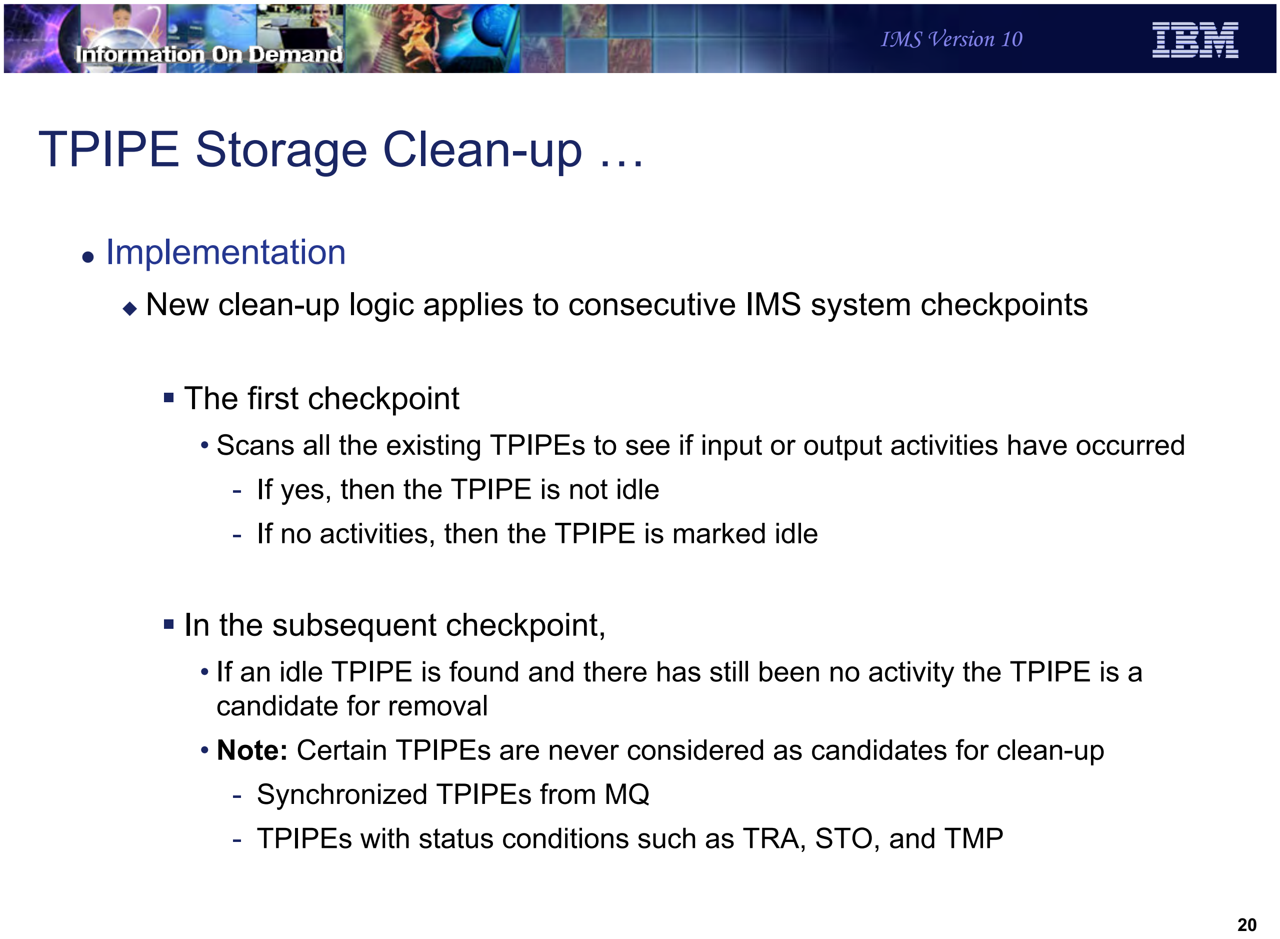
TPIPE Storage Clean-up

- Enhancement to release unused storage
  - Supports the removal of unused TPIPEs
    - Idle for two checkpoints
- This capability is available in IMS V8 and IMS V9
  - IMS V8: PQ99983, IMS V9: PK00386

Figure 12-18. TPIPE Storage Clean-up CMA01.0

Notes:

The next enhancement provides a more efficient way to control unused storage associated with idle TPIPEs. TPIPEs (Transaction Pipes) are OTMA control blocks that represent logical connections between the client and IMS. They are analogous to an IMS logical terminal (LTERM) and allow IMS to associate all input and output with a particular OTMA client. Once created, they occupy storage whether or not they are used again. This clean-up enhancement determines whether or not an inactive TPIPE can be deleted and its storage released. A TPIPE is considered inactive if it has been idle for 2 consecutive checkpoints.


Information On Demand
IMS Version 10
IBM

TPIPE Storage Clean-up …

- Implementation
  - New clean-up logic applies to consecutive IMS system checkpoints
    - The first checkpoint
      - Scans all the existing TPIPEs to see if input or output activities have occurred
        - If yes, then the TPIPE is not idle
        - If no activities, then the TPIPE is marked idle
    - In the subsequent checkpoint,
      - If an idle TPIPE is found and there has still been no activity the TPIPE is a candidate for removal
      - **Note:** Certain TPIPEs are never considered as candidates for clean-up
        - Synchronized TPIPEs from MQ
        - TPIPEs with status conditions such as TRA, STO, and TMP

Figure 12-19. TPIPE Storage Clean-up … CMA01.0

Notes:

The TPIPE storage clean-up function is activated when OTMA is activated in an IMS system. Logic added to IMS system checkpoint processing determines whether a TPIPE is active or idle based upon whether or not there are any input or output messages associated with the control block in addition to whether or not any TPIPE status conditions exist. TPIPEs that are idle for two consecutive checkpoints are deleted.

Active TPIPES include those that are processing commit-then-send (CM0) messages in a shared queues environment, have incomplete send-then-commit (CM1) messages, or have queued commit-then-send (CM0) output messages.

Certain TPIPEs are never considered for removal. These include synchronized MQ TPIPEs and TPIPEs with outstanding status indicators.

Member Level Security

- New /SECURE OTMA command capability
  - Allows each OTMA member to define its own security setting
    - FULL, CHECK, NONE, or PROFILE
  - Dynamic change of security level
    - Note - Messages are processed with the security level that was in effect when the message was received

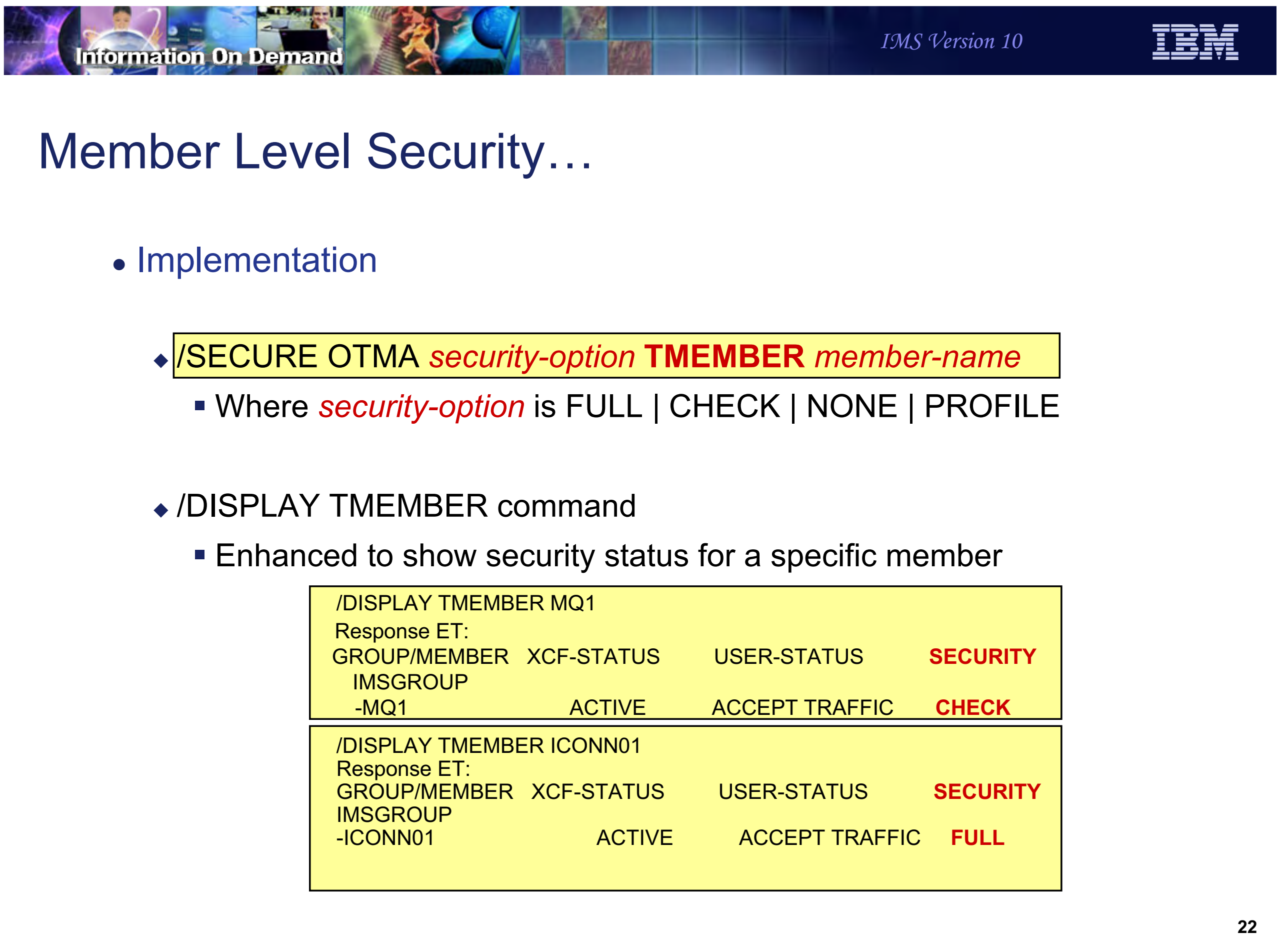
- Prior Releases
  - OTMA security was a system-wide setting for all OTMA members

Figure 12-20. Member Level Security CMA01.0

Notes:

Prior to IMS V10, OTMA does not allow different security levels defined for various members. The security setting requested is considered a system-wide setting for all of OTMA members. In V10, the OTMA command, /SECURE OTMA, has been enhanced to allow specification of member security so that each OTMA client can have its own security level.

Note - Messages are always processed with the security level that was in effect when the message was received. Even if a new security level is introduced by command, the security level associated with the message is based on the level in effect at the time of message receipt.



Information On Demand
IMS Version 10
IBM

Member Level Security…

- Implementation
  - /SECURE OTMA *security-option* **TMEMBER** *member-name*
    - Where *security-option* is FULL | CHECK | NONE | PROFILE
  - /DISPLAY TMEMBER command
    - Enhanced to show security status for a specific member

```
/DISPLAY TMEMBER MQ1
Response ET:
GROUP/MEMBER   XCF-STATUS       USER-STATUS          SECURITY
   IMSGROUP
   -MQ1                 ACTIVE        ACCEPT TRAFFIC      CHECK
```

```
/DISPLAY TMEMBER ICONN01
Response ET:
GROUP/MEMBER   XCF-STATUS       USER-STATUS          SECURITY
IMSGROUP
-ICONN01                  ACTIVE        ACCEPT TRAFFIC     FULL
```

Figure 12-21. Member Level Security… CMA01.0

Notes:

The addition of the TMEMBER keyword to the /SECURE OTMA command provides the ability to define any of the security options for a specific OTMA member.

The /DISPLAY command has also been enhanced to provide a mechanism to display the security option in effect for a specific member.

Asynchronous Message Enhancements

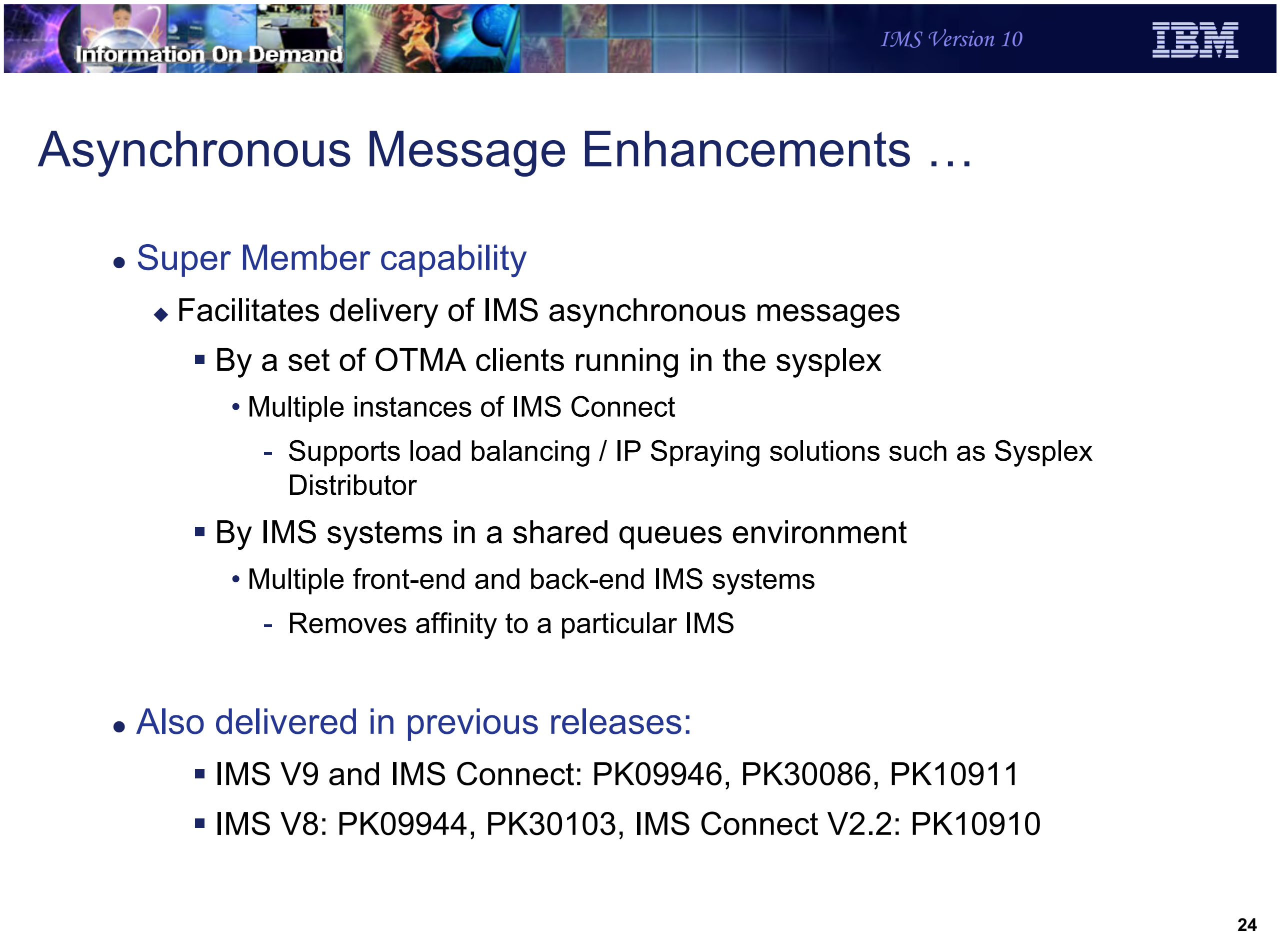
- Asynchronous messages
  - CM0 - Commit-then-Send messages
  - Undelivered IOPCB and all ALTPCB messages

- Enhancements - only apply to IMS Connect
  - Super Member capability
  - Purge
  - Reroute

Figure 12-22. Asynchronous Message Enhancements CMA01.0

Notes:

Several enhancements are available to address asynchronous CM0 messages in OTMA environments that use IMS shared queues as well as those that implement load balancing or IP spraying techniques such as Sysplex Distributor or the WebSphere Edge Server. These enhancements extend the capabilities of IMS Connect.


Information On Demand
IMS Version 10
IBM

Asynchronous Message Enhancements …

- Super Member capability
  - Facilitates delivery of IMS asynchronous messages
    - By a set of OTMA clients running in the sysplex
      - Multiple instances of IMS Connect
        - Supports load balancing / IP Spraying solutions such as Sysplex Distributor
    - By IMS systems in a shared queues environment
      - Multiple front-end and back-end IMS systems
        - Removes affinity to a particular IMS
- Also delivered in previous releases:
    - IMS V9 and IMS Connect: PK09946, PK30086, PK10911
    - IMS V8: PK09944, PK30103, IMS Connect V2.2: PK10910

Figure 12-23. Asynchronous Message Enhancements … CMA01.0

Notes:

The Super Member capability in OTMA facilitates the delivery of asynchronous (CM0 Commit-then-send) messages by any instance within a set of IMS Connects and IMS subsystems. With this function, affinity to a particular IMS or IMS Connect is removed and the use of shared queues as well as solutions such as Sysplex Distributor become more viable.

Asynchronous Message Enhancements …

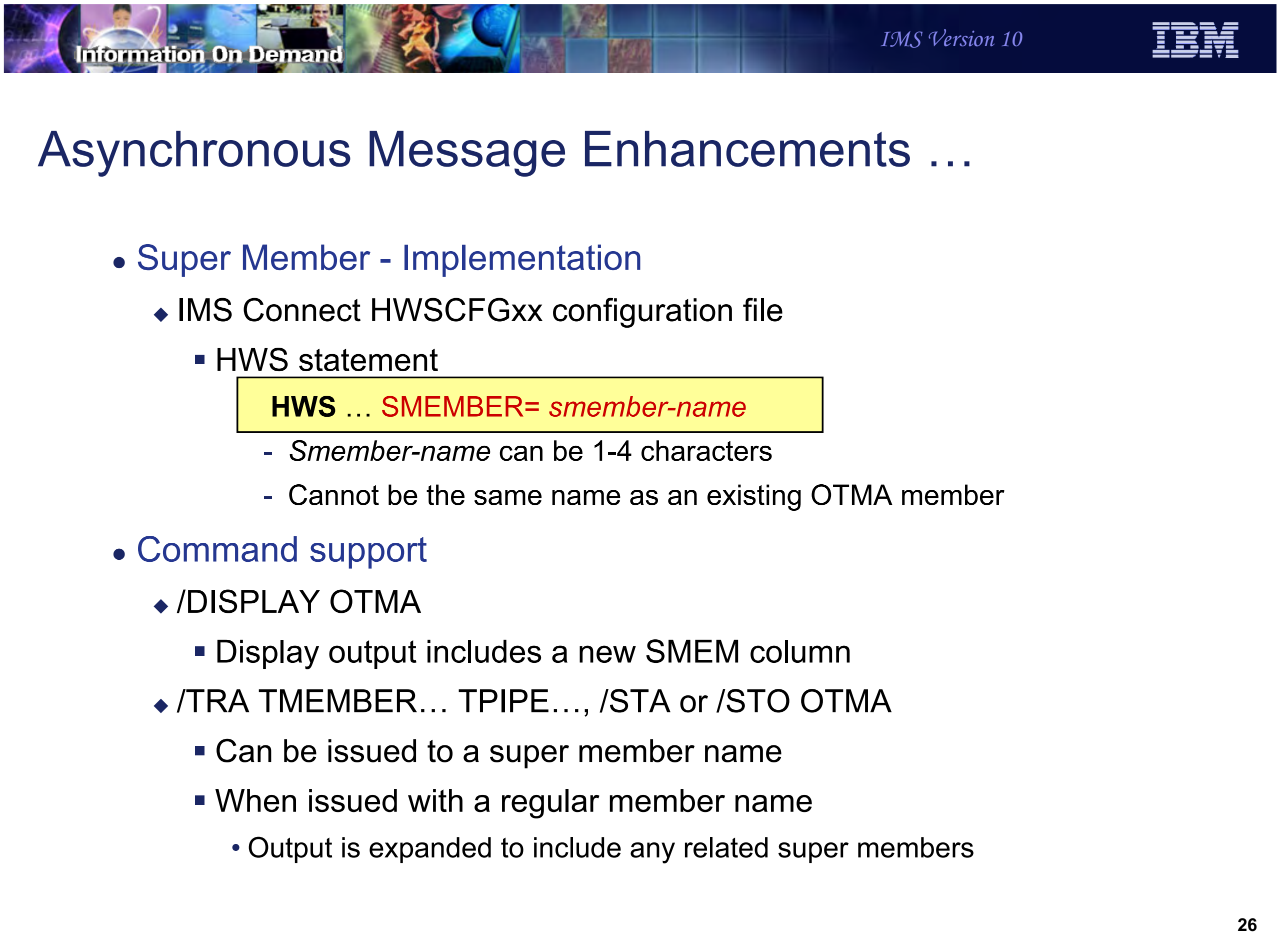
- Super Member capability …
  - A group name given to a set of OTMA members
    - E.g., IMS Connect instances
      - Allows any IMS Connect instance to retrieve the message
  - Generic structure name in IMS on which the messages are queued
    - OTMA members that share a super member name
      - Recognized by IMS by both their specific name and super member name
    - For Shared Queues, no affinity to an IMS system
      - Any IMS can deliver the message

Figure 12-24. Asynchronous Message Enhancements … CMA01.0

Notes:

A super member is a special OTMA member name which can be shared by a set of IMS Connects to handle the CM0 hold queue messages.

When an IMS Connect attaches to IMS, OTMA creates a regular member structure name unique to that instance to track the connection status and to record the connection options for later transaction processing. If a super member name, which could be considered a group name for a set of IMS Connects using Sysplex Distributor or similar mechanism, is given by an IMS Connect during the connection time, OTMA will additionally create a member structure called the super member structure. If the super member structure already exists, OTMA will use it instead of creating a new one. A regular member structure is dedicated exclusively to the IMS Connect for which it was created. However, a super member structure is shared among a set of IMS Connects so that a Resume TPIPE can be issued from any IMS Connect. The role of the super member is to store and deliver the CM0 hold queue messages.


Information On Demand
IMS Version 10
IBM

Asynchronous Message Enhancements …

- Super Member - Implementation
  - IMS Connect HWSCFGxx configuration file
    - HWS statement

```
HWS … SMEMBER= smember-name
```

      - *Smember-name* can be 1-4 characters
      - Cannot be the same name as an existing OTMA member
- Command support
  - /DISPLAY OTMA
    - Display output includes a new SMEM column
  - /TRA TMEMBER… TPIPE…, /STA or /STO OTMA
    - Can be issued to a super member name
    - When issued with a regular member name
      - Output is expanded to include any related super members

Figure 12-25. Asynchronous Message Enhancements … CMA01.0

Notes:

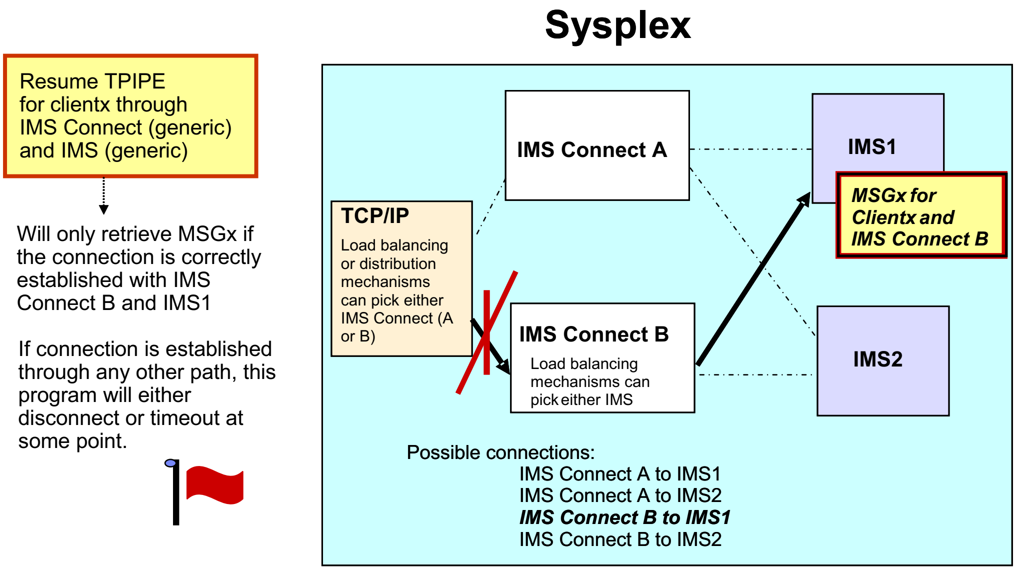
IMS Connect has been enhanced to support the Super Member capability. The HWSCFGxx configuration file member provides a new SMEMBER= parameter in the HWS statement. The value provided must be different than the value provided in the MEMBER= parameter of the DATASTORE statement and must follow the OTMA naming conventions.

When the client-bid protocol message is sent from an IMS Connect member that has specified a Super Member value, the state data of the OTMA prefix carries the defined SMEMBER value in a new 4-byte field at the offset of x’36’ in the state data section. In addition to the super member name in the client-bid message, a new flag TMAMFGSM (x’08’) at the offset x’2D’ of the state data informs OTMA that the super member processing is requested.

Additionally, the /DIS, /TRA, /STA and /STO commands have been enhanced to support the Super Member capability. The /DISPLAY OTMA command output includes a new SMEM column to display the Super Member name if one exists. The /TRA TMEMBER … TPIPE, /STA OTMA and /STO OTMA commands can be issued against either a Super Member name or regular IMS Connect.

Asynchronous Message Enhancements …

- Super member capability - the issue



Sysplex

Resume TPIPE for clientx through IMS Connect (generic) and IMS (generic)

Will only retrieve MSGx if the connection is correctly established with IMS Connect B and IMS1

If connection is established through any other path, this program will either disconnect or timeout at some point.

IMS Connect A
IMS1
MSGx for Clientx and IMS Connect B
TCP/IP
Load balancing or distribution mechanisms can pick either IMS Connect (A or B)
IMS Connect B
Load balancing mechanisms can pick either IMS
IMS2

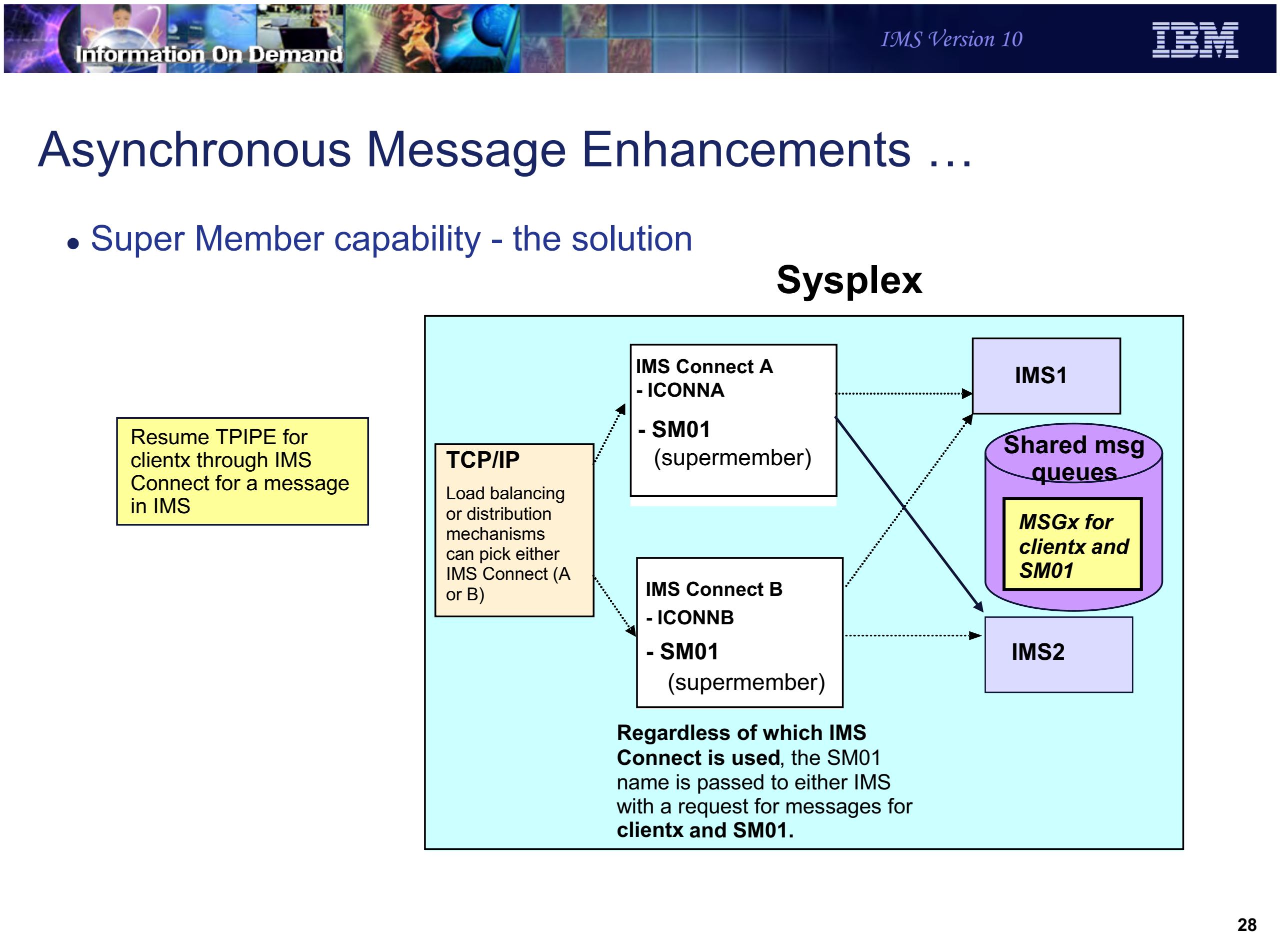
Possible connections:
IMS Connect A to IMS1
IMS Connect A to IMS2
IMS Connect B to IMS1
IMS Connect B to IMS2

Figure 12-26. Asynchronous Message Enhancements … CMA01.0

Notes:

This and the next visual attempt to illustrate the value of the Super Member support.

Asynchronous CM0 output messages in IMS are queued to a message queue construct that is identified by TMEMBER and TPIPE and, therefore, associated with a specific IMS Connect instance. As a result, there is a potential issue when using any of the load balancing or sysplex distribution mechanisms or even when a message is processed in a back-end IMS in a shared queues group. To retrieve the message, the remote program needs to establish a connection through the appropriate IMS Connect to the actual IMS system that queued the message. This can be a challenge because there is no easy mechanism for a remote program to discover the required connection path, i.e., a specific IMS Connect to a specific IMS. Additionally, the remote programs may not want to know specific connection paths to IMS because that would negate the value of using load balancing and distribution mechanisms.



Asynchronous Message Enhancements …

- Super Member capability - the solution

Sysplex

Resume TPIPE for clientx through IMS Connect for a message in IMS

TCP/IP

Load balancing or distribution mechanisms can pick either IMS Connect (A or B)

IMS Connect A
- ICONNA
- SM01
(supermember)

IMS Connect B
- ICONNB
- SM01
(supermember)

IMS1

Shared msg queues

MSGx for clientx and SM01

IMS2

Regardless of which IMS Connect is used, the SM01 name is passed to either IMS with a request for messages for **clientx and SM01.**

Figure 12-27. Asynchronous Message Enhancements … CMA01.0

Notes:

The OTMA Super Member function resolves the issue by allowing a RESUME TPIPE request to retrieve CM0 output across all combinations of IMS Connect and IMS systems. If multiple IMS systems are involved, then those IMS systems must also have IMS shared queues implemented. If there is only one IMS system but multiple IMS Connects, then shared queues support is not required. As shown in the illustration on this visual, the Resume TPIPE request for clientx can be routed through any load balancing or distribution mechanism to either IMS Connect A or IMS Connect B. Both systems are identified to IMS1 and IMS2 by their unique XCF member names as well as the global Super Member name of SM01. The request to retrieve the output message for TPIPE clientx can be sent to either IMS1 or IMS2 because both have access to the shared queues and, more specifically, to all the messages under the shared queues construct for SM01 and TPIPE clientx.

Information On Demand
IMS Version 10
IBM

Asynchronous Message Enhancements …

- REROUTE and PURGE Support
  - For OTMA hold queue capable clients ONLY
    - OTMA members, e.g. IMS Connect
      - That keep asynchronous messages on the IMS message queue until requested by the remote application
  - Extended support that allows remote applications to
    - Reject (NAK) or request action on asynchronous messages
      - **Purge** from the IMS message queue
      - **Reroute** to an alternate destination
      - When both REROUTE and PURGE are specified for a message
        - Neither action is performed
        - OTMA issues DFS2407W message
        - Stores the message onto the hold queue of the inputting TPIPE
- Also delivered in previous releases

Figure 12-28. Asynchronous Message Enhancements … CMA01.0

Notes:

OTMA asynchronous message enhancements has been extended to provide greater flexibility for messages that cannot be delivered or are rejected by the remote client. The two actions, which are mutually exclusive of each other, are to reroute messages that cannot be delivered to an alternate destination or to purge them.

The REROUTE enhancement provides a mechanism for an IMS Connect client to request that undeliverable Commit Mode Zero (CM0) output associated with a send/receive from the client application be rerouted to an alternate IMS Connect destination. When a Client Reroute Request is made, IMS Connect notifies OTMA to remove the message from the current queue and requeue it to the provided Reroute name queue.

The Purge Not Deliverable extension allows the remote client application or the appropriate User Message Exit to specify whether or not the output should be purged if it is not deliverable. Note that if both capabilities are requested, neither action is performed and DFS2407W message is issued.

This capability was delivered in previous releases as follows: IMS V9 and IMS Connect: PK16934, PK22480, PK24907, PK09543, PK12013; and for

IMS V8: PK21868, PK09542, IMS Connect V2.2: PK12012

Information On Demand
IMS Version 10
IBM

Asynchronous Message Enhancements …

- REROUTE and PURGE - Implementation …
  - REROUTE
    - OTMA

OMUSR_FLAG1	new setting of OMUSR_REROUT (X’01’)
OMUSR_ARCLEV	new setting of OMUSR_AL02 (X’02’)
Field OMUSR_REROUT_NM	holds the reroute name

    - IMS Connect

IRM_ARCH	new setting of IRM_ARCH1 (X’01’)
IRM_F3	new setting of IRM_F3_REROUT (X’08’)
New field IRM_REROUT_NM	holds the reroute name

  - PURGE
    - OTMA

OMHDRPND EQU X'10'	purge if not deliverable

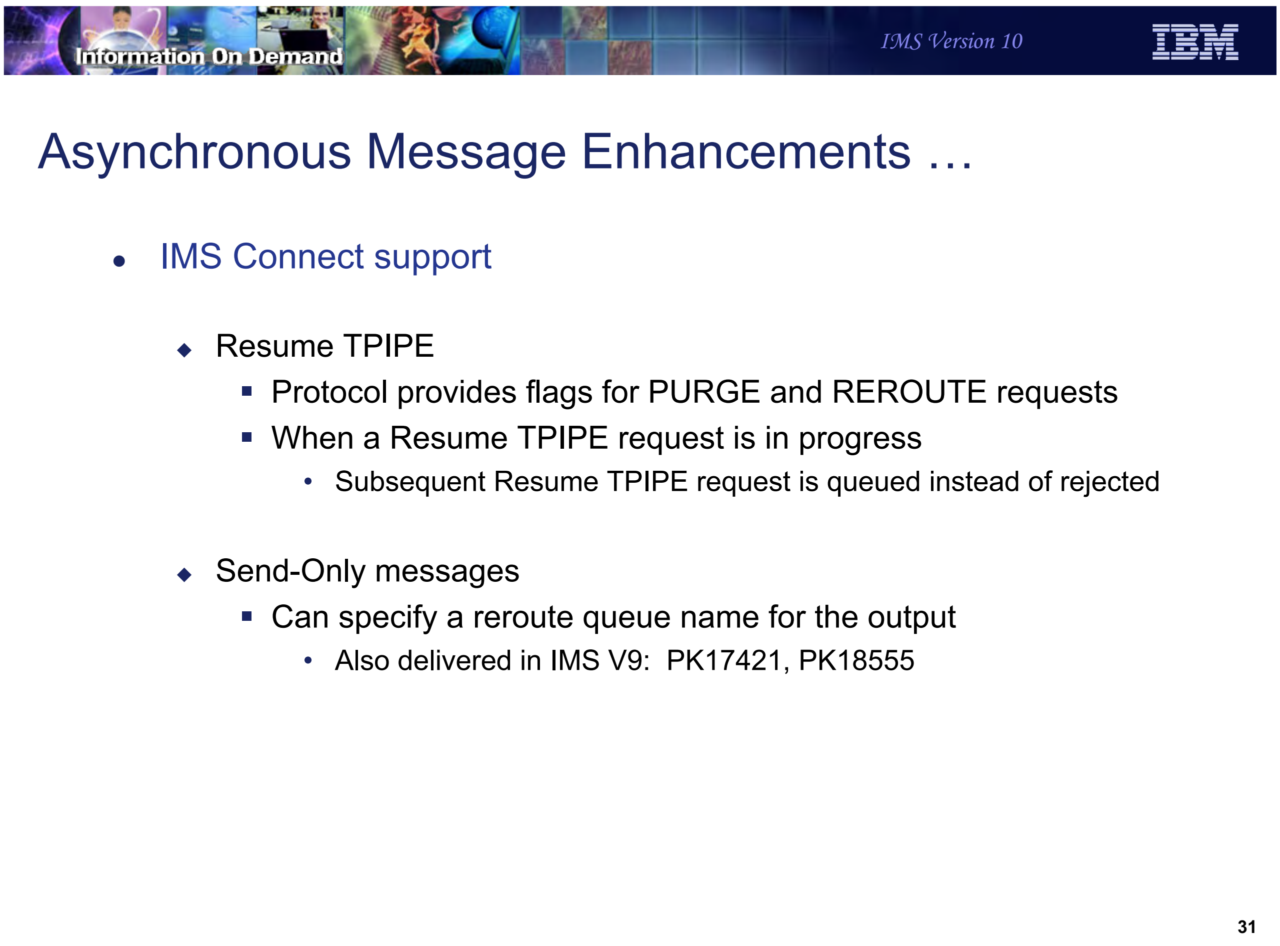
    - IMS Connect

      IRM_F3 has a setting of IRM_F3_PURGE (X’04’)

Figure 12-29. Asynchronous Message Enhancements … CMA01.0

Notes:

Both the OTMA headers and the IMS Connect headers have been enhanced to request either the PURGE or REROUTE capability.



Asynchronous Message Enhancements …

- IMS Connect support
    - Resume TPIPE
        - Protocol provides flags for PURGE and REROUTE requests
        - When a Resume TPIPE request is in progress
            - Subsequent Resume TPIPE request is queued instead of rejected
    - Send-Only messages
        - Can specify a reroute queue name for the output
            - Also delivered in IMS V9: PK17421, PK18555

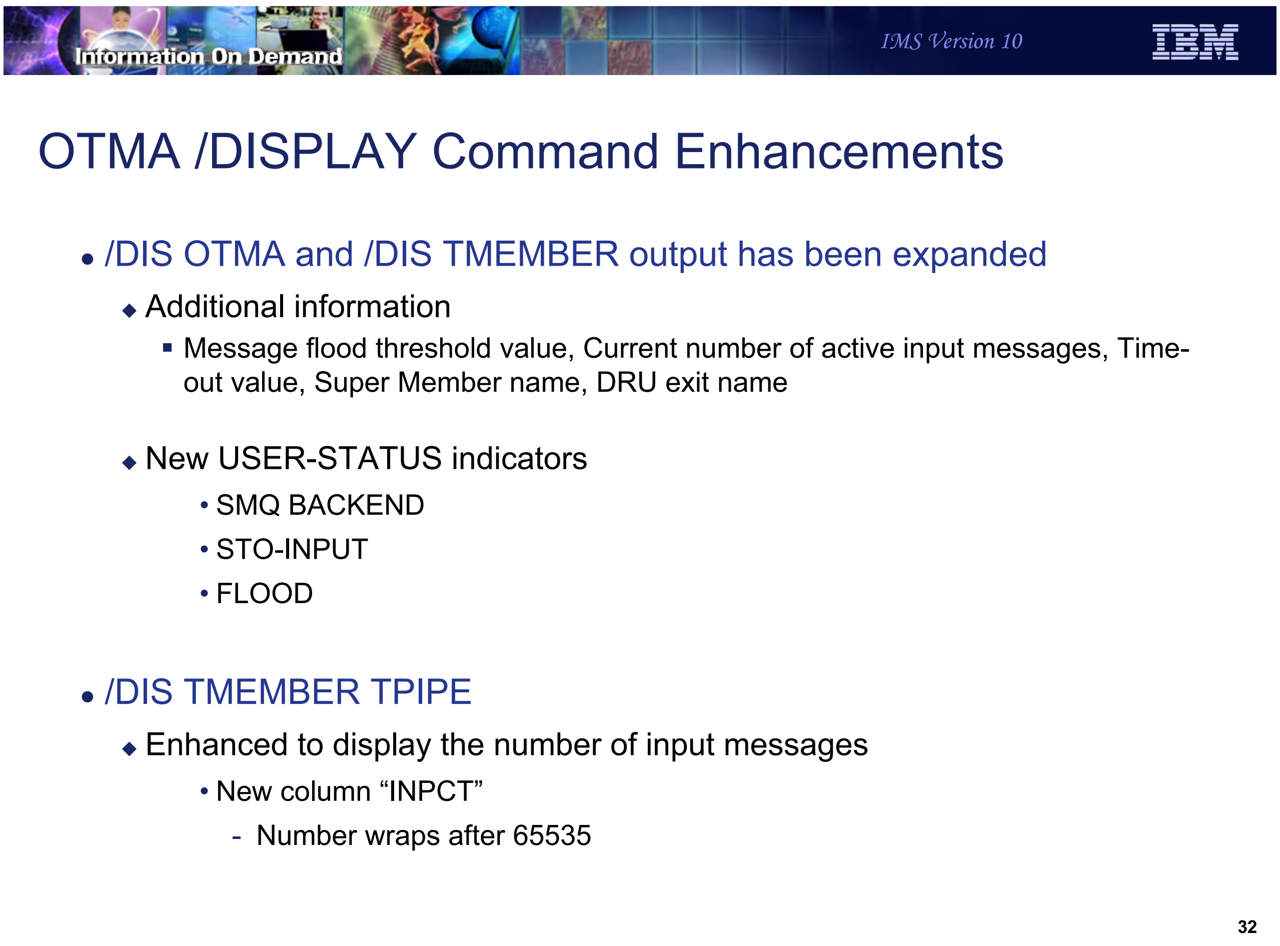
Figure 12-30. Asynchronous Message Enhancements … CMA01.0

Notes:

IMS Connect has been enhanced to take advantage of the new capabilities. The Resume TPIPE protocol provides flags to indicate one or the other type of request.

Another enhancement in this area allows RESUME TPIPE requests to be queued when requested for the same TPIPE name.

Additionally, the Send-Only protocol allows specification of a reroute queue name for the output in the case of an initial rejection of the output reply. This capability allows IOPCB output from the Send-Only input transaction to be rerouted to a dedicated TPIPE instead of the inputting TPIPE. The user of the IMS Connect Send-Only transaction will need to turn on the reroute flag and specify a reroute TPIPE name in the input stream to activate the capability. Note that this is not supported by the local option capability, HWSIMSO0, HWSIMS01 and HWSJAVA0.


Information On Demand
IMS Version 10
IBM

OTMA /DISPLAY Command Enhancements

- /DIS OTMA and /DIS TMEMBER output has been expanded
  - Additional information
    - Message flood threshold value, Current number of active input messages, Time-out value, Super Member name, DRU exit name
  - New USER-STATUS indicators
    - SMQ BACKEND
    - STO-INPUT
    - FLOOD
- /DIS TMEMBER TPIPE
  - Enhanced to display the number of input messages
    - New column “INPCT”
      - Number wraps after 65535

Figure 12-31. OTMA /DISPLAY Command Enhancements CMA01.0

Notes:

The /DISPLAY OTMA and /DISPLAY TMEMBER command outputs have been enhanced to provide more information about the OTMA environment and specific TMEMBERs. To contain all the information, the single line display output has been increased to two lines.

Several new USER-STATUS indicators provide the following information:

- SMQ BACKEND - This status indicator on a TMEMBER line shows up on a back-end IMS in a shared queues group. It shows that OTMA has duplicated the specific TMEMBER environment and control blocks needed on the back-end to process a message that was received from that TMEMBER which is attached on the front-end. The same member name on the front-end IMS is shown in ‘connected’ state and no SMQ BACKEND indicator. Whereas the corresponding control blocks on the back-end IMS which processes the message is shown in ‘disconnected’ state with an SMQ BACKEND indicator.
- STO-INPUT - This status shows that the /STOP TMEMBER command has been issued for a specific member-name and no new input can be accepted.

- FLOOD - This status indicator shows that a specific TMEMBER is in a message flood condition and that the maximum input message count that was specified has been reached.

Additionally, the /DIS TMEMBER TPIPE command has been enhanced to show the number of input messages currently on the queue.

OTMA Processing during Restart

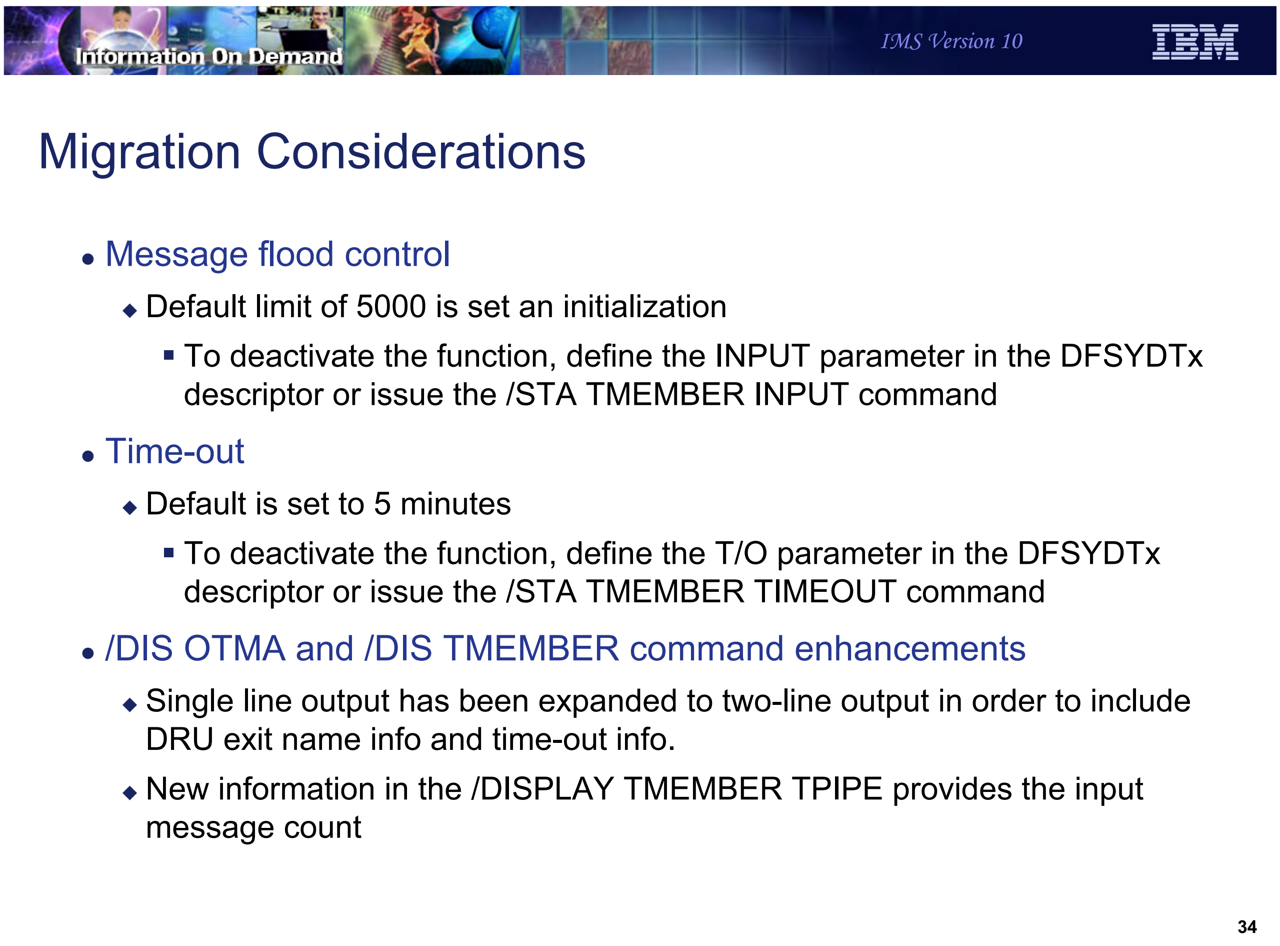
- New OTMA=M option in DFSPBxx
  - Option to control OTMA functionality during all restarts including ERE
    - IMS does not enable OTMA during the system initialization
    - /START OTMA commands are not recovered during restarts
- /START OTMA NOCHECK
  - Command to start OTMA as a non-recoverable request during restart
    - Capability is introduced for OTMA=N users
- Also delivered in IMS V8 and IMS V9
  - IMS V8: PK14679, IMSV9: PK14680

Figure 12-32. OTMA Processing during Restart CMA01.0

Notes:

In addition to the existing OTMA values of Y and N, IMS V10 introduces the option of OTMA=M (manual). When IMS first initializes, the value of OTMA=M functions similarly to OTMA=N such that OTMA is not started. Once IMS is up and running, the /STA OTMA command can be issued but becomes non-recoverable. The setting of OTMA=M, therefore, takes effect when IMS terminates either normally or abnormally and has to be restarted. During restart processing, OTMA is not automatically restarted. This capability was introduced to prevent looping abend situations where IMS may have terminated as a result of OTMA error conditions.

An additional impact on OTMA restart processing is introduced for environments where IMS is initialized with OTMA=N. If a /START OTMA NOCHECK command is issued, the command is also not recovered during either a warm start or emergency restart.


Information On Demand
IMS Version 10
IBM

Migration Considerations

- Message flood control
  - Default limit of 5000 is set an initialization
    - To deactivate the function, define the INPUT parameter in the DFSYDTx descriptor or issue the /STA TMEMBER INPUT command
- Time-out
  - Default is set to 5 minutes
    - To deactivate the function, define the T/O parameter in the DFSYDTx descriptor or issue the /STA TMEMBER TIMEOUT command
- /DIS OTMA and /DIS TMEMBER command enhancements
  - Single line output has been expanded to two-line output in order to include DRU exit name info and time-out info.
  - New information in the /DISPLAY TMEMBER TPIPE provides the input message count

Figure 12-33. Migration Considerations CMA01.0

Notes:

The considerations listed on this visual address issues that should be considered when migrating from a previous release of IMS. The assumption is that migration to IMS V10 is based on existing functionality without adding any new capabilities during the migration process.

The message flood control enhancement in IMS V10 is automatically enabled with a default limit of 5000 messages. To provide compatibility with previous releases and deactivate the support, either specify an INPUT value of 0 in a descriptor or issue the /STA TMEMBER INPUT command.

Likewise, the timeout support for synchronous CM1 message is automatically enabled with a default value of 5 minutes. To provide compatibility with previous releases and deactivate the support, specify T/O value of 0 in a descriptor or issuing the /STA TMEMBER TIMEOUT command.

Note that the /DISPLAY command output associated with OTMA and TMEMBER requests has been expanded to two output lines and includes new information. As a migration consideration, this is a key issue for automated operations.

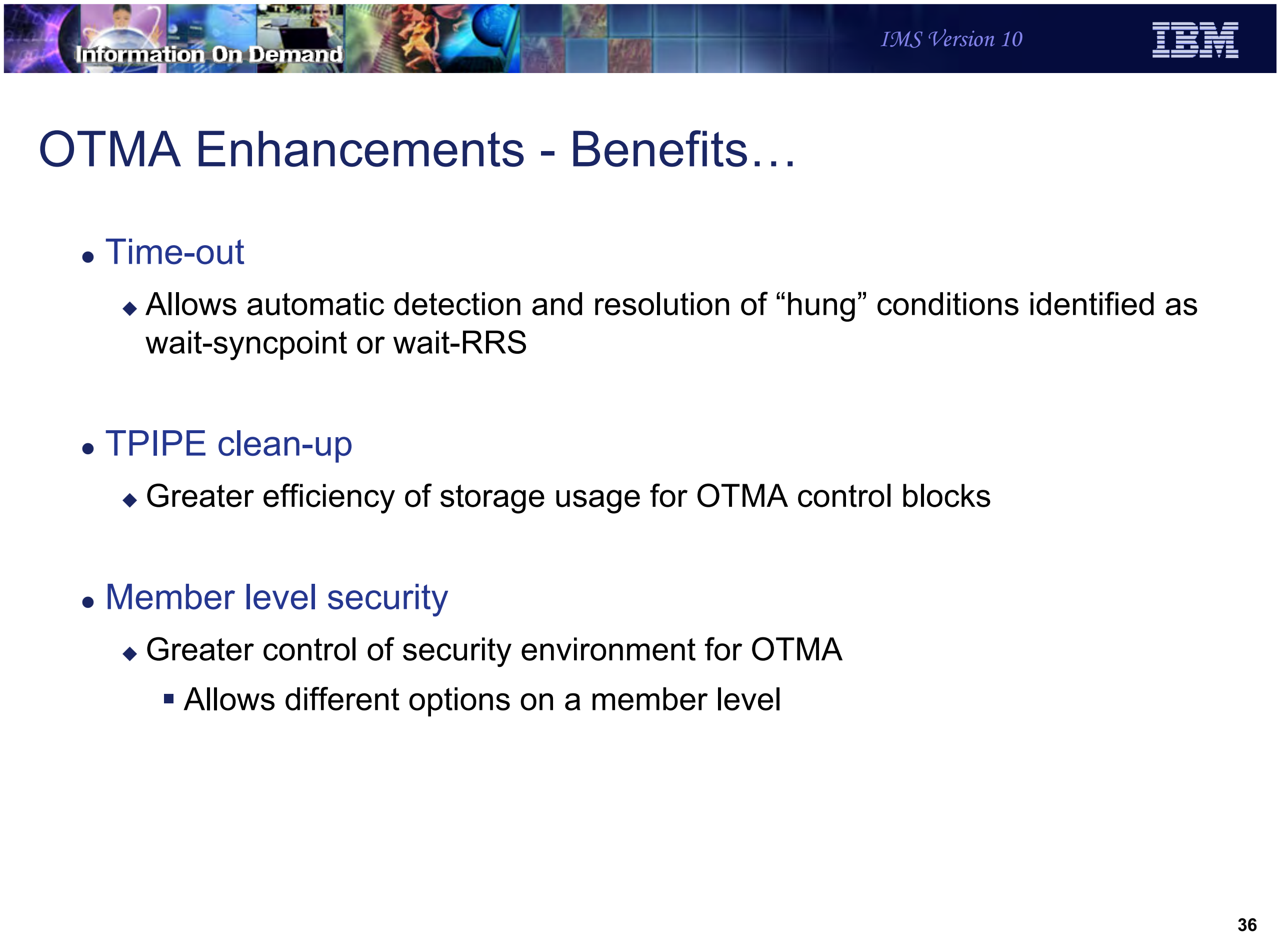
OTMA Enhancements - Benefits

- Destination Routing support
  - Facilitates outbound messages through an architected OTMA layer
    - Minimizes the need for coding OTMA exit routines through descriptors
    - Enables growth of e-business environments using the IMS Connector for Java and the IMS SOAP Gateway
- Resume TPIPE security
  - Ensures that only authorized users can retrieve output messages in the hold queue
    - Supports IMS Connect clients using Resume TPIPE commands
- Message Flood protection
  - Provides a mechanism to prevent OTMA clients from flooding the IMS message queues and causing S40D abends

Figure 12-34. OTMA Enhancements - Benefits CMA01.0

Notes:

The next three visuals summarize the enhancements in this section.



Information On Demand

IMS Version 10

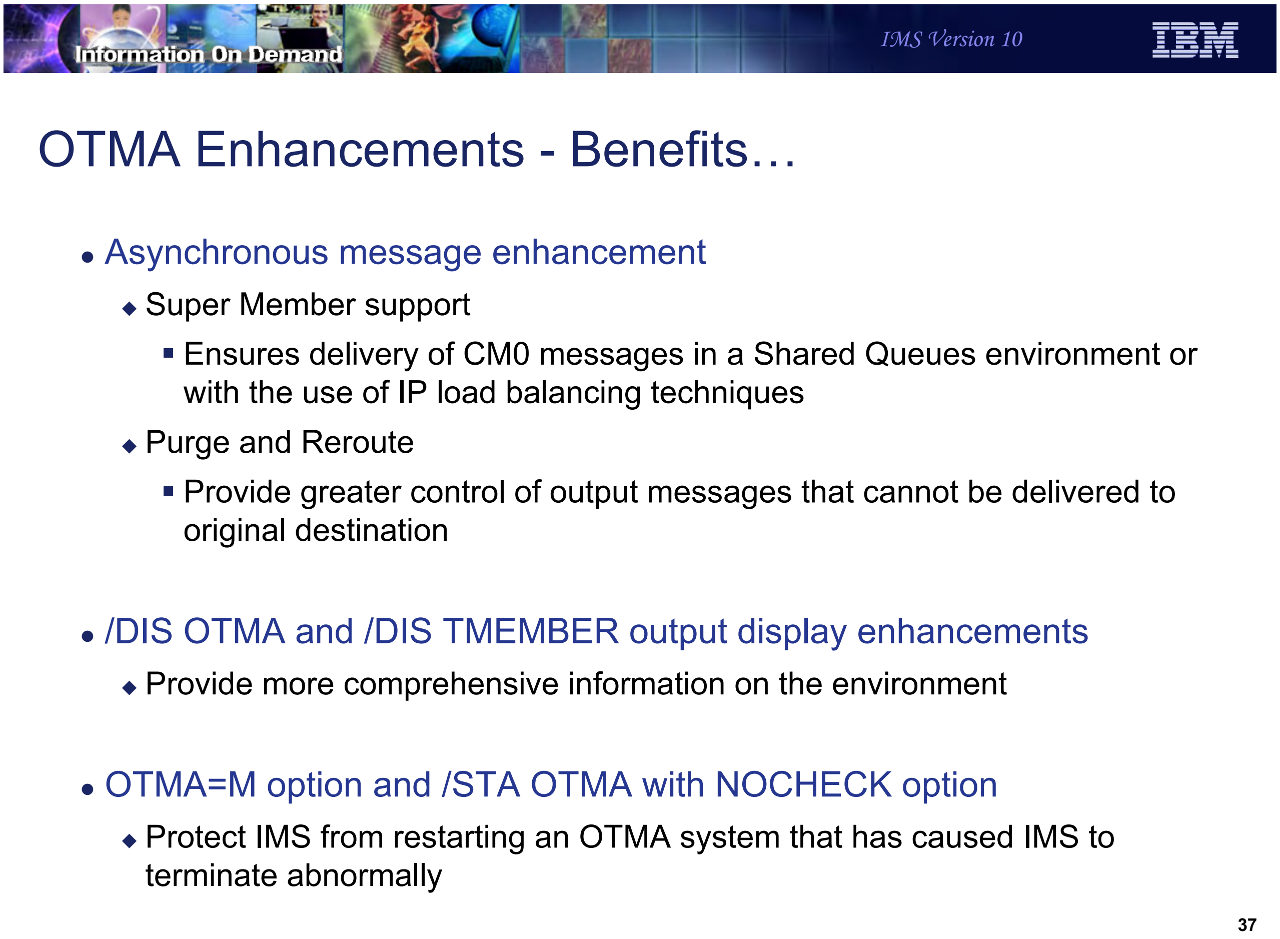
OTMA Enhancements - Benefits…

- Time-out
  - Allows automatic detection and resolution of “hung” conditions identified as wait-syncpoint or wait-RRS
- TPIPE clean-up
  - Greater efficiency of storage usage for OTMA control blocks
- Member level security
  - Greater control of security environment for OTMA
    - Allows different options on a member level

36

Figure 12-35. OTMA Enhancements - Benefits… CMA01.0

Notes:



OTMA Enhancements - Benefits…

- Asynchronous message enhancement
    - Super Member support
        - Ensures delivery of CM0 messages in a Shared Queues environment or with the use of IP load balancing techniques
    - Purge and Reroute
        - Provide greater control of output messages that cannot be delivered to original destination
- /DIS OTMA and /DIS TMEMBER output display enhancements
    - Provide more comprehensive information on the environment
- OTMA=M option and /STA OTMA with NOCHECK option
    - Protect IMS from restarting an OTMA system that has caused IMS to terminate abnormally

Figure 12-36. OTMA Enhancements - Benefits… CMA01.0

Notes:

Information On Demand
IMS Version 10

IMS Connect Enhancements

Figure 12-37. [No title] CMA01.0

Notes:

Highlights

- ACEE aging value support
- Client password change request
- RACF mixed case password
- CM1 timeout
- Message flood control
- Asynchronous message enhancements
  - Super member, Reroute and Purge Not Deliverable
  - Port affinity
  - Alternate clientid
- XML Adapter support
- IMS SOA Composite Business Application Support

Figure 12-38. Highlights CMA01.0

Notes:

IMS Connect provides several usability, availability and security enhancements.

ACEE Aging Value Support

- New OAAV parameter in the DATASTORE statement of the HWSCFGxx file
  - OTMA ACEE aging value in seconds
    - 0 to <u>2147483647</u> (x' 7FFFFFFF' in OTMA)
      - Values between 0 and 300 are reset to 300
    - Example:

```
DATASTORE ID=…. OAAV=360
```

- OTMA aging value in effect can be displayed
  - VIEWHWS/VIEWDS command output
  - MVS Modify Command QUERY MEMBER and QUERY DATASTORE output
    - Example:

```
 RACF APPL NAME=
OTMA ACEE AGING VALUE=360
```

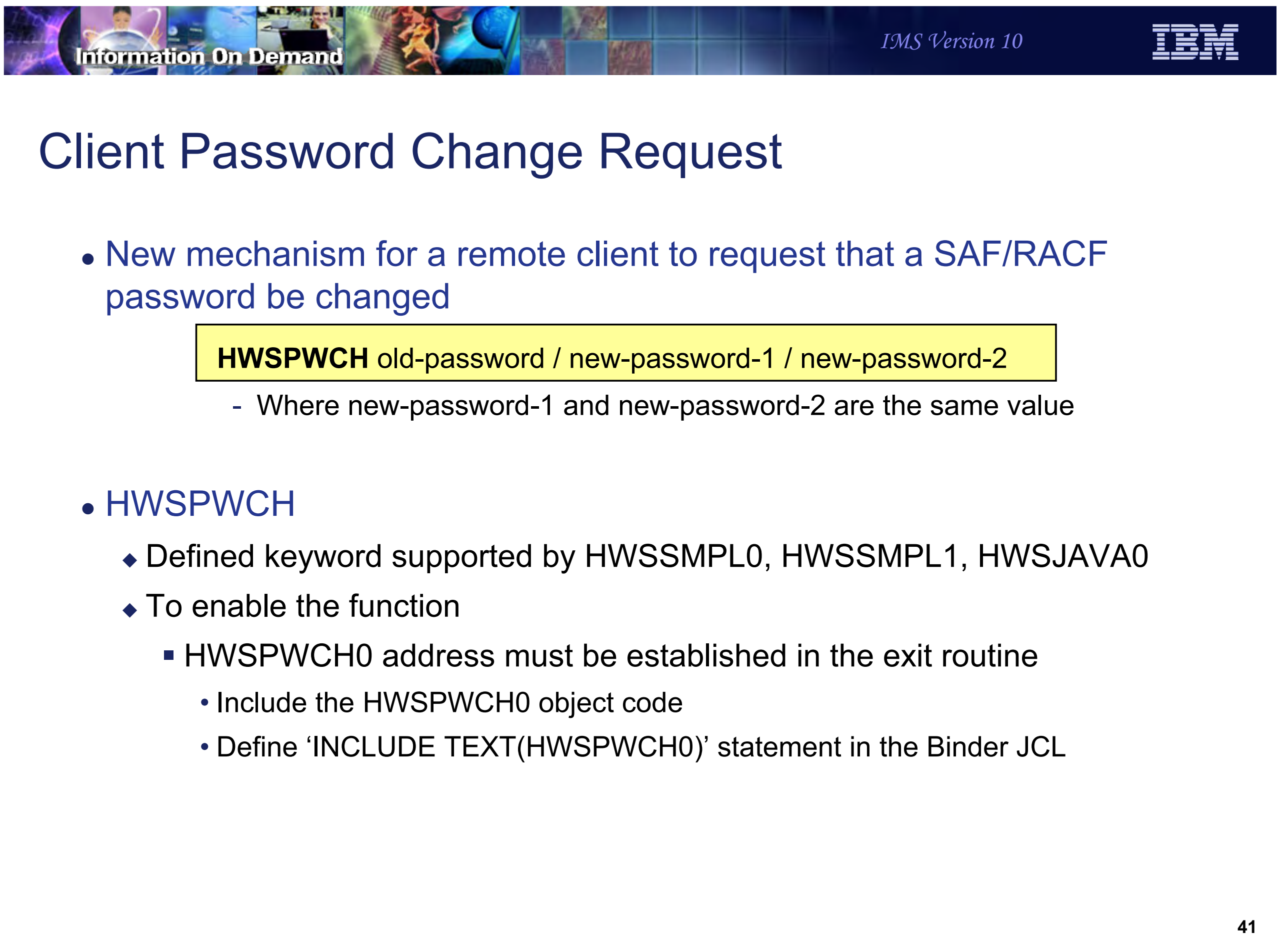
Figure 12-39. ACEE Aging Value Support CMA01.0

Notes:

The Access Control Environment Element (ACEE) is a control block that represents a verified userid to IMS. The ACEE is used to determine the user's authorization to the IMS command or IMS transaction requested in the input message. Once built in OTMA, the ACEE for each userid is cached and the aging value associated with each OTMA client, e.g., IMS Connect, is kept in a table. The aging value is then used to determine when the cached control block should expire and be refreshed. IMS re-creates the ACEE if a message associated with the userid is received but the age of the current ACEE is greater than the aging value. The aging value is used to balance performance (possible RACF I/O to refresh the ACEE) and integrity. For IMS Connect, the ACEE expiration value is specified during the client-bid process and is set to a default of no expiration.

IMS V10 provides a new parameter, OAAV, in the DATASTORE statement of the HWSCFGxx file for specification of an OTMA ACEE aging value. If not specified, the default continues to be 2147483647 which, in essence, means no expiration and is the maximum value supported by OTMA.

The VIEWHWS,VIEWDS, QUERY MEMBER and QUERY DATASTORE command output displays have all been enhanced to show the aging value that is in effect for the associated environment.


Information On Demand
IMS Version 10
IBM

Client Password Change Request

- New mechanism for a remote client to request that a SAF/RACF password be changed

HWSPWCH old-password / new-password-1 / new-password-2

  - Where new-password-1 and new-password-2 are the same value

- HWSPWCH
  - Defined keyword supported by HWSSMPL0, HWSSMPL1, HWSJAVA0
  - To enable the function
    - HWSPWCH0 address must be established in the exit routine
      - Include the HWSPWCH0 object code
      - Define 'INCLUDE TEXT(HWSPWCH0)' statement in the Binder JCL

Figure 12-40. Client Password Change Request CMA01.0

Notes:

IMS Connect provides a new mechanism that allows a remote client to request that the SAF/RACF password associated with a userid be changed.

As provided, the new capability will be supported in the HWSSMPL0, HWSSMPL1 and HWSJAVA0 exit routines. The routines check for a leading keyword of 'HWSPWCH' to determine whether it is a request to change the password. This 'HWSPWCH' string can be viewed as a transaction code but new logic in the routine, HWSPWCH0, is called to process the special request. HWSSMPL0, HWSSMPL1 and HWSJAVA0 can be modified to define the HWSPWCH constant with a unique keyword value other than 'HWSPWCH'.

The exit routines pass HWSPWCH0 the request keyword length in IMSEA_PWCHKEYL field. This allows HWSPWCH0 to process the password change request independently from the request keyword. After regaining control, the exit routines check the return code in register 15. A zero return code means the request is successful. If it is a non-zero return code, the exit routine checks the IMSEA_ERCD field for a valid error code, and replies back to the client with an error message specified in IMSEA_MSGTEXT and IMSEA_MSGLEN fields.

In order to establish the HWSPWCH0 address, the HWSPWCH0 object code must be included in the exit routine and an ‘INCLUDE   TEXT(HWSPWCH0)’ statement added to the exit routine JCL for the binder (link-edit) step. During execution, if a request for password change is received and the HWSPWCH0 address does not exist, the exit routine will send a message back to the client stating that the password change function is not supported.

Information On Demand
IMS Version 10

Client Password Change Request …

- Remote client support
  - IMS TM Resource Adapter (formerly IC4J)

LLLL | IRM | OTMA | LLZZ **HWSPWCH** *old-password / new-password1 / new-password2 /* EOM

  - Other clients

LLLL | IRM | LLZZ **HWSPWCH** *old-password / new-password1 / new-password2 /* EOM

Figure 12-41. Client Password Change Request … CMA01.0

Notes:

The password change support in HWSJAVA0 can be invoked by the IMS TM Resource adapter (formerly called IMS Connector for JAVA or IC4J) client. After the OTMA headers, the message sent begins with the defined keyword HWSPWCH followed by the old and new passwords.

Similarly, other clients that invoke exit routines based on HWSSMPL0 or HWSSMPL1 can supply the HWSPWCH request after the IRM header.

Information On Demand
IMS Version 10
IBM

RACF Mixed Case Password Support

- Enhancement to enable RACF mixed case passwords
  - PSWDMC parameter in HWS statement in IMS Connect HWSCFGxx

```
HWS …PSWDMC = Y | N
```

  - IMS Connect command

```
SETPWMC ON|OFF
```

  - IMS Connect UPDATE command

```
F imsconnproc, UPDATE MEMBER TYPE(IMSCON)SET(PSWDMC(ON|OFF)
```

- Requires that RACF support is enabled

```
RACF SETROPTS(MIXEDCASE)
```

Figure 12-42. RACF Mixed Case Password Support CMA01.0

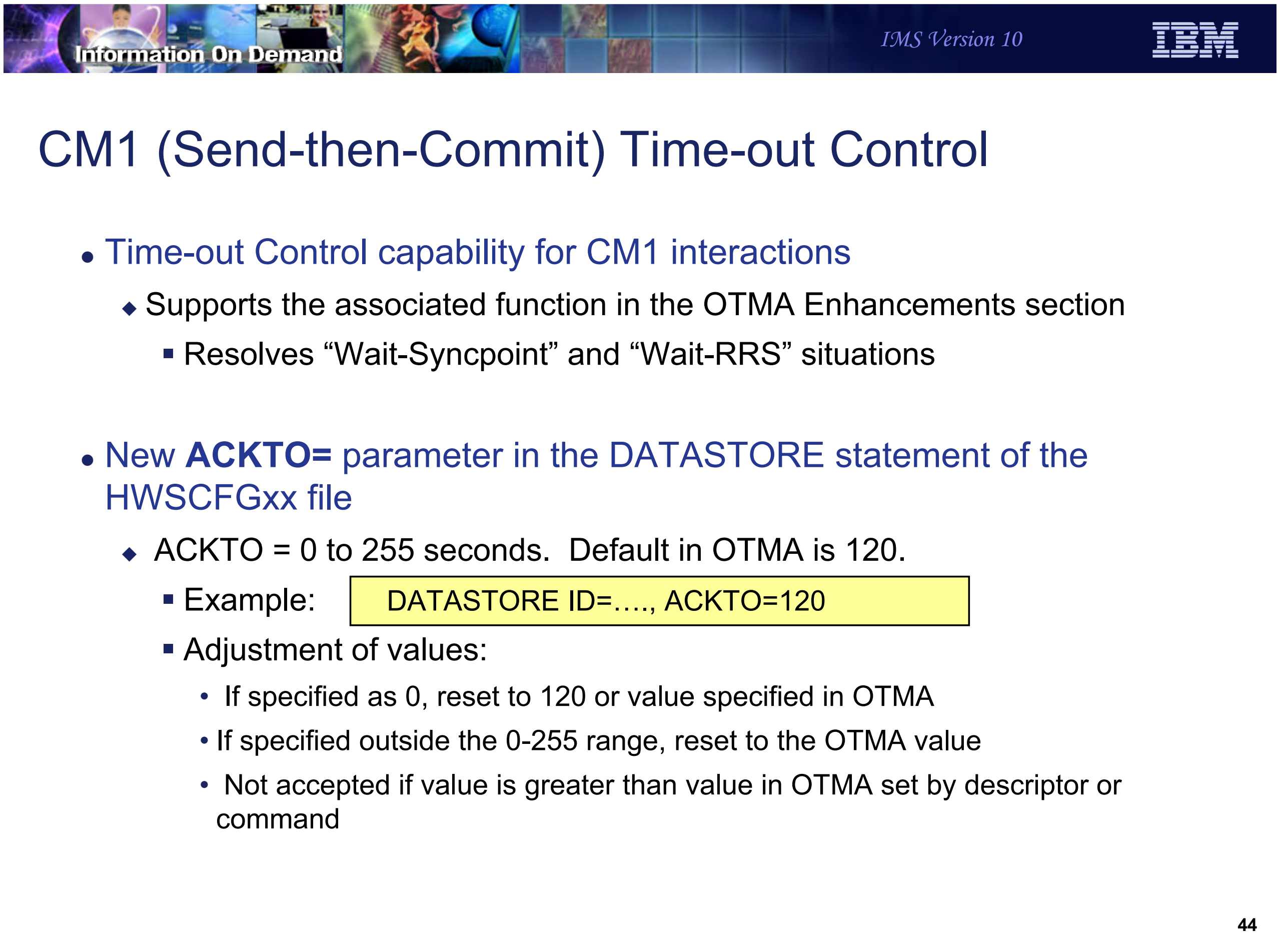
Notes:

The support for RACF mixed case password in IMS Connect is aligned with the IMS V10 support for mixed case passwords. The capability in IMS Connect allows the password to be preserved exactly as the remote client provided and pass the string to RACF without translation to upper case.

The function can be turned on as follows:

> A new parameter, PSWDMC= in the HWS= statement, defines the option of mixed case passwords where PSWDMC=N is the default. This setting can be changed using the IMS Connect SETPWMC or UPDATE command.
>
> IMS Connect also provides a command, SETPWMC, that can override the HWSCFGxx specification.
>
> The PSWDMC keyword is also available to the IMS Connect UPDATE command as another way to request the support.

The IMS Connect support requires that RACF enable mixed case passwords through the RACF SETROPTS(MIXEDCASE) command. Note that the RACF enablement of this support does not constitute the IMS Connect usage of this support. Also note that the mixed case support for IMS Connect can only take effect when RACF is enabled.


Information On Demand
IMS Version 10
IBM

CM1 (Send-then-Commit) Time-out Control

- Time-out Control capability for CM1 interactions
  - Supports the associated function in the OTMA Enhancements section
    - Resolves “Wait-Syncpoint” and “Wait-RRS” situations
- New **ACKTO=** parameter in the DATASTORE statement of the HWSCFGxx file
  - ACKTO = 0 to 255 seconds. Default in OTMA is 120.
    - Example: DATASTORE ID=…., ACKTO=120
    - Adjustment of values:
      - If specified as 0, reset to 120 or value specified in OTMA
      - If specified outside the 0-255 range, reset to the OTMA value
      - Not accepted if value is greater than value in OTMA set by descriptor or command

44

Figure 12-43. CM1 (Send-then-Commit) Time-out Control CMA01.0

Notes:

IMS Connect supports the new OTMA time-out control function for send-then-commit CM1 interactions. OTMA provides a default value of 120 seconds after which transactions that are held in “Wait-Syncpoint” or “Wait-RRS” status are released and backed out. If provided, the value in the ACKTO parameter of the IMS Connect configuration DATASTORE statement is passed to OTMA during client-bid processing.

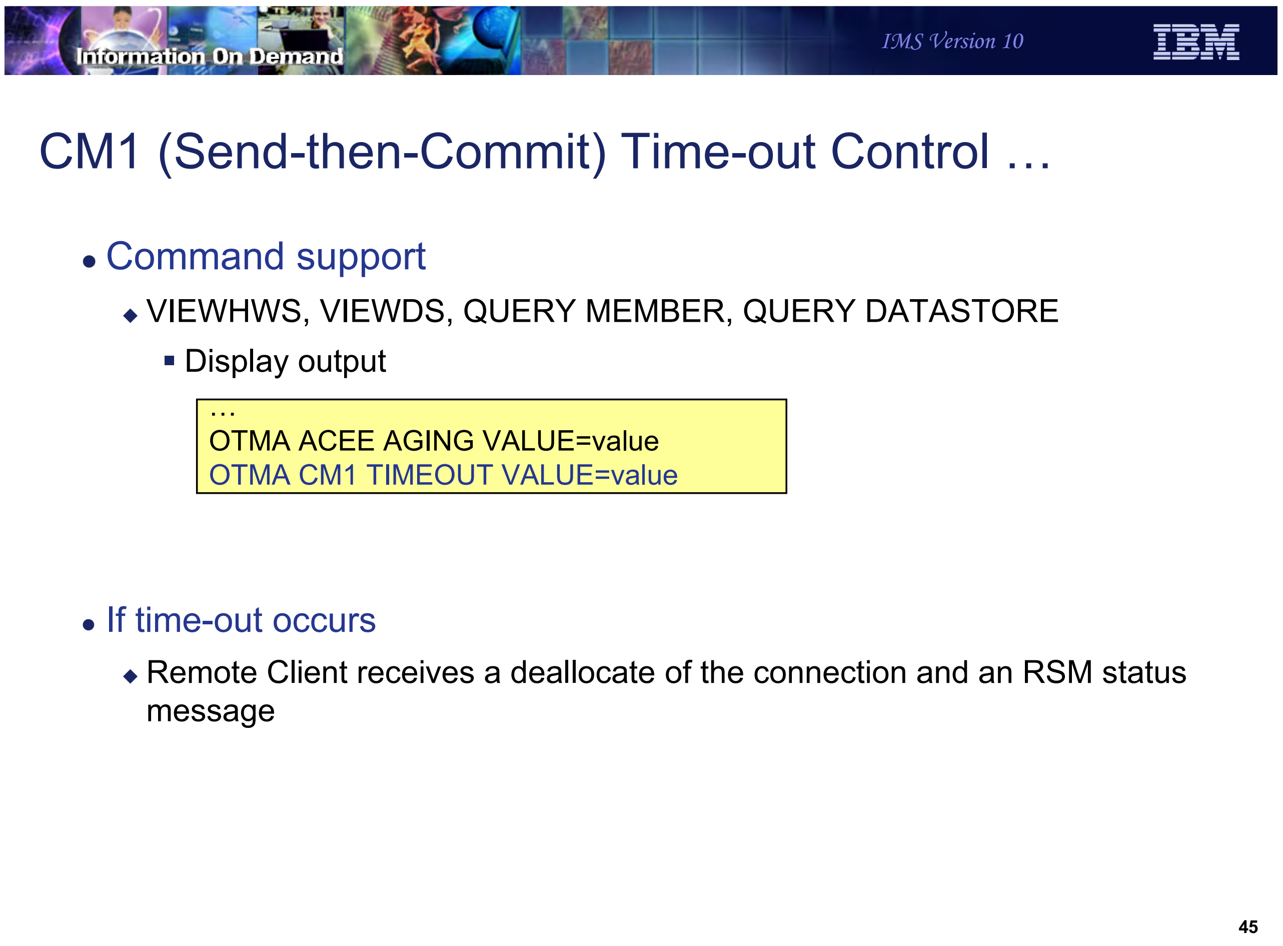
Note the following considerations:

A specified value of 0 is reset to 5.

Any value specified in error between 60 and 255 is reset to 60.

Any value outside the 0-255 range Results in an Abend U3401.

Value cannot be greater than what is defined in OTMA - If the CM1TO value is equal to or greater than the time-out specified in IMS by an OTMA descriptor or /STA TMEMBER command, OTMA will ignore the IMS Connect request. On the other hand, if the CM1TO value is less than the current time-out value set by the OTMA descriptor or command then the value which is passed to IMS by the IMS Connect client-bid process will be used for the time-out action for this member.


Information On Demand
IMS Version 10

CM1 (Send-then-Commit) Time-out Control …

- Command support
  - VIEWHWS, VIEWDS, QUERY MEMBER, QUERY DATASTORE
    - Display output

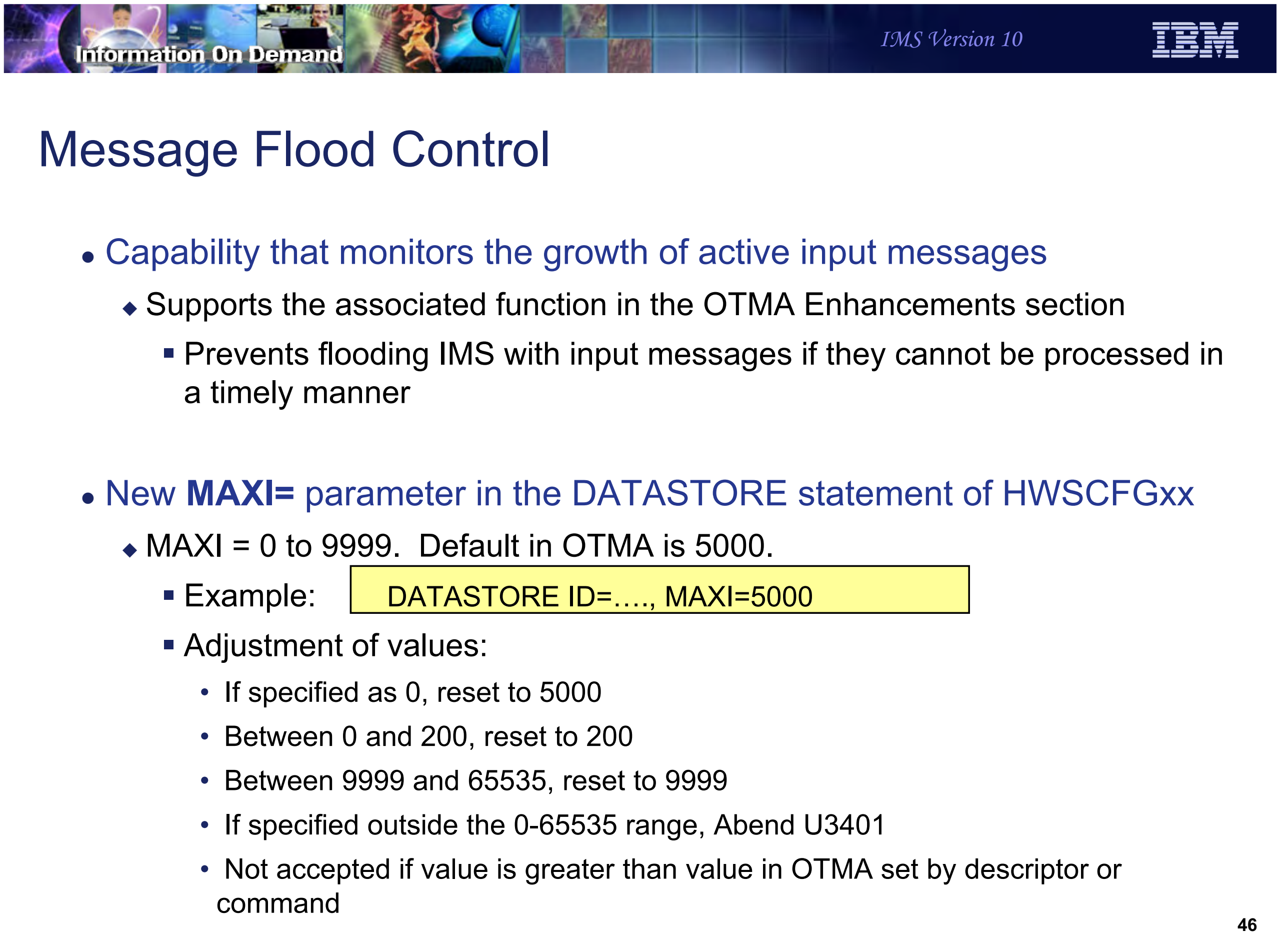
```
…
OTMA ACEE AGING VALUE=value
OTMA CM1 TIMEOUT VALUE=value
```

- If time-out occurs
  - Remote Client receives a deallocate of the connection and an RSM status message

Figure 12-44. CM1 (Send-then-Commit) Time-out Control … CMA01.0

Notes:

IMS Connect command output has been enhanced to display the CM1 timeout value. The applicable commands include: VIEWHWS and VIEWDS output command and the MVS MODIFY command for QUERY MEMBER and QUERY DATASTORE.


Information On Demand
IMS Version 10
IBM

Message Flood Control

- Capability that monitors the growth of active input messages
  - Supports the associated function in the OTMA Enhancements section
    - Prevents flooding IMS with input messages if they cannot be processed in a timely manner
- New **MAXI=** parameter in the DATASTORE statement of HWSCFGxx
  - MAXI = 0 to 9999. Default in OTMA is 5000.
    - Example: `DATASTORE ID=…., MAXI=5000`
    - Adjustment of values:
      - If specified as 0, reset to 5000
      - Between 0 and 200, reset to 200
      - Between 9999 and 65535, reset to 9999
      - If specified outside the 0-65535 range, Abend U3401
      - Not accepted if value is greater than value in OTMA set by descriptor or command

Figure 12-45. Message Flood Control	CMA01.0

Notes:

IMS Connect also takes advantage of the OTMA Message Flood Control capability to automatically monitor the growth of active input messages. OTMA provides a default value of 5000 messages after which input messages from a specific IMS Connect instance will be rejected. If provided, an override value in the MAXI parameter of the IMS Connect configuration DATASTORE statement is passed to OTMA during client-bid processing.

Note the following considerations:

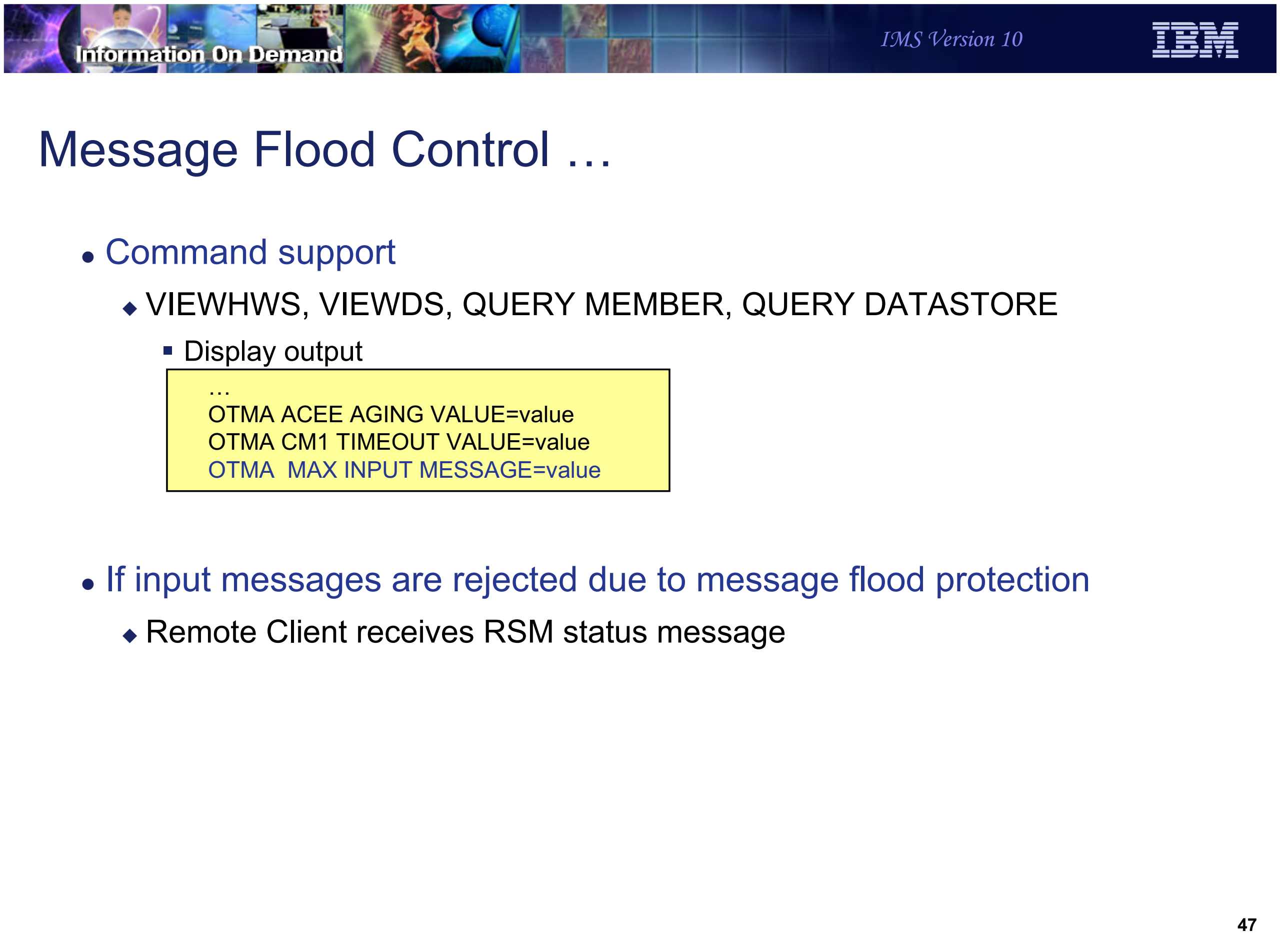
A specified value of 0 is reset to 5000.

A value between 0 and 200 is reset to 200.

Any value specified in error between 9999 and 65535 is reset to 9999.

Any value outside the 0-65535 range Results in an Abend U3401.

Value cannot be greater than what is defined in OTMA - If the MAXI value is equal to or greater than the INPUT value specified in IMS by an OTMA descriptor or /STA TMEMBER command, OTMA will ignore the IMS Connect request. On the other hand, if the MAXI value is between 1 and 200, a value of 200 will be sent to IMS. If the IMS Connect value is less than the value specified in OTMA, then the IMS Connect MAXI value will be used.



Information On Demand
IMS Version 10

Message Flood Control …

- Command support
  - VIEWHWS, VIEWDS, QUERY MEMBER, QUERY DATASTORE
    - Display output

```
…
OTMA ACEE AGING VALUE=value
OTMA CM1 TIMEOUT VALUE=value
OTMA  MAX INPUT MESSAGE=value
```

- If input messages are rejected due to message flood protection
  - Remote Client receives RSM status message

Figure 12-46. Message Flood Control … CMA01.0

Notes:

IMS Connect command output has been enhanced to display the maximum input message value. The applicable commands include: VIEWHWS and VIEWDS output command and to the MVS MODIFY command for QUERY MEMBER and QUERY DATASTORE.

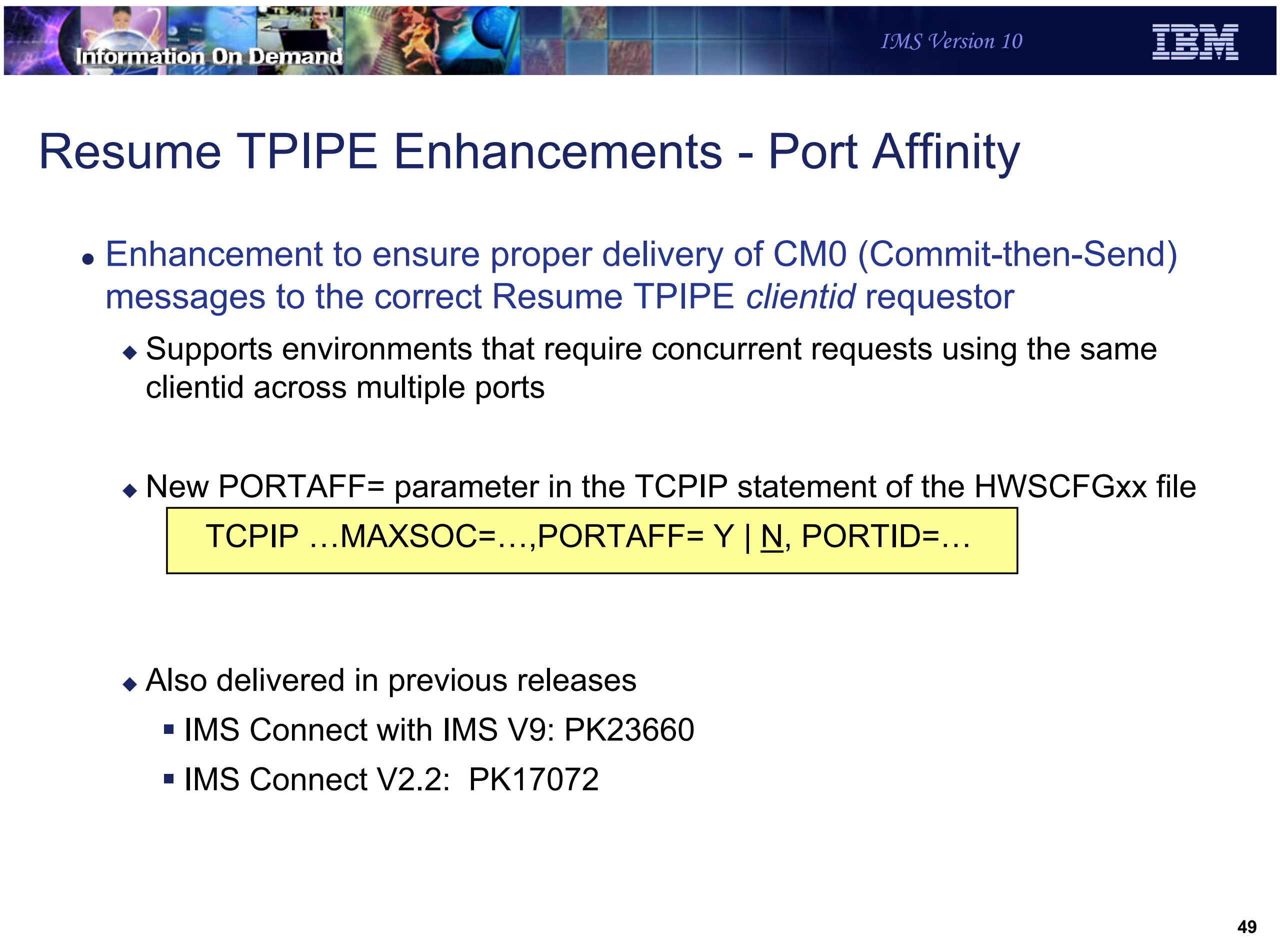
Asynchronous Message (CM0) Enhancements

- Functions that are documented and described in the OTMA Enhancements section
  - Super member support
  - Reroute
  - Purge

Figure 12-47. Asynchronous Message (CM0) Enhancements CMA01.0

Notes:

There are several enhancements for CM0 processing that are described in detail in the OTMA Enhancements section. The detail in the OTMA section includes specifics for IMS Connect.



Information On Demand
IMS Version 10
IBM

Resume TPIPE Enhancements - Port Affinity

- Enhancement to ensure proper delivery of CM0 (Commit-then-Send) messages to the correct Resume TPIPE *clientid* requestor
  - Supports environments that require concurrent requests using the same clientid across multiple ports
  - New PORTAFF= parameter in the TCPIP statement of the HWSCFGxx file

```
TCPIP …MAXSOC=…,PORTAFF= Y | N, PORTID=…
```

  - Also delivered in previous releases
    - IMS Connect with IMS V9: PK23660
    - IMS Connect V2.2: PK17072

Figure 12-48. Resume TPIPE Enhancements - Port Affinity CMA01.0

Notes:

Using concurrent Resume TPIPE connection requests of the same clientid across several ports may cause problems such as keeping an IMS OTMA TPIPE in WAIT-H status. To address the issue and support this concurrency requirement, IMS Connect provides a new parameter to enforce all the correlated interaction such as the retrieval of a message and associated ACK or NAK to the same remote client instance. The PORTAFF parameter in the TCPIP statement controls whether commit-then-send (CM0) output messages sent by IMS to an IMS Connect system have affinity to the port on which IMS Connect received the original input message.

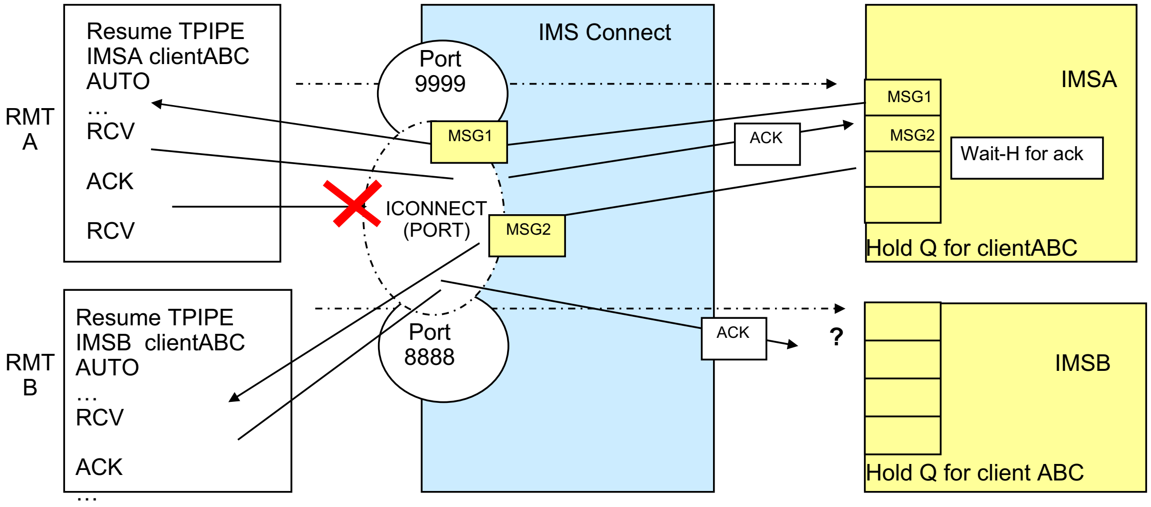
When PORTAFF=Y is specified, IMS Connect returns all CM0 output for this IMS Connect client through the same port on which it received the original input message.

When PORTAFF=N is specified, IMS Connect attempts to return the CM0 output to the first client it finds on any available port with an outstanding request from this clientid.

NOTE: If running in a sysplex environment that has implemented redundancy, load balancing, and supermember support, etc., PORTAFF=N is a reasonable choice. Using the same instance of a single clientid across multiple ports is not recommended.

Resume TPIPE Enhancements - Port Affinity …

- Consideration - concurrent requests using the same clientid
  - PORTAFF=N (default)



RMT A

Resume TPIPE
IMSA clientABC
AUTO
…
RCV
ACK
RCV

RMT B

Resume TPIPE
IMSB clientABC
AUTO
…
RCV
ACK
…

IMS Connect
Port 9999
MSG1
ICONNECT (PORT)
MSG2
Port 8888
ACK
ACK
?

IMSA
MSG1
MSG2
Wait-H for ack
Hold Q for clientABC

IMSB
Hold Q for client ABC

Recommendation: specify PORTAFF=Y
or do not use concurrent Resume_TPIPE requests with the same clientid

Figure 12-49. Resume TPIPE Enhancements - Port Affinity … CMA01.0

Notes:

As mentioned in the previous visual, when PORTAFF=N, IMS Connect attempts to return the CM0 output to the first port found on which the client ID of this IMS connect is present. PORTAFF=N is the default. There is a consideration in this area because if a message is retrieved from the Hold Queue, then the PORTID in OMUSER_PORTID is ignored and IMS Connect assumes the portid is the generic ICONNECT port. This assumption tends not to be a problem unless an error such as a connection failure occurs. When that happens and a message has been received in IMS Connect for delivery, IMS Connect scans all the ports under the generic ICONNECT port and delivers the message to the first one that it finds.

The example on this visual shows a situation where two Resume_TPIPE AUTO requests specifying the same clientid, clientABC, are sent into a single IMS Connect. One request (from RMTA) is sent to IMSA and the other (from RMTB) is sent to IMSB. IMSA has two messages, MSG1 and MSG2, on the Hold Queue. IMSB has no messages at the moment and so clientABC on RMTB just waits. IMSA sends MSG1 to IMS Connect which delivers the message to the outstanding request for clientABC on RMTA which responds with an ACK. In this scenario the connection fails for one of several reasons - ACK timeout, network problem, etc. Since the Resume_TPIPE had originally specified AUTO, IMSA sends MSG2 after the ack for MSG1 is received. When IMS Connect receives the message, it detects that the connection to RMTA is no longer there and scans the ports under ICONNECT to find the first one available. IMS Connect sends MSG2 to the waitiing clientABC on RMTB. This instance of ClientABC retrieves the message and sends an ACK back to IMSB which is not expecting an ACK. Meanwhile, IMSA's Hold Queue for clientABC is in WAIT-H status waiting for an ACK that will never be received.

Information On Demand

IMS Version 10

Resume TPIPE Enhancements - Port Affinity …

- In a Sysplex Distributor, Super Member, Shared Queues environment
  - PORTAFF=N (default) is feasible

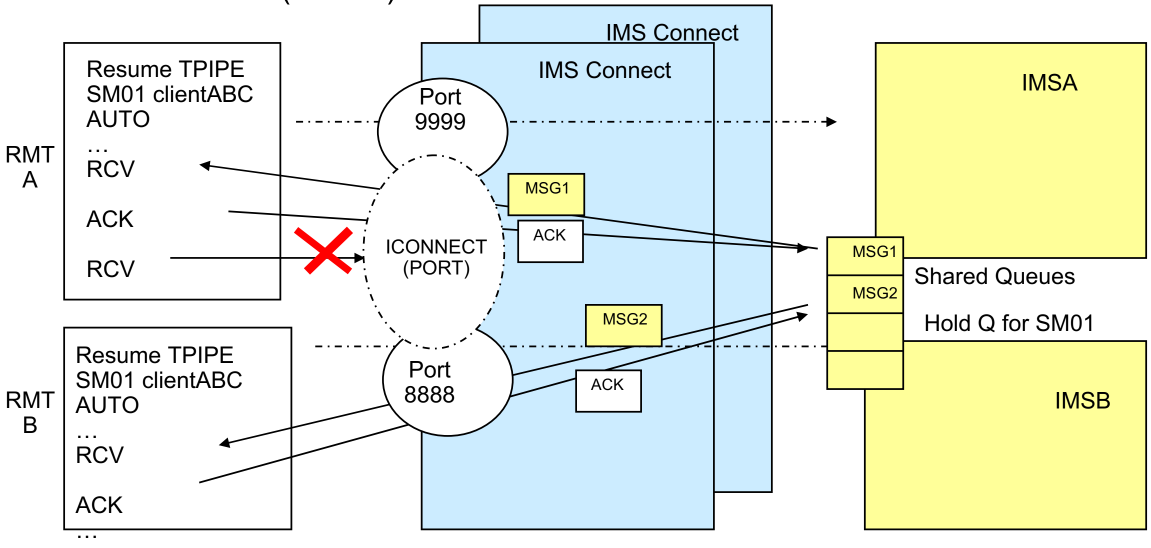
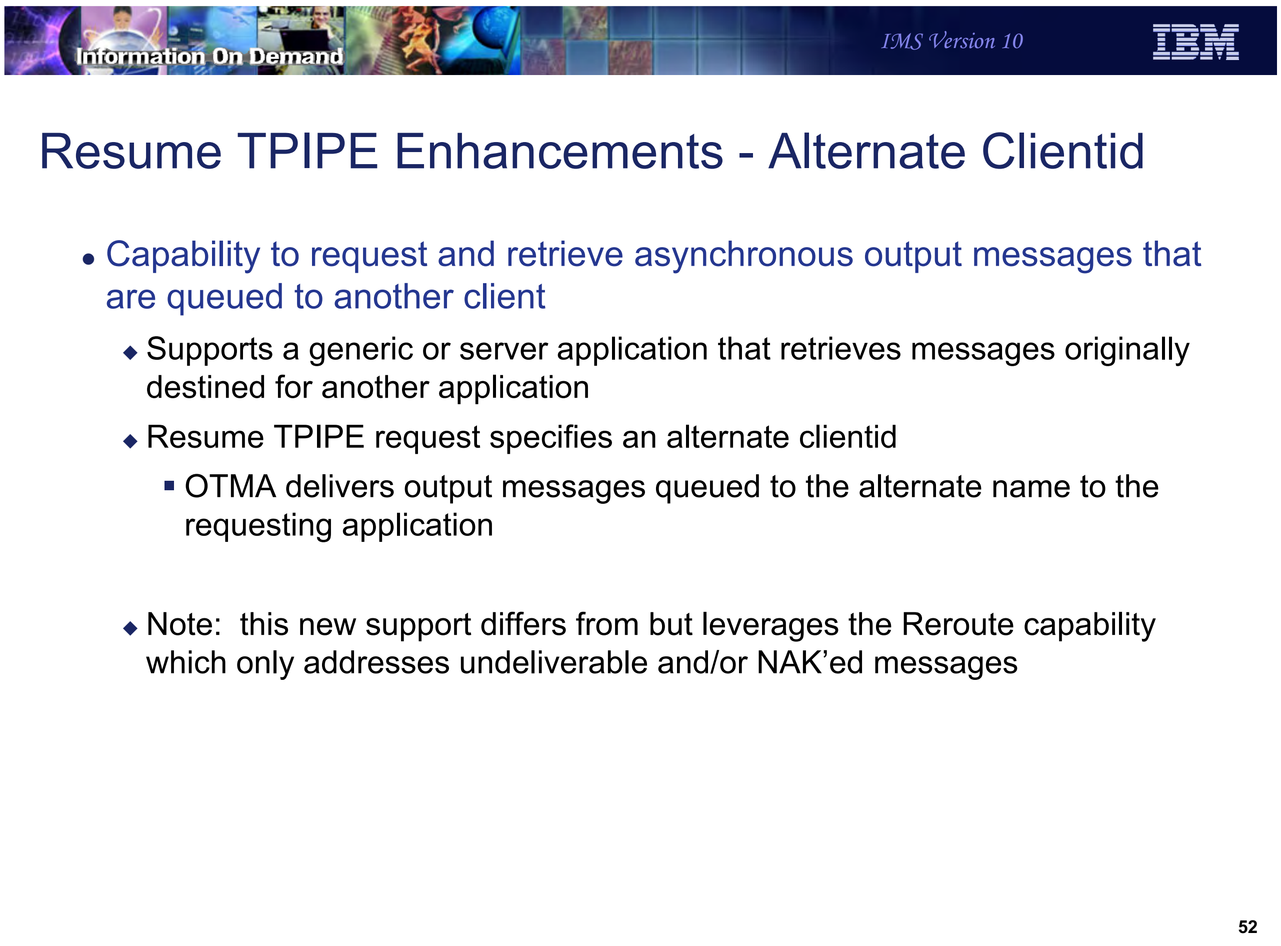

RMT A
Resume TPIPE
SM01 clientABC
AUTO
…
RCV
ACK
RCV
RMT B
Resume TPIPE
SM01 clientABC
AUTO
…
RCV
ACK
…
IMS Connect
IMS Connect
Port 9999
ICONNECT (PORT)
Port 8888
MSG1
ACK
MSG2
ACK
IMSA
MSG1
MSG2
Shared Queues
Hold Q for SM01
IMSB

Figure 12-50. Resume TPIPE Enhancements - Port Affinity … CMA01.0

Notes:

In an environment that supports full redundancy including Shared Queues, Super Member support, Sysplex Distributor, etc., specifying PORTAFF=N is feasible. This configuration with concurrent Resume_TPIPE clients using the same clientid presumes that any of the remote clients can retrieve any of the messages.



Information On Demand
IMS Version 10
IBM

Resume TPIPE Enhancements - Alternate Clientid

- Capability to request and retrieve asynchronous output messages that are queued to another client
  - Supports a generic or server application that retrieves messages originally destined for another application
  - Resume TPIPE request specifies an alternate clientid
    - OTMA delivers output messages queued to the alternate name to the requesting application
  - Note: this new support differs from but leverages the Reroute capability which only addresses undeliverable and/or NAK'ed messages

Figure 12-51. Resume TPIPE Enhancements - Alternate Clientid CMA01.0

Notes:

IMS Connect introduces a new protocol that allows client applications to specify an alternate clientid in the RESUME TPIPE request. IMS Connect forwards the alternate clientid to OTMA, and OTMA returns the asynchronous messages that are queued to the TPIPE of the alternate clientid name to the client application that issued the Resume TPIPE request.

This Alternate Clientid function differs from the Reroute capability that was discussed earlier. Reroute requests address the situations when messages cannot be delivered or are NAK'ed. In these situations, OTMA queues the message onto the TPIPE name associated with the reroute request when the undeliverable condition occurs. The Alternate Clientid function, on the other hand, supports Resume TPIPE requests that retrieve messages which are already queued to a TPIPE name but the name is different . This new capability could be used to provide a programmable solution in the OTMA/IMS Connect environment that is comparable to the way that IMS users can assign an LTERM and all messages queued to it to a different node.

Information On Demand
IMS Version 10
IBM

Resume TPIPE Enhancements - Alternate Clientid …

- Supported by IMS Connect user message exits
  - HWSSMPL0, HWSSMPL1, HWSSOAP1, HWSJAVA0
    - IRM Header
      - IRM_RT_ALTCID   8 bytes specifying the alternate clientid
        - Occupies same offset as reroute name IRM_REROUT_NM field
    - OTMA header
      - OMUSR_RT_ALTCID   8 bytes specifying the alternate clientid
        - Occupies the same offset of OMUSR_REROUT_NM field
- Resume TPIPE security (discussed in the OTMA section)
  - IF RIMS|Rxxxxxxx resource class is enabled
    - Required userid access to the alternate clientid (TPIPE)

Figure 12-52. Resume TPIPE Enhancements - Alternate Clientid … CMA01.0

Notes:

This support applies to: remote client applications that invoke user message exits HWSSMPL0 and HWSSMPL1; IMS SOAP Gateway client applications using user message exit HWSSOAP1; IMS Connector for Java (IC4J) client applications using user message exit HWSJAVA0; and any other user-written IMS Connect message exits. No support is provided for the local option, HWSIMSO0 and HWSIMSO1.

To take advantage of the new capability, optional fields have been added to both the IRM and OTMA headers. Either field, IRM_RT_ALTCID in the IRM or OMUSR_RT_ALTCID can be used to set a valid alternate clientid value. The feature is not enabled if the fields are both set to blanks.

XML Adapter Support

- Capability that supports translation between XML messages and IMS messages
  - IMS Connect client, e.g., IMS SOAP Gateway (available with IMS V9)
    - Sends an XML message with a request for translation
  - IMS Connect
    - Inbound: invokes the XML Adapter to translate message for IMS
      - Removes XML tags
      - If necessary, convert from UNICODE to EBCDIC
    - Outbound: invokes the XML Adapter to prepare an XML message
      - If necessary, convert from EBCDIC to appropriate UNICODE encoding schema
      - Create XML tags
- IMS V9 Support - PK24912, PK29938

Figure 12-53. XML Adapter Support

Notes:

The XML Adapter support opens up IMS Connect to: receive and recognize messages containing XML tags; provide the conversion into a message format for IMS including unicode to EBCDIC translation if needed; receive IMS Application reply messages that do not contain XML tags; and perform the conversion to XML as well as any EBCDIC to unicode translation prior to replying to the remote client.

XML Adapter Support - Overview

- Overview

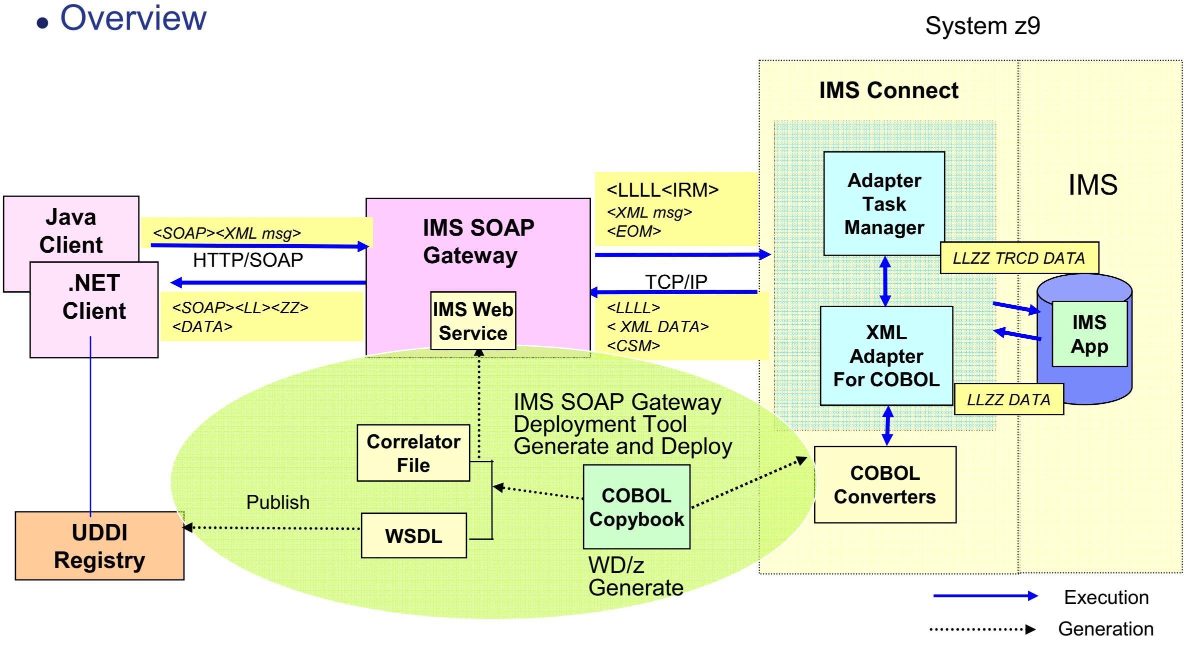

System z9
IMS Connect
IMS
Java Client
.NET Client
<SOAP><XML msg>
HTTP/SOAP
<SOAP><LL><ZZ><DATA>
IMS SOAP Gateway
IMS Web Service
<LLLL<IRM>
<XML msg>
<EOM>
TCP/IP
<LLLL>
< XML DATA>
<CSM>
Adapter Task Manager
LLZZ TRCD DATA
XML Adapter For COBOL
LLZZ DATA
IMS App
COBOL Converters
IMS SOAP Gateway Deployment Tool Generate and Deploy
Correlator File
WSDL
COBOL Copybook
WD/z Generate
Publish
UDDI Registry
Execution
Generation

Figure 12-54. XML Adapter Support - Overview CMA01.0

Notes:

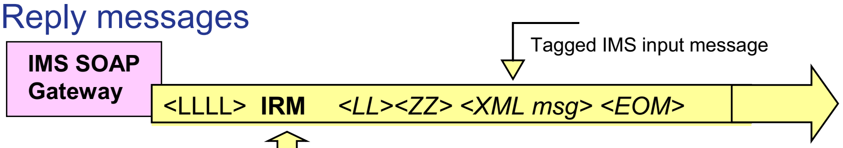
This overview visual shows the current supported configuration for the XML Adapter Support in IMS Connect. The IMS Soap Gateway is an IBM-provided function that must be at a minimum level of V9.2 to take advantage of the XML Adapter in IMS Connect.

WebSphere Developer for zSeries (WDz) supports the generation of XML Converters for COBOL applications using COBOL copybooks. The XML Adapter support in IMS Connect, in conjunction with the generated WDz XML Converters, facilitates the conversion of XML transactional requests into byte stream application data structures, and vice versa. Clients can send transactional requests in XML to IMS Connect, where the XML Adapter calls the user specified WDz XML Converter to convert it to byte streams that the IMS application understands, and then IMS Connect sends it to IMS. Responses when received by IMS Connect will be converted to XML before being sent back to the clients. Each IMS application expects its messages to be in a certain data structure; therefore, one XML Converter is needed for each IMS application.

Information On Demand IMS Version 10

XML Adapter Support - Implementation

- Input and Reply messages



- IRM Header - provides new specifications for the XML adapter

LL	- 2 byte Length
zz -->	byte 1 - Architecture Level, Set to Binary '00000010' to signal that the storage space in the IRM is present for IRM_TAG_ADAPT and IRM_TAG_MAP byte 2 - Set to Binary '00000001' if Tagged TranCode and Data Set to Binary '00000010' if Tagged Data only
IRMID	identifies the user message exit, e.g., *HWSOA1*
…..	
New fields added at the end of the user data section	
IRM_TAG_ADAPT	specifies the Adapter name that is to receive control to perform the XML removal on input and addition on output.,e.g, HWSXMLA0
IRM_TAG_MAP	specifies the Tag name that is to be used by the ADAPTER to process the XML tags on input and output messages. (XML Converter Driver name)

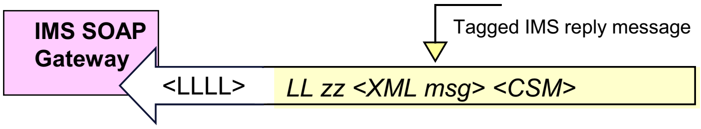


Figure 12-55. XML Adapter Support - Implementation CMA01.0

Notes:

Input messages must adhere to the application protocol, as defined and documented for IMS Connect, by providing an IRM header that provides the name of the message exit routine that is to be called. For XML Adapter support, the IRM provides flags and values that can be used to invoke the appropriate functions.

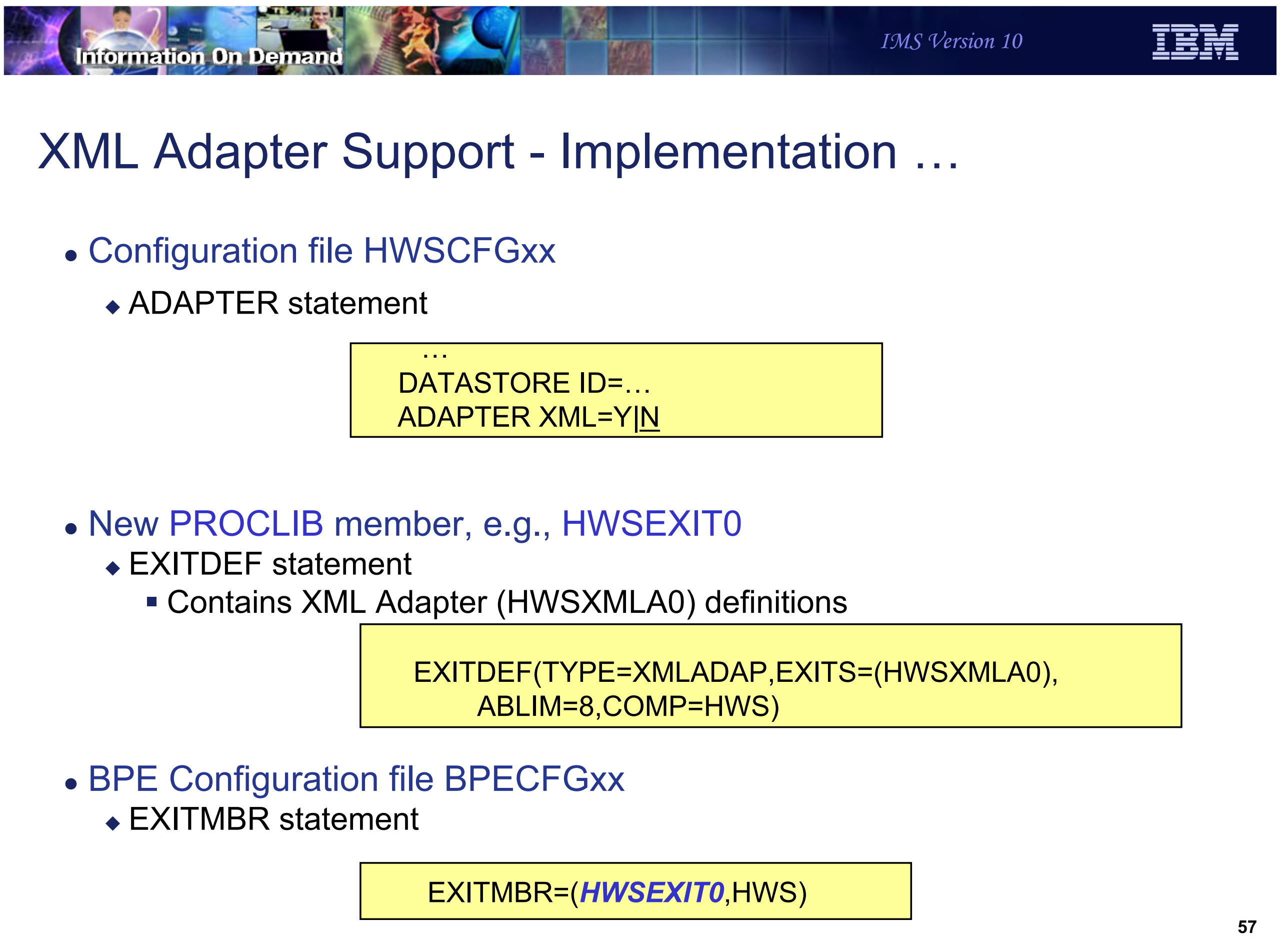
The IRM header always begins with an LL (2 byte length field) follows by 2 bytes that are usually zeroes for non-XML message processing. When the XML Adapter is to be invoked, the first zz byte can be used to signal the inclusion of 16 bytes in the user data section of the IRM header for specification of the TAG and ADAPTER names. The second byte details whether the tagged data includes the trancode or not.

At the end of the of the user data section, two fields provide the information for the XML environment:

> The IRM_TAG_MAP field provides 8 bytes to specify the TAG name. This is the name of the COBOL Driver, the XMI or whatever is required for the Adapter to call or use to perform the XML transformation.

> The IRM_TAG_ADAPT field provides 8 bytes to specify the ADAPTER name. This is the name of the routine that is to gain control from IMS Connect to remove/add the XML tags, defined by the TAG name. The same Adapter that processes the input message (removes the XML Tags) will be required to process the output message (add the XML Tags).

Reply messages are sent back by IMS connect in tagged format. When the message is sent by IMS to IMS Connect, the user message exit invokes the Adapter to process the byte array data and convert it to XML tagged data before sending the message to the remote client.


Information On Demand
IMS Version 10
IBM

XML Adapter Support - Implementation …

- Configuration file HWSCFGxx
  - ADAPTER statement

```
 …
DATASTORE ID=…
ADAPTER XML=Y|N
```

- New PROCLIB member, e.g., HWSEXIT0
  - EXITDEF statement
    - Contains XML Adapter (HWSXMLA0) definitions

```
EXITDEF(TYPE=XMLADAP,EXITS=(HWSXMLA0),
      ABLIM=8,COMP=HWS)
```

- BPE Configuration file BPECFGxx
  - EXITMBR statement

```
EXITMBR=(HWSEXIT0,HWS)
```

Figure 12-56. XML Adapter Support - Implementation … CMA01.0

Notes:

To enable the support in IMS Connect, several definitions have to be created:

A new ADAPTER statement in the HWSCFGx file provides the option of identifying the presence or absence of the XML adapter support. The parameter XML= provides two values. XML=Y/N. Y(es) requests the XML adapter function. The default of N(o) disables the function.

The only adapter that is currently supported is the IBM-provided Cobol Adapter HWSXMLA0. The definitions for the adapter are provided in an EXITDEF statement in a special PROCLIB member. The name of the member can be HWSEXIT0, as shown in this example, or it can be any name that is meaningful to the installation. The name that is chosen must also be defined to IMS in the BPE configuration file statement EXITMBR. The values of the EXITDEF statement are as follows:

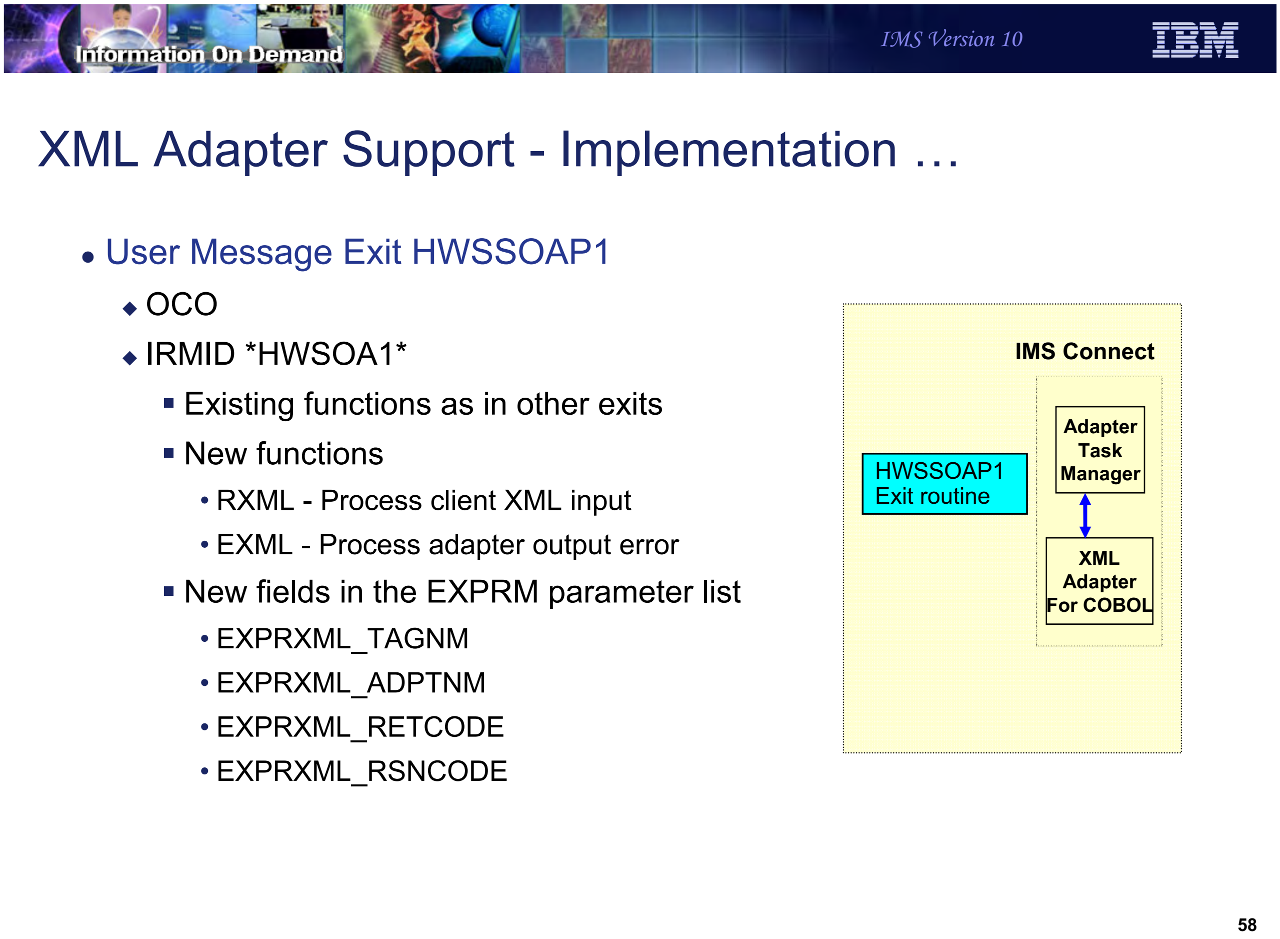
TYPE=XMLADAP must be coded as is and defines the exit as an XML adapter plug-in to IMS Connect.

EXITS=HWSXMLA0 also must be coded as shown for the IBM-supplied Cobol Adapter.

ABLIM = 0 (unlimited), 1, up to 2,147,483,647 defines the abend limit which is the number of times the XML ADAPTER can abend before it is disabled

COMP=HWS must also be coded as is and requests the HWS component of IMS Connect.

IMS Connect detects the name of the PROCLIB member which contains the adapter information in the EXITDEF statement in the BPE configuration file.



XML Adapter Support - Implementation …

- User Message Exit HWSSOAP1
  - OCO
  - IRMID *HWSOA1*
    - Existing functions as in other exits
    - New functions
      - RXML - Process client XML input
      - EXML - Process adapter output error
    - New fields in the EXPRM parameter list
      - EXPRXML_TAGNM
      - EXPRXML_ADPTNM
      - EXPRXML_RETCODE
      - EXPRXML_RSNCODE

Figure 12-57. XML Adapter Support - Implementation … CMA01.0

Notes:

IMS Connect provides a new user message exit routine that specifically understands the XML interface. In addition to the existing message functions of READ, XMIT, INIT and TERM, the new exit routine, HWSSOAP1, includes new interfaces: RXML and EXML. On input from the IMS Connect Client Application, IMS Connect calls HWSSOAP1 with the function type set to ‘RXML’ which translates the IRM header and returns the ADAPTER name and the TAG name back to IMS Connect. Upon returning to IMS Connect, error analysis will be performed. If an error exists, function ‘EXML’ is invoked to process the error message. If no error exists, IMS Connect calls and passes control to the ADAPTER passing a set of parameters which determine: if this ADAPTER should process the message; if translation is required or not; and what XML processing is required. When the reply is ready to be sent back, IMS Connect receives the message and invokes the ‘XMIT’ function. Again the ADAPTER is called to prepare the reply in a tagged format.

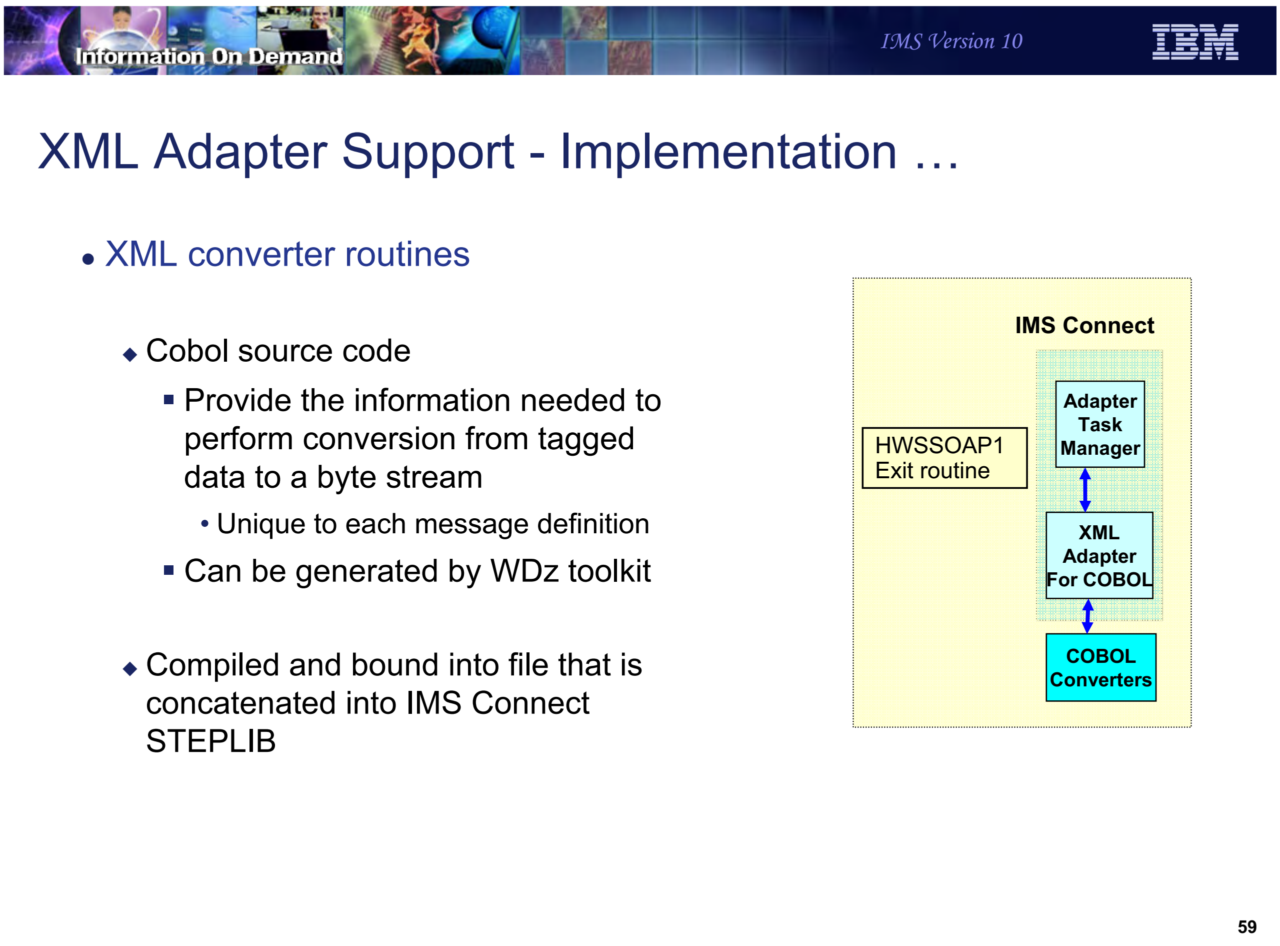
New fields in the EXPRM parameter list include the following:
- EXPRXML_TAGNM Defines an output field and contains the TAG NAME to be used by the ADAPTER, the TAG name would represent one of the following: XMI name for user

purposes, COBOL Driver name for IMS Connect support of COBOL ADAPTER, user Map Name for user purposes

- EXPRXML_ADPTNM Defines an output field and contains the Adapter name to be used by IMS Connect to load the correct ADAPTER.

- EXPRXML_RETCODE - Return code returned by the User Message Exit back to IMS Connect.

-EXPRXML_RSNCODE - Reason code returned by the User Message Exit.


Information On Demand
IMS Version 10
IBM

XML Adapter Support - Implementation …

- XML converter routines
  - Cobol source code
    - Provide the information needed to perform conversion from tagged data to a byte stream
      - Unique to each message definition
    - Can be generated by WDz toolkit
  - Compiled and bound into file that is concatenated into IMS Connect STEPLIB

IMS Connect
HWSSOAP1
Exit routine
Adapter
Task
Manager
XML
Adapter
For COBOL
COBOL
Converters

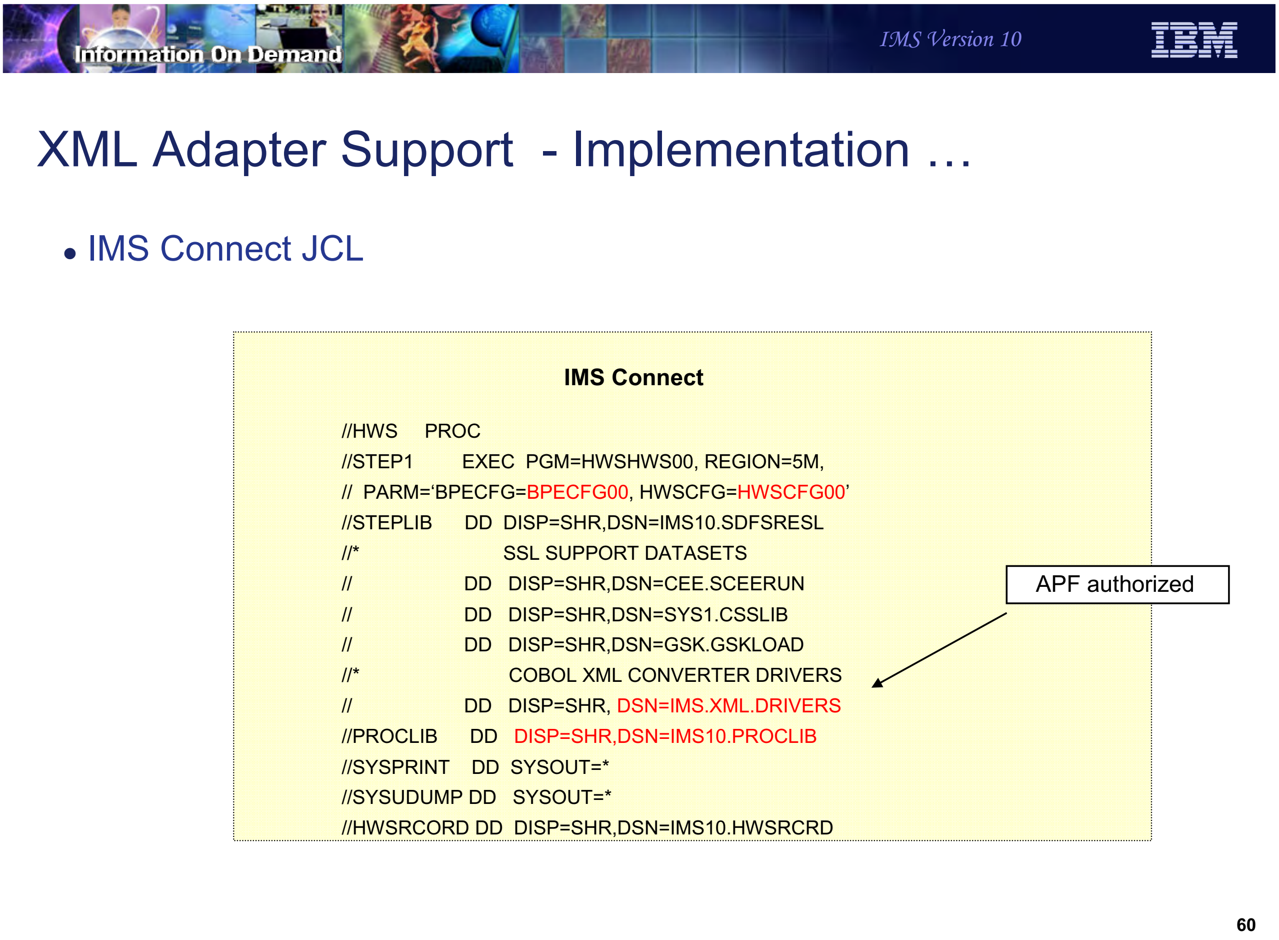
Figure 12-58. XML Adapter Support - Implementation … CMA01.0

Notes:

To perform correct conversion for each message, the Adapter requires information on how this is to be done. This information is called an XML Converter. XML Converters are Cobol programs generated from WebSphere Developer for zSeries (WDz) using the target IMS COBOL application's copybook. For each copybook, WDz generates three COBOL programs - an Inbound Converter (XML schema), an Outbound Converter (XML schema), and a Driver (Cobol code). These three programs are combined into one file and are referred to as an XML Converter. An XML Converter has to be created for each IMS COBOL transaction application that wants to support XML transaction messages. For details on generating XML Converters using WDz, see the WDz documentation.

On an inbound request to IMS, the XML Adapter calls the COBOL XML Converter Driver and passes it the inbound function code. The COBOL XML Converter Driver calls the Inbound Converter to parse the incoming XML message, convert the parsed message into the COBOL data structure byte streams, and then return it to the XML Adapter. On an outbound reply from the IMS application program, the XML Adapter calls the COBOL XML Converter Driver with an outbound function code, which then calls the Outbound Converter.

The Outbound Converter converts the IMS output message into XML and passes it back to the XML Adapter which then sends it to IMS Connect to be sent back to the client.



XML Adapter Support - Implementation …

- IMS Connect JCL

IMS Connect

```
//HWS     PROC
//STEP1        EXEC  PGM=HWSHWS00, REGION=5M,
//  PARM=‘BPECFG=BPECFG00, HWSCFG=HWSCFG00’
//STEPLIB      DD  DISP=SHR,DSN=IMS10.SDFSRESL
//*                  SSL SUPPORT DATASETS
//                DD   DISP=SHR,DSN=CEE.SCEERUN
//                DD   DISP=SHR,DSN=SYS1.CSSLIB
//                DD   DISP=SHR,DSN=GSK.GSKLOAD
//*                   COBOL XML CONVERTER DRIVERS
//                DD   DISP=SHR, DSN=IMS.XML.DRIVERS
//PROCLIB      DD   DISP=SHR,DSN=IMS10.PROCLIB
//SYSPRINT    DD  SYSOUT=*
//SYSUDUMP DD   SYSOUT=*
//HWSRCORD DD  DISP=SHR,DSN=IMS10.HWSRCRD
```

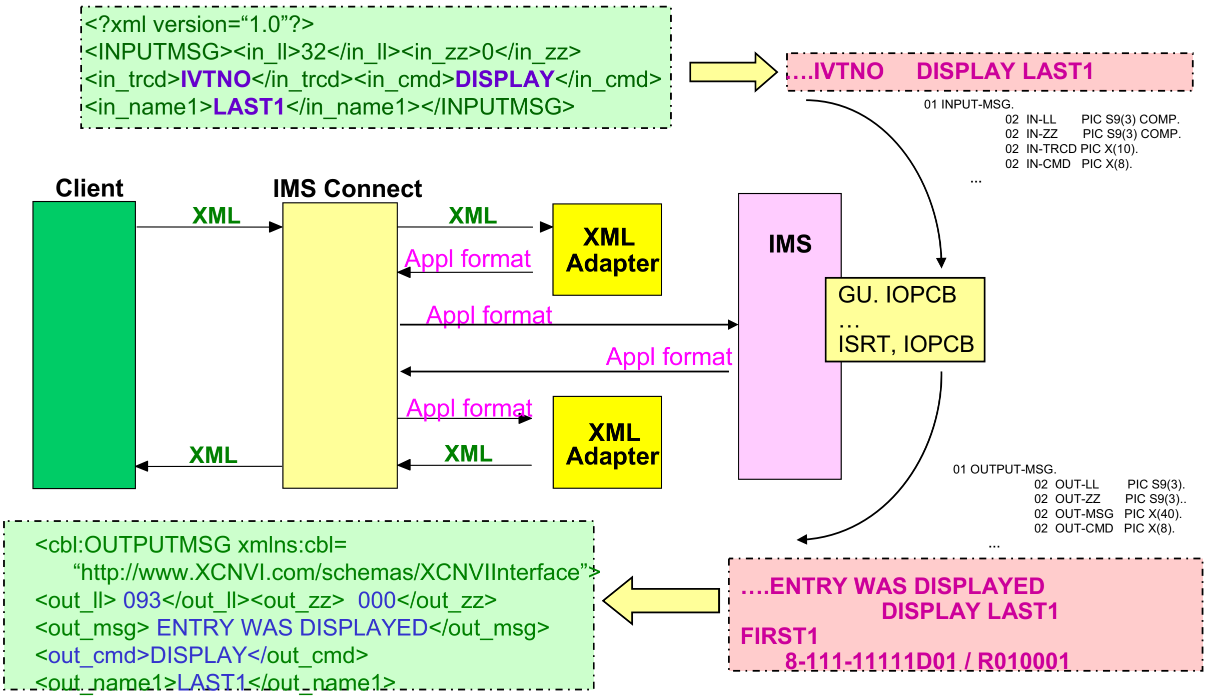
APF authorized

Figure 12-59. XML Adapter Support - Implementation … CMA01.0

Notes:

The Cobol XML Converter routines must be added into an APF-authorized library that it concatenated into the STEPLIB list of the IMS Connect startup JCL.

XML Adapter Support - Flow



```
<?xml version="1.0"?>
<INPUTMSG><in_ll>32</in_ll><in_zz>0</in_zz>
<in_trcd>IVTNO</in_trcd><in_cmd>DISPLAY</in_cmd>
<in_name1>LAST1</in_name1></INPUTMSG>
```

....IVTNO DISPLAY LAST1

```
01 INPUT-MSG.
        02  IN-LL     PIC S9(3) COMP.
        02  IN-ZZ     PIC S9(3) COMP.
        02  IN-TRCD PIC X(10).
        02  IN-CMD   PIC X(8).
   ...
```

```
GU. IOPCB
…
ISRT, IOPCB
```

```
01 OUTPUT-MSG.
        02  OUT-LL      PIC S9(3).
        02  OUT-ZZ      PIC S9(3)..
        02  OUT-MSG   PIC X(40).
        02  OUT-CMD   PIC X(8).
   ...
```

```
....ENTRY WAS DISPLAYED
            DISPLAY LAST1
FIRST1
    8-111-11111D01 / R010001
```

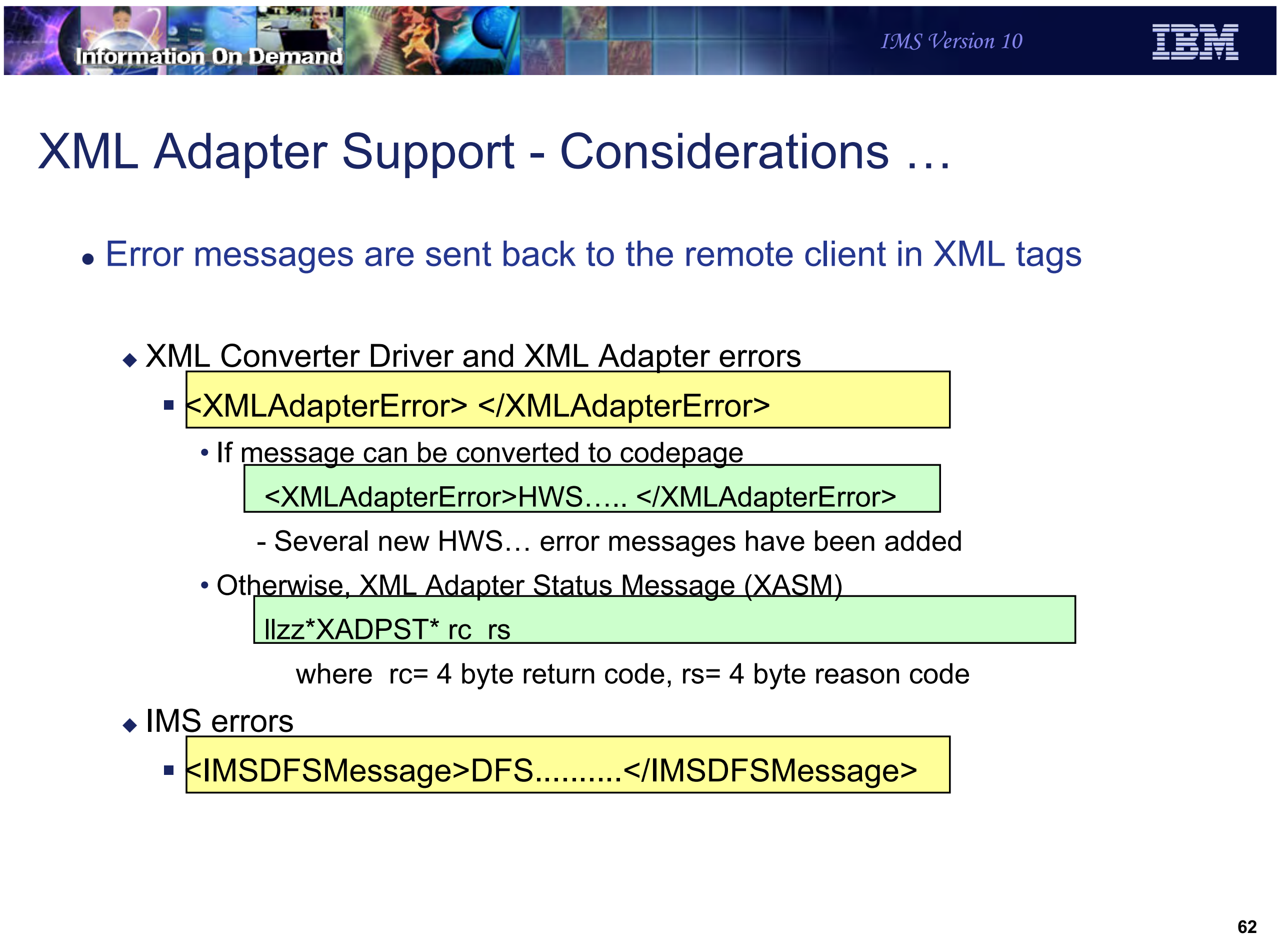
```
  <cbl:OUTPUTMSG xmlns:cbl=
     "http://www.XCNVI.com/schemas/XCNVIInterface">
  <out_ll> 093</out_ll><out_zz>  000</out_zz>
  <out_msg> ENTRY WAS DISPLAYED</out_msg>
  <out_cmd>DISPLAY</out_cmd>
  <out_name1>LAST1</out_name1>
```

Figure 12-60. XML Adapter Support - Flow CMA01.0

Notes:

This visual gives a high level view of the flow of messages. When an IMS Connect Client sends a transaction message in XML to IMS Connect, IMS Connect receives the input message and checks a flag in the IRM that indicates whether the input message is in XML or not. If it is XML, IMS Connect calls the XML Adapter. The XML Adapter converts the XML input message into the COBOL Application input data structure, sends it back to IMS Connect. IMS Connect processes the returned stream and forwards it on as an LLzz_trancode_data message to IMS so that the IMS Application can process the input message. The IMS COBOL Application returns an output message to IMS Connect. IMS Connect again calls the Adapter. The XML Adapter converts the IMS COBOL Application output message into an XML output message, sends it back to IMS Connect. IMS Connect processes the message from the XML Adapter and sends the XML output message to its Client.

The IMS COBOL Application is not aware of any message conversions. It receives messages in the same application specific format as it normally expects. The diagram below shows the message formats received by each component.


Information On Demand
IMS Version 10
IBM

XML Adapter Support - Considerations …

- Error messages are sent back to the remote client in XML tags
  - XML Converter Driver and XML Adapter errors
    - <XMLAdapterError> </XMLAdapterError>
      - If message can be converted to codepage
        <XMLAdapterError>HWS….. </XMLAdapterError>
        - Several new HWS… error messages have been added
      - Otherwise, XML Adapter Status Message (XASM)
        llzz*XADPST* rc rs
        where rc= 4 byte return code, rs= 4 byte reason code
  - IMS errors
    - <IMSDFSMessage>DFS..........</IMSDFSMessage>

Figure 12-61. XML Adapter Support - Considerations …   CMA01.0

Notes:

There are two types of runtime errors generated in the XML Adapter - the errors from WDz COBOL XML Converter Drivers, and the errors from the XML Adapter. Both types of error messages are returned from the XML Adapter in XML-tagged messages if the output codepage is available. IMS Connect does not distinguish between the two types of errors. If the output codepage is not available, the XML Adapter returns a XML Adapter Status Message (XASM). The XASM includes a return and reason code to indicate the type of error.

Besides these two types of errors, the XML Adapter also passes other error messages from/to IMS Connect, for example, the DFS error messages generated by IMS transactions. For any DFS messages, either errors or non-errors, the XML Adapter wraps the messages with XML tags and passes them back to the caller. The DFS message in XML format is defined as <IMSDFSMessage>DFS..........</IMSDFSMessage>.

XML Adapter Support - Considerations …

- IMS Connect Restart
  - Required when new XML Converters are to be changed
    - Adding or deleting an XML Converter does not require a restart

- BPE UPDATE command to refresh the Adapter exit

  F ICONN01, REF USRX NAME(XMLADAP)

    - Value:
      - Refresh request to bring in a new version of the ADAPTER plug-in user exit routine

- LE environment is required
  - XML Adapter brings up an LE environment during initialization

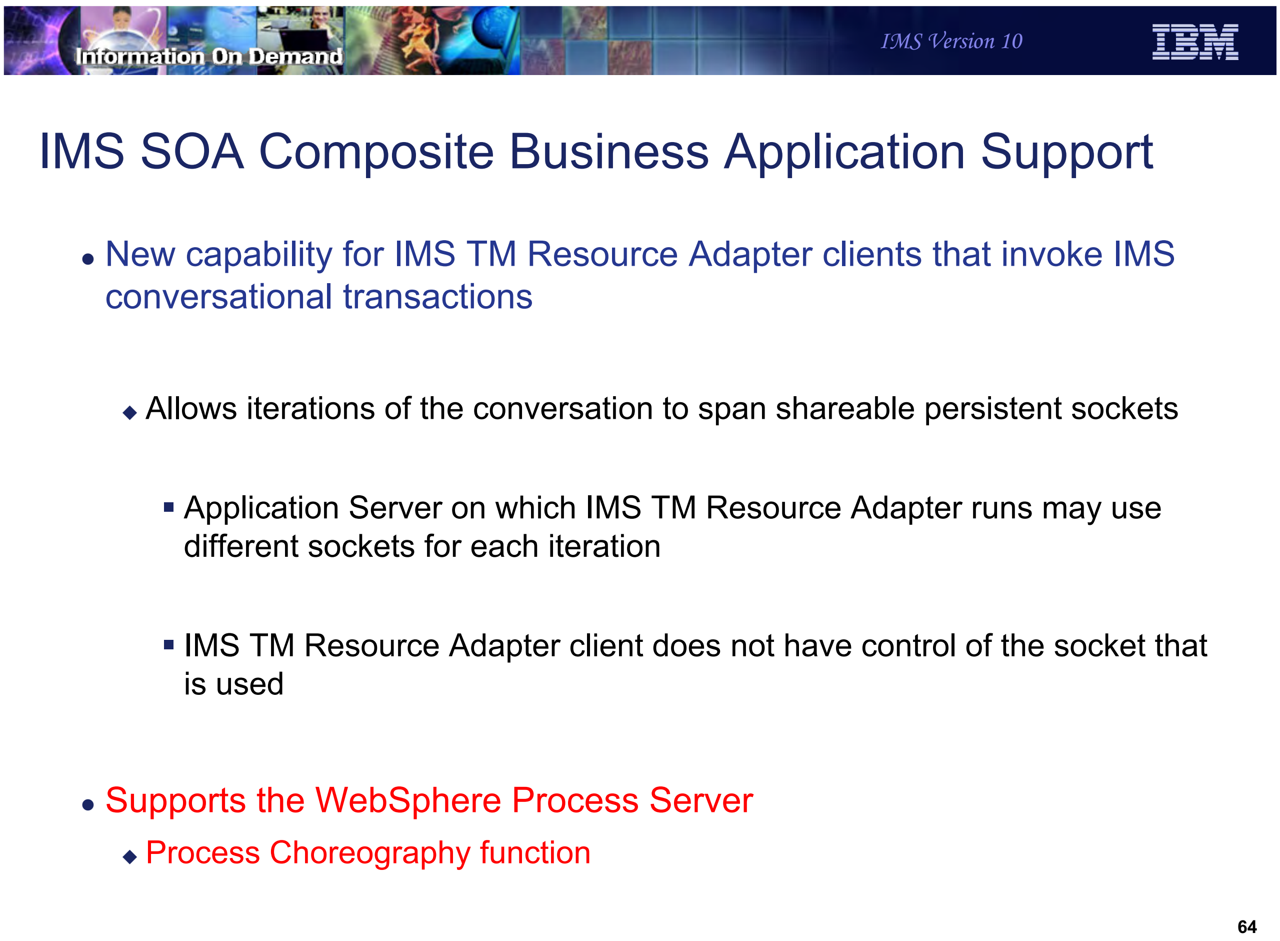
- Restriction
  - Single-segment messages only (multi-segment support is coming)

Figure 12-62. XML Adapter Support - Considerations … CMA01.0

Notes:

An IMS Connect restart is required to recognize any changes that have been made to the dataset containing the XML Converters. On the other hand, a refresh of the Adapter exit routine can be done while IMS Connect is active by using the BPE refresh capability.

Note that the COBOL XML Converter Driver requires an LE environment to run. The XML Adapter brings up an LE environment during its initialization processing.



Information On Demand

IMS Version 10

IMS SOA Composite Business Application Support

- New capability for IMS TM Resource Adapter clients that invoke IMS conversational transactions
  - Allows iterations of the conversation to span shareable persistent sockets
    - Application Server on which IMS TM Resource Adapter runs may use different sockets for each iteration
    - IMS TM Resource Adapter client does not have control of the socket that is used
- Supports the WebSphere Process Server
  - Process Choreography function

64

Figure 12-63. IMS SOA Composite Business Application Support CMA01.0

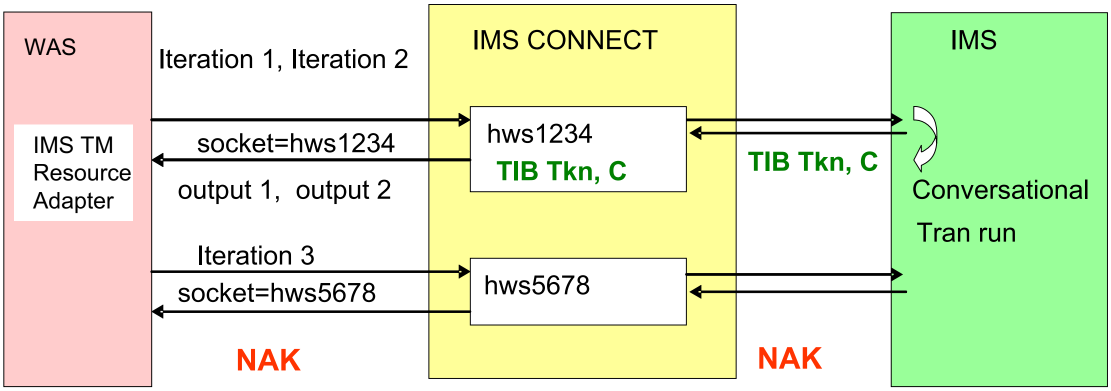
Notes:

IMS TM Resource Adapter (formerly IC4J) applications running in a WebSphere Application Server (WAS) can take advantage of either dedicated or shareable persistent sockets. A dedicated persistent socket remains dedicated to a particular user-specified clientid for commit-then-send CM0 interactions until released by a client application's request. Shareable persistent sockets, on the other hand, can be shared (serially reused) by multiple applications in the WAS server running either send-then-commit CM1 or commit-then-send CM0 interactions. For the latter type of socket, IC4J generates a clientid and does not allow user-specified clientid's.

Although the IMS TM Resource Adapter has supported interactions with IMS conversational transactions, IMS V10 enhances this capability by allowing the iterations between a conversation to span multiple shareable sockets.

IMS SOA Composite Business Application Support …

- The situation in previous releases



WAS

IMS TM Resource Adapter

Iteration 1, Iteration 2

socket=hws1234

output 1, output 2

Iteration 3

socket=hws5678

NAK

IMS CONNECT

hws1234

TIB Tkn, C

hws5678

TIB Tkn, C

NAK

IMS

Conversational

Tran run

TIB Tkn: 8-byte timestamp
C: conversational transaction indicator

Figure 12-64. IMS SOA Composite Business Application Support … CMA01.0

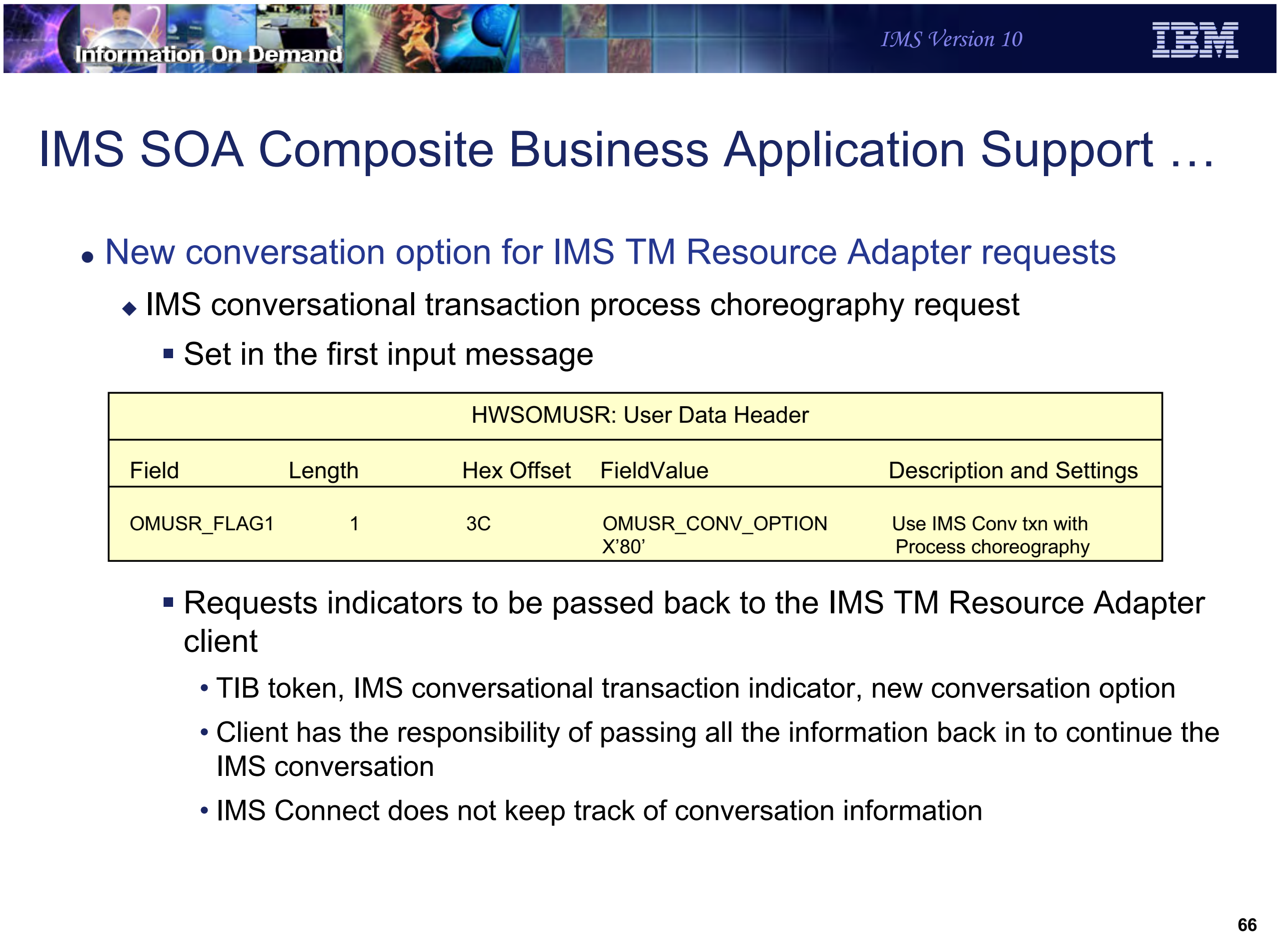
Notes:

This visual discusses the flow and a potential issue in the existing implementation.

The first request from the IMS TM Resource Adapter application invokes the first iteration of an IMS conversational transaction flow. The IMS TM Resource Adapter either uses an existing shareable persistent socket connection or generates a new one. For this example, the generated clientid is hws1234. At this point, neither the remote client nor IMS Connect are aware of whether the input is a conversational transaction. When the first reply message for this conversational iteration is created, IMS includes a conversational indicator (C) and a Transaction Instance Block token (TIB Tkn) in the message prefix that is sent back to IMS Connect. IMS Connect then saves these two pieces of conversational information in an internal control block associated with the clientid, hws1234. IMS Connect also sends the C indicator and the TIB Tkn with the output of the first iteration to the IMS TM Resource Adapter which returns them to the Java application.

When the second iteration of the same conversation begins, the message is routed to one of the shareable sockets in the pool. If the second iteration comes in through the same socket hws1234, IMS Connect uses the saved information in its internal control block to associate this conversation iteration with the previous one.

If the third iteration is routed to a different socket in the WAS pool, for example, HWS5678, IMS Connect cannot associate the request with the intended internal control block (hws1234) and the conversation information saved in it. IMS Connect, therefore, creates a different internal control block for hws5678 which has no knowledge of the existing conversation. When IMS receives the the input, it rejects the request with a NAK.


Information On Demand
IMS Version 10

IMS SOA Composite Business Application Support …

- New conversation option for IMS TM Resource Adapter requests
    - IMS conversational transaction process choreography request
        - Set in the first input message

HWSOMUSR: User Data Header

Field	Length	Hex Offset	FieldValue	Description and Settings
OMUSR_FLAG1	1	3C	OMUSR_CONV_OPTION X’80’	Use IMS Conv txn with Process choreography

- Requests indicators to be passed back to the IMS TM Resource Adapter client
    - TIB token, IMS conversational transaction indicator, new conversation option
    - Client has the responsibility of passing all the information back in to continue the IMS conversation
    - IMS Connect does not keep track of conversation information

Figure 12-65. IMS SOA Composite Business Application Support … CMA01.0

Notes:

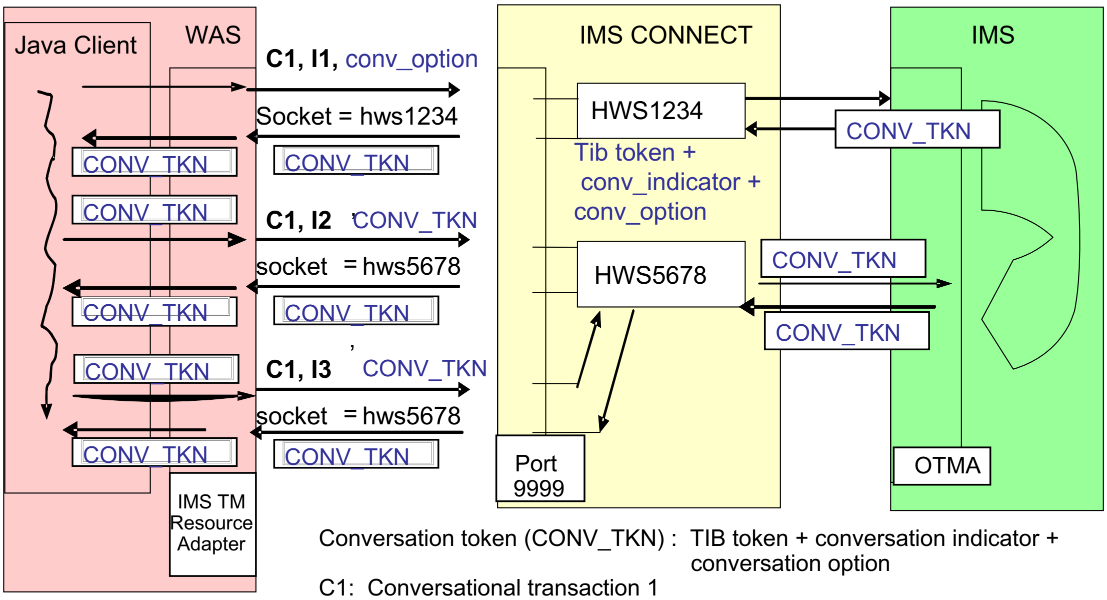
IMS Connect defines a new conversational option which is available to IMS TM Resource Adapter applications. Setting the OMUSR_CONV_OPTION to true (X’80’) invokes the function. If the function is not enabled, IMS Connect uses the existing conversational logic. Note that setting the OMSUSR_CONV_OPTION to false (X’00’) does not imply a non-conversational transaction.

IMS Connect creates the conversation token (the TIB token, Conversation indicator and conversation option flag), cleans up the internal control blocks, and passes the conversation token to the IMS TM Resource Adapter. The Adapter passes the information to the application which has to maintain the information to pass back on the next iteration of the conversation. This architecture allows the conversation to be continued by IMS Connect and passed to IMS on any of the sockets that are available to the remote client.

Information On Demand IMS Version 10 IBM

IMS SOA Composite Business Application Support …

- The new capability – allows the conversation to be resumed on a different connection



Java Client
WAS
IMS TM Resource Adapter
C1, I1, conv_option
Socket = hws1234
CONV_TKN
CONV_TKN
CONV_TKN
C1, I2 CONV_TKN
socket = hws5678
CONV_TKN
CONV_TKN
CONV_TKN
C1, I3 CONV_TKN
socket = hws5678
CONV_TKN
CONV_TKN
IMS CONNECT
HWS1234
Tib token + conv_indicator + conv_option
HWS5678
Port 9999
IMS
CONV_TKN
CONV_TKN
CONV_TKN
OTMA

Conversation token (CONV_TKN) : TIB token + conversation indicator + conversation option

C1: Conversational transaction 1

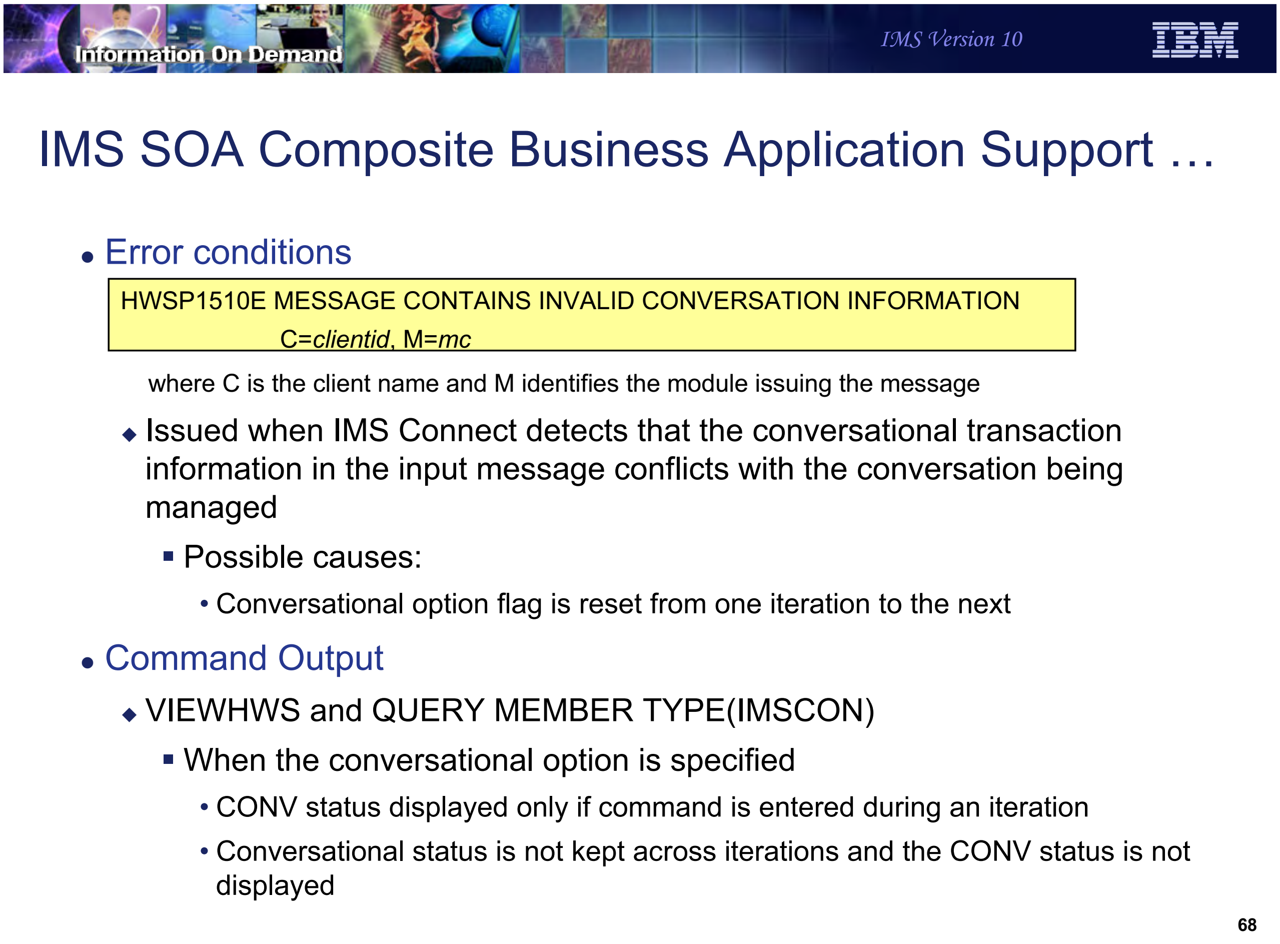
I1, I2, I3: Iterations of the Conversational transaction

67

Figure 12-66. IMS SOA Composite Business Application Support … CMA01.0

Notes:

This visual illustrates the new capability. When receiving a transaction reply from OTMA, IMS Connect checks to see whether the reply message contains the conversational transaction indicator, the Transaction Instance Block (TIB) token, and the new conversational option flag. If so, IMS Connect passes the conversation token back to the IMS TM Resource Adapter and cleans up all the internal knowledge about this conversational iteration. The Adapter then returns the conversation token back to the Java client. The Java client maintains responsibility for sending the conversation token back in the subsequent iterations of a conversation or in the SYNC_END_CONVERSATION interaction to end a conversation. The conversation token is composed of the 8-byte timestamp of the TIB token, the conversational transaction indicator, and the new conversational option indicator. The conversation token is passed back and forth between the Java client, the IMS TM Resource Adapter, IMS Connect, and OTMA. IMS Connect uses the conversation token to track the conversation within a single iteration.



Information On Demand
IMS Version 10

IMS SOA Composite Business Application Support …

- Error conditions

```
HWSP1510E MESSAGE CONTAINS INVALID CONVERSATION INFORMATION
               C=clientid, M=mc
```

where C is the client name and M identifies the module issuing the message

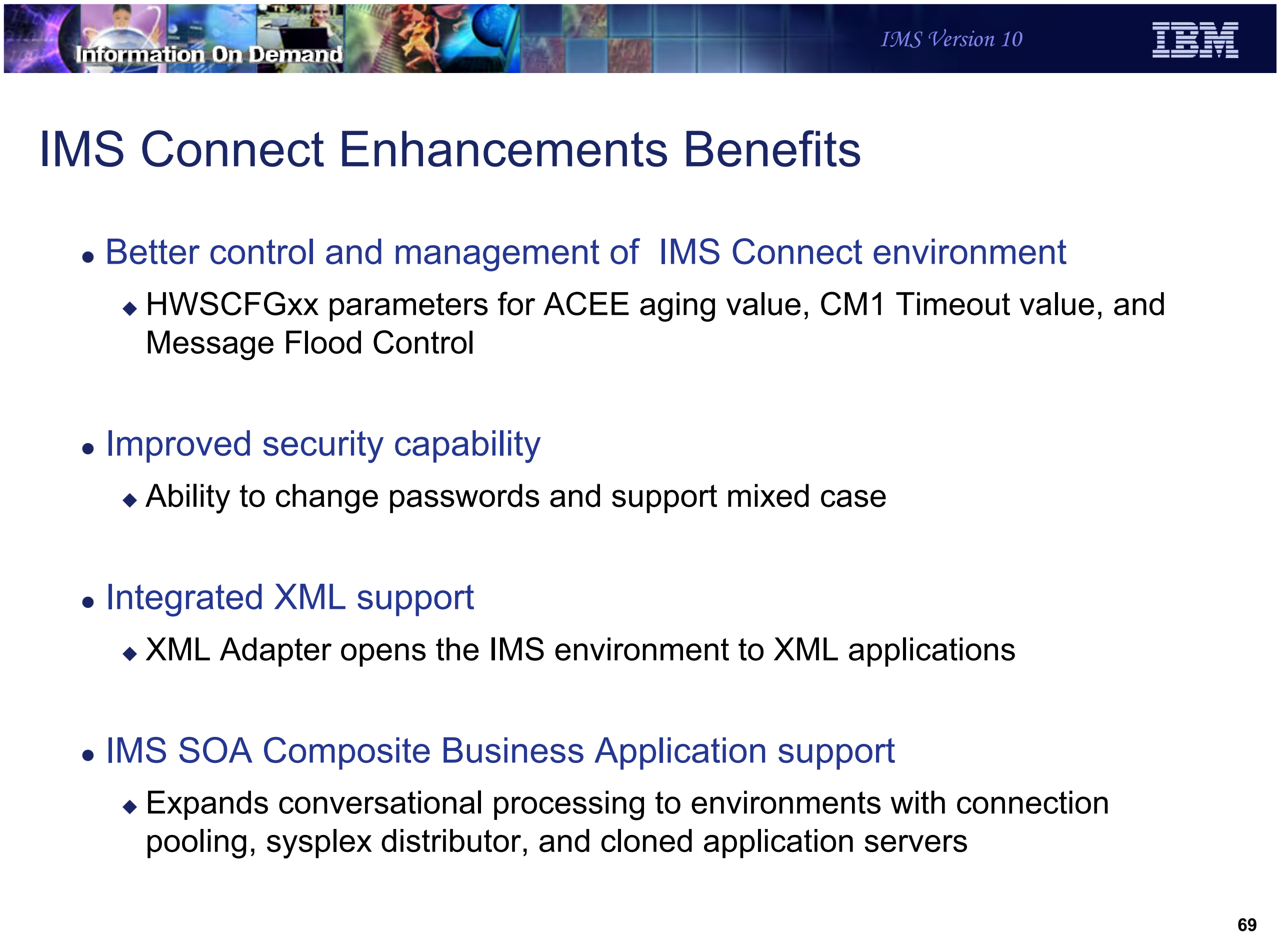
  - Issued when IMS Connect detects that the conversational transaction information in the input message conflicts with the conversation being managed
    - Possible causes:
      - Conversational option flag is reset from one iteration to the next
- Command Output
  - VIEWHWS and QUERY MEMBER TYPE(IMSCON)
    - When the conversational option is specified
      - CONV status displayed only if command is entered during an iteration
      - Conversational status is not kept across iterations and the CONV status is not displayed

Figure 12-67. IMS SOA Composite Business Application Support … CMA01.0

Notes:

If the new conversational option flag and other conversational information are set in an input message, IMS Connect assumes that this is a subsequent iteration of a conversational transaction and uses the conversational information to resume the conversation. If IMS Connect determines that the conversational transaction information from the input message does not conflict with any conversation it manages, IMS Connect continues processing the input. Otheriwse, IMS Connect issues the HWSP1510E error message. This scenario could happen when the conversational option flag is flip-flopped from FALSE in one conversational iteration to TRUE in a subsequent one.

The IMS Connect commands VIEWHWS and QUERY MEMBER TYPE(IMSCON) display the CONV status if a client is running an IMS conversational transaction. When the new conversational option is specified, IMS Connect no longer keeps track of the conversational status of the client across iterations of the conversational transaction. The output of the commands, therefore, will not be able to display the CONV status. Only when the display command is entered during the instance that a conversational iteration is in progress will the CONV status be displayed.


Information On Demand
IMS Version 10

IMS Connect Enhancements Benefits

- Better control and management of IMS Connect environment
  - HWSCFGxx parameters for ACEE aging value, CM1 Timeout value, and Message Flood Control
- Improved security capability
  - Ability to change passwords and support mixed case
- Integrated XML support
  - XML Adapter opens the IMS environment to XML applications
- IMS SOA Composite Business Application support
  - Expands conversational processing to environments with connection pooling, sysplex distributor, and cloned application servers

Figure 12-68. IMS Connect Enhancements Benefits CMA01.0

Notes:

Unit 13.IMS Integration Suite Enhancements

Information On Demand
IMS Version 10
IBM

IMS Integration Suite

- IMS TM Resource Adapter
  - IMS Connector for Java
- IMS SOAP Gateway
- IMS MFS Web Support
- IMS DB Resource Adapter
  - IMS JDBC Connector
- IMS DLIModel utility
- IMS XML DB

Figure 13-1. IMS Integration Suite CMA01.0

Notes:

The IMS™ Integration Suite is a collection of IMS middleware functions and tools that support your IMS on demand systems and your distributed IMS application environment. Some of these middleware functions and tools are generally available, whereas others might be part of an open beta program or technology preview program. All of these functions and tools require IMS Version 9 (or later), and are not supported for previous releases of IMS.

The IMS Integration Suite includes the following IMS middleware functions and tools:

IMS TM Resource Adapter - formally know as IMS Connector for Java

IMS SOAP Gateway

IMS MFS Web support

IMS DB Resource Adapter – formally known as IMS JDBC Connector

IMS DLIModel utility

IMS XML DB

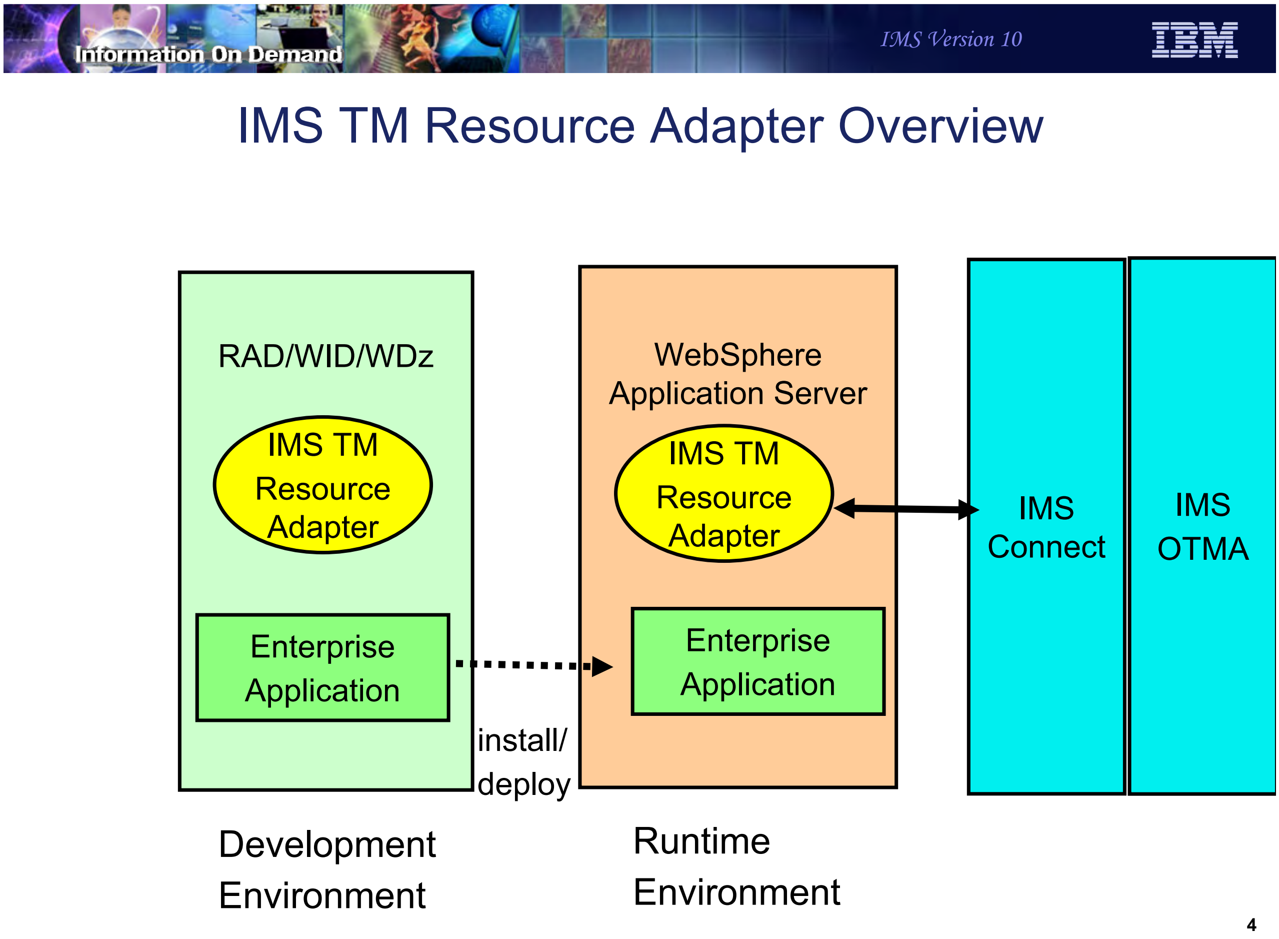
Information On Demand
IMS Version 10
IBM

IMS TM Resource Adapter

3

Figure 13-2. [No title] CMA01.0

Notes:



IMS TM Resource Adapter Overview

RAD/WID/WDz

IMS TM Resource Adapter

Enterprise Application

install/ deploy

WebSphere Application Server

IMS TM Resource Adapter

Enterprise Application

IMS Connect

IMS OTMA

Development Environment

Runtime Environment

Figure 13-3. [No title] CMA01.0

Notes:

IMS TM Resource Adapter is a J2EE Connector Architecture (J2C) resource adapter that supports IBM Java integrated development environments (IDEs) to create Java applications that can access IMS transactions. The IDEs import C, COBOL and PL/I source to generate the Java data bindings for the IMS transaction input/output message segments.

This diagram shows the dual functions of IMS TM Resource Adapter. To provide IDE support and to provide runtime environment support to access IMS applications using IMS Connect as the interface to IMS via OTMA.

The following IDEs are supported by IMS TM Resource Adapter:

Rational Application Developer (RAD)

Rational Software Architect

WebSphere Integration Developer (WID)

WebSphere Developer for zSeries (WDz)

WebSphere Studio Application Developer Integration Edition (WSAD-IE)

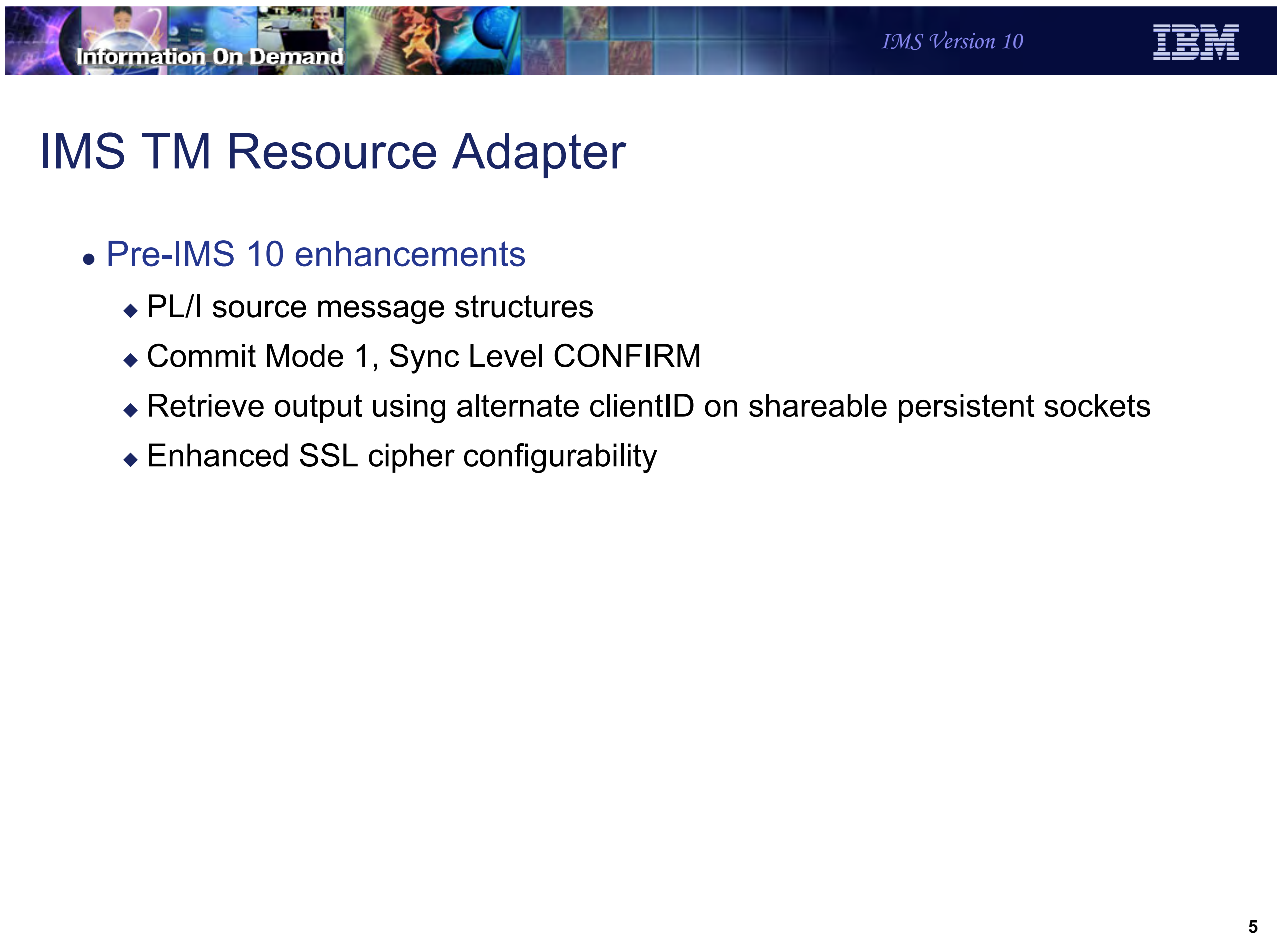
The IMS TM Resource Adapter also provides a plug-in for a J2EE application server by using the standardized system contracts. IMS TM Resource Adapter provides runtime support for WebSphere Application Server on distributed and z/OS platforms

A J2EE Application Server provides the runtime environment for application code.

WebSphere Process Server (WPS) – extends an application server by providing a

business process engine that executes business processes.

WID is the development tool used to create standards based business processes that are deployed to WPS.



Information On Demand
IMS Version 10

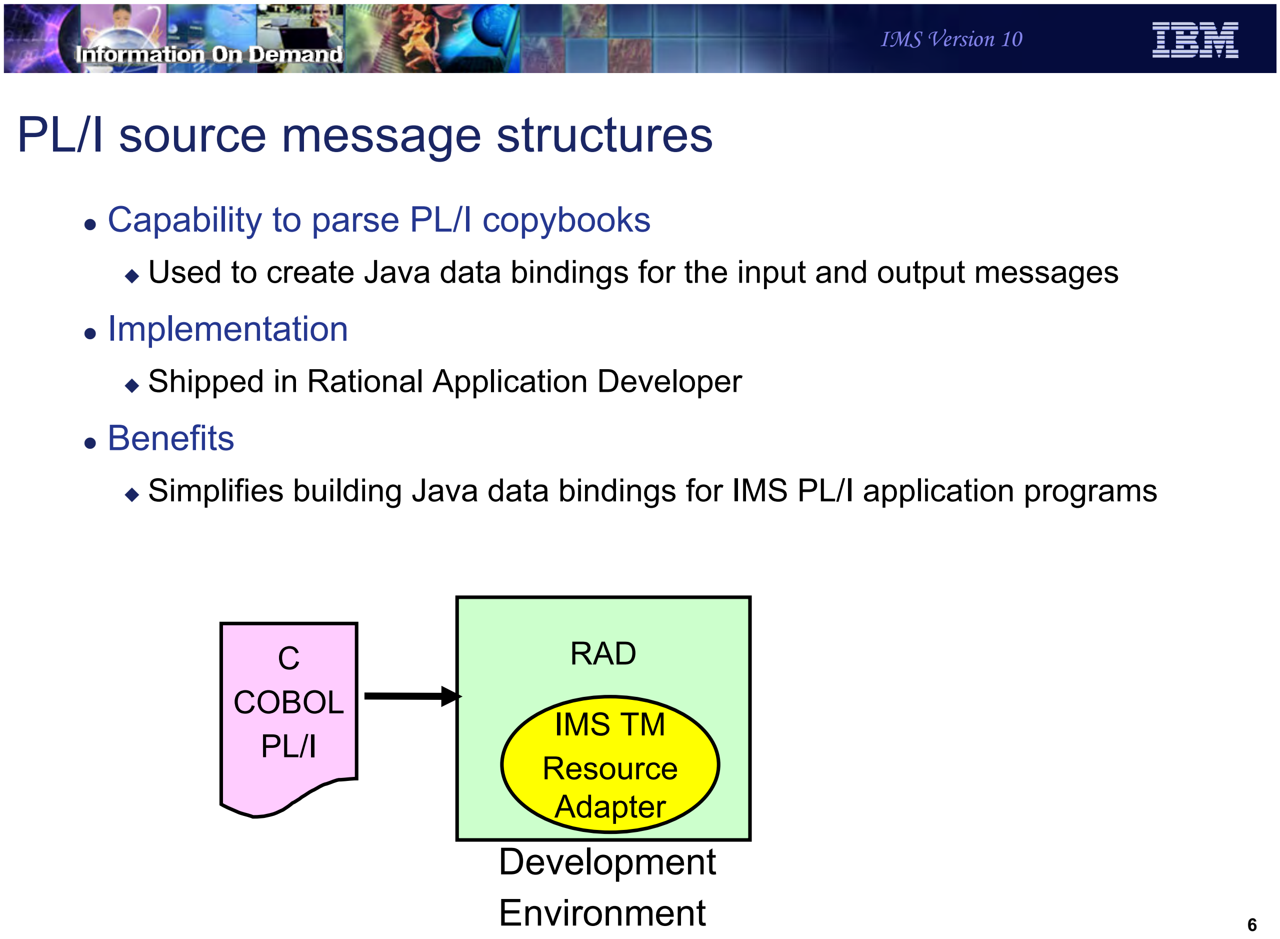
IMS TM Resource Adapter

- Pre-IMS 10 enhancements
  - PL/I source message structures
  - Commit Mode 1, Sync Level CONFIRM
  - Retrieve output using alternate clientID on shareable persistent sockets
  - Enhanced SSL cipher configurability

Figure 13-4. IMS TM Resource Adapter CMA01.0

Notes:

This is the list of enhancements IMS TM Resource Adapter provided after IMS V9 and before IMS V10.



Information On Demand

IMS Version 10

PL/I source message structures

- Capability to parse PL/I copybooks
  - Used to create Java data bindings for the input and output messages
- Implementation
  - Shipped in Rational Application Developer
- Benefits
  - Simplifies building Java data bindings for IMS PL/I application programs

C
COBOL
PL/I
RAD
IMS TM
Resource
Adapter
Development
Environment

6

Figure 13-5. PL/I source message structures CMA01.0

Notes:

RAD imports C , COBOL or PL/I definitions of the IMS transaction input and output messages and creates Java data bindings for the input and output messages.

Prior to this support the customer would have to create a COBOL copybook version of the IMS PL/I input/output message structure.

IMS TM Resource Adapter
- Commit Mode 1, Sync Level CONFIRM

- Shareable Persistent Socket

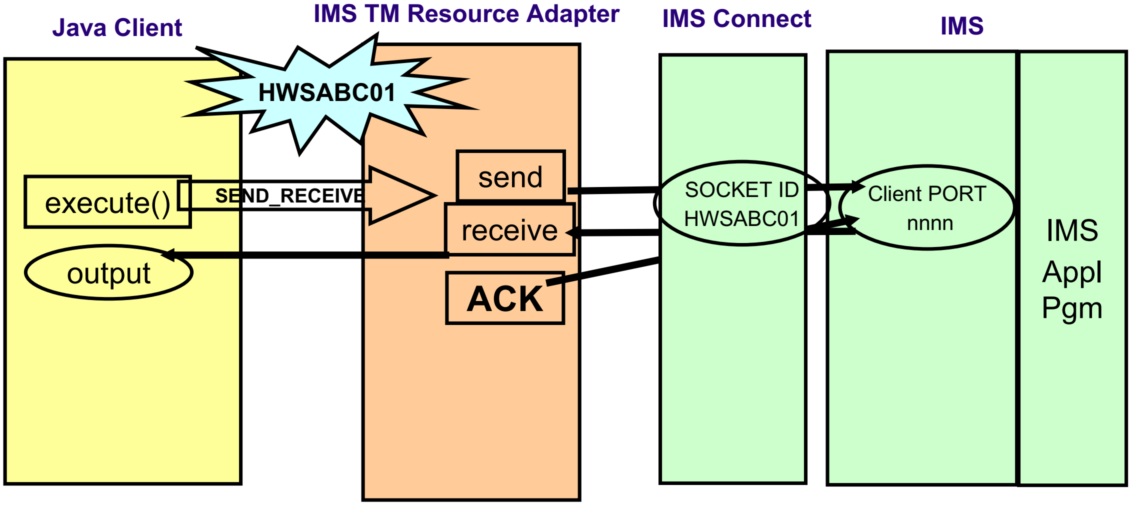

Java Client
IMS TM Resource Adapter
IMS Connect
IMS
HWSABC01
execute()
SEND_RECEIVE
send
receive
ACK
output
SOCKET ID
HWSABC01
Client PORT
nnnn
IMS
Appl
Pgm

Figure 13-6. [No title] CMA01.0

Notes:

Commit Mode 1 is only supported using a shareable persistent socket.

IMS TM Resource Adapter uses persistent socket connections to communicate with IMS Connect. A persistent socket connection prevents the overhead of opening and closing the socket for each use by an application. IMS TM Resource Adapter provides two types of persistent socket connections.

1.Dedicated persistent – a dedicated persistent socket connection is restricted to Commit Mode 0 (Commit then Send protocol) and the client application provides the clientid to be used as the socket identifier.

2.Shareable persistent – a shareable persistent socket connection supports Commit Mode 0 and Commit Mode 1 (Send then Commit protocol). However, The client application does not provide a clientid for a socket identifier. IMS TM Resource Adapter generates a clientid.

The support for CONFIRM provides the ability for IMS TM Resource Adapter to respond ACK to the output message which activates the IMS sync-point process. In this case the client application is not running under a Global Transaction.

This provides the ability for IMS TM Resource Adapter to notify IMS that it has received the IMS application output message without requiring a Global Transaction.

In this example the Java client is using a shareable persistent socket Commit Mode 1 Sync Level CONFIRM. IMS TM Resource Adapter generates HWSABC01 for a socket ID and sends the input message to the IMS application program via IMS Connect and OTMA. To provide a message queue anchor for the Commit Mode 1 output message OTMA creates a transaction pipe (TPIPE) and will use the TCP/IP generated Client Port number provided by IMS Connect for the TPIPE name. When IMS TM Resource Adapter receives the output message from IMS Connect it responds with an ACK. IMS Connect will send the ACK to OTMA to activate the IMS sync point process.

Note the Java Client application cannot respond with an ACK.

Information On Demand
IMS Version 10

Commit Mode 1, Sync Level CONFIRM

- Implementation
  - Shipped in Rational Application Developer
- Considerations
  - IMS TM Resource Adapter sends ACK
  - CM1 (Send-then-Commit) Time-out Control
- Benefits
  - IMS TM Resource Adapter can notify IMS that it has received the IMS application output message without requiring a Global Transaction
  - Reduce the need of a compensating transaction
  - IMS TM Resource Adapter compatible with APPC sync level support

Figure 13-7. Commit Mode 1, Sync Level CONFIRM CMA01.0

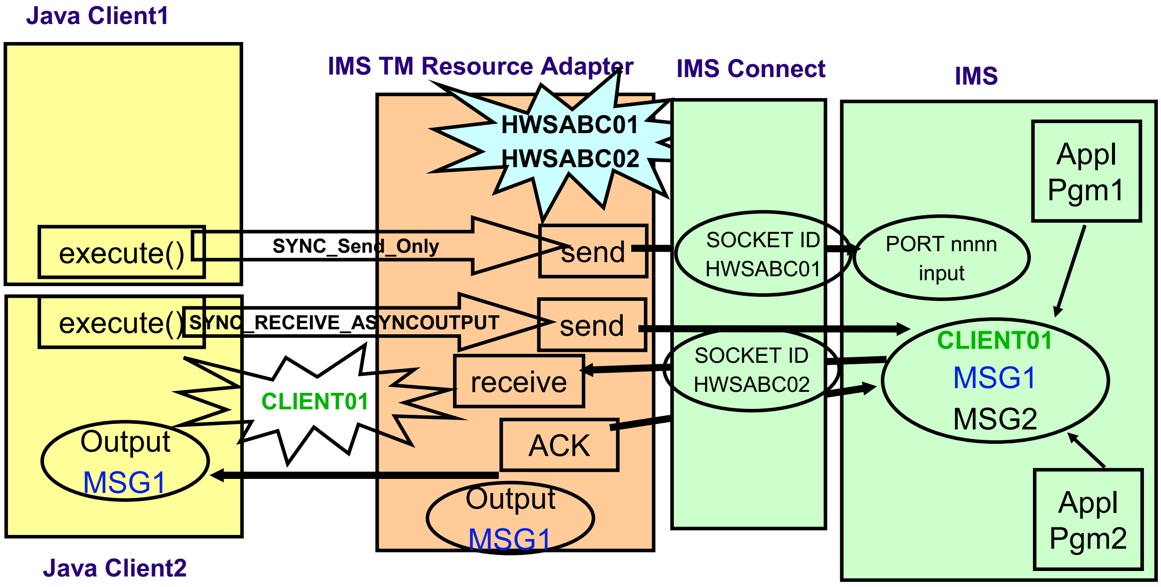
Notes:

Commit Mode One Synclevel None has the potential to discard an output message without a ROLLBack of the database in the event IMS Connect is not able to deliver the output message to IMS TM Resource Adapter. As a result, a compensation transaction may have to be executed to reverse the database changes. Adding the Commit Mode One Synclevel Confirm could reduce the necessity of using a compensation transaction because an ACK will be performed by IMS TM Resource Adapter.

APPC provides CM1 sync levels NONE,CONFIRM and SYNCPT. Customers who have APPC client applications accessing IMS using CM1 sync level CONFRIM can now provide IMS TM Resource Adapter clients with the same sync levels.

Information On Demand IMS Version 10 IBM

IMS TM Resource Adapter - Retrieve output using alternate clientID


Java Client1
IMS TM Resource Adapter
IMS Connect
IMS
HWSABC01
HWSABC02
Appl Pgm1
execute()
SYNC_Send_Only
send
SOCKET ID HWSABC01
PORT nnnn input
execute()
SYNC_RECEIVE_ASYNCOUTPUT
send
SOCKET ID HWSABC02
CLIENT01
MSG1
MSG2
receive
CLIENT01
ACK
Output MSG1
Output MSG1
Appl Pgm2
Java Client2

"Single wait " and "Single no wait"

9

Figure 13-8. [No title] CMA01.0

Notes:

For Shareable Persistent Sessions the client application can not specify a clientID since it is generated by IMS TM Resource Adapter. This can prevent the retrieving of failed IOPCB output or ALTPCB output messages. For the current IMS TM Resource Adapter support the message originating client application can only request asynchronous output while the generated id for the socket remains connected.

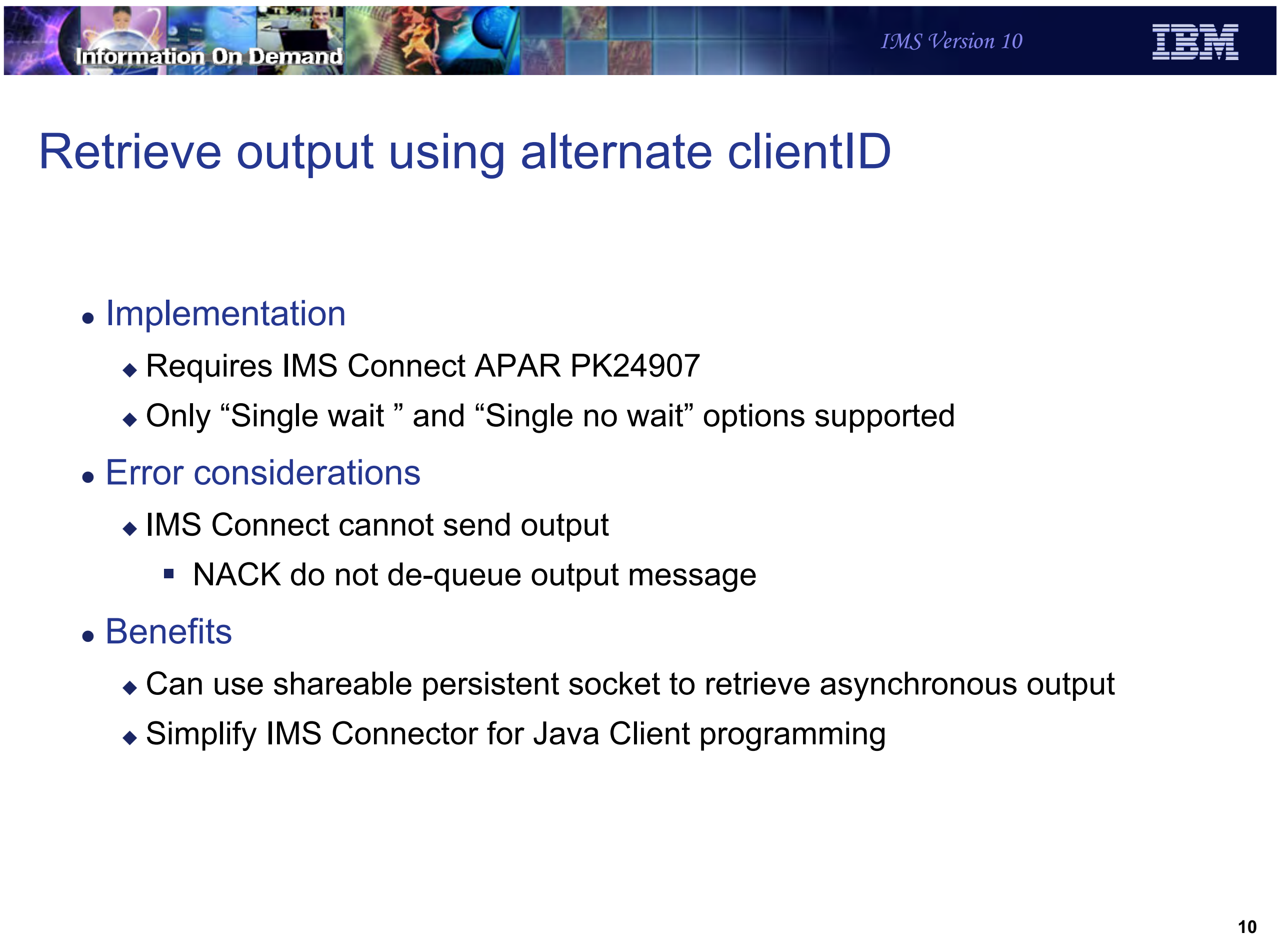
In this example Java Client1 submits a Send_Only input message to IMS. IMS TM Resource Adapter generates HWSABC01 Socket ID and sends the input message to IMS Connect. The IMS application program does not respond to the I/O PCB but does insert a message to ALTPCB using CLIENT01 for a TPIPE name. Java Client2 submits a SYNC_RECEIVE_ASYNCOUTPUT to IMS TM Resource Adapter to obtain the output queued to CLIENT01 TPIPE. IMS TM Resource Adapter generates HWSABC02 socket ID and uses the new "Resume tpipe with alternate clientid“ protocol to retrieve the output message.

Notice with this protocol a different Java Client application can be used to retrieve asynchronous output generated from processing the originating java client application request.

IMS TM Resource Adapter supports the “Single wait ” and “Single no wait” options.

For the Single message with wait option only one message is returned for a RESUME TPIPE request. If no message is present OTMA waits for a message to arrive and then sends that single message to IMS Connect. The timer set on RESUME TPIPE can expire before a message is returned to IMS Connect. If that occurs, IMS Connect will NAK the message when sent to IMS Connect by OTMA.

For Single message nowait only one message is returned following the RESUME TPIPE. If no message is present OTMA does not wait for a message and the IMS Connect timer causes a timeout to occur based on the timeout value specified.



Information On Demand
IMS Version 10
IBM

Retrieve output using alternate clientID

- Implementation
  - Requires IMS Connect APAR PK24907
  - Only “Single wait ” and “Single no wait” options supported
- Error considerations
  - IMS Connect cannot send output
    - NACK do not de-queue output message
- Benefits
  - Can use shareable persistent socket to retrieve asynchronous output
  - Simplify IMS Connector for Java Client programming

Figure 13-9. Retrieve output using alternate clientID CMA01.0

Notes:

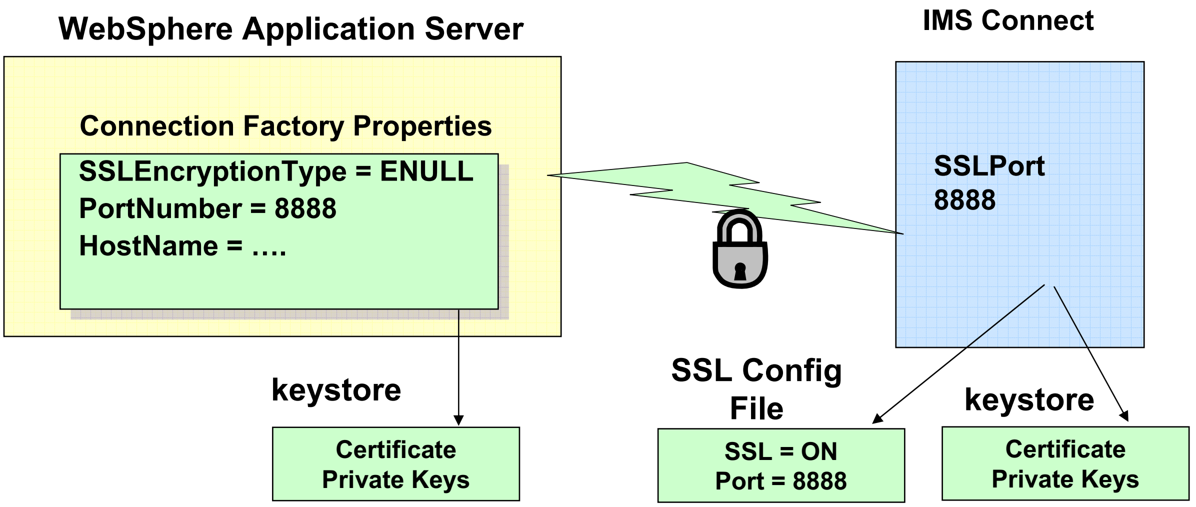
IMS Connect APAR PK24907 supports a new protocol "Resume tpipe with alternate clientid“ that provides the IMS TM Resource Adapter client application the ability to specify an alternate clientid in the SYNC_RECEIVE_ASYNCOUTPUT interaction.

If IMS Connect cannot send the output message to IMS TM Resource Adapter then IMS Connect will respond NACK to OTMA. OTMA will not de-queue the output message.

Information On Demand
IMS Version 10
IBM

IMS TM Resource Adapter

- Enhanced SSL cipher configurability
  - ENULL value for the SSLEncryptionType property.



WebSphere Application Server

Connection Factory Properties

SSLEncryptionType = ENULL
PortNumber = 8888
HostName = ….

keystore

Certificate
Private Keys

IMS Connect

SSLPort
8888

SSL Config File

SSL = ON
Port = 8888

keystore

Certificate
Private Keys

Figure 13-10. IMS TM Resource Adapter CMA01.0

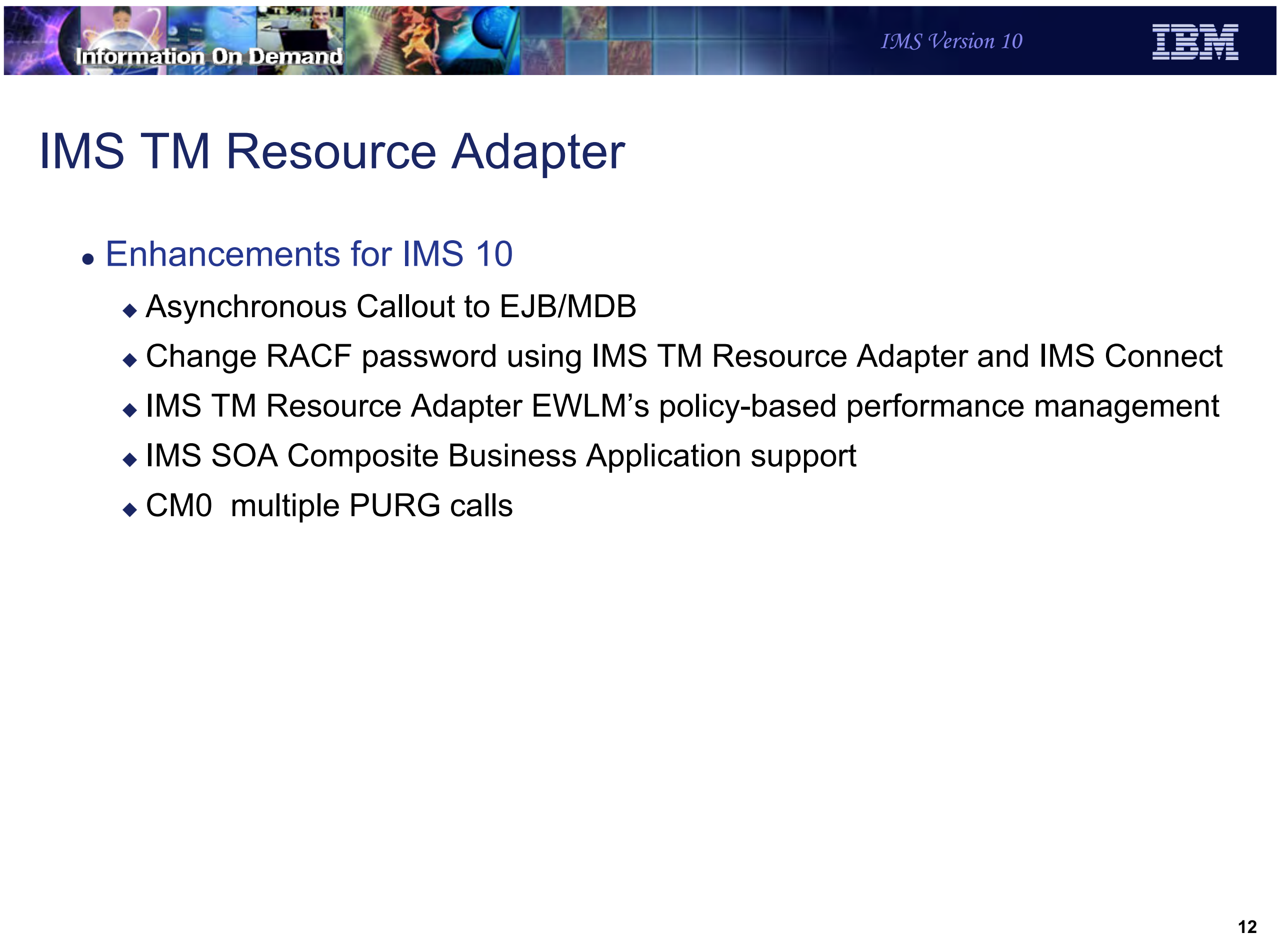
Notes:

The Secure Socket Layer (SSL) protocol ensures that the transfer of sensitive information over the Internet is secure.

Currently, IMS TM Resource Adapter allows customers to specify only “strong” or “weak” as the SSLEncryptionType for all connections created by an IMS connection factory. Enhanced SSL cipher configurability allows customers to specify a new value, “ENULL” for SSLEncryptionType. When “ENULL” is specified, IMS TM Resource Adapter will attempt to use a cipher spec whose name contains the string “NULL”

Null encryption will allow for authentication to take place during the SSL handshaking process as is currently the case. Once the handshaking process for a socket has completed including authentication as required, all messages will flow in the clear over that socket.

An example is the passing of non-confidential data between IMS TM Resource Adapter and IMS Connect, both of which are located behind a firewall.



IMS TM Resource Adapter

- Enhancements for IMS 10
  - Asynchronous Callout to EJB/MDB
  - Change RACF password using IMS TM Resource Adapter and IMS Connect
  - IMS TM Resource Adapter EWLM’s policy-based performance management
  - IMS SOA Composite Business Application support
  - CM0 multiple PURG calls

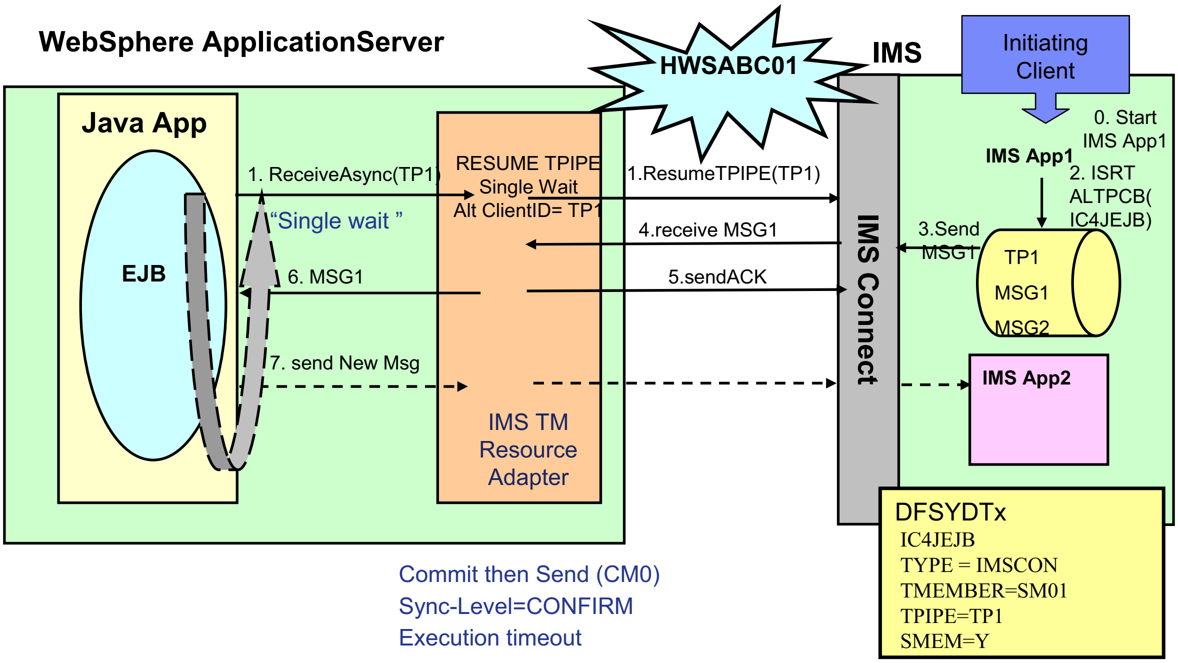
Figure 13-11. IMS TM Resource Adapter CMA01.0

Notes:

This is the list of enhancements IMS TM Resource Adapter provides for IMS V10.

Information On Demand IMS Version 10 IBM

Asynchronous Callout to EJB


WebSphere ApplicationServer
HWSABC01
IMS
Initiating Client
Java App
EJB
1. ReceiveAsync(TP1)
“Single wait ”
6. MSG1
7. send New Msg
RESUME TPIPE
Single Wait
Alt ClientID= TP1
IMS TM Resource Adapter
1.ResumeTPIPE(TP1)
4.receive MSG1
5.sendACK
IMS Connect
0. Start IMS App1
IMS App1
2. ISRT ALTPCB(IC4JEJB)
3.Send MSG1
TP1
MSG1
MSG2
IMS App2
DFSYDTx
IC4JEJB
TYPE = IMSCON
TMEMBER=SM01
TPIPE=TP1
SMEM=Y

Commit then Send (CM0)
Sync-Level=CONFIRM
Execution timeout

Figure 13-12. Asynchronous Callout to EJB CMA01.0

Notes:

IMS TM Resource Adapter has been enhanced to support access to Enterprise Java Beans (EJB) from an IMS application program. In this scenario an EJB uses the SYNC_RECEIVE_ASYNC_OUTPUT_SINGLE_WAIT (1) alt-clientID=TP1 to request a message queued to TP1. IMS TM Resource Adapter uses the IMS Connect "Resume tpipe with alternate clientid“ protocol to wait for a message to retrieve (1). IMS Connect receives the request and notifies OTMA to send a message. An already scheduled (0) IMS Application program does an insert to an ALTPCB for a destination defined in an OTMA destination routing descriptor (2). The OTMA destination routing descriptor is used to define the OTMA TPIPE for the en-queue of the message (TP1). OTMA sends the message to IMS Connect (3) and IMS Connect will return the message to IMS TM Resource Adapter (4). IMS TM Resource Adapter receives the message and notifies IMS Connect with an ACK(5). IMS Connect will forward the ACK to OTMA and the message will be de-queued.

IMS TM Resource Adapter receives the message and returns it to the EJB (6).

An optional step is the EJB creates a response to the message request. IMS TM Resource Adapter sends the response to IMS (7). This results in the scheduling of an IMS Application program to process the response.

Asynchronous Callout to MDB

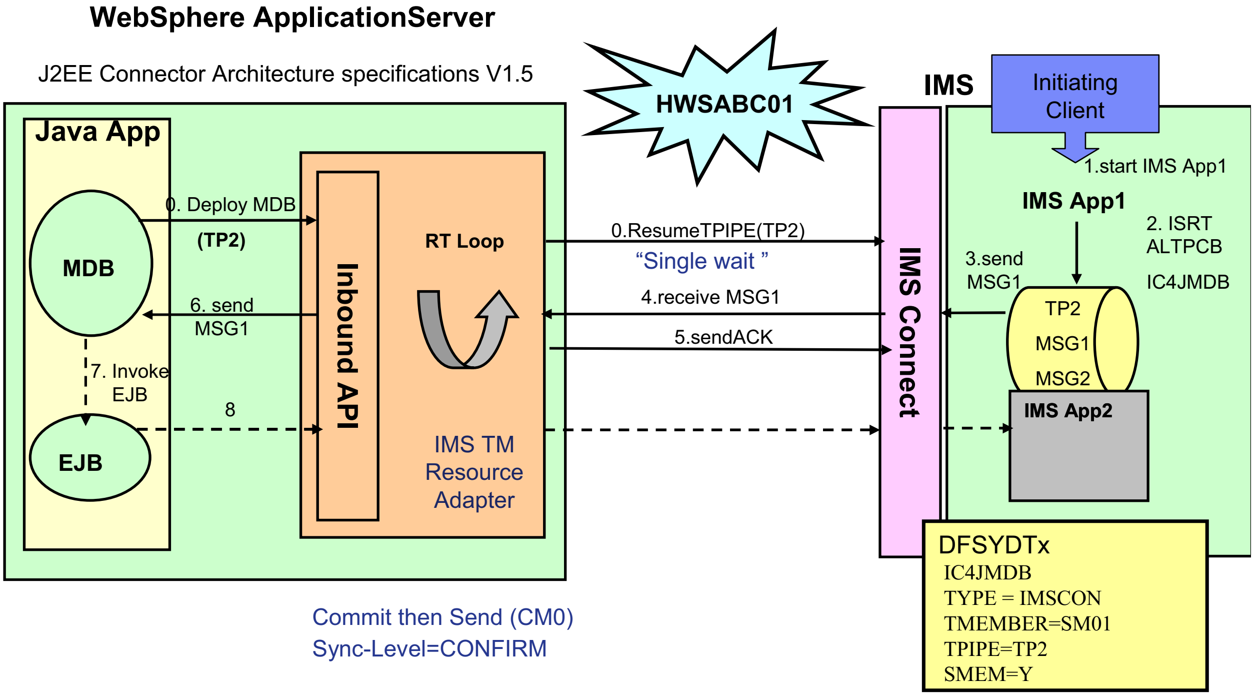

WebSphere ApplicationServer
J2EE Connector Architecture specifications V1.5
Java App
MDB
0. Deploy MDB
(TP2)
6. send MSG1
7. Invoke EJB
EJB
8
Inbound API
RT Loop
IMS TM Resource Adapter
HWSABC01
0.ResumeTPIPE(TP2)
"Single wait"
4.receive MSG1
5.sendACK
IMS Connect
IMS
Initiating Client
1.start IMS App1
IMS App1
2. ISRT ALTPCB
IC4JMDB
3.send MSG1
TP2
MSG1
MSG2
IMS App2
DFSYDTx
IC4JMDB
TYPE = IMSCON
TMEMBER=SM01
TPIPE=TP2
SMEM=Y
Commit then Send (CM0)
Sync-Level=CONFIRM

Figure 13-13. Asynchronous Callout to MDB CMA01.0

Notes:

IMS TM Resource Adapter also supports access to Message Driven Beans (MDB) from an IMS application program. The J2EE Connector Architecture specifications V1.5 describes how an enterprise application can access a J2EE application environment. This is known as Inbound processing. IMS TM Resource Adapter is providing Inbound processing for IMS application programs. From an IMS application perspective this is Asynchronous Callout processing.

In this scenario IMS TM Resource Adapter uses the IMS Connect "Resume tpipe with alternate clientid“ protocol to wait for a message to retrieve (0).

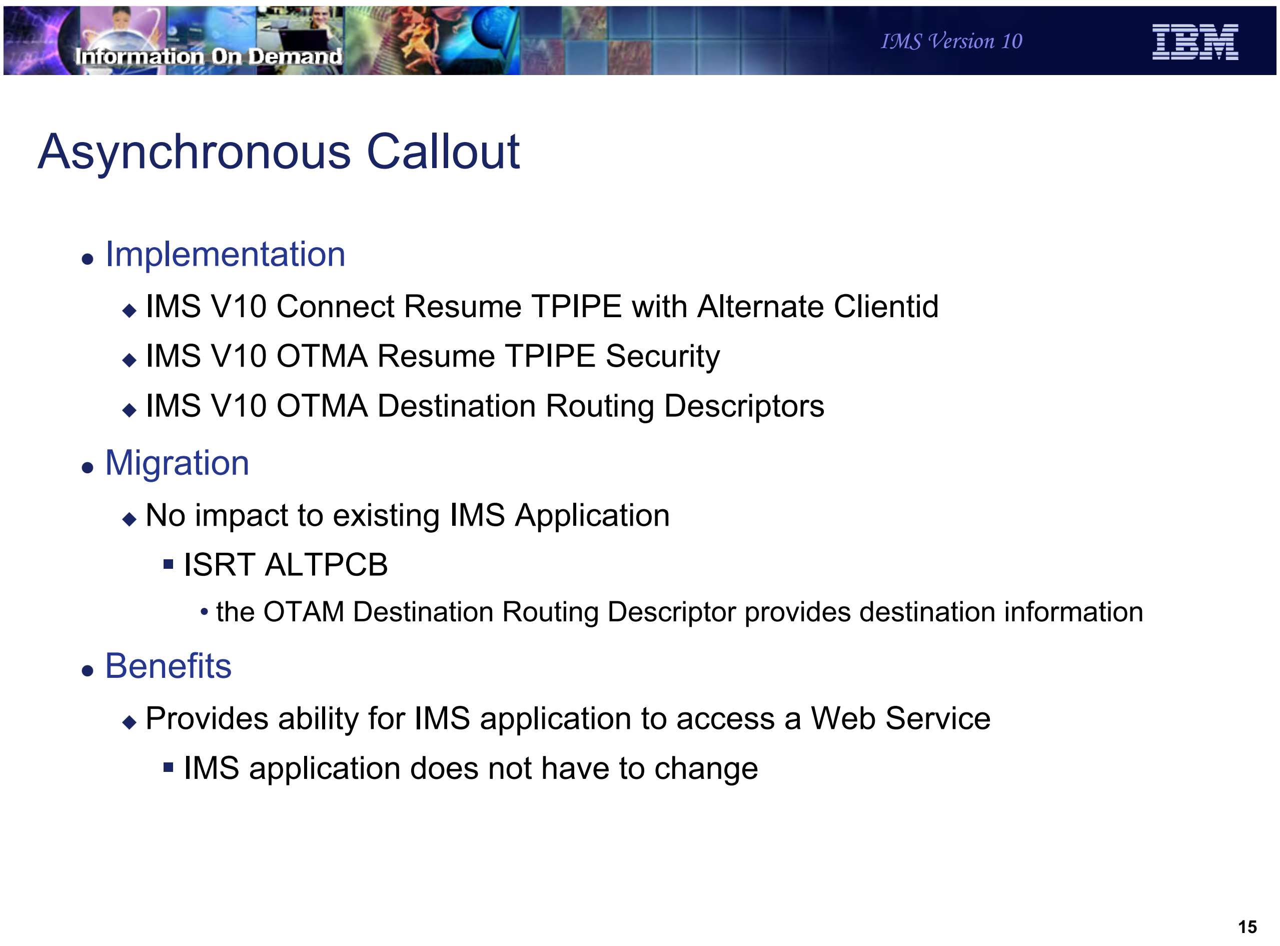
An already scheduled(1) IMS Application program does an insert to an ALTPCB for a destination defined in an OTMA Descriptor (2). The OTMA Descriptor is used to define the OTMA TPIPE for en-queue of the message.

OTMA sends the message to IMS Connect (3) and IMS Connect will return the message to IMS TM Resource Adapter (4). IMS TM Resource Adapter receives the message and notifies IMS Connect with an ACK(5). IMS Connect will forward the ACK to OTMA and the message will be de-queued.

An MDB running inside WebSphere Application server will use the standard JCA 1.5 interfaces to listen to callout request from IMS applications. IMS TM Resource Adapter receives the message send it to the MDB (6).

The MDB will invoke the EJB (7) to process the message.

An optional step is the EJB creates a response to the message request. IMS TM Resource Adapter sends the response to IMS (8). This results in the scheduling of an IMS Application program to process the response.

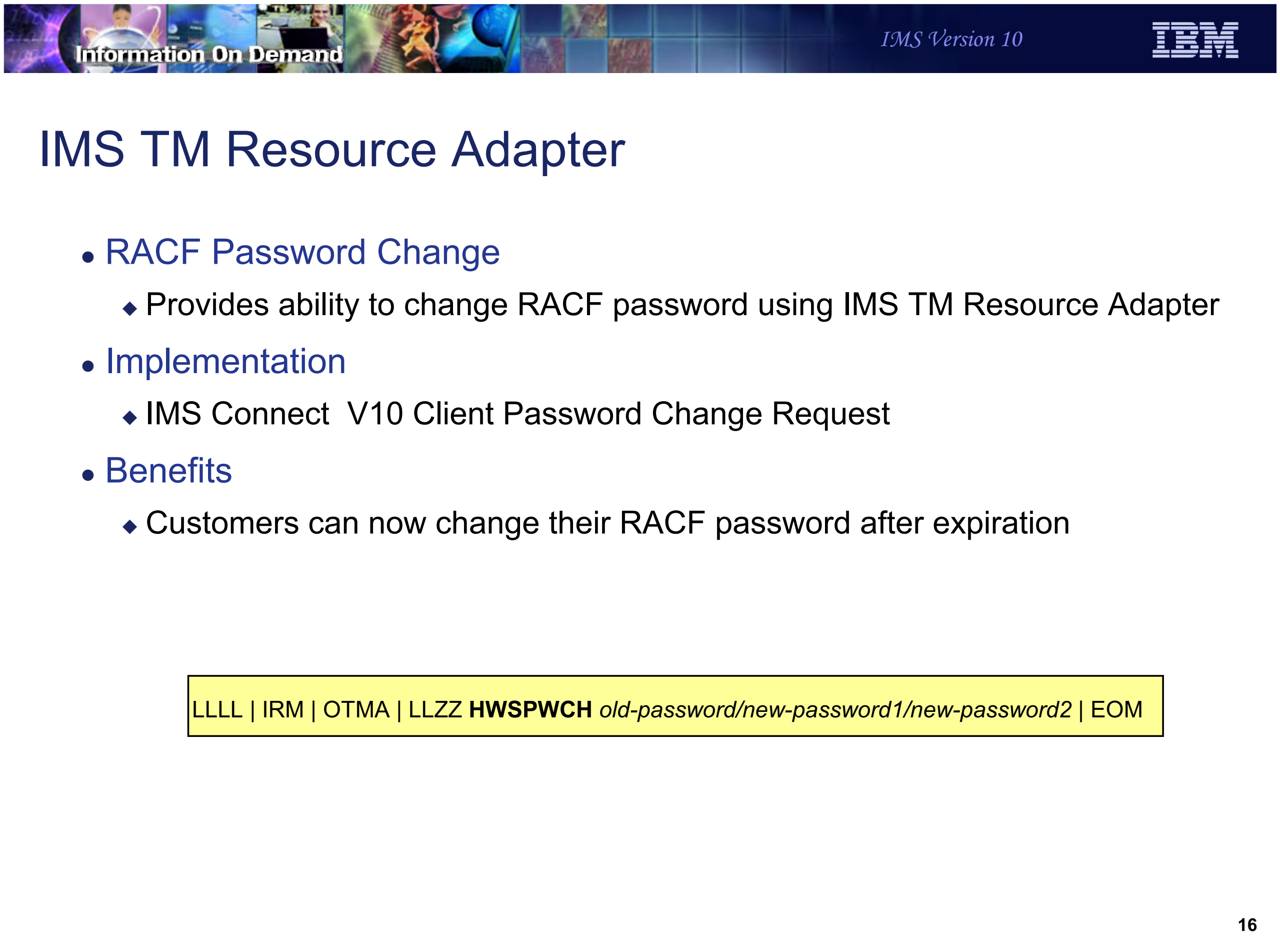


Asynchronous Callout

- Implementation
  - IMS V10 Connect Resume TPIPE with Alternate Clientid
  - IMS V10 OTMA Resume TPIPE Security
  - IMS V10 OTMA Destination Routing Descriptors
- Migration
  - No impact to existing IMS Application
    - ISRT ALTPCB
      - the OTAM Destination Routing Descriptor provides destination information
- Benefits
  - Provides ability for IMS application to access a Web Service
    - IMS application does not have to change

Figure 13-14. Asynchronous Callout CMA01.0

Notes:


Information On Demand
IMS Version 10
IBM

IMS TM Resource Adapter

- RACF Password Change
  - Provides ability to change RACF password using IMS TM Resource Adapter
- Implementation
  - IMS Connect V10 Client Password Change Request
- Benefits
  - Customers can now change their RACF password after expiration

LLLL | IRM | OTMA | LLZZ **HWSPWCH** *old-password/new-password1/new-password2* | EOM

16

Figure 13-15. IMS TM Resource Adapter CMA01.0

Notes:

IMS TM Resource Adapter has been enhanced to change a RACF password during interactions with IMS Connect.

When a J2EE application requires access to an IMS transaction that is password protected the RACF password may have expired. Using this facility the J2EE application can prompt the client for an new password and provide the new password to facilitate completion of the business process.

Information On Demand
IMS Version 10
IBM

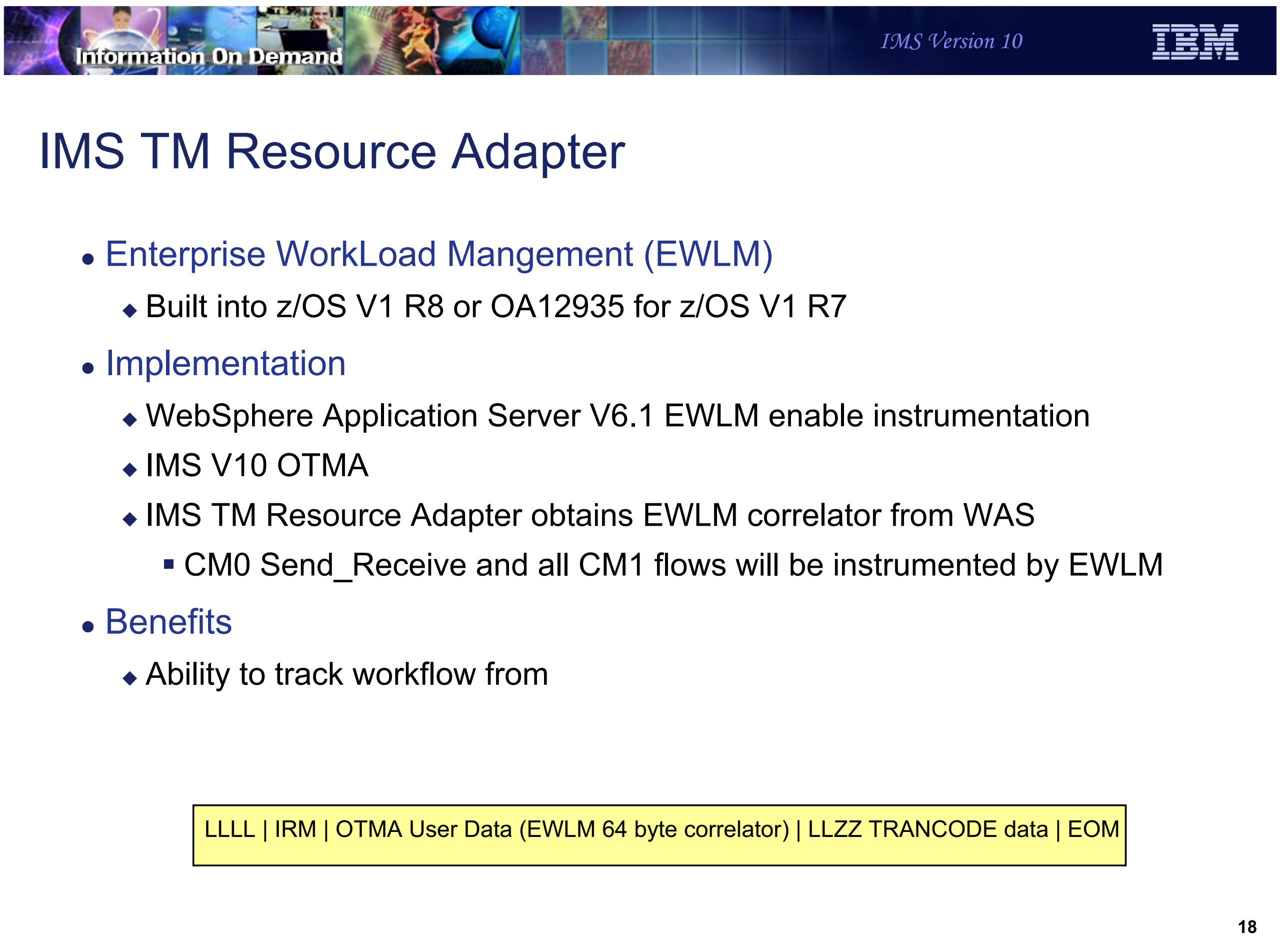
IMS TM Resource Adapter

- Client sample

```
package sample.ims.hwspwcdh;
     import com.ibm.connector2.ims.ico.*;

     public class HWSPWCHIMSConnect
     {
        public static void main(String[] args)
        {
           try
           {
              INMSG input = new INMSG();
              input.setInll((short) 31);
              input.setInzz((short) 0);
              input.setIncmd("HWSPWCH CURPSWD/NEWPSWD/NEWPSWD");
              HWSPWCHIMSProxy proxy = new HWSPWCHCmdIMSProxy();
              proxy.setINMSG(input);
              // Run the IMS service.
              proxy.execute();

     }
```

Figure 13-16. IMS TM Resource Adapter CMA01.0

Notes:


Information On Demand
IMS Version 10
IBM

IMS TM Resource Adapter

- Enterprise WorkLoad Mangement (EWLM)
  - Built into z/OS V1 R8 or OA12935 for z/OS V1 R7
- Implementation
  - WebSphere Application Server V6.1 EWLM enable instrumentation
  - IMS V10 OTMA
  - IMS TM Resource Adapter obtains EWLM correlator from WAS
    - CM0 Send_Receive and all CM1 flows will be instrumented by EWLM
- Benefits
  - Ability to track workflow from

LLLL | IRM | OTMA User Data (EWLM 64 byte correlator) | LLZZ TRANCODE data | EOM

Figure 13-17. IMS TM Resource Adapter CMA01.0

Notes:

The Enterprise Workload Management (EWLM) support requires the EWLM CICS/IMS Support that is built into z/OS Version 1 Release 8 or provided in APAR OA12935 for z/OS Version 1 Release 7.

IMS TM Resource Adapter obtains the EWLM correlator from WebSphere Application Server, then passes the correlator to IMS by inserting the correlator into the OTMA prefix for the transaction. When the transaction returns to IMS TM Resource Adapter after processing is complete, IMS TM Resource Adapter reports the status of the transaction to WAS.

If the incoming transaction does not contain an EWLM correlator, the existing function is unchanged.

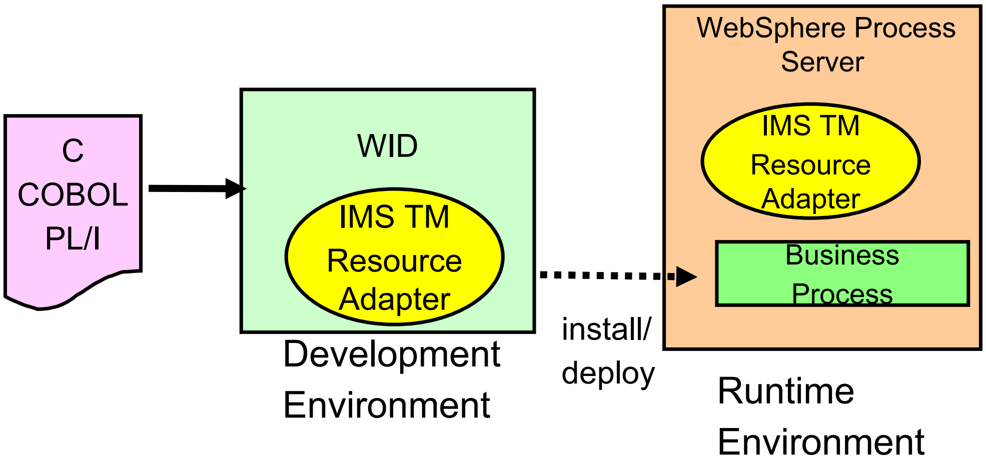
CM0 Send Only flows will not be instrumented by EWLM. IMS TM Resource Adapter does not receive a correlator for CM0 send only transactions. Therefore, the transaction will be sent to IMS with the correlator field zeroed out, and IMS will call the existing non-EWLM version of the WLM services

If the correlator is provided then IMS TM Resource Adapter will pass it to IMS. If the correlator is not provided then the OTMA prefix field will be set to zero.

If an EWLM enabled transaction is routed from a V10 IMS to a backlevel IMS, the transaction passes through the backlevel system with the EWLM data unchanged and the existing WLM function is unchanged on the backlevel system. No APAR is required for prior versions of IMS.

IMS TM Resource Adapter

- IMS SOA Composite Business Application support
  - Provides multi-connection IMS Conversational Transaction support
    - Only available to IMS TM Resource Adapter clients
    - Supports WebSphere Process Server Business Process Flows



C
COBOL
PL/I

WID
IMS TM Resource Adapter
Development Environment

install/ deploy

WebSphere Process Server
IMS TM Resource Adapter
Business Process
Runtime Environment

Figure 13-18. IMS TM Resource Adapter CMA01.0

Notes:

IMS SOA Composite Business Application support

- Migration
  - Replace Connection ID with Conversation Token
  - Only supports CM1 Synch level NONE or CONFIRM
  - IMS TM Resource Adapter
    - Used to pass IMS conversation token
- Considerations
  - IMS TM Resource Adapter Client must manage IMS conversation token
  - Supports only one IMS
- Benefits
  - Provides multiple connections IMS conversational transaction support

Figure 13-19. IMS SOA Composite Business Application support CMA01.0

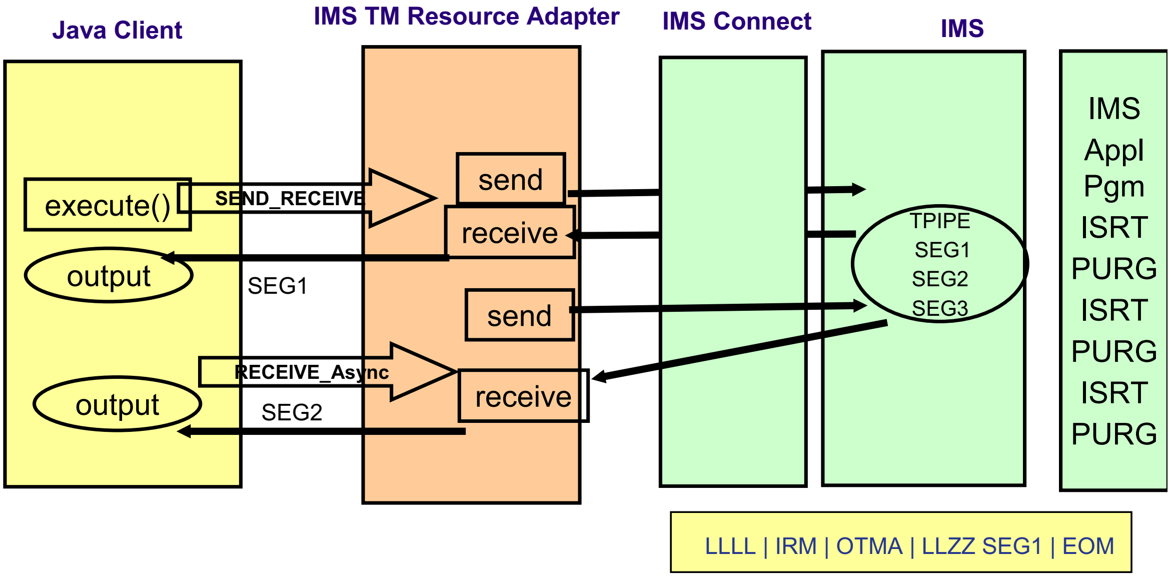
Notes:

Because an IMS Conversational transaction is a commit mode 1 type of transaction, IMS Connect only supports commit mode 1 conversational input messages with synch level of none or confirm from IMS TM Resource Adapter

client applications.

IMS TM Resource Adapter - CM0 multiple Purge calls

- Current Support



Java Client

IMS TM Resource Adapter

IMS Connect

IMS

execute()

SEND_RECEIVE

send

receive

output

SEG1

send

RECEIVE_Async

receive

output

SEG2

TPIPE
SEG1
SEG2
SEG3

IMS
Appl
Pgm
ISRT
PURG
ISRT
PURG
ISRT
PURG

LLLL | IRM | OTMA | LLZZ SEG1 | EOM

Figure 13-20. [No title] CMA01.0

Notes:

When there are multiple ISRT and PURG calls in the IMS application, OTMA sends one response message with multiple output segments if the client application uses Commit Mode 1 interaction. If the client application uses Commit Mode 0, OTMA will send multiple output response messages, one for each PURG call. OTMA introduces an enhancement so that the customer using CM0 with multiple PURG calls in the TP PCB can ignore the PURG calls to generate one output message with multiple segments. This provides consistent output message processing for CM0 and CM1.

A new property called ignorePURGCall will be added to the IMSInteractionSpec.

IMS TM Resource Adapter - CM0 multiple Purge calls

- New optional support

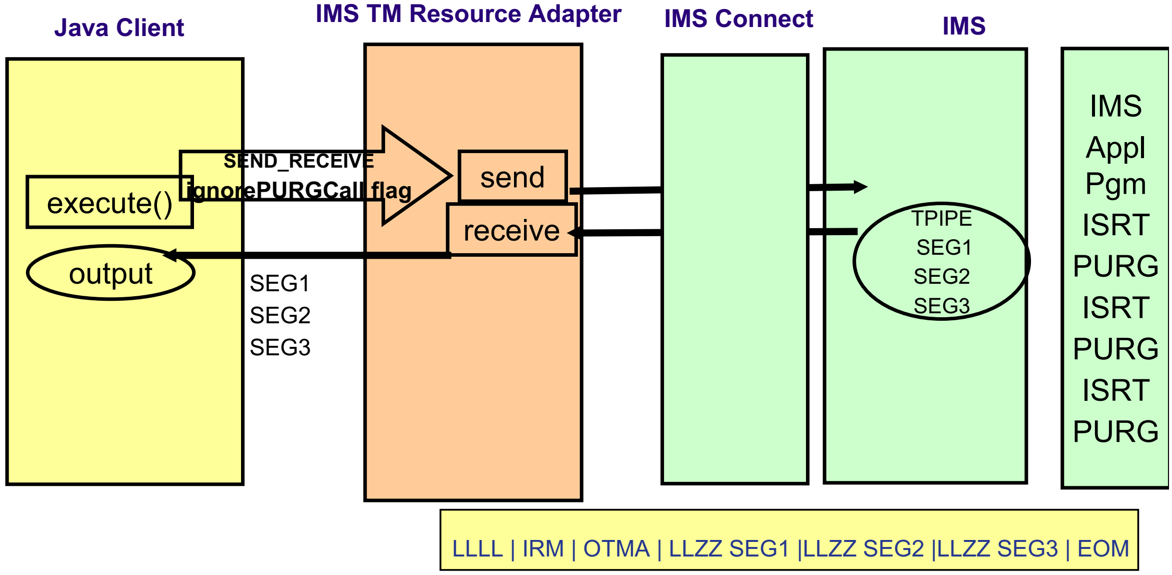

Java Client
IMS TM Resource Adapter
IMS Connect
IMS
execute()
SEND_RECEIVE
ignorePURGCall flag
send
receive
output
SEG1
SEG2
SEG3
TPIPE
SEG1
SEG2
SEG3
IMS
Appl
Pgm
ISRT
PURG
ISRT
PURG
ISRT
PURG
LLLL | IRM | OTMA | LLZZ SEG1 |LLZZ SEG2 |LLZZ SEG3 | EOM

Figure 13-21. [No title] CMA01.0

Notes:

Commit Mode 1 is only supported using a shareable persistent socket.

IMS Connector for Java uses persistent socket connections to communicate with IMS Connect. A persistent socket connection prevents the overhead of opening and closing the socket for each use by an application. IMS Connector for Java provides two types of persistent socket connections.

1.Dedicated persistent – a dedicated persistent socket connection is restricted to Commit Mode 0 (Commit then Send protocol) and the client application provides the clientid to be used as the socket identifier.

2.Shareable persistent – a shareable persistent socket connection supports Commit Mode 0 and Commit Mode 1 (Send then Commit protocol). However, The client application does not provide a clientid for a socket identifier. IC4J generates a clientid.

The support for CONFIRM provides the ability for IC4J to respond ACK to the output message which activates the IMS sync-point process. In this case the client application is not running under a Global Transaction.

This provides the ability for IMS Connector for Java to notify IMS that it has received the IMS application output message without requiring a Global Transaction.

In this example the Java client is using a shareable persistent socket Commit Mode 1 Sync Level CONFIRM. IC4J generates HWSABC01 for a socket ID and sends the input message to the IMS application program via IMS Connect and OTMA. To provide a message queue anchor for the Commit Mode 1 output message OTMA creates a transaction pipe (TPIPE) and will use the TCP/IP generated Client Port number provided by IMS Connect for the TPIPE name. When IC4J receives the output message from IMS Connect is responds with an ACK. IMS Connect will send the ACK to OTMA to activate the IMS sync point process.

Note the Java Client application cannot respond with an ACK.

IMS TM Resource Adapter

- CM0 multiple PURG calls
  - Provides CM1 and CM0 consistent output message processing
  - IMS V10 OTMA optional output processing enhancement
- Implementation
  - IMS TM Resource Adapter Client
    - IMSInteractionSpec new ignorePURGCall property

Figure 13-22. IMS TM Resource Adapter CMA01.0

Notes:

When there are multiple ISRT and PURG calls in the IMS application, OTMA sends one response message with multiple output segments if the client application uses Commit Mode 1 interaction. If the client application uses Commit Mode 0, OTMA will send multiple output response messages, one for each PURG call. OTMA introduces an enhancement so that the customer using CM0 with multiple PURG calls in the TP PCB can ignore the PURG calls to generate one output message with multiple segments. This provides consistent output message processing for CM0 and CM1.

A new property called ignorePUGRCall will be added to the IMSInteractionSpec.

IMS SOAP Gateway

Figure 13-23. [No title] CMA01.0

Notes:

IMS SOAP Gateway is a separate product that is available free of charge from the IMS Web site

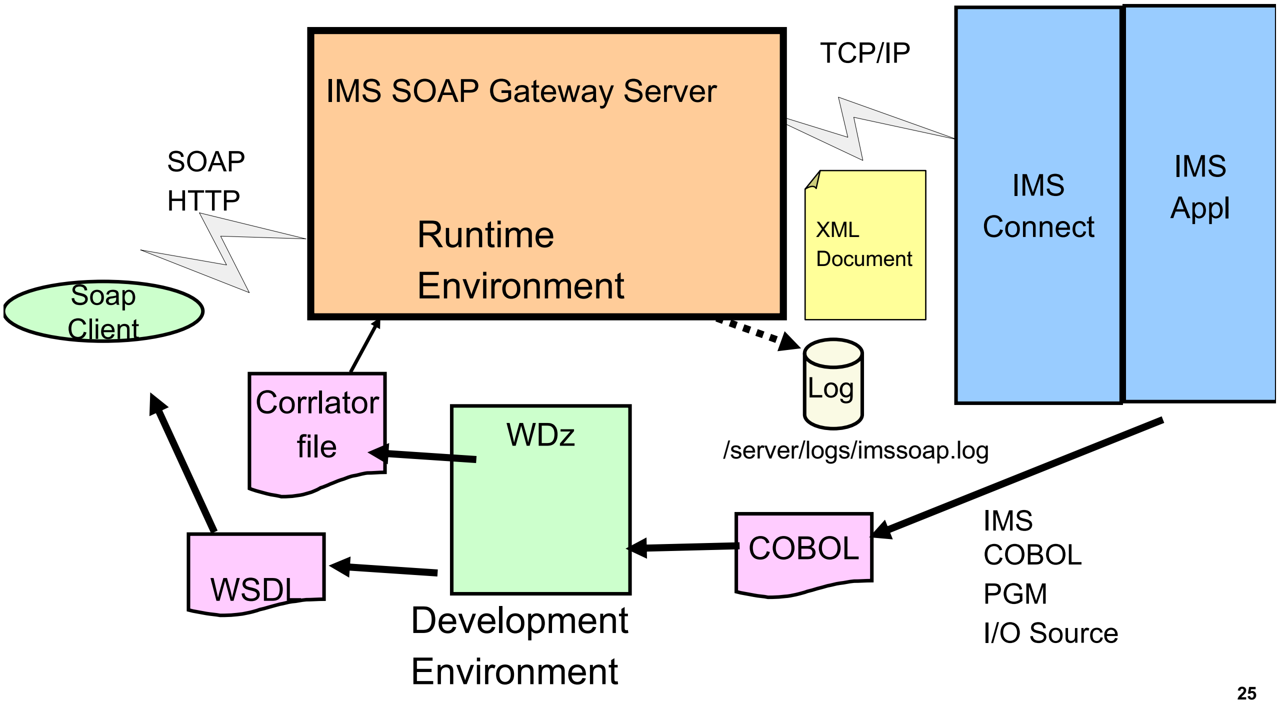
Information On Demand

IMS Version 10

IBM

IMS SOAP Gateway overview

Enables IMS applications to inter-operate outside of the IMS environment using SOAP protocols



IMS SOAP Gateway Server

Runtime Environment

TCP/IP

SOAP
HTTP

Soap Client

XML Document

IMS Connect

IMS Appl

Log

/server/logs/imssoap.log

Corrlator file

WDz

WSDL

COBOL

IMS
COBOL
PGM
I/O Source

Development Environment

Figure 13-24. [No title] CMA01.0

Notes:

The IMS SOAP Gateway is an XML-based connectivity Web service solution that is an HTTP endpoint for a SOAP client and supports access to IMS applications in a Service-Oriented Architecture (SOA) environment. This diagram shows a SOAP client accessing IMS via the IMS SOAP Gateway Server. SOAP (simple object access protocol) is a protocol for exchanging the XML-based messages over a network using HyperText Transfer Protocol (HTTP).

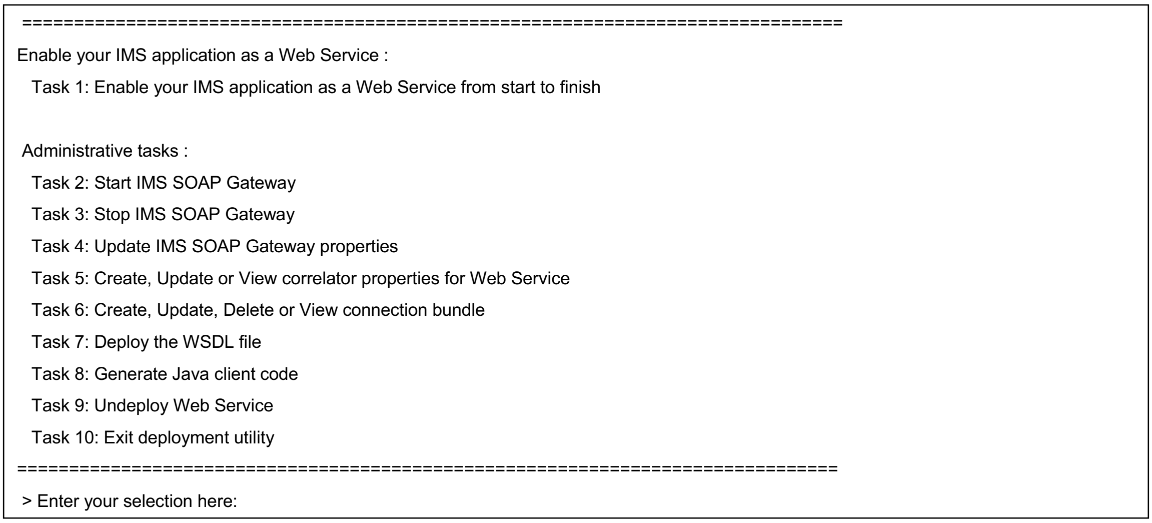
To deploy an IMS application as a Web service, you need to create a WSDL (Web service description language) file.

A WSDL file is an XML document that describes a Web service. WSDL files are used by others (for example, the client that invokes the service) to discover the service and to understand how to invoke the service. It specifies the location of the service and the operations that the service exposes.

WebSphere Developer for zSeries (WDz) can be used to generate a WSDL file.

IMS SOAP Gateway Deployment Utility

- The Deployment Utility has the following Options.



```
==============================================================================
Enable your IMS application as a Web Service :
  Task 1: Enable your IMS application as a Web Service from start to finish

 Administrative tasks :
  Task 2: Start IMS SOAP Gateway
  Task 3: Stop IMS SOAP Gateway
  Task 4: Update IMS SOAP Gateway properties
  Task 5: Create, Update or View correlator properties for Web Service
  Task 6: Create, Update, Delete or View connection bundle
  Task 7: Deploy the WSDL file
  Task 8: Generate Java client code
  Task 9: Undeploy Web Service
  Task 10: Exit deployment utility
==============================================================================
 > Enter your selection here:
```

Figure 13-25. IMS SOAP Gateway Deployment Utility CMA01.0

Notes:

The IMS SOAP Gateway also provides a deployment utility that generates the connection bundle, the correlator file, and the Server properties file. The connection bundle specifies the connection and security properties between IMS SOAP Gateway, IMS Connect, and IMS. The correlator file specifies transaction, and runtime properties and the information that the IMS SOAP Gateway needs to match incoming requests to the appropriate back-end IMS application. It also identifies the Connection Bundle. The Server Properties file is used to configure the IMS SOAP Gateway Server runtime environment

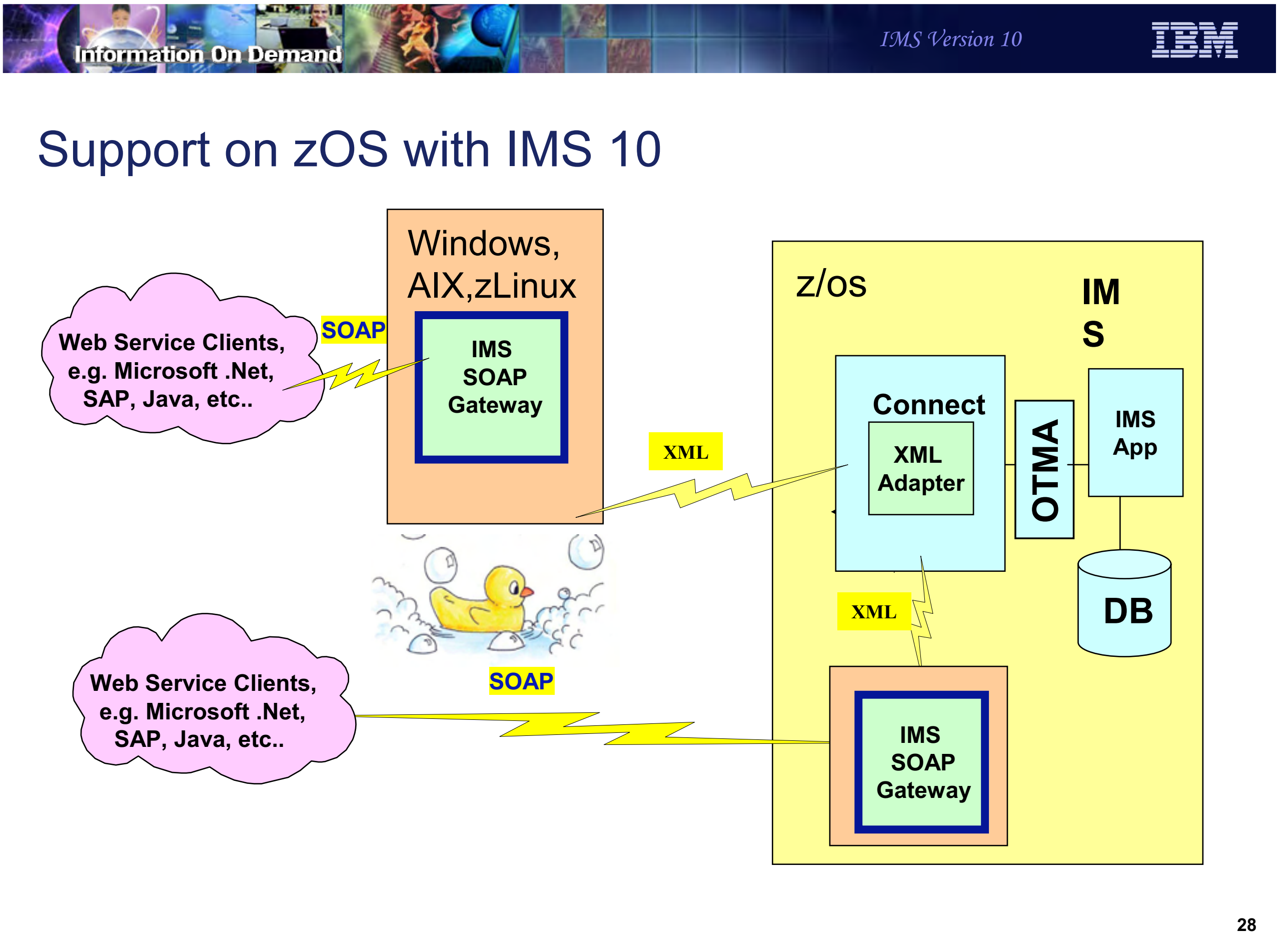
The Deployment Utility runs in the Windows DOS Command prompt.

IMS SOAP Gateway V10 Enhancements

- Support on z/OS
- Asynchronous Callout to Web Service
  - Requires IMS V10
    - IMS Connect Resume TPIPE with Alternate Clientid
    - IMS OTMA Resume TPIPE Security
    - IMS OTMA Destination Routing Descriptors

Figure 13-26. IMS SOAP Gateway V10 Enhancements CMA01.0

Notes:



Information On Demand

IMS Version 10

Support on zOS with IMS 10

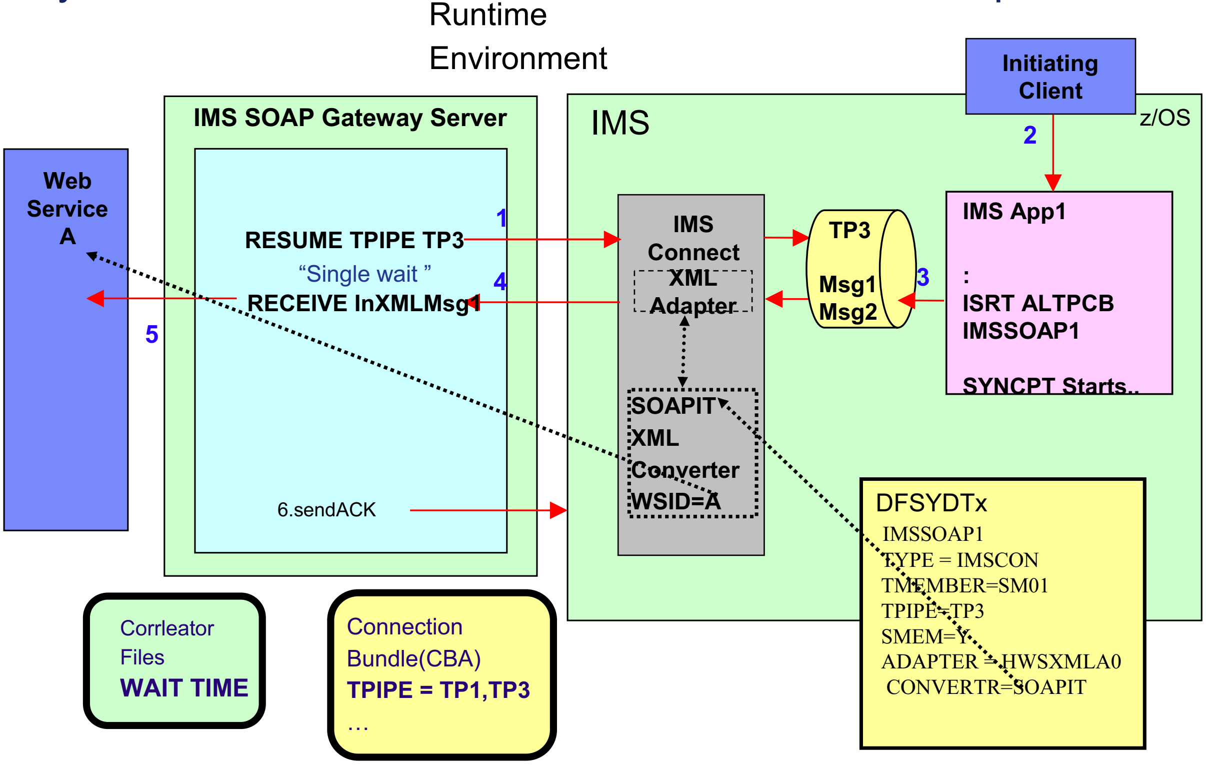
Windows,
AIX,zLinux
IMS
SOAP
Gateway
Web Service Clients,
e.g. Microsoft .Net,
SAP, Java, etc..
SOAP
XML
z/os
IM
S
Connect
XML
Adapter
OTMA
IMS
App
DB
XML
SOAP
Web Service Clients,
e.g. Microsoft .Net,
SAP, Java, etc..
IMS
SOAP
Gateway

Figure 13-27. Support on zOS with IMS 10

CMA01.0

Notes:

Asynchronous Callout to Web Service with XML Adapter



Runtime

Environment

Initiating Client

Web Service A

IMS SOAP Gateway Server

RESUME TPIPE TP3

"Single wait "

RECEIVE InXMLMsg1

6.sendACK

IMS

z/OS

IMS Connect

XML Adapter

SOAPIT

XML

Converter

WSID=A

TP3

Msg1

Msg2

IMS App1

:

ISRT ALTPCB

IMSSOAP1

SYNCPT Starts..

1

2

3

4

5

DFSYDTx

IMSSOAP1

TYPE = IMSCON

TMEMBER=SM01

TPIPE=TP3

SMEM=Y

ADAPTER = HWSXMLA0

CONVERTR=SOAPIT

Corrleator

Files

WAIT TIME

Connection

Bundle(CBA)

TPIPE = TP1,TP3

…

Figure 13-28. Asynchronous Callout to Web Service with XML Adapter CMA01.0

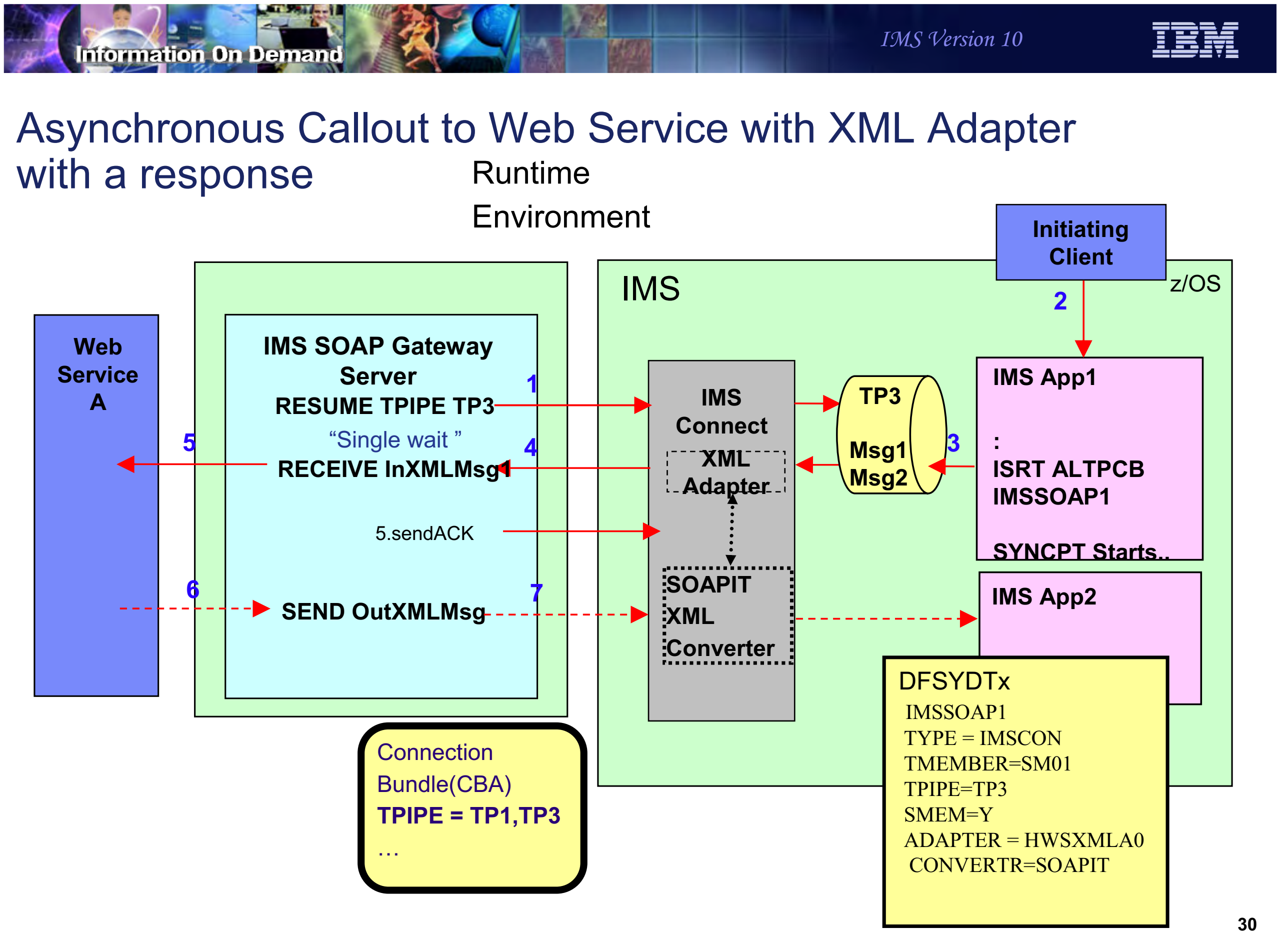
Notes:

In this scenario an IMS Application program does an insert to an ALTPCB whose destination is an OTMA Descriptor. The OTMA Descriptor is used to define the OTMA TPIPE for en-queue of the message.

Steps 1, 2 and 3 schedule the IMS application which results in the insert to ALTPCB defined by the OTMA Descriptor that represents IMS SOAP Gateway.

IMS SOAP Gateway will use the Resume TPIPE Alt-ClientID protocol to retrieve the message(1). IMS Connect will invoke the COBOL adapter to perform XML transformation of the message and send the message to the IMS SOAP Gateway(4). IMS SOAP Gateway receives the messages and sends it to the Web Service(5).

An optional step is the Web Service creates a response to the message request. IMS SOAP Gateway sends the response to IMS (6). This results in the scheduling of an IMS Application program to process the response.



Information On Demand

IMS Version 10

Asynchronous Callout to Web Service with XML Adapter with a response

Runtime Environment

Web Service A

IMS SOAP Gateway Server

RESUME TPIPE TP3

"Single wait"

RECEIVE InXMLMsg1

5.sendACK

SEND OutXMLMsg

IMS

IMS Connect

XML Adapter

SOAPIT XML Converter

TP3

Msg1

Msg2

Initiating Client

z/OS

IMS App1

:

ISRT ALTPCB IMSSOAP1

SYNCPT Starts..

IMS App2

1

2

3

4

5

6

7

Connection Bundle(CBA)

TPIPE = TP1,TP3

…

DFSYDTx

IMSSOAP1

TYPE = IMSCON

TMEMBER=SM01

TPIPE=TP3

SMEM=Y

ADAPTER = HWSXMLA0

CONVERTR=SOAPIT

Figure 13-29. Asynchronous Callout to Web Service with XML Adapterwith a response CMA01.0

Notes:

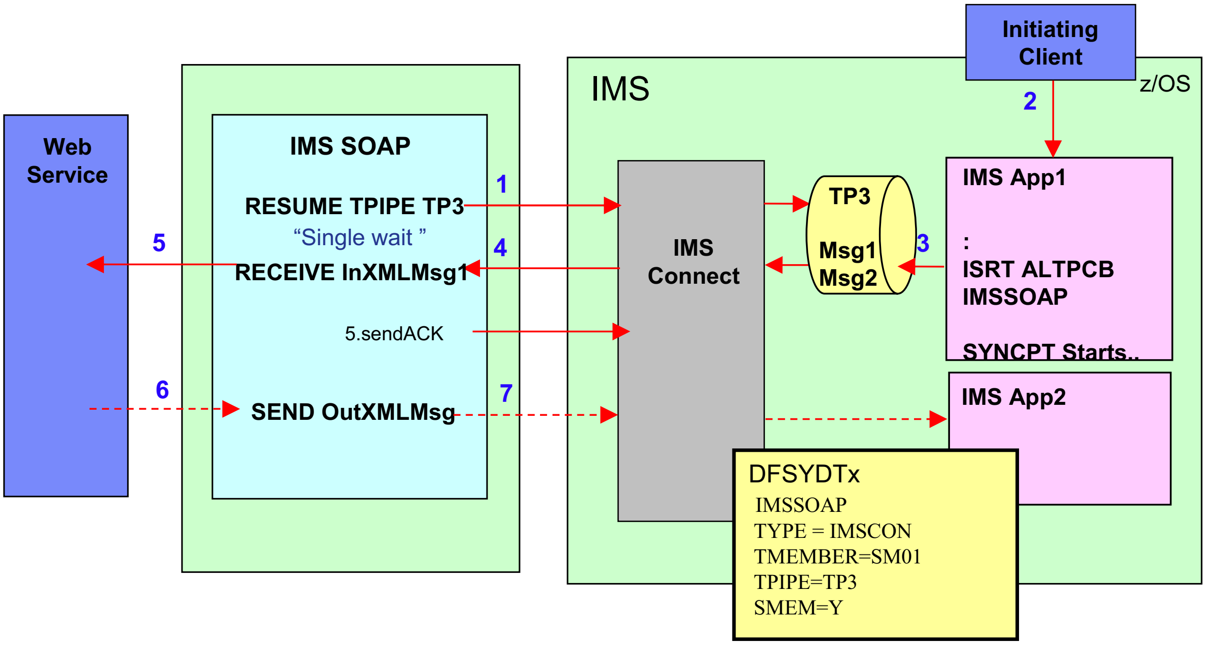
In this scenario an IMS Application program does an insert to an ALTPCB whose destination is an OTMA Descriptor. The OTMA Descriptor is used to define the OTMA TPIPE for en-queue of the message.

Steps 1, 2 and 3 schedule the IMS application which results in the insert to ALTPCB defined by the OTMA Descriptor that represents IMS SOAP Gateway.

IMS SOAP Gateway will use the Resume TPIPE Alt-ClientID protocol to retrieve the message(1). IMS Connect will invoke the COBOL adapter to perform XML transformation of the message and send the message to the IMS SOAP Gateway(4). IMS SOAP Gateway receives the messages and sends it to the Web Service(5).

An optional step is the Web Service creates a response to the message request. IMS SOAP Gateway sends the response to IMS (6). This results in the scheduling of an IMS Application program to process the response.

Asynchronous Callout to Web Service without XML Adapter


Initiating Client
z/OS
2
IMS
Web Service
IMS SOAP
1
RESUME TPIPE TP3
"Single wait "
5
4
RECEIVE InXMLMsg1
5.sendACK
6
SEND OutXMLMsg
7
IMS Connect
TP3
Msg1
Msg2
3
IMS App1
:
ISRT ALTPCB
IMSSOAP
SYNCPT Starts..
IMS App2
DFSYDTx
IMSSOAP
TYPE = IMSCON
TMEMBER=SM01
TPIPE=TP3
SMEM=Y

Note: IMS application needs to create XML data stream

Figure 13-30. Asynchronous Callout to Web Service without XML Adapter CMA01.0

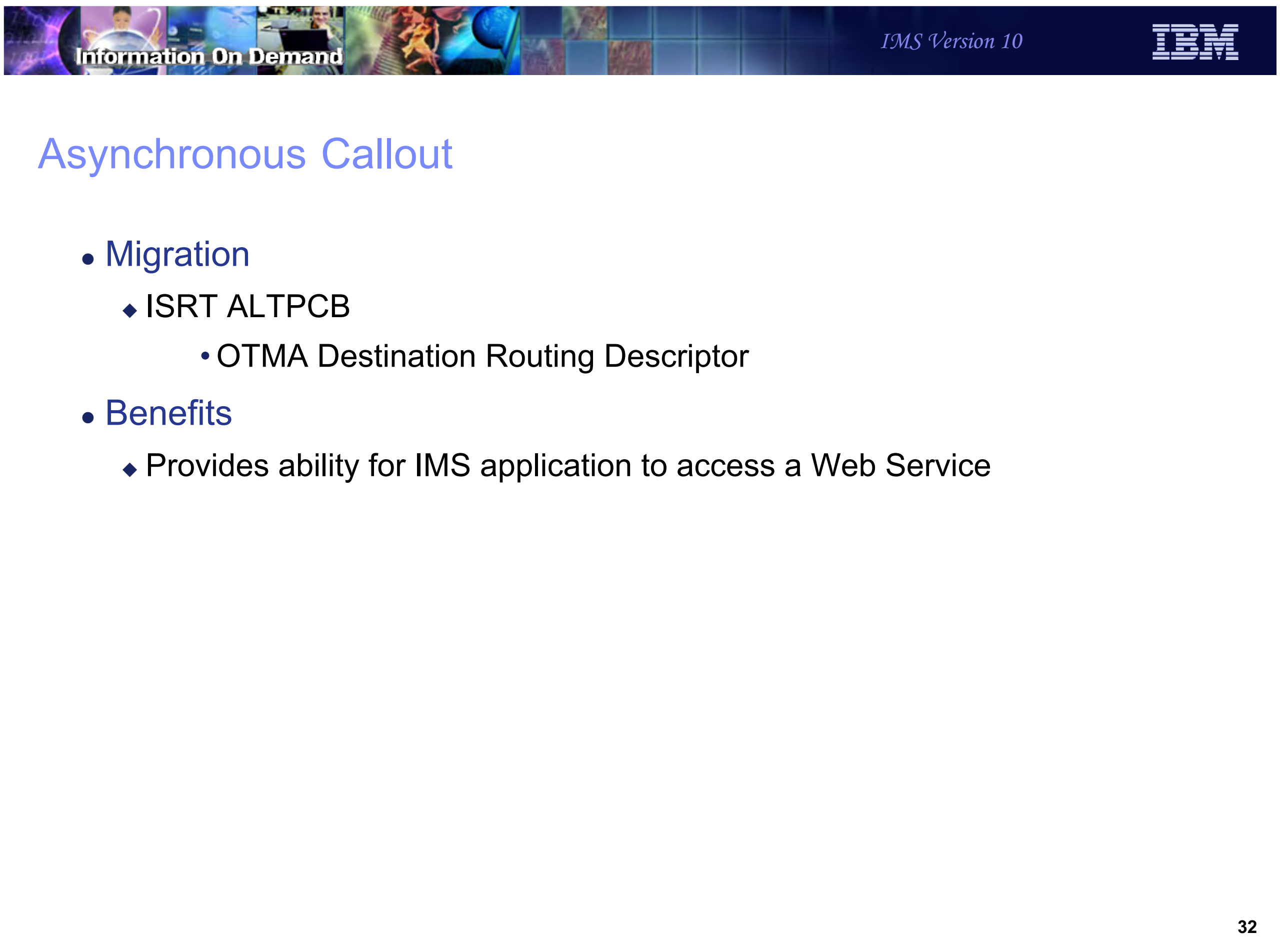
Notes:

In this scenario an IMS Application program does an insert to an ALTPCB whose destination is an OTMA Descriptor. The OTMA Descriptor is used to define the OTMA TPIPE for en-queue of the message.

Steps 1, 2 and 3 schedule the IMS application which results in the insert to ALTPCB defined by the OTMA Descriptor that represents IMS SOAP Gateway.

IMS SOAP Gateway will use the Resume TPIPE Alt-ClientID protocol to retrieve the message(1). IMS Connect will invoke the COBOL adapter to perform XML transformation of the message and send the message to the IMS SOAP Gateway(4). IMS SOAP Gateway receives the messages and sends it to the Web Service(5).

An optional step is the Web Service creates a response to the message request. IMS SOAP Gateway sends the response to IMS (6). This results in the scheduling of an IMS Application program to process the response.


Information On Demand
IMS Version 10
IBM

Asynchronous Callout

- Migration
  - ISRT ALTPCB
    - OTMA Destination Routing Descriptor
- Benefits
  - Provides ability for IMS application to access a Web Service

Figure 13-31. Asynchronous Callout CMA01.0

Notes:

The OTMA Callout Descriptors need to be created.

IMS MFS Web Support

Figure 13-32. [No title] CMA01.0

Notes:

IMS MFS Web Support
MFS Web Services and MFS Web Enablement

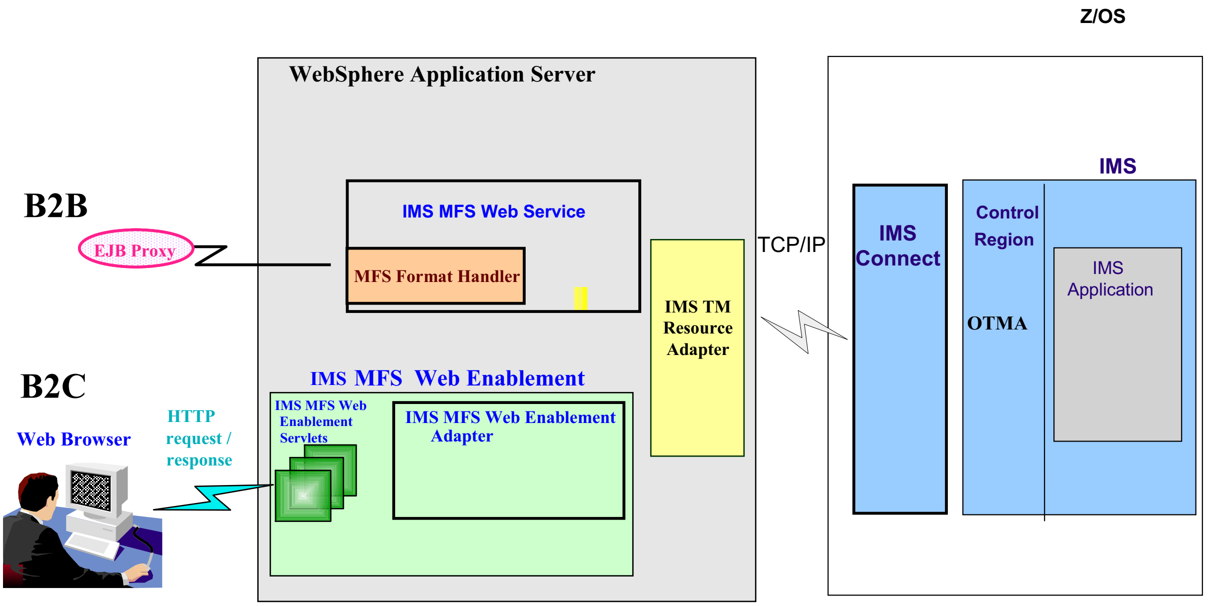

Z/OS
WebSphere Application Server
B2B
EJB Proxy
IMS MFS Web Service
MFS Format Handler
IMS TM Resource Adapter
TCP/IP
IMS Connect
IMS
Control Region
OTMA
IMS Application
B2C
Web Browser
HTTP request / response
IMS MFS Web Enablement
IMS MFS Web Enablement Servlets
IMS MFS Web Enablement Adapter

Figure 13-33. IMS MFS Web SupportMFS Web Services and MFS Web Enablement CMA01.0

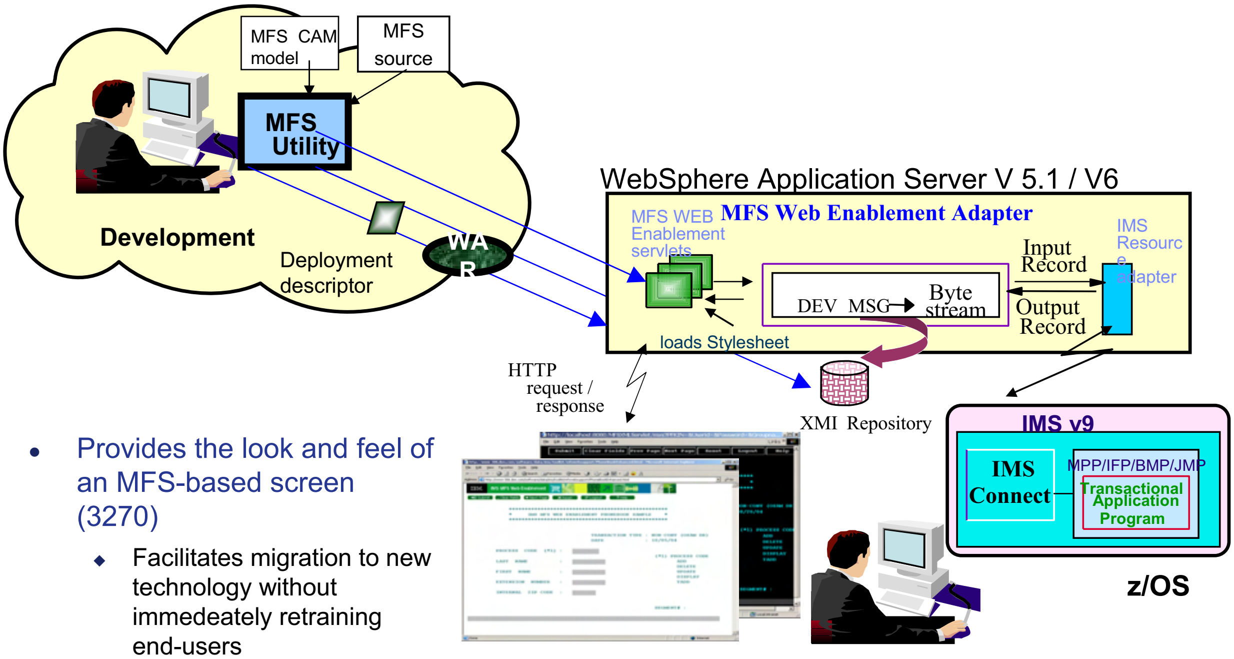
Notes:

IMS MFS Web support provides two solutions to re-use existing MFS-based IMS business logic.

1. IMS MFS Web services support allows you to submit MFS-based IMS transactions as Web services.
2. IMS MFS Web enablement support replaces the MFS online processor and allows you to seamlessly navigate MFS-based Web pages that all have a similar look and feel.

Information On Demand
IMS Version 10
IBM

MFS Web Enablement


MFS CAM model
MFS source
MFS Utility
Development
Deployment descriptor
WAR
WebSphere Application Server V 5.1 / V6
MFS WEB Enablement servlets
MFS Web Enablement Adapter
IMS Resource adapter
Input Record
Output Record
DEV MSG
Byte stream
loads Stylesheet
HTTP request / response
XMI Repository
IMS v9
IMS Connect
MPP/IFP/BMP/JMP
Transactional Application Program
z/OS

- Provides the look and feel of an MFS-based screen (3270)
  - Facilitates migration to new technology without immedeately retraining end-users

35

Figure 13-34. MFS Web Enablement CMA01.0

Notes:

MFS Web Services – SOA support

- A capability that takes advantage of MFS source to generate web services

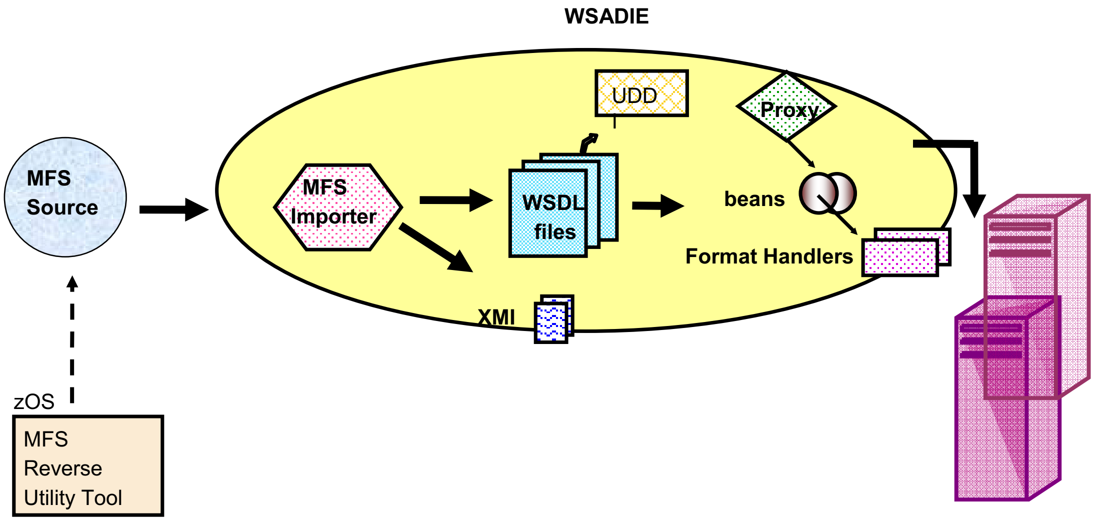

WSADIE
MFS Source
MFS Importer
UDD
Proxy
WSDL files
beans
Format Handlers
XMI
zOS
MFS Reverse Utility Tool

Figure 13-35. MFS Web Services – SOA support CMA01.0

Notes:

IMS DB Resource Adapter

Figure 13-36. [No title] CMA01.0

Notes:

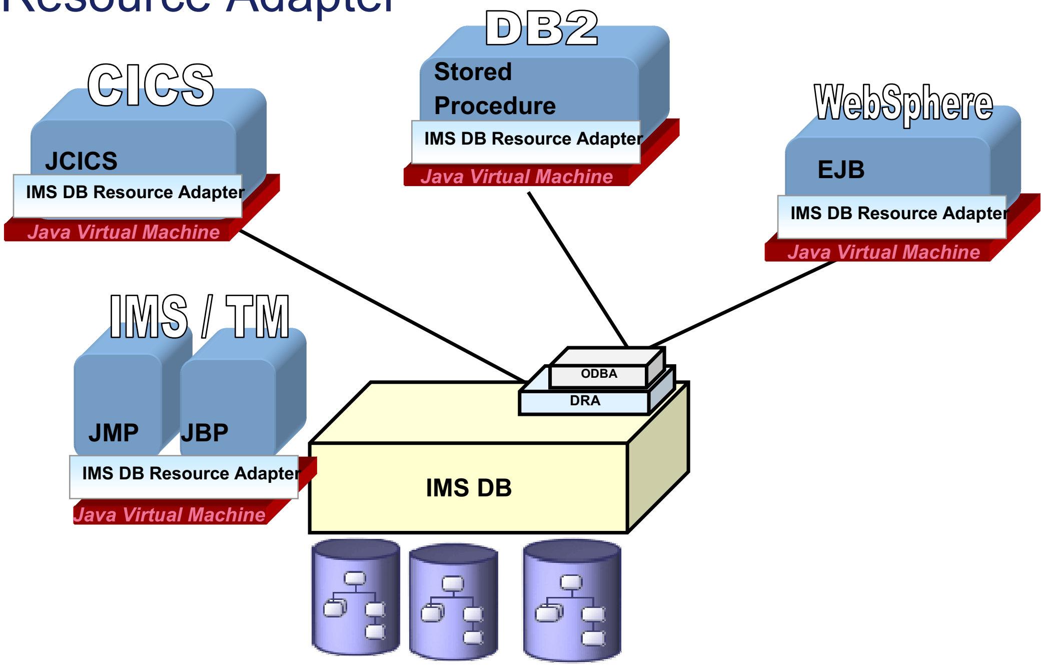
The IMS DB Resource Adapter is a function of IMS that enables a programmer with minimal IMS knowledge to write Java application programs that access IMS databases

Information On Demand

IMS Version 10

IBM

IMS DB Resource Adapter



DB2

Stored Procedure

IMS DB Resource Adapter

Java Virtual Machine

CICS

JCICS

IMS DB Resource Adapter

Java Virtual Machine

WebSphere

EJB

IMS DB Resource Adapter

Java Virtual Machine

IMS / TM

JMP

JBP

IMS DB Resource Adapter

Java Virtual Machine

ODBA

DRA

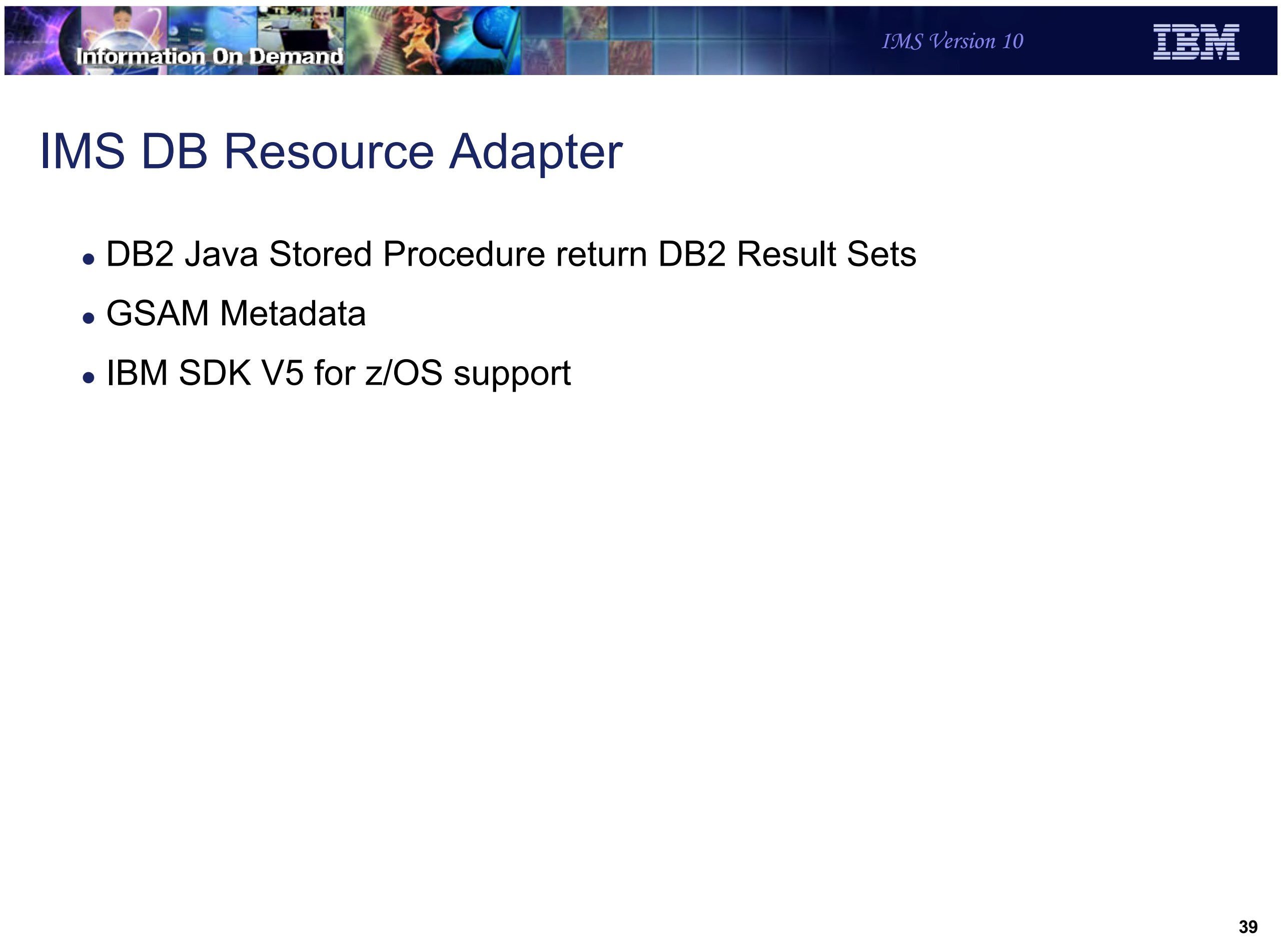
IMS DB

Figure 13-37. IMS DB Resource Adapter

CMA01.0

Notes:

The IMS DB Resource Adapter enables JDBC access to IMS DB from IMS TM JMP/JBP environments, CICS Java application, DB2 Java Stored procedure, and Enterprise Java Beans running on WebSphere distributed and z/OS environments.



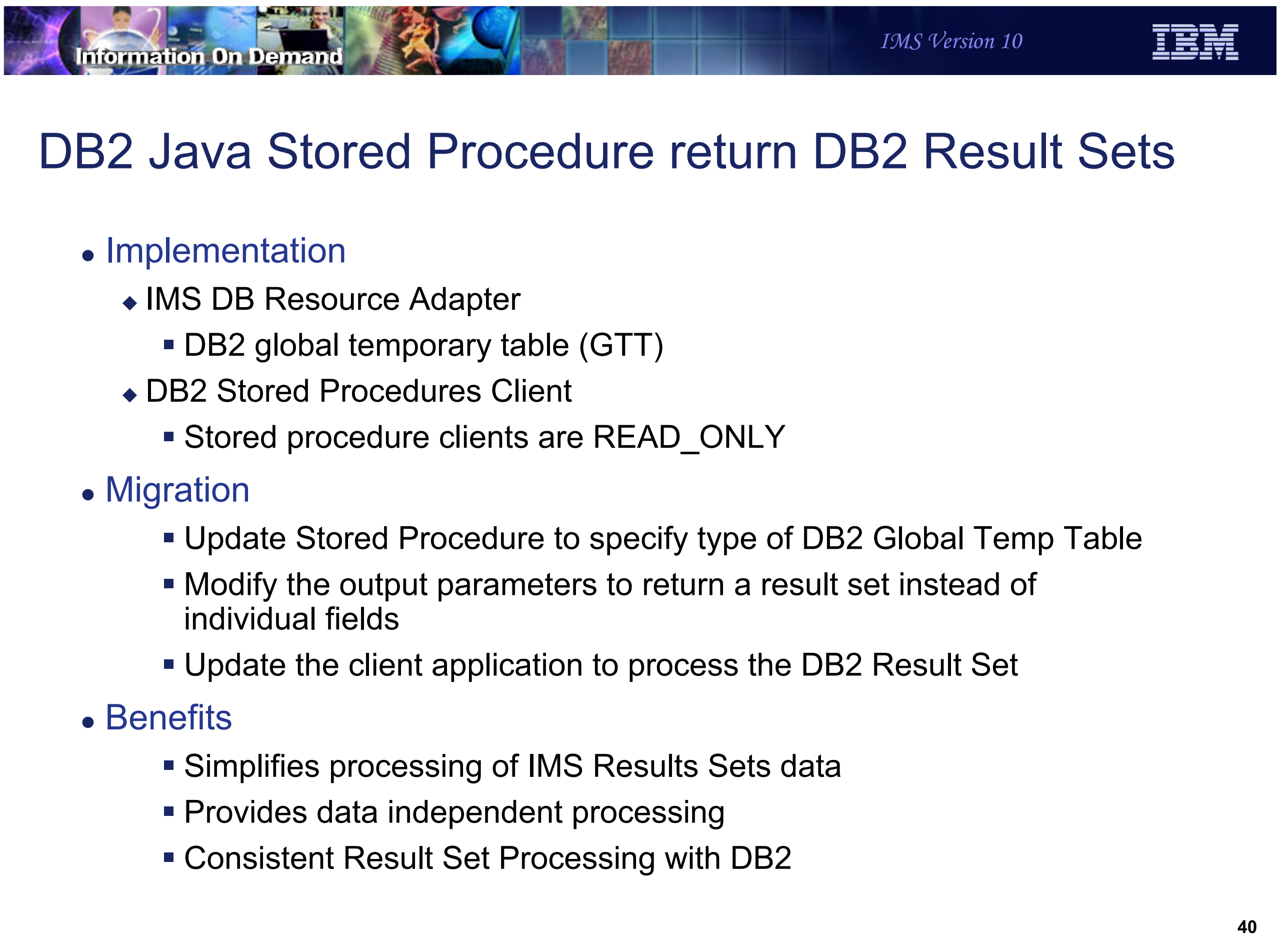
IMS DB Resource Adapter

- DB2 Java Stored Procedure return DB2 Result Sets
- GSAM Metadata
- IBM SDK V5 for z/OS support

Figure 13-38. IMS DB Resource Adapter CMA01.0

Notes:

The IMS DB Resource Adapter has been enhanced to support DB2 Java Stored Procedures returning DB2 result sets instead of returning IMS result sets. GSAM access has been enhanced to use IMS Java Metadata generated by DLIModel Utility. Changes were also made to support IBM SDK V5 for z/OS.


Information On Demand
IMS Version 10

DB2 Java Stored Procedure return DB2 Result Sets

- Implementation
  - IMS DB Resource Adapter
    - DB2 global temporary table (GTT)
  - DB2 Stored Procedures Client
    - Stored procedure clients are READ_ONLY
- Migration
  - Update Stored Procedure to specify type of DB2 Global Temp Table
  - Modify the output parameters to return a result set instead of individual fields
  - Update the client application to process the DB2 Result Set
- Benefits
  - Simplifies processing of IMS Results Sets data
  - Provides data independent processing
  - Consistent Result Set Processing with DB2

Figure 13-39. DB2 Java Stored Procedure return DB2 Result Sets CMA01.0

Notes:

The IMS DB Resource Adapter has been enhanced to create a DB2 global temporary table (GTT) in the DB2 Stored Procedure Address Space and moves the IMS data returned from IMS ResultSet into the DB2 GTT. The DB2 Result Set returned to the client application will iterate the DB2 GTT for result set processing.

The stored procedure clients are READ_ONLY. This means that the client cannot update or delete IMS data using the DB2 result set returned to it by the stored procedure because it is not possible send a change in a DB2 temp table back to an IMS database

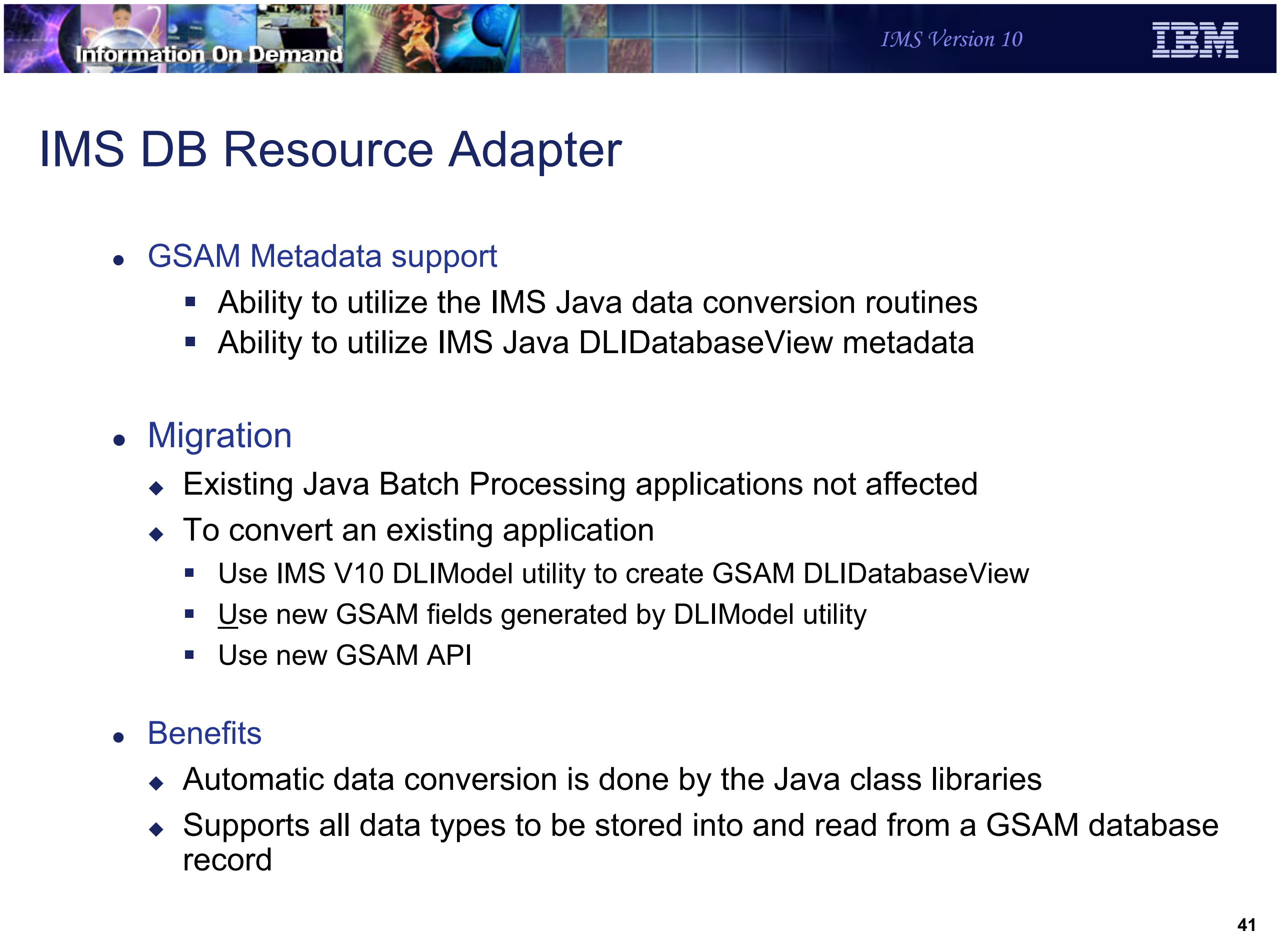
Existing DB2 Stored Procedures using the IMS V10 DB Resource Adapter can continue to run without modifications

To update existing stored procedures to return DB2 result sets instead of individual fields you need to update the stored procedure to specify the use of a created or declared type of DB2 Global Temp Table. After the Stored Procedure application queries the IMS database the IMS DB Resource Adapter API is used to convert the IMS ResultSet to a DB2 Result Set using the GTT.

The output parameters defined for the Stored Procedure need to be changed to return a result set instead of individual fields.

The client application needs to be modified to accept a DB2 result set instead of IMS individual fields.

The use of DB2 Results Sets provides a consistent and easier programming interface for processing output from a DB2 Stored Procedure that uses IMS JDBC Connector to access IMS data.



Information On Demand

IMS Version 10

IBM

IMS DB Resource Adapter

- GSAM Metadata support
    - Ability to utilize the IMS Java data conversion routines
    - Ability to utilize IMS Java DLIDatabaseView metadata
- Migration
    - Existing Java Batch Processing applications not affected
    - To convert an existing application
        - Use IMS V10 DLIModel utility to create GSAM DLIDatabaseView
        - Use new GSAM fields generated by DLIModel utility
        - Use new GSAM API
- Benefits
    - Automatic data conversion is done by the Java class libraries
    - Supports all data types to be stored into and read from a GSAM database record

Figure 13-40. IMS DB Resource Adapter CMA01.0

Notes:

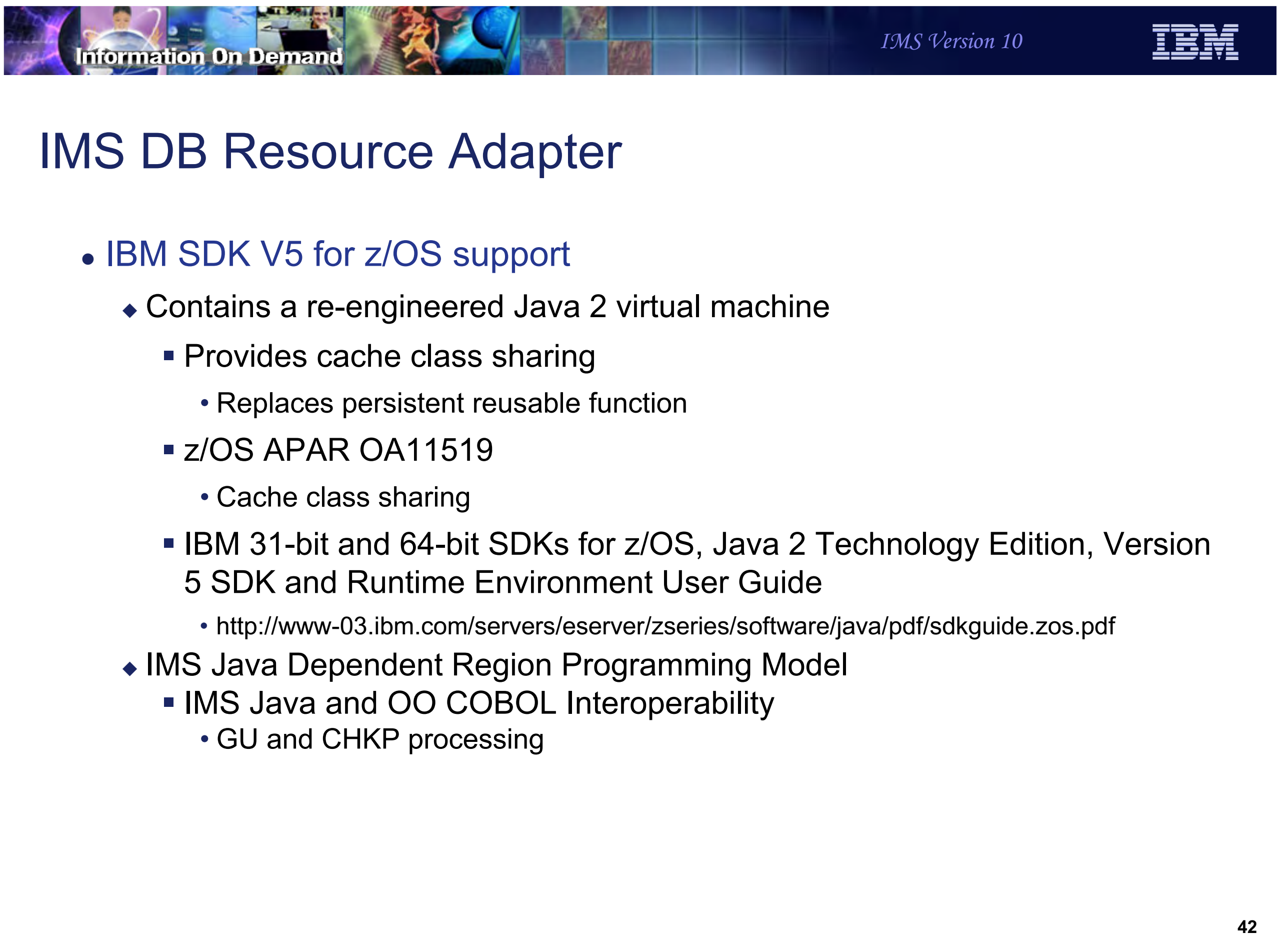
IMS DB Resource Adapter support for GSAM was enabled by Small Programming Enhancement PQ93785/UQ93241. This provided a Java Batch Processing application program with a Java API to open read write and close GSAM databases. However , the application was required to know the complete details of the GSAM database

Given a DLIDatabaseView name, the IMS Java class library will use the metadata information to capture the correct sequence of bytes in the record, read that sequence of bytes, and convert that sequence of bytes into the appropriate data type as defined by the metadata.

Java Batch Processing applications that use the IMS V9 GSAM database support will not be affected

The DLIModel utility for IMS V10 provides new GSAM record & field options. This will be described in the DLIModel utility section.

IMS V10 GSAM enables all of the supported data types to be stored into and read from a GSAM database record and provides automatic data conversion on behalf of the application. GSAM database support is now consistent with the IMS Java class libraries support for all other IMS database types.


Information On Demand
IMS Version 10
IBM

IMS DB Resource Adapter

- IBM SDK V5 for z/OS support
  - Contains a re-engineered Java 2 virtual machine
    - Provides cache class sharing
      - Replaces persistent reusable function
    - z/OS APAR OA11519
      - Cache class sharing
    - IBM 31-bit and 64-bit SDKs for z/OS, Java 2 Technology Edition, Version 5 SDK and Runtime Environment User Guide
      - http://www-03.ibm.com/servers/eserver/zseries/software/java/pdf/sdkguide.zos.pdf
  - IMS Java Dependent Region Programming Model
    - IMS Java and OO COBOL Interoperability
      - GU and CHKP processing

Figure 13-41. IMS DB Resource Adapter CMA01.0

Notes:

z/OS delivers a complete Java 2 Software Developer Kit (SDK).

Previous versions of the SDK provided a reusable function to support transactional runtime environments like IMS. This capability allowed the Java Virtual Machine (JVM) to be initialized during IMS Java dependent region startup and to be “re-set” after the IMS application program completed processing. This avoided the overhead of loading the JVM for each IMS appplication program schedule.

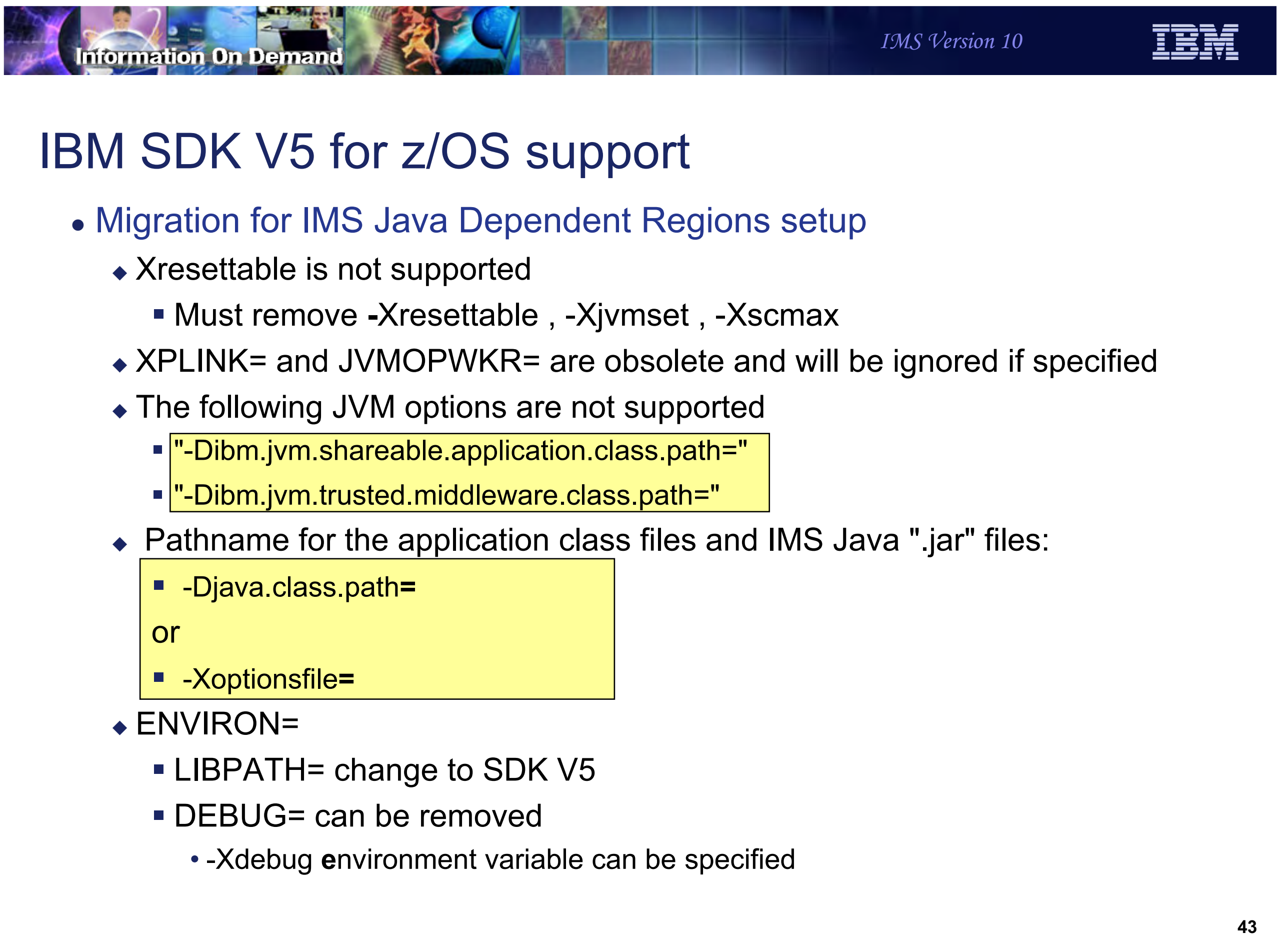
The new SDK provides a Class Sharing capability to replace the persistent reusable function.

z/OS APAR OA11519 is recommended for cache class sharing.

The User Guide can be downloaded from the specified URL.

IMS message GU/CHKP processing was altered for Java Dependent Regions (JDRs) when JDRs were initially implemented. For JDRs, an IMS message GU meant to get the message only, do not perform sync point processing. CHKP processing meant perform sync point processing only, do not get the next message. In the standard IMS model, message GU and CHKP meant perform sync point and retrieve the next input message.

For IMS V10, IMS JDRs GU/CHKP processing will be consistent with the IMS standard model.



IBM SDK V5 for z/OS support

- Migration for IMS Java Dependent Regions setup
  - Xresettable is not supported
    - Must remove **-**Xresettable , -Xjvmset , -Xscmax
  - XPLINK= and JVMOPWKR= are obsolete and will be ignored if specified
  - The following JVM options are not supported
    - "-Dibm.jvm.shareable.application.class.path="
    - "-Dibm.jvm.trusted.middleware.class.path="
  - Pathname for the application class files and IMS Java ".jar" files:
    - -Djava.class.path**=**

    or

    - -Xoptionsfile**=**
  - ENVIRON=
    - LIBPATH= change to SDK V5
    - DEBUG= can be removed
      - -Xdebug **e**nvironment variable can be specified

Figure 13-42. IBM SDK V5 for z/OS support CMA01.0

Notes:

The serial reusability feature of the IBM SDK for z/OS, version 1.4.2 (31-bit) and earlier is not supported. If you specify –Xresettable -Xjvmset or -Xscmax the JVM will issue an error message and will not start.

The JVM shared library libjvm.so is installed in directory jre/bin/j9vm.

The IMS Java dependent region ENVIRON= member needs to have the LIBPATH changed to LIBPATH=JavaHome/bin/j9vm:JavaHome/bin:ImsjavaPath

ENVIRON= can now be used to specify any Java environment variables. This can be used to replace the DEBUG= parameter. The Java environment variables are presented as Java system properties at runtime and are therefore accessible by a Java application running in the JDR.

-Xdebug starts the JVM with the debugger enabled. By default, the debugger is disabled

Information On Demand IMS Version 10 IBM

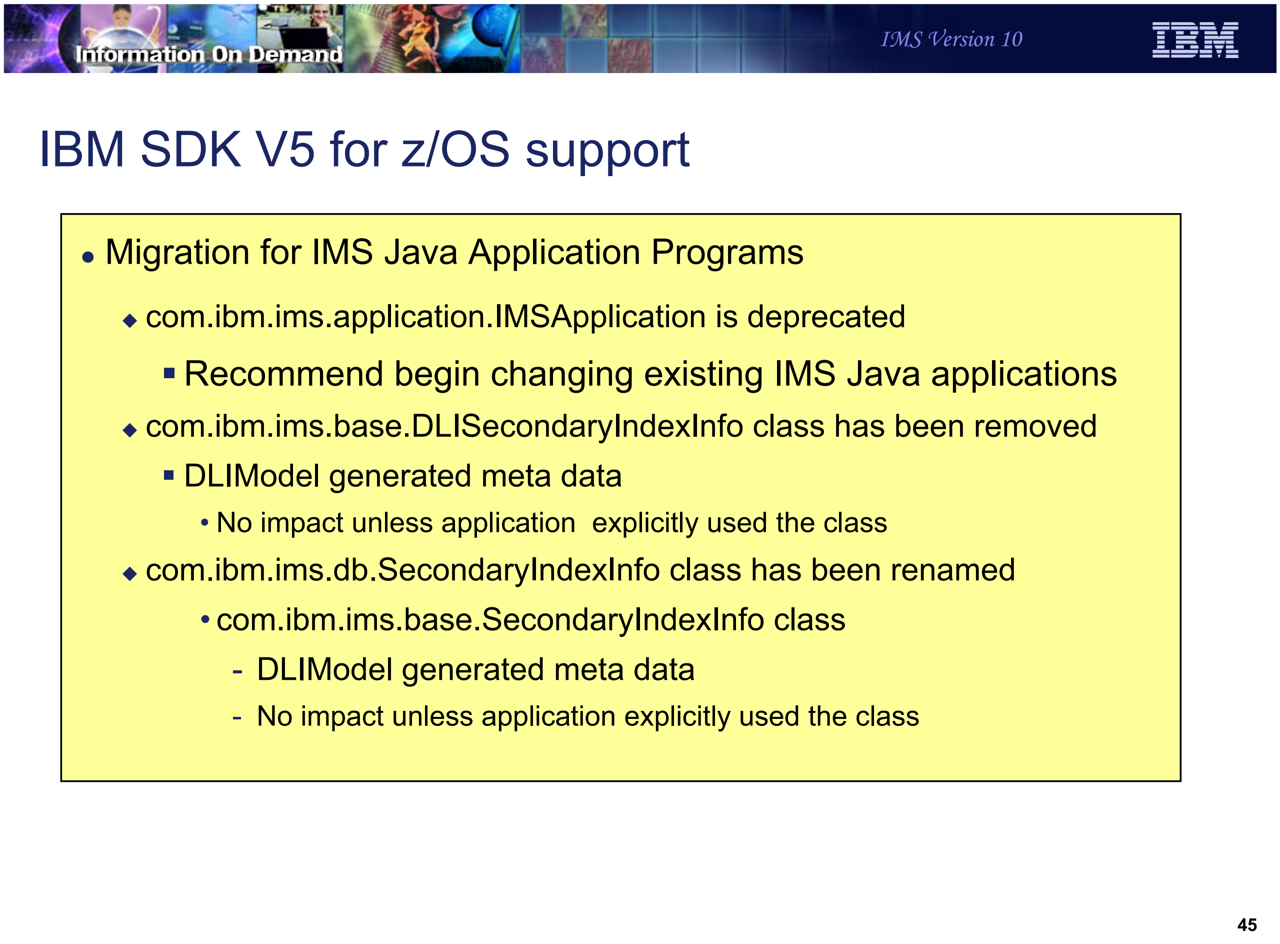
IBM SDK V5 for z/OS support

- Migration for IMS Java Dependent Regions programming model
  - Current applications do not need to change unless:
    - Applications that use U118 Abend for ROLLBACK
    - Applications that use explicit CHKP call
      - Now receives next input message after CHKP call

Figure 13-43. IBM SDK V5 for z/OS support CMA01.0

Notes:

An IMS Java application that does not issue explicit ROLLBACK but uses U118 abend for rollback processing must be changed. An IMS Java application that uses the IMS Java hierarchical database interface for CHKP call processing must be able to process the next message off the message queue.



Information On Demand

IMS Version 10

IBM SDK V5 for z/OS support

- Migration for IMS Java Application Programs
  - com.ibm.ims.application.IMSApplication is deprecated
    - Recommend begin changing existing IMS Java applications
  - com.ibm.ims.base.DLISecondaryIndexInfo class has been removed
    - DLIModel generated meta data
      - No impact unless application explicitly used the class
  - com.ibm.ims.db.SecondaryIndexInfo class has been renamed
      - com.ibm.ims.base.SecondaryIndexInfo class
        - DLIModel generated meta data
        - No impact unless application explicitly used the class

45

Figure 13-44. IBM SDK V5 for z/OS support CMA01.0

Notes:

The IMS DB Resource Adapter class com.ibm.ims.application.IMSApplication is deprecated in IMS Version 10.

A class marked as deprecated is considered obsolete but still works in IMS Version 10.

You do not have to change your applications, but it is highly recommended.

The com.ibm.ims.base.DLISecondaryIndexInfo class has been removed from the library. This will only impact you if you did not use DLIModel to generate the meta data classes (the database view) or if you use the DLISecondaryIndexInfo class explicitly in your code.

The com.ibm.ims.db.SecondaryIndexInfo class has been renamed to com.ibm.ims.base.SecondaryIndexInfo.

You will be impacted only if you use the class directly in your code. The metadata that is generated by the DLIModel utility is not affected.

Information On Demand

IMS Version 10

IBM SDK V5 for z/OS support
- IMS Java Application sample API

Current

```
public class CustomerApplication extends IMSApplication {
 public static void main(String args[]) {
        CustomerApplication myapp = new CustomerApplication();
        myapp.begin();
```

remove

remove

Modified

```
public class CustomerApplication  {
 public static void main(String args[]) {
        CustomerApplication myapp = new CustomerApplication();
```

Figure 13-45. IBM SDK V5 for z/OS support - IMS Java Application sample API CMA01.0

Notes:

The applications that subclass IMSApplication can be modified as follows:

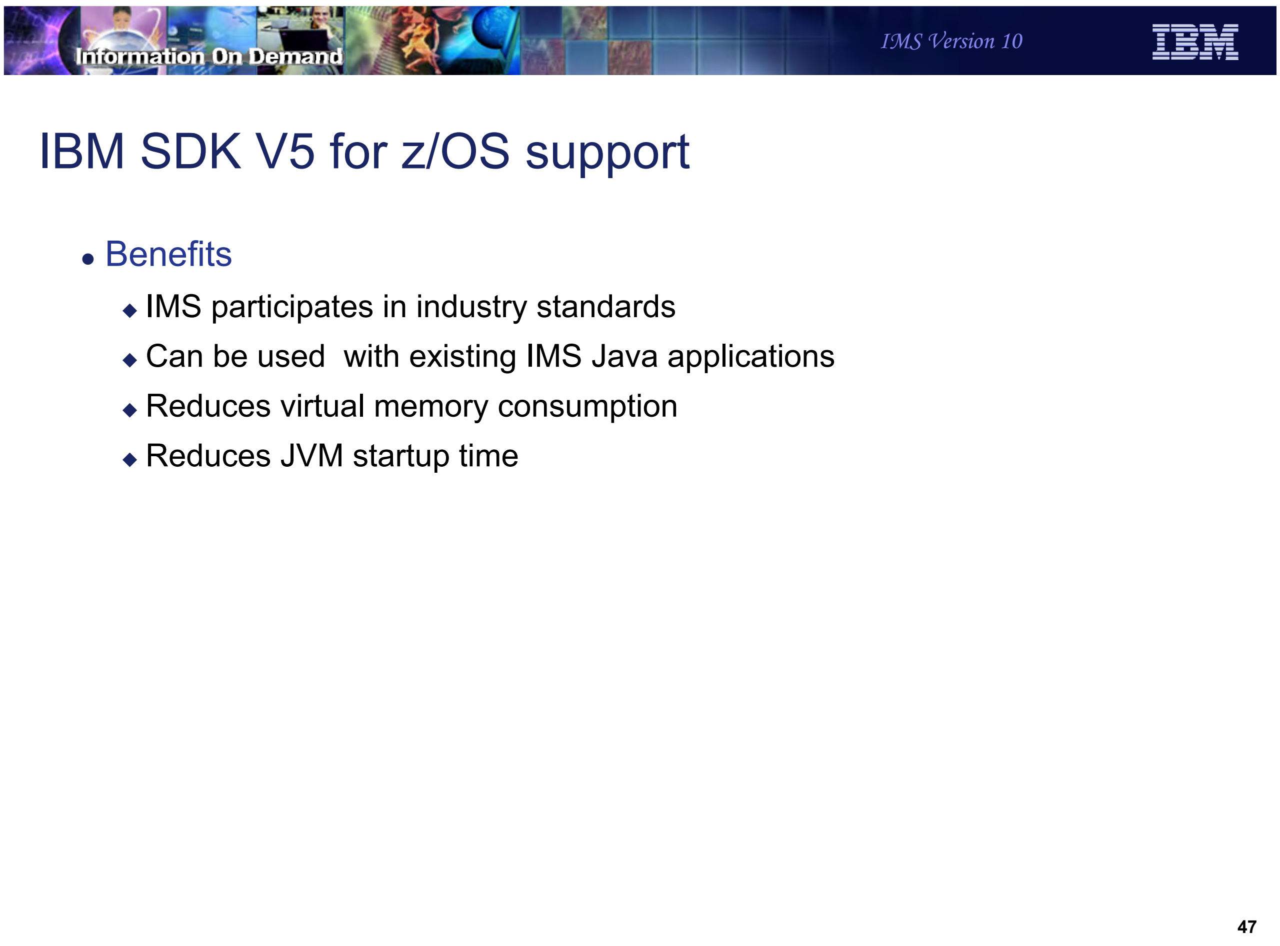
Remove the "extends IMSApplication" from the class declaration line.

For example, "public class CustomerApplication extends IMSApplication"

becomes "public class CustomerApplication".

The main method of the application no longer needs to call the IMSApplication.begin() method.

Instead, the main method can directly call the public void doBegin() method or simply move the logic from the doBegin() method to the main method and delete the doBegin() method.



Information On Demand
IMS Version 10

IBM SDK V5 for z/OS support

- Benefits
  - IMS participates in industry standards
  - Can be used with existing IMS Java applications
  - Reduces virtual memory consumption
  - Reduces JVM startup time

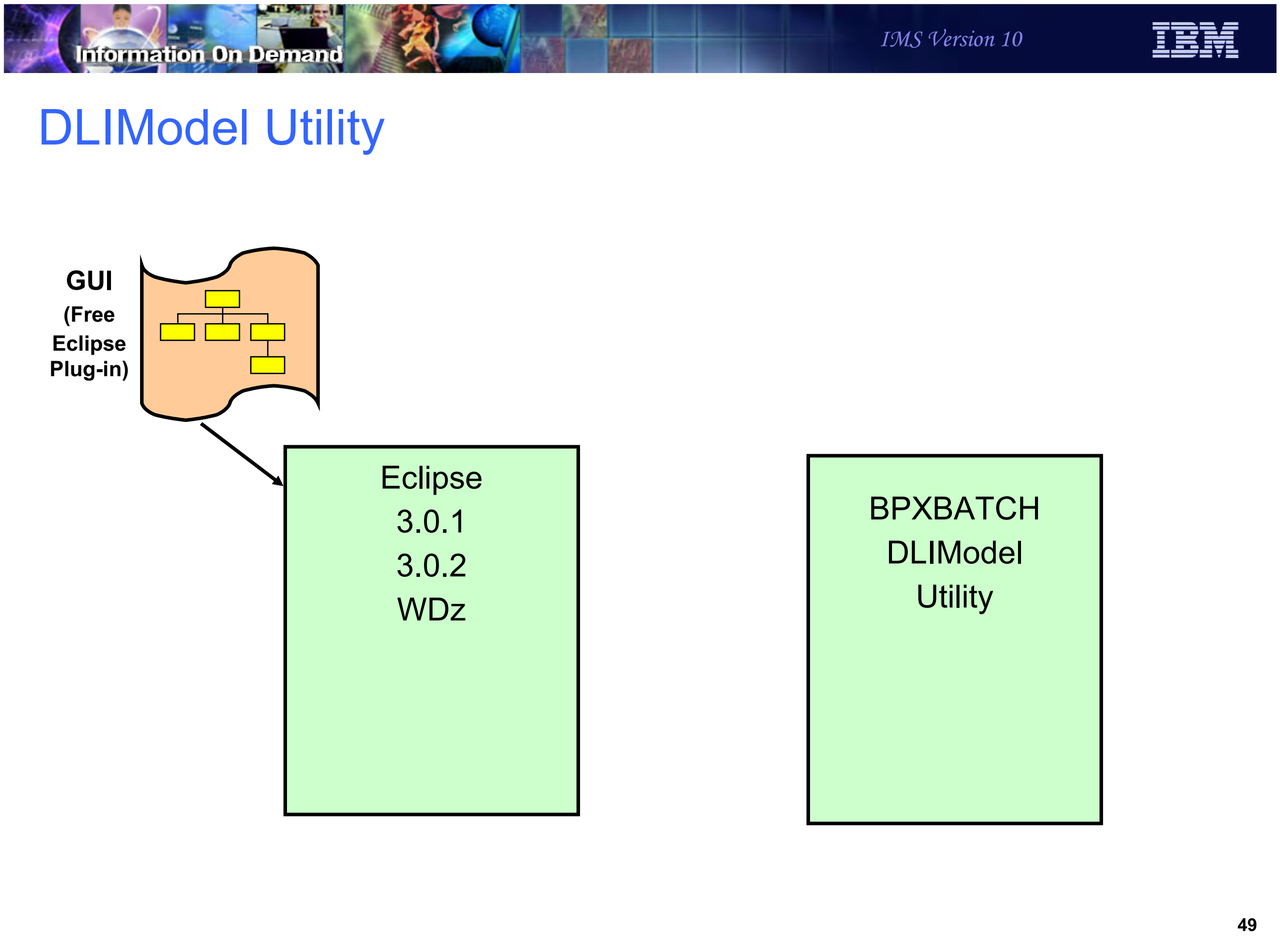
Figure 13-46. IBM SDK V5 for z/OS support CMA01.0

Notes:

IMS DLIModel utility

Figure 13-47. [No title] CMA01.0

Notes:



DLIModel Utility

GUI
(Free
Eclipse
Plug-in)

Eclipse
3.0.1
3.0.2
WDz

BPXBATCH
DLIModel
Utility

49

Figure 13-48. DLIModel Utility

CMA01.0

Notes:

Two versions of the IMS DLIModel utility are available:

1.An IMS-shipped version that runs from System Services or from the z/OS® BPXBATCH utility

2.A technology preview IMS Web free download version that runs as a plug-in to Eclipse

The GUI can be installed in an Eclipse 3.0.1 or 3.0.2 level tool. It can also be installed in WebSphere Developer for z IDE.

DLIModel Utility

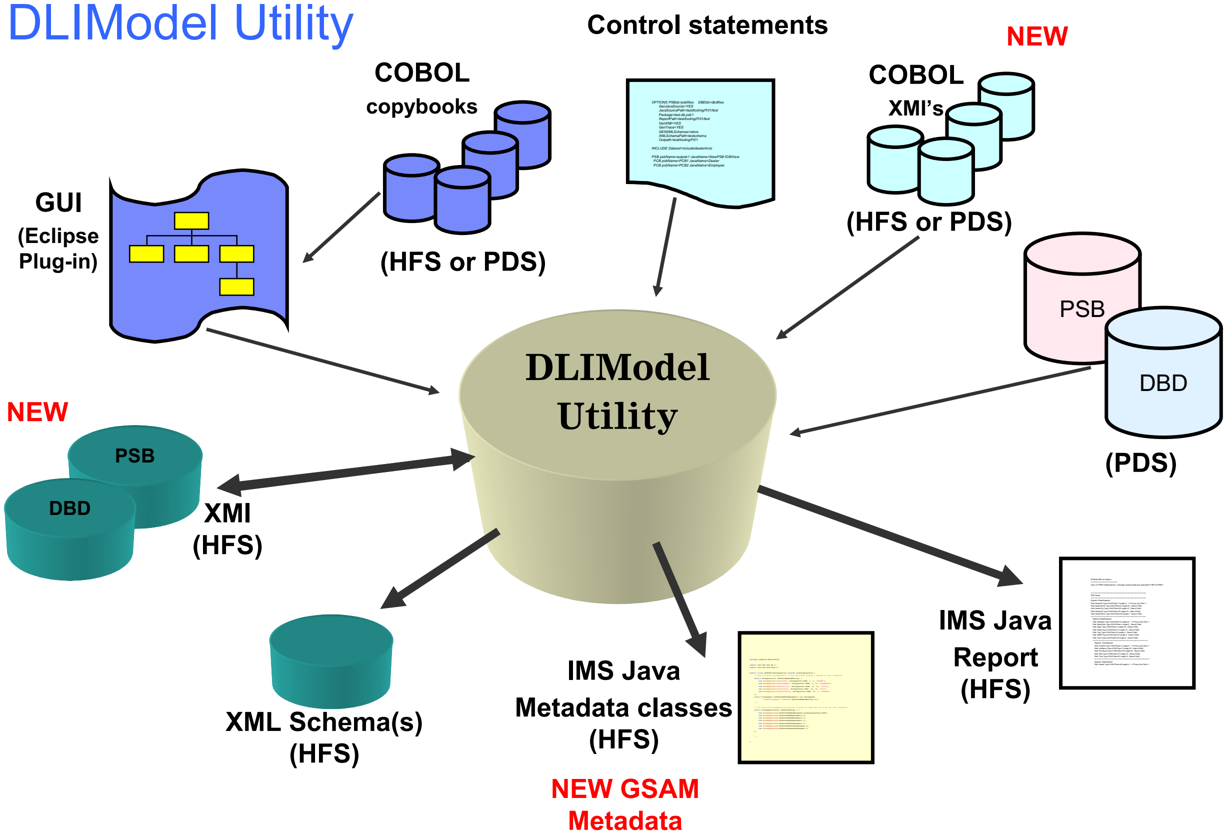

Control statements
NEW
COBOL
copybooks
COBOL
XMI's
GUI
(Eclipse
Plug-in)
(HFS or PDS)
(HFS or PDS)
PSB
DBD
DLIModel
Utility
NEW
PSB
DBD
XMI
(HFS)
(PDS)
IMS Java
Report
(HFS)
XML Schema(s)
(HFS)
IMS Java
Metadata classes
(HFS)
NEW GSAM
Metadata

Figure 13-49. DLIModel Utility CMA01.0

Notes:

The IMS DLIModel utility has been enhanced to generate XMI from PSB and DBD source. The generated XMI can also be used as input to the DLIModel utility.

GSAM now uses the GSAMDLIDatabaseView IMS Java class for metadata information about the GSAM database. The DLIModel Utility now supports GSAM databases.

Input:

Shows the inputs and outputs of the DLIModel utility. The actions of the utility are directed by control statements that you supply. PSB and DBD source members are read from their PDS or PDSE data sets and parsed by the utility to build an in-memory object model of the database structure and the PSB's view of that structure.

Note IMS COBOL copybooks can only be processed by the GUI and the BPXBATCH utility can only process COBOL XMI representations of the COBOL copybooks.

Output:

The utility generates various outputs that were requested through control statements. You can specify to have an IMS Java metadata class be generated for the PSB processed, together with a corresponding easy-to-read DLIModel Java Report for the Java programmer to use.

You can specify an XMI description of the entire in-memory model (one description covers the PSB and all DBDs processed in the run).

You can also request a detailed trace file of the utility execution if one is necessary for problem resolution.

IMS Java metadata classes

The DLIModel utility produces the necessary metadata classes needed to develop IMS Java applications. However, the Java developer needs only to reference the DLIModel Java Report for information about the classes.

DLI Model Java Report

The DLIModel Java Report summarizes the structure of the IMS databases in a way that allows you to create IMS Java applications and to code SQL queries against the databases. With the DLIModel Java Report, you do not have to interpret the syntax of the IMS Java classes or refer to the DBD or PSB source.

XMI Description of databases

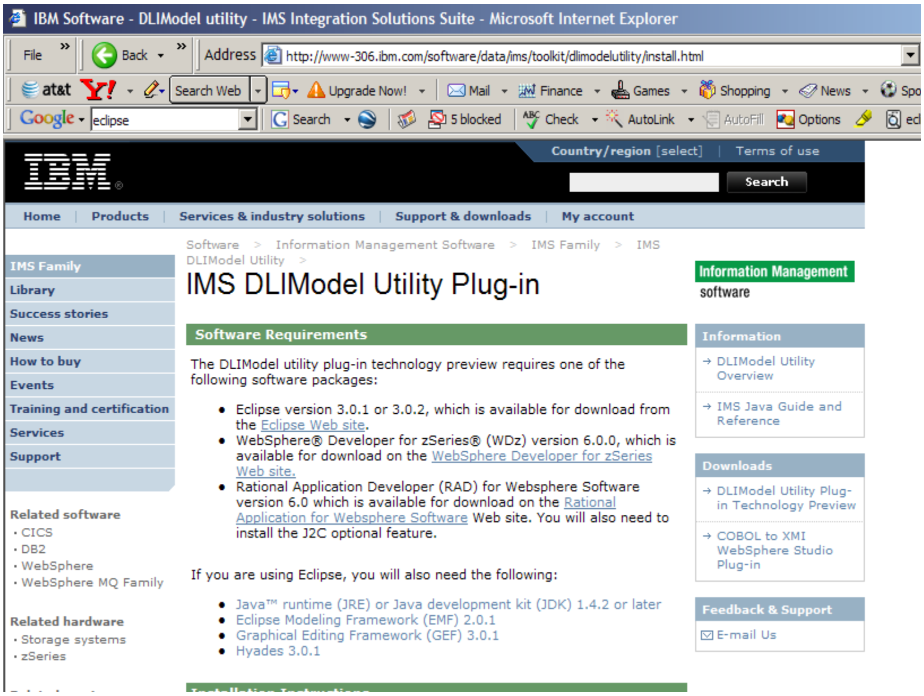
An XMI file, written in UTF-8 encoding, is produced by the utility if you specify genXMI=YES in the OPTIONS control statement. It describes all of the PCBs and their referenced DBDs processed in the entire run of the utility. The XMI that is produced by the utility is based on a metamodel of IMS database defined in UML. This model is a package with a number of inheritance relationships to the OMG Common Warehouse Metamodel (CWM). However, only the IMS package itself is included and used in the DLIModel utility.

XML Schema

The generated XML schema, written in UTF-8 encoding, is an XML document that describes an IMS database based on a PCB. An XML schema is required to retrieve or store XML in IMS. IMS uses an XML schema to validate an XML document that is either being stored into IMS or being retrieved from IMS. The XML schema, not the application program, determines the structural layout of the parsed XML document in the database during storage and the of the generated XML document during retrieval.

Information On Demand
IMS Version 10
IBM

DLIModel Utility GUI Download


File
Back
at&t
Search Web
Upgrade Now!
Mail
Finance
Games
Shopping
News
Spo
Google
eclipse
Search
5 blocked
Check
AutoLink
AutoFill
Options
ed
IBM
Country/region [select]
Terms of use
Search
Home
Products
Services & industry solutions
Support & downloads
My account
IMS Family
Library
Success stories
News
How to buy
Events
Training and certification
Services
Support
Information Management
software

IBM Software - DLIModel utility - IMS Integration Solutions Suite - Microsoft Internet Explorer

Address http://www-306.ibm.com/software/data/ims/toolkit/dlimodelutility/install.html

Software > Information Management Software > IMS Family > IMS DLIModel Utility >

IMS DLIModel Utility Plug-in

Software Requirements

The DLIModel utility plug-in technology preview requires one of the following software packages:

- Eclipse version 3.0.1 or 3.0.2, which is available for download from the Eclipse Web site.
- WebSphere® Developer for zSeries® (WDz) version 6.0.0, which is available for download on the WebSphere Developer for zSeries Web site.
- Rational Application Developer (RAD) for Websphere Software version 6.0 which is available for download on the Rational Application for Websphere Software Web site. You will also need to install the J2C optional feature.

If you are using Eclipse, you will also need the following:

- Java™ runtime (JRE) or Java development kit (JDK) 1.4.2 or later
- Eclipse Modeling Framework (EMF) 2.0.1
- Graphical Editing Framework (GEF) 3.0.1
- Hyades 3.0.1

Related software
- CICS
- DB2
- WebSphere
- WebSphere MQ Family

Related hardware
- Storage systems
- zSeries

Information
- → DLIModel Utility Overview
- → IMS Java Guide and Reference

Downloads
- → DLIModel Utility Plug-in Technology Preview
- → COBOL to XMI WebSphere Studio Plug-in

Feedback & Support
- E-mail Us

Figure 13-50. DLIModel Utility GUI Download CMA01.0

Notes:

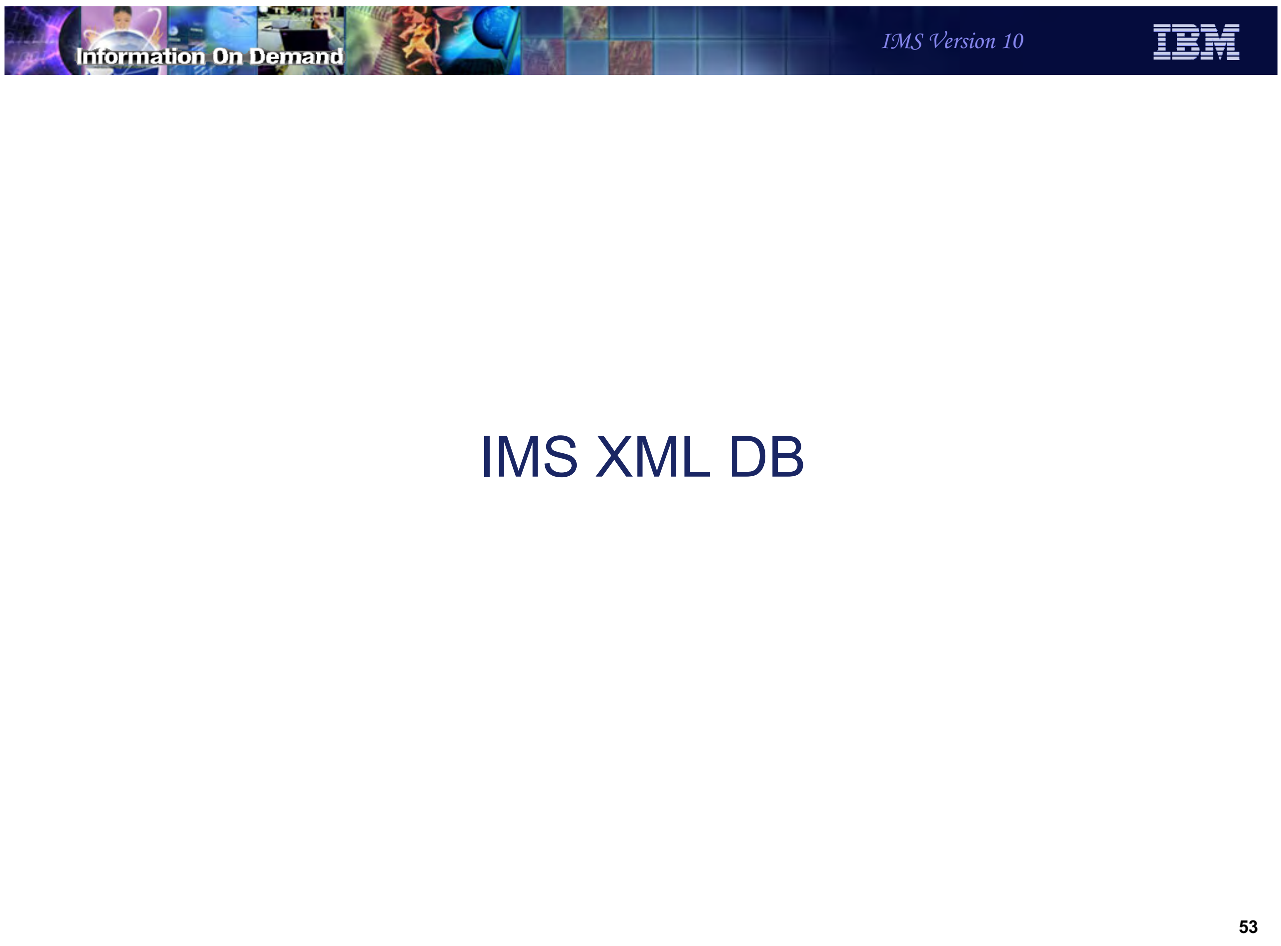
This is the IMS web site to download the DLIModel Utility Plug-in

DLIModel Utility GUI

- Migration
  - Eclipse version 3.0.1 or 3.0.2
  - Download DLIModel Utility plug-in
- Benefits
  - Direct import of COBOL Copybook
  - IMS Perspective for Integrated Development Environment

Figure 13-51. DLIModel Utility GUI CMA01.0

Notes:



IMS XML DB

Figure 13-52. [No title] CMA01.0

Notes:

IMS XML DB allows applications to view a traditional IMS database as an XML database and to use an IMS database to store XML documents.

XML Database - Review

- Introduces a way to view/map native IMS hierarchical data to XML documents
  - Aligns IMS Database (DBD) with XML Schema
- Allows the retrieval and storage of IMS Records as XML documents with no change to existing IMS databases
  - Enables query of IMS data using XQuery

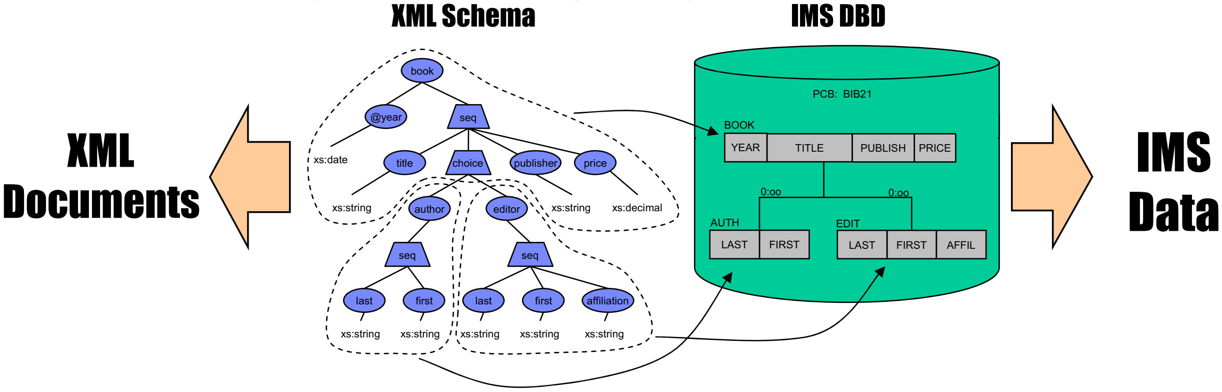
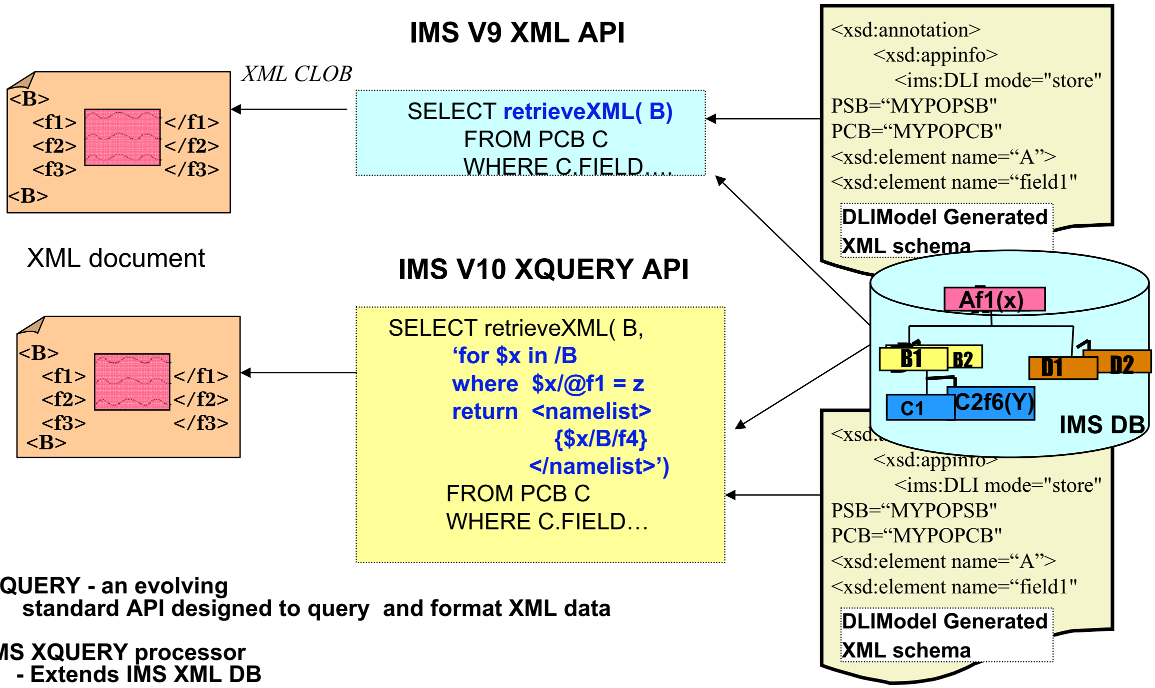

XML Schema
IMS DBD
XML Documents
IMS Data
book
@year
seq
xs:date
title
choice
publisher
price
xs:string
author
editor
xs:string
xs:decimal
seq
seq
last
first
last
first
affiliation
xs:string
xs:string
xs:string
xs:string
xs:string
PCB: BIB21
BOOK
YEAR
TITLE
PUBLISH
PRICE
0:oo
0:oo
AUTH
LAST
FIRST
EDIT
LAST
FIRST
AFFIL

Figure 13-53. XML Database - Review CMA01.0

Notes:

Information On Demand
IMS Version 10
IBM

IMS DB Resource Adapter XML APIs



IMS V9 XML API

XML CLOB

```
<B>
  <f1>  </f1>
  <f2>  </f2>
  <f3>  </f3>
<B>
```

SELECT **retrieveXML(B)**
FROM PCB C
WHERE C.FIELD….

XML document

IMS V10 XQUERY API

```
<B>
  <f1>  </f1>
  <f2>  </f2>
  <f3>  </f3>
<B>
```

SELECT retrieveXML(B,
'for $x in /B
where $x/@f1 = z
return <namelist>
{$x/B/f4}
</namelist>')
FROM PCB C
WHERE C.FIELD…

```
<xsd:annotation>
    <xsd:appinfo>
        <ims:DLI mode="store"
PSB="MYOPSB"
PCB="MYOPCB"
<xsd:element name="A">
<xsd:element name="field1"
```

DLIModel Generated XML schema

```
<xsd:...
    <xsd:appinfo>
        <ims:DLI mode="store"
PSB="MYOPSB"
PCB="MYOPCB"
<xsd:element name="A">
<xsd:element name="field1"
```

DLIModel Generated XML schema

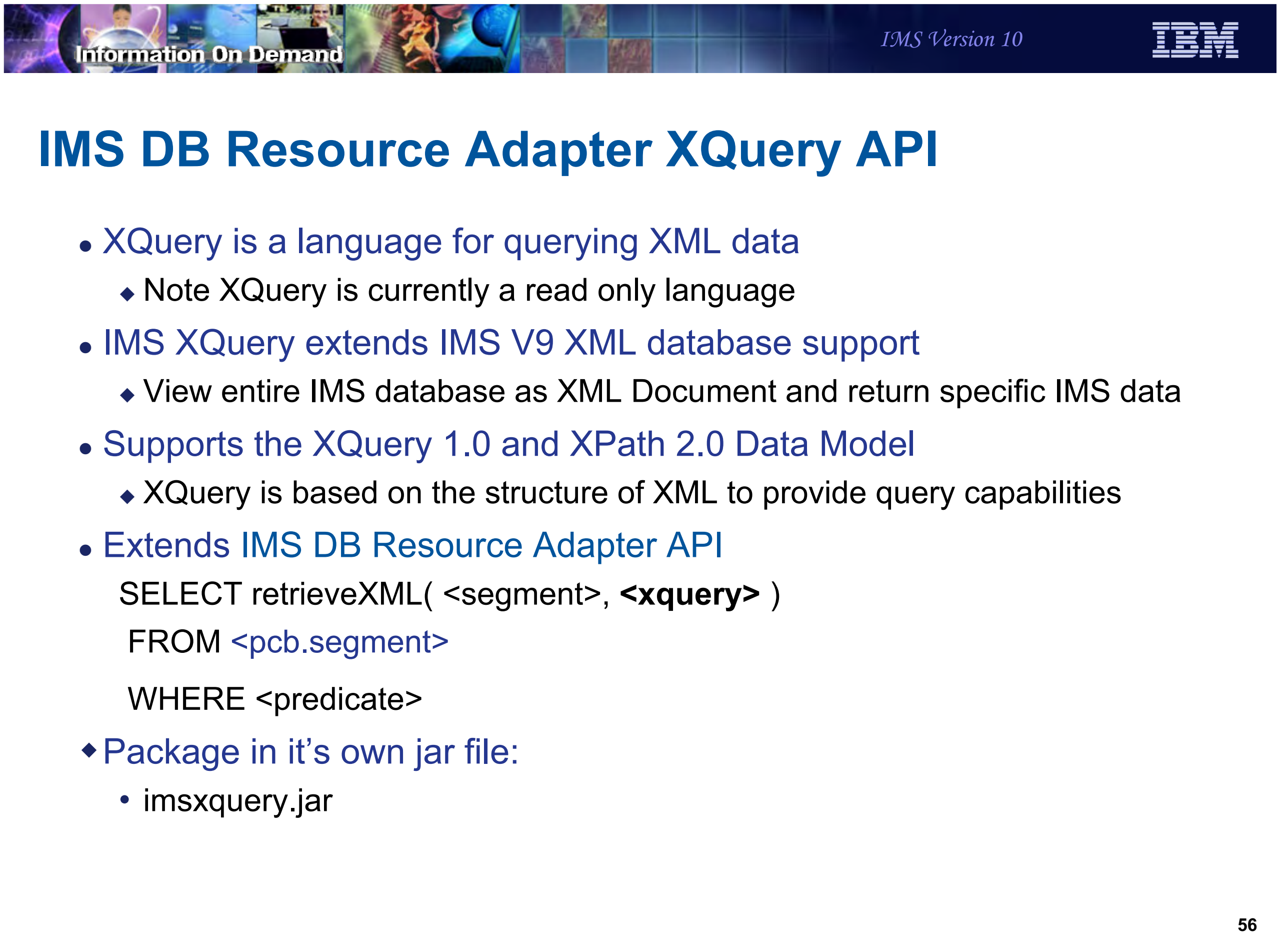
Af1(x)
B1
B2
D1
D2
C1
C2f6(Y)
IMS DB

XQUERY - an evolving standard API designed to query and format XML data

IMS XQUERY processor
- **Extends IMS XML DB**
- **Uses XML schemas generated by DLIModel**

Figure 13-54. [No title] CMA01.0

Notes:



IMS DB Resource Adapter XQuery API

- XQuery is a language for querying XML data
  - Note XQuery is currently a read only language
- IMS XQuery extends IMS V9 XML database support
  - View entire IMS database as XML Document and return specific IMS data
- Supports the XQuery 1.0 and XPath 2.0 Data Model
  - XQuery is based on the structure of XML to provide query capabilities
- Extends IMS DB Resource Adapter API

```
SELECT retrieveXML( <segment>, <xquery> )
 FROM <pcb.segment>
 WHERE <predicate>
```

- Package in it’s own jar file:
  - imsxquery.jar

Figure 13-55. IMS DB Resource Adapter XQuery API CMA01.0

Notes:

XQuery is a functional programming language that was designed by the World Wide Web Consortium (W3C) to meet specific requirements for querying XML data.

XQuery is based on the structure of XML and leverages this structure to provide query capabilities for the same range of data that XML stores.

The IMS DB Resource Adapter XQuery support extends the *retrieveXML* User Defined Function (UDF) by adding a second parameter. The second parameter allows the passing of an XQuery 1.0 expression.

The expression is evaluated relative to the retrieveXML context and returned to the result set as a CLOB value.

This implementation views the entire IMS DB as an XML document and enables the return of specific IMS data based on the XQuery.

For IMS XQuery support the XQuery 1.0 and XPath 2.0 Data Model serves two purposes.

First, it defines the information contained in the input to be used by the IMS XQuery processor.

Second, it defines all permissible values of expressions in the XQuery, and XPath languages that can be evaluated by the IMS XQuery processor.

The IMS DB Resource Adapter is packaged in imsjava.jar.

The IMS XQuery function resides in a separate package (imsxquery.jar).

Information On Demand
IMS Version 10
IBM

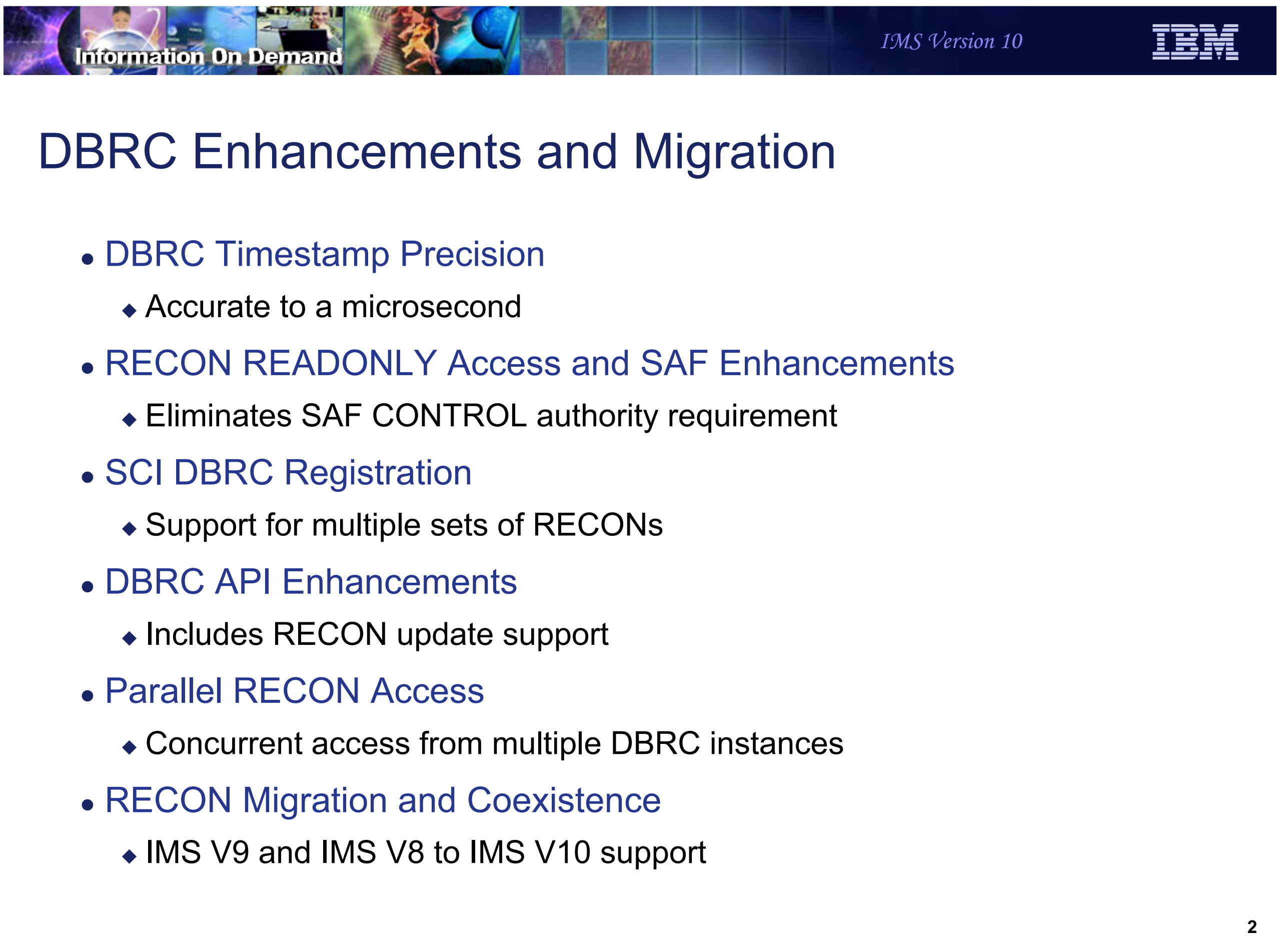
IMS DB Resource Adapter XQuery

- Benefits
  - IMS participates in industry standards
  - Can use Application Development tools that support XQuery
  - Can be used with existing IMS data and IMS XML Schemas

Figure 13-56. IMS DB Resource Adapter XQuery CMA01.0

Notes:

Unit 14.DBRC



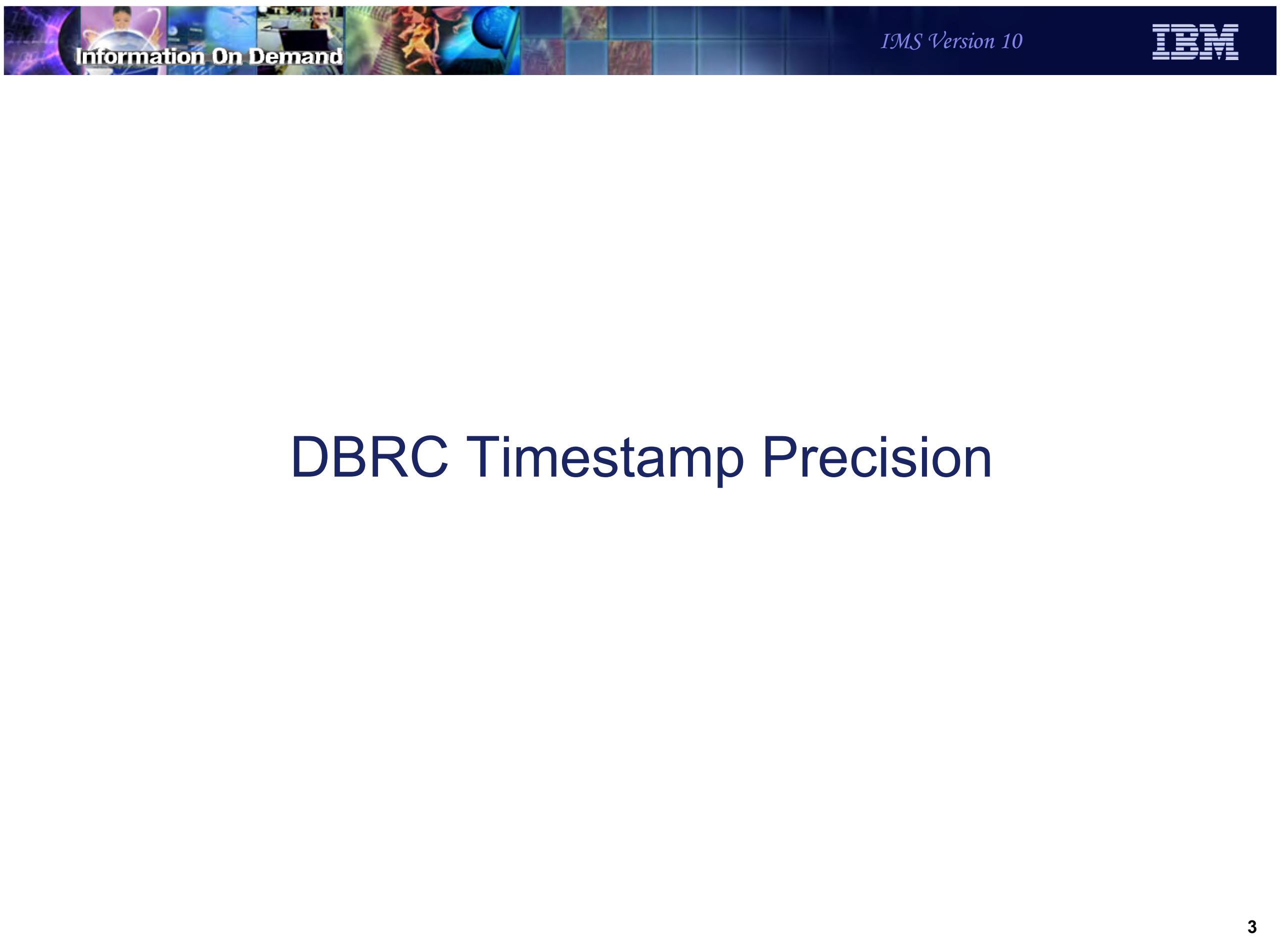
Information On Demand
IMS Version 10
IBM

DBRC Enhancements and Migration

- DBRC Timestamp Precision
  - Accurate to a microsecond
- RECON READONLY Access and SAF Enhancements
  - Eliminates SAF CONTROL authority requirement
- SCI DBRC Registration
  - Support for multiple sets of RECONs
- DBRC API Enhancements
  - Includes RECON update support
- Parallel RECON Access
  - Concurrent access from multiple DBRC instances
- RECON Migration and Coexistence
  - IMS V9 and IMS V8 to IMS V10 support

Figure 14-1. DBRC Enhancements and Migration CMA01.0

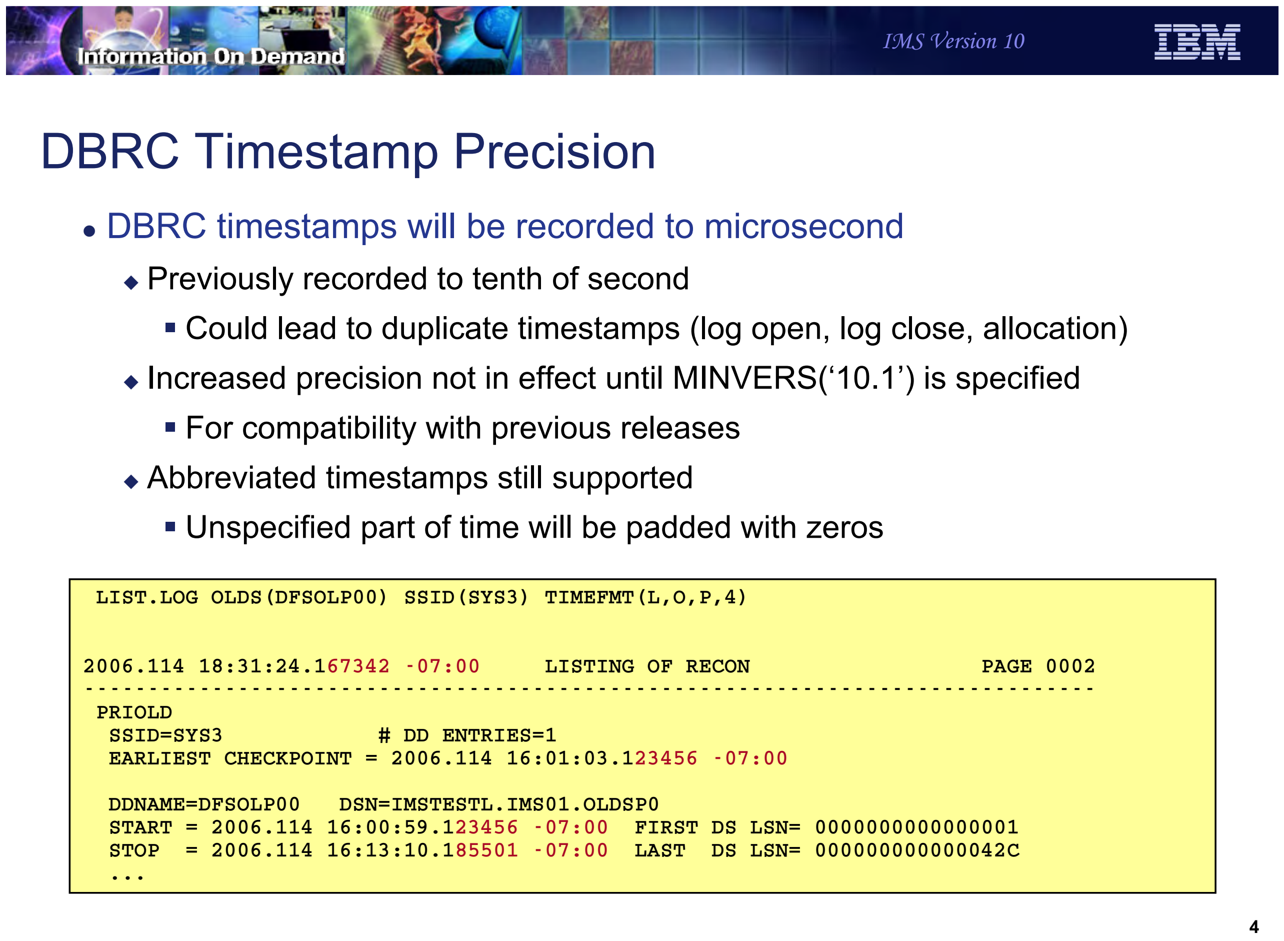
Notes:


Information On Demand
IMS Version 10
IBM

DBRC Timestamp Precision

Figure 14-2. [No title] CMA01.0

Notes:



DBRC Timestamp Precision

- DBRC timestamps will be recorded to microsecond
  - Previously recorded to tenth of second
    - Could lead to duplicate timestamps (log open, log close, allocation)
  - Increased precision not in effect until MINVERS(‘10.1’) is specified
    - For compatibility with previous releases
  - Abbreviated timestamps still supported
    - Unspecified part of time will be padded with zeros

```
 LIST.LOG OLDS(DFSOLP00) SSID(SYS3) TIMEFMT(L,O,P,4)


2006.114 18:31:24.167342 -07:00     LISTING OF RECON                  PAGE 0002
--------------------------------------------------------------------------------
 PRIOLD
  SSID=SYS3             # DD ENTRIES=1
  EARLIEST CHECKPOINT = 2006.114 16:01:03.123456 -07:00

  DDNAME=DFSOLP00   DSN=IMSTESTL.IMS01.OLDSP0
  START = 2006.114 16:00:59.123456 -07:00  FIRST DS LSN= 0000000000000001
  STOP  = 2006.114 16:13:10.185501 -07:00  LAST  DS LSN= 000000000000042C
  ...
```

Figure 14-3. DBRC Timestamp Precision CMA01.0

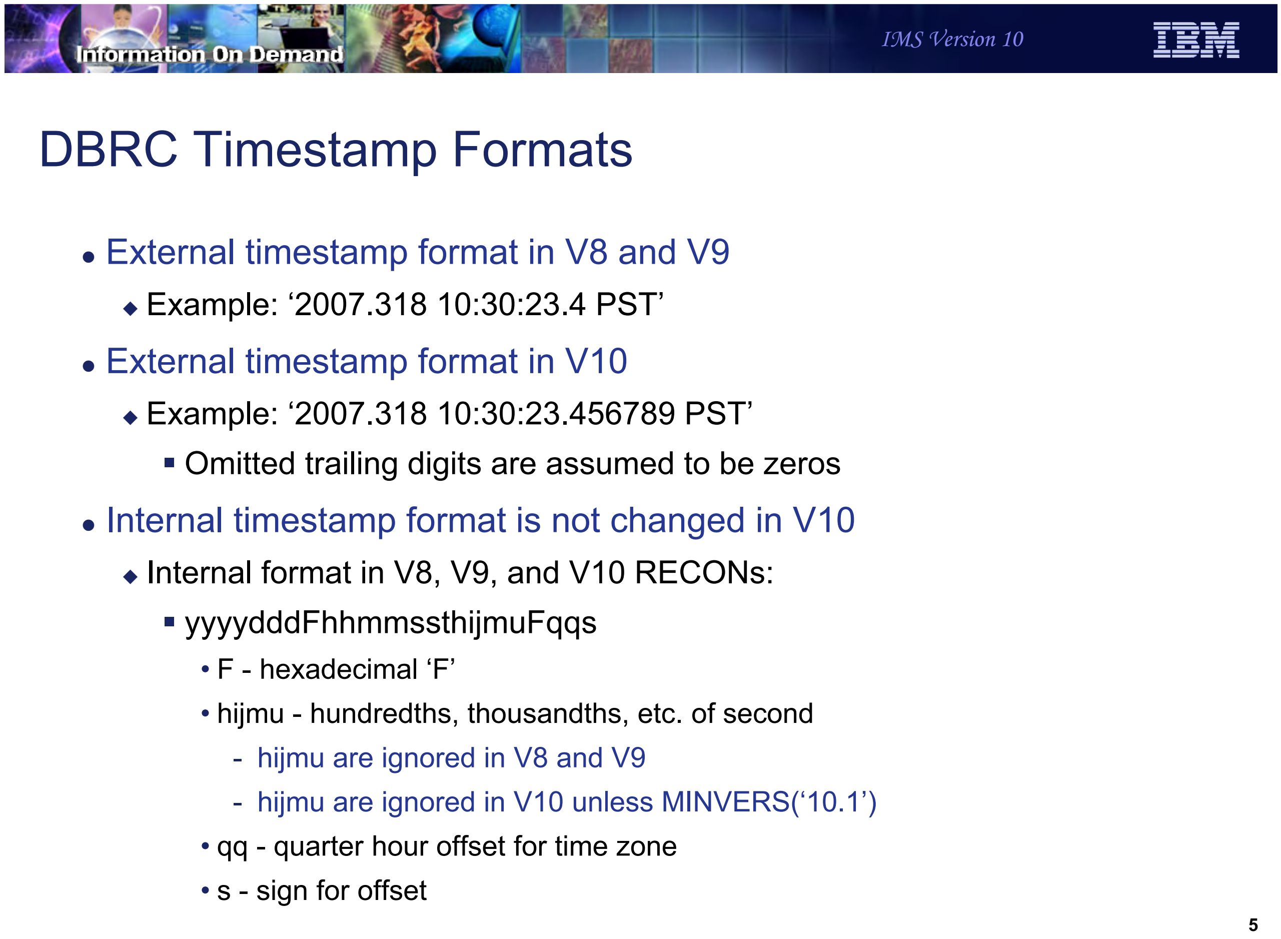
Notes:

IMS V10 will record timestamps to the microsecond when MINVERS('10.1') is in effect. Previous releases record timestamps to the tenth of a second. When a user specifies a timestamp, it may be abbreviated. That means that times to the microsecond do not have to be specified. Unspecified parts of the time are padded with zeros.

When using GENJCL, the precision value may be coded on the TIMEFMT parameter of the %SET statement in skeletal JCL. It is a value from 1 to 6. The default in previous releases was 1. In IMS V10 the default depends on the MINVERS value. MINVERS('10.1') sets the default to 6. MINVERS values less than '10.1' sets the default to 1.

The output from a DBRC LIST command includes full precision timestamps. In previous releases, the timestamp in the listing included only tenths of seconds. If you request the time zone offset to be listed, it follows the timestamp. Since the timestamp is larger by five bytes, the offset is moved by five bytes. In general there were at least five blanks following timestamps in listings from previous releases. This means that other information did not have to be moved for full precision timestamps in V10. The exception is the timeline information in the listing created by a LIST.HISTORY command. The timeline at the end of

LIST.HISTORY outputs still has only tenths of seconds. There was not room to add the additional five bytes to these timestamps in the listing.


Information On Demand
IMS Version 10
IBM

DBRC Timestamp Formats

- External timestamp format in V8 and V9
  - Example: ‘2007.318 10:30:23.4 PST’
- External timestamp format in V10
  - Example: ‘2007.318 10:30:23.456789 PST’
    - Omitted trailing digits are assumed to be zeros
- Internal timestamp format is not changed in V10
  - Internal format in V8, V9, and V10 RECONs:
    - yyyydddFhhmmssthijmuFqqs
      - F - hexadecimal ‘F’
      - hijmu - hundredths, thousandths, etc. of second
        - hijmu are ignored in V8 and V9
        - hijmu are ignored in V10 unless MINVERS(‘10.1’)
      - qq - quarter hour offset for time zone
      - s - sign for offset

Figure 14-4. DBRC Timestamp Formats CMA01.0

Notes:

The timestamp used in DBRC commands and utility control statements is expanded in V10. In previous releases it may include times to one tenth of a second. In V10 it may include times to a microsecond.

Compressed and punctuated timestamp formats continue to be used. The change is the additional significant digits for hundredths through millionths of a second. The examples shown here are punctuated timestamps. The corresponding compressed timestamps would be:

V8 or V9: 063181030234 -8

V10: 06318103023456789 -8

The internal format of the timestamp in the RECONs is not changed in V10. The format has included microseconds since IMS V6. The PRILOG start times have had zeros in these positions. Most other RECON records have had actual values. Nevertheless, they have been ignored by other processing. IMS V10 stores actual values in all time fields.

When MINVERS(‘10.1’) is used with IMS V10, these positions become significant and are not ignored.

The internal timestamp format is used by the DBRC API. It has not changed in V10.

DBRC Commands

- Abbreviated timestamps are supported
  - Year and day are required
  - Other values are padded with zeros
  - Examples:

```
LIST.LOG TOTIME('2007.178')
LIST.LOG TOTIME('2007.178 16:23')
```

- Commands using timestamps to identify a record or information within a record require full precision
  - Example:

```
CHANGE.IC DBD(ABC) DDN(ABC01) -
RECTIME('2007.178 16:23:31.123456 -08:00') -
ICDSN(NEW.DSN)
```

Figure 14-5. DBRC Commands CMA01.0

Notes:

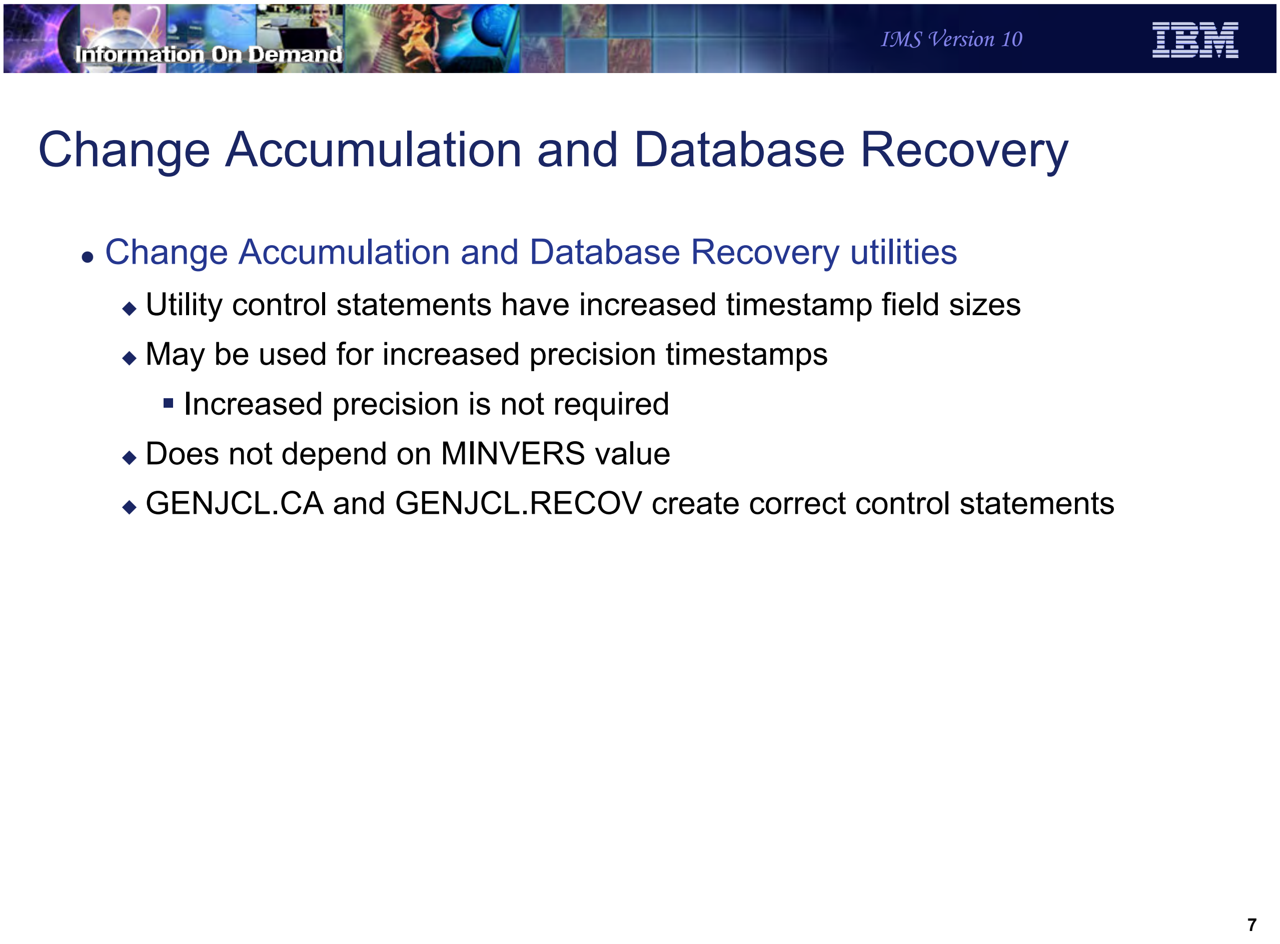
The first example lists all log records or OLDS entries that have start times on or before day 178 in 2007. Full precision is not required on the TOTIME value. The next example lists all log records or OLDS entries that have start time on or before day 16:28 (4:28 p.m.) on day 178 in 2007.

Many RECON records include the timestamp as their keys. When a RECON record is recorded with a full precision timestamp, commands for that specific record require full precision in their timestamps. The second example shows a command where the full precision timestamp is required. An image copy record was created with a full precision timestamp. The timestamp is part of the key. The CHANGE.IC command in the example changes the data set name of the image copy of the database data set with DDNAME ABC01 in the ABC database which was run at the time specified in the RECTIME parameter.

Commands requiring full precision timestamps are

- CHANGE.BKOUT, CHANGE.CA, CHANGE.IC, CHANGE.PRILOG RLDS, CHANGE.PRILOG SLDS, CHANGE.PRILOG TSLDS, CHANGE.SECLOG RLDS, CHANGE.SECLOG SLDS, CHANGE.SECLOG TSLDS, and CHANGE.UIC
- DELETE.CA, DELETE.IC, DELETE.RECOV, DELETE.REORG, and DELETE.UIC

If you enter a command with a timestamp that has with less than six digits for microseconds, DBRC uses zeros for the missing digits. Since NOTIFY commands create new information in the RECONs, their timestamps do not have to match existing data. You can use timestamps without all six digits for microseconds for NOTIFY commands. Zeros will be used for any of the missing digits.



Information On Demand
IMS Version 10

Change Accumulation and Database Recovery

- Change Accumulation and Database Recovery utilities
  - Utility control statements have increased timestamp field sizes
  - May be used for increased precision timestamps
    - Increased precision is not required
  - Does not depend on MINVERS value
  - GENJCL.CA and GENJCL.RECOV create correct control statements

Figure 14-6. Change Accumulation and Database Recovery CMA01.0

Notes:

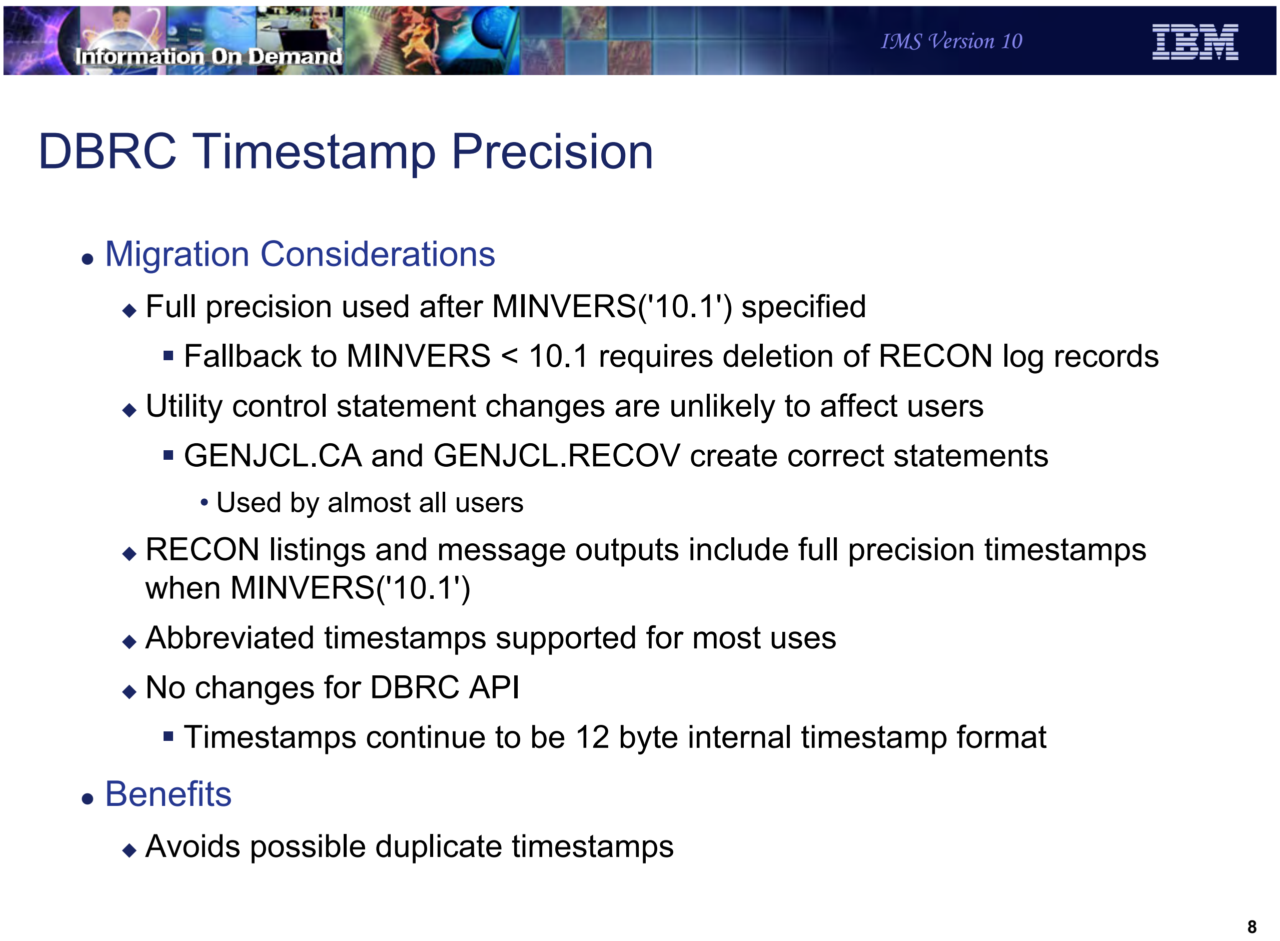
The Database Change Accumulation utility DB0 and DB1 control statements have been modified to support timestamps with greater precision. The new expanded timestamp format may be used in V10 but it is not required. DB0 control statements are used to specify the database data sets that are accumulated to the new change accumulation data set. DB1 control statements are used to specify the database data sets that are written to the new output log data set. Since the timestamps now may occupy more columns in the control statements, the position of the database data set DDNAMEs have moved and only three may be specified on one control statement. Previous IMS versions allowed four to be specified on a control statement.

The S control statement for the Database Recovery utility is used to specify the database and DDNAME for the database data set that is to be recovered. If the recovery is a timestamp recovery, the timestamp is also specified. The new expanded timestamp format may be used in V10 but it is not required. In previous releases column 57 was used for an indicator. The indicator could specify that a user image copy had been restored or that an

RSR receive was done. Since the timestamp may use column 57, the indicator is now coded in column 63 when needed. There is another small change in the coding of the timestamp. In previous releases there was not a space between the time and the sign for the time zone offset.

GENJCL.CA has been updated to create the new format of the control statements. These changes to the control statements will have no effects on users who create Change Accumulation JCL and control statements with GENJCL.CA. This is the vast majority of IMS installations.

GENJCL.RECOV has been updated to create the new format of the control statement for timestamp recovery. The change in the control statement will have no effects on users who create Database Recovery JCL and control statements with GENJCL.RECOV. This is the vast majority of IMS installations.


Information On Demand
IMS Version 10
IBM

DBRC Timestamp Precision

- Migration Considerations
  - Full precision used after MINVERS('10.1') specified
    - Fallback to MINVERS < 10.1 requires deletion of RECON log records
  - Utility control statement changes are unlikely to affect users
    - GENJCL.CA and GENJCL.RECOV create correct statements
      - Used by almost all users
  - RECON listings and message outputs include full precision timestamps when MINVERS('10.1')
  - Abbreviated timestamps supported for most uses
  - No changes for DBRC API
    - Timestamps continue to be 12 byte internal timestamp format
- Benefits
  - Avoids possible duplicate timestamps

Figure 14-7. DBRC Timestamp Precision CMA01.0

Notes:

Full precision timestamps are not implemented unless the RECONs have MINVERS('10.1') specified. Even when MINVERS is set to a lower value, the IMS V10 Change Accumulation and Database Recovery utility control statements require new formats which accommodate full precision timestamps. Nevertheless, this is unlikely to be a concern to users since GENJCL.CA and GENJCL.RECOV in IMS V10 always produce control statements with the V10 formats.

If you wish, you may specify abbreviated timestamps for most uses. DBRC will interpret the time correctly. Full precision timestamps are required in CHANGE and DELETE commands when a full precision timestamp is part of the RECON record key.

When MINVERS('10.1') is set log records are created with the full precision timestamps. If you fallback to MINVERS('9.1') or MINVERS('8.1'), the log records created with full precision timestamps must be deleted before the CHANGE.RECON MINVERS(…) command is issued. The records that must be deleted include PRILOG, PRISLD, PRIOLD, etc. and the ALLOC records associated with these log records.

Information On Demand
IMS Version 10
IBM

RECON READONLY Access and SAF Enhancements

Figure 14-8. [No title]

CMA01.0

Notes:

SAF (RACF) Support for RECONs

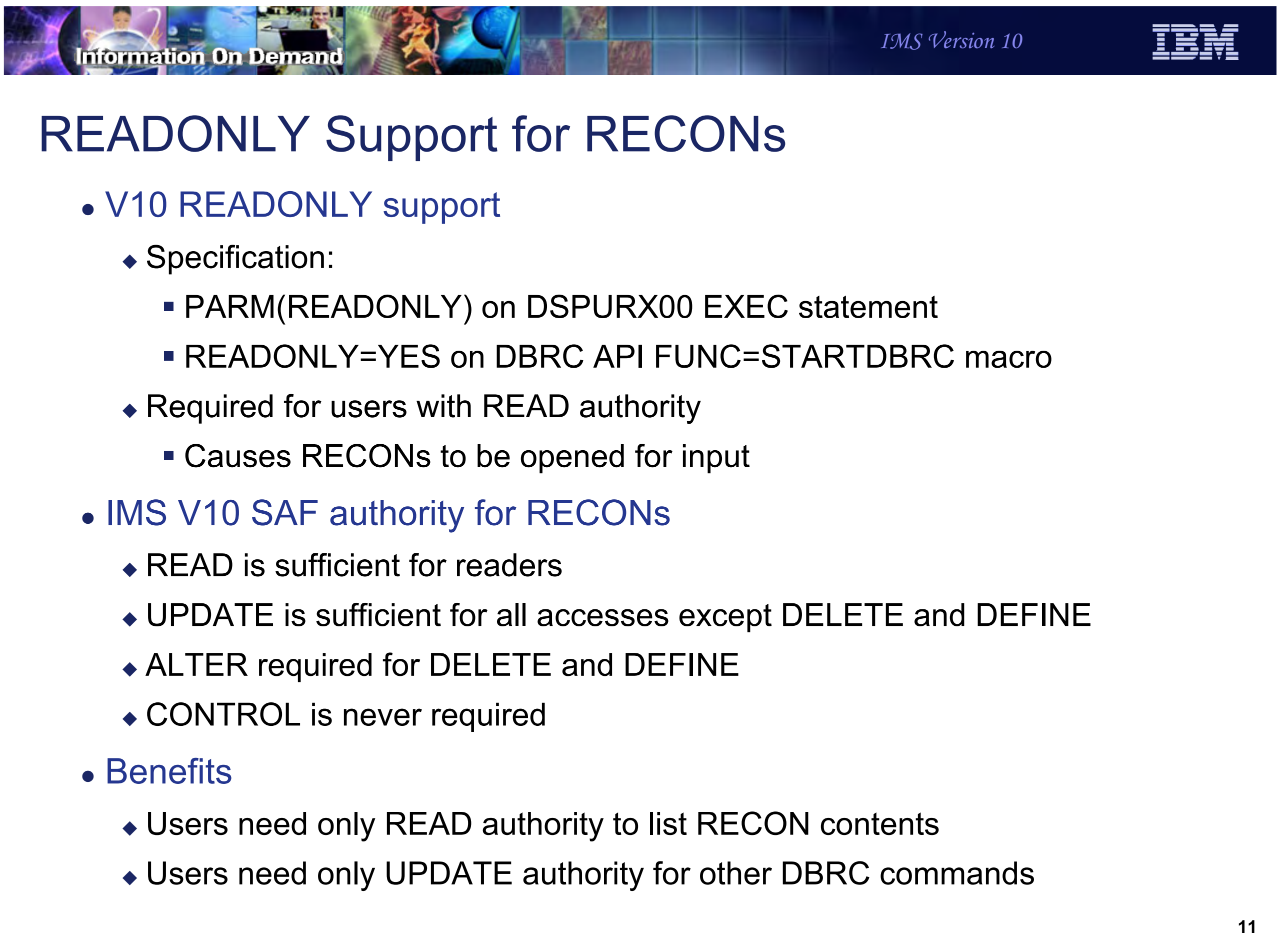
- SAF supports four levels of data set authority
    - READ, UPDATE, CONTROL, and ALTER
- Previous releases of IMS
    - CONTROL or higher (ALTER) was required for RECONs
        - ALTER required for DELETE and DEFINE
    - Users only wanting to read data from RECONs required CONTROL authority

Figure 14-9. SAF (RACF) Support for RECONs CMA01.0

Notes:

System Authorization Facility (SAF) products, such as RACF, support four levels of data set authority. In ascending sequence of authority these are READ, UPDATE, CONTROL, and ALTER. Previous releases of IMS required at least CONTROL authority for all users of the RECONs. IDCAMS DEFINE and DELETE of a RECON data set required ALTER.

Previous IMS releases opened the RECONs for update with CONTROL specified in the VSAM ACB for the RECONs. This required CONTROL authority for open. This was true even if the user only wanted to read the RECONs as would be done for a LIST command.


Information On Demand
IMS Version 10
IBM

READONLY Support for RECONs

- V10 READONLY support
  - Specification:
    - PARM(READONLY) on DSPURX00 EXEC statement
    - READONLY=YES on DBRC API FUNC=STARTDBRC macro
  - Required for users with READ authority
    - Causes RECONs to be opened for input
- IMS V10 SAF authority for RECONs
  - READ is sufficient for readers
  - UPDATE is sufficient for all accesses except DELETE and DEFINE
  - ALTER required for DELETE and DEFINE
  - CONTROL is never required
- Benefits
  - Users need only READ authority to list RECON contents
  - Users need only UPDATE authority for other DBRC commands

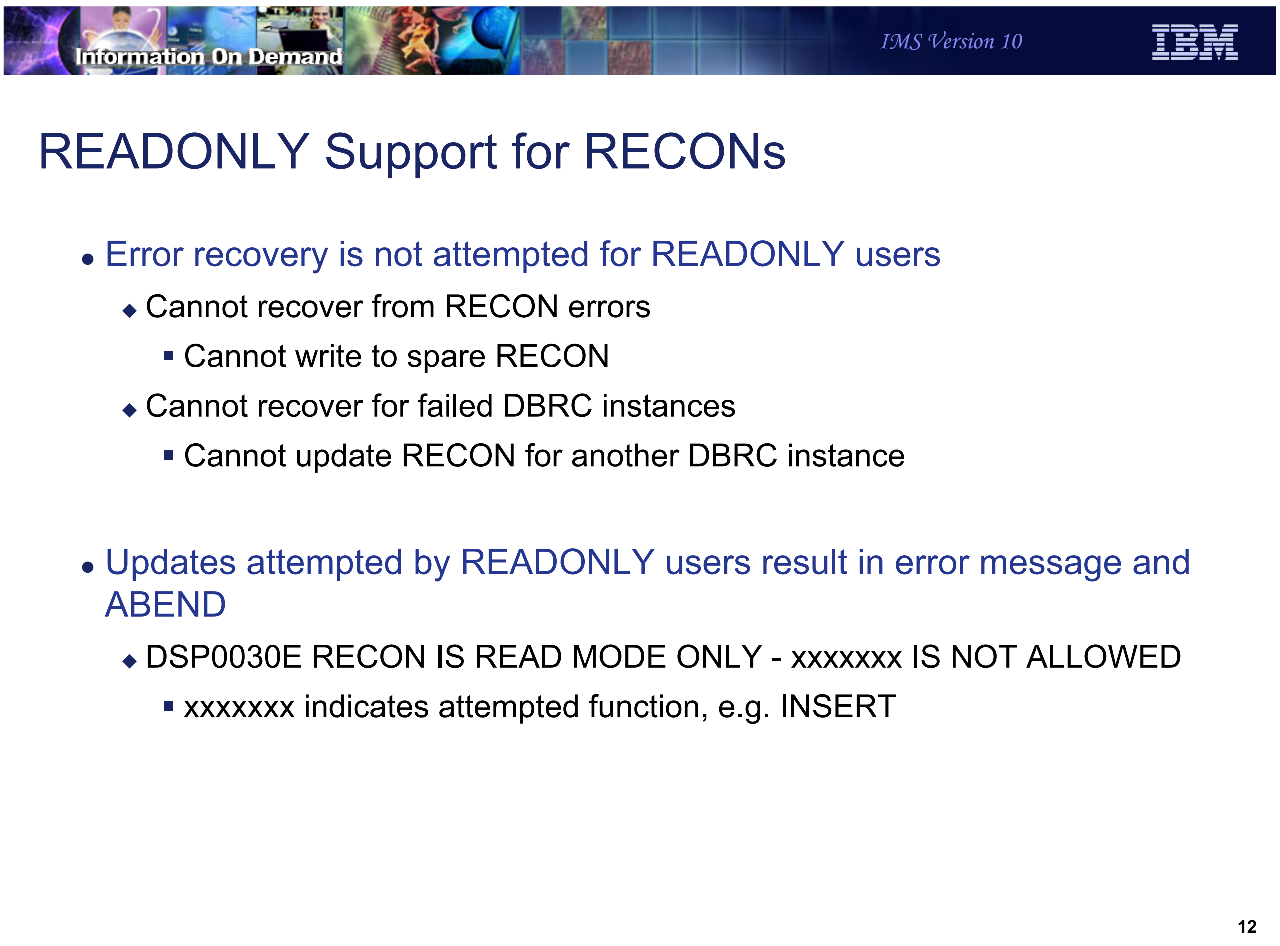
Figure 14-10. READONLY Support for RECONs CMA01.0

Notes:

IMS V10 adds READONLY support for the RECONs. This is invoked with PARM(READONLY) on the EXEC statement for the DBRC utility (DSPURX00) or by specifying the new READONLY=YES parameter on the DBRC API FUNC=STARTDBRC macro. When READONLY is specified, the RECONs are opened for read. This means that only READ authority is required in SAF (RACF).

IMS V10 has made another change to open. Due to this change, CONTROL does not have to be specified for users who update the RECONs. Only UPDATE authority is required. Of course, ALTER is still required for users who DELETE and DEFINE the data sets. In previous releases DBRC opened the RECONs with CONTROL specified in the VSAM ACB for the RECONs. This required CONTROL authority for open. In IMS V10 the open has changed. If READONLY is not specified, the open is done for update but CONTROL is not specified in the ACB. This means that only UPDATE authority is required.

As with previous releases, if you invoke the DBRC utility from your program you may use the DSPURXRT entry point. IMS V10 has added the capability to specify READONLY through a parameter passed to the entry point in the first word of the argument list.


Information On Demand
IMS Version 10
IBM

READONLY Support for RECONs

- Error recovery is not attempted for READONLY users
  - Cannot recover from RECON errors
    - Cannot write to spare RECON
  - Cannot recover for failed DBRC instances
    - Cannot update RECON for another DBRC instance
- Updates attempted by READONLY users result in error message and ABEND
  - DSP0030E RECON IS READ MODE ONLY - xxxxxxx IS NOT ALLOWED
    - xxxxxxx indicates attempted function, e.g. INSERT

Figure 14-11. READONLY Support for RECONs CMA01.0

Notes:

Since READONLY causes IMS to open the RECONs for input, users of READONLY cannot invoke recovery processes for the RECONs. There are two kinds of recovery processes. The first is recovery from RECON errors. If an I/O error occurs on a RECON data set, updaters can reconfigure the RECONs. This includes copying the good RECON to the spare. READONLY users cannot do writes, so they cannot do this recovery process. The second recovery process is recovering from a failed DBRC instance. When a DBRC instance (batch job, utility, or online system) updates multiple RECON records it first writes a Multiple Update Record (MUP) to the RECONs. It then does the updates and, finally, deletes the MUP record. If it fails in the middle of this process, another DBRC instance recovers. The other DBRC reads the MUP record and either completes or backs out the changes. If the other DBRC instance is a READONLY user, it cannot perform this recovery because it cannot write.

If a READONLY execution attempts to update the RECONs, message DSP0030E is issued and the application abends. The variable text in the message indicates the operation that was attempted. The possible values are:

CONFIG An attempt was made to reconfigure the RECON data sets.

DELETE An attempt was made to delete a record from the RECON.

INSERT An attempt was made to create a new RECON record.

UPDATE An attempt was made to update an existing RECON record.

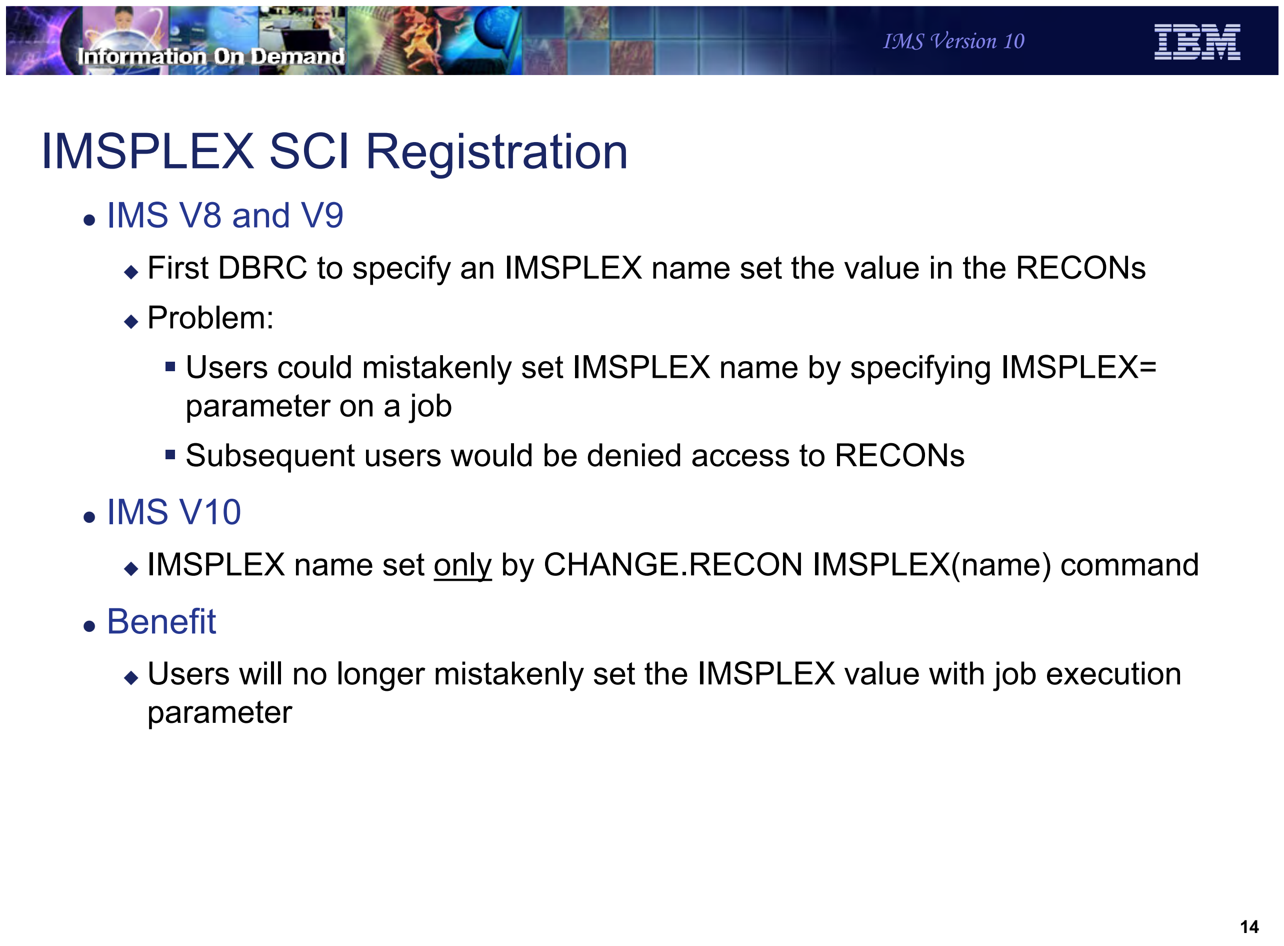
UPGRADE An attempt was made to upgrade the RECON data sets.

MODE-SW An attempt was made to switch accessing mode.

IMSPLEX SCI Registration

Figure 14-12. [No title]

Notes:



Information On Demand

IMS Version 10

IMSPLEX SCI Registration

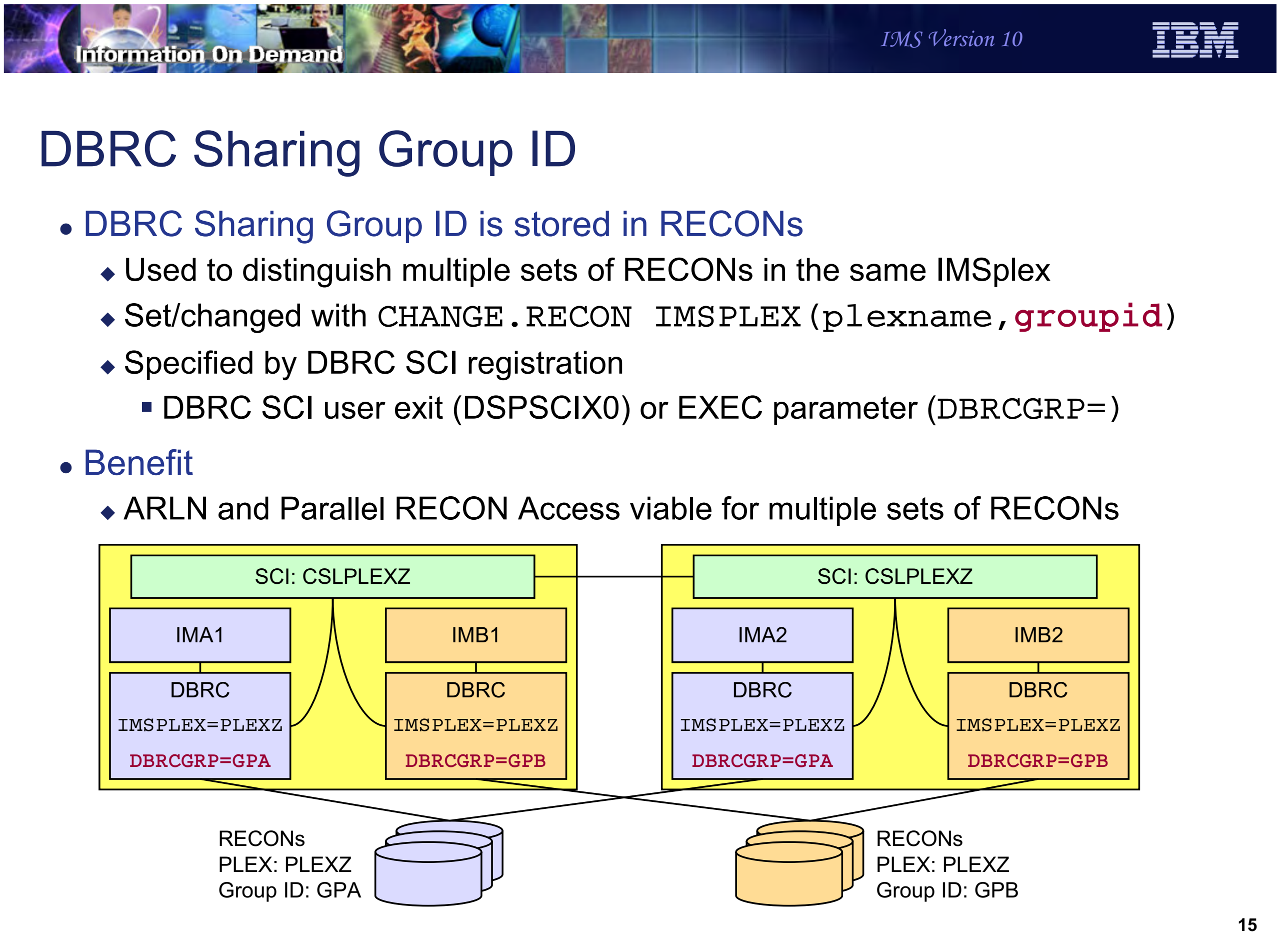
- IMS V8 and V9
  - First DBRC to specify an IMSPLEX name set the value in the RECONs
  - Problem:
    - Users could mistakenly set IMSPLEX name by specifying IMSPLEX= parameter on a job
    - Subsequent users would be denied access to RECONs
- IMS V10
  - IMSPLEX name set only by CHANGE.RECON IMSPLEX(name) command
- Benefit
  - Users will no longer mistakenly set the IMSPLEX value with job execution parameter

Figure 14-13. IMSPLEX SCI Registration CMA01.0

Notes:

When the IMSPLEX name is set in the RECONs, all subsequent users must specify the same IMSPLEX name. They must either include the IMSPLEX= execution parameter with the correct value or the DBRC SCI Registration exit must specify the correct value.

In previous releases some installations reported that users mistakenly specified the IMSPLEX= execution parameter. This caused subsequent jobs to fail until the IMSPLEX name could be removed from the RECONs with a CHANGE.RECON NOPLEX command. The change in IMS V10 will prevent this situation from occurring. Now the IMSPLEX name will only be set by the CHANGE.RECON IMSPLEX(name) command.



Information On Demand
IMS Version 10
IBM

DBRC Sharing Group ID

- DBRC Sharing Group ID is stored in RECONs
  - Used to distinguish multiple sets of RECONs in the same IMSplex
  - Set/changed with `CHANGE.RECON IMSPLEX(plexname,groupid)`
  - Specified by DBRC SCI registration
    - DBRC SCI user exit (DSPSCIX0) or EXEC parameter (`DBRCGRP=`)
- Benefit
  - ARLN and Parallel RECON Access viable for multiple sets of RECONs

SCI: CSLPLEXZ
IMA1
DBRC
IMSPLEX=PLEXZ
DBRCGRP=GPA
IMB1
DBRC
IMSPLEX=PLEXZ
DBRCGRP=GPB
SCI: CSLPLEXZ
IMA2
DBRC
IMSPLEX=PLEXZ
DBRCGRP=GPA
IMB2
DBRC
IMSPLEX=PLEXZ
DBRCGRP=GPB
RECONs
PLEX: PLEXZ
Group ID: GPA
RECONs
PLEX: PLEXZ
Group ID: GPB

Figure 14-14. DBRC Sharing Group ID CMA01.0

Notes:

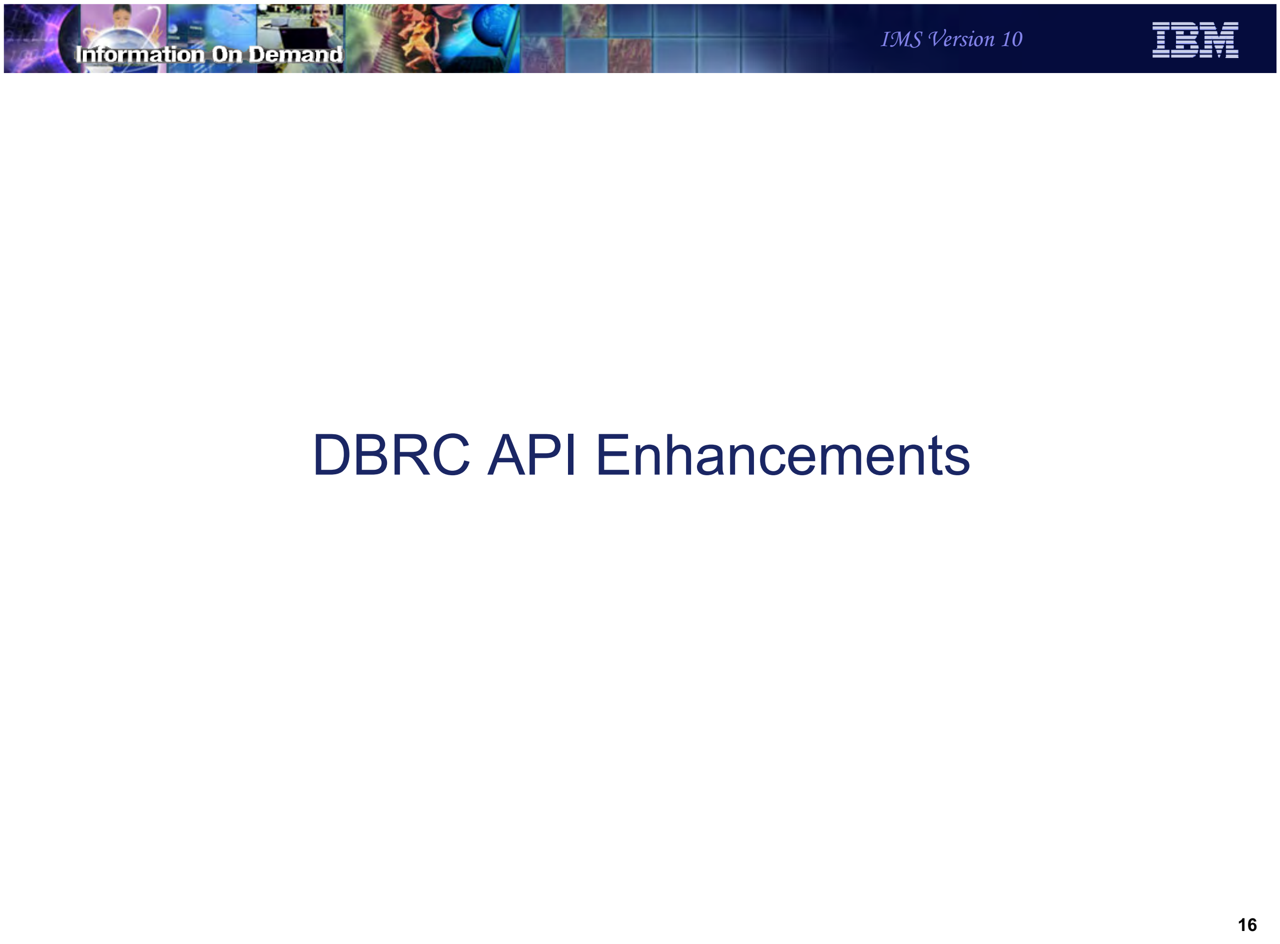
The DBRC Group ID is new in IMS V10. It is used so that Automatic RECON Loss Notification (ARLN) can distinguish between the different sets of RECONs. The IDs ensure that a reconfiguration of the RECONs is only processed by the DBRCs using that set of RECONs.

When upgrading RECONs from a previous release where the plexname was specified, the group ID defaults to 001. If there is no plexname in the RECONs when they are upgraded, the DBRC Group ID is not set.

The DBRC Group ID is set or changed with CHANGE.RECON IMSPLEX(plexname,groupid) command. If the DBRC Group ID is not set before the command is issued, it defaults to 001.

DBRC SCI registration must include the DBRC Group ID along with the IMSplex name.

The example on this slide shows an IMSPLEX with two sets of RECONs. One set uses the DBRC Group ID of GPA. The other set uses GPB. The DBRCs using the GPA RECONs must specify GPA to SCI registration. The DBRCs using the GPB RECONs must specify GPB to SCI registration.



DBRC API Enhancements

Figure 14-15. [No title] CMA01.0

Notes:

IMS V10 introduces several enhancements to the DBRC API. This class provides an overview of these enhancements. You should refer to the IMS V10 System Programming API Reference manual for specific information on using this API.

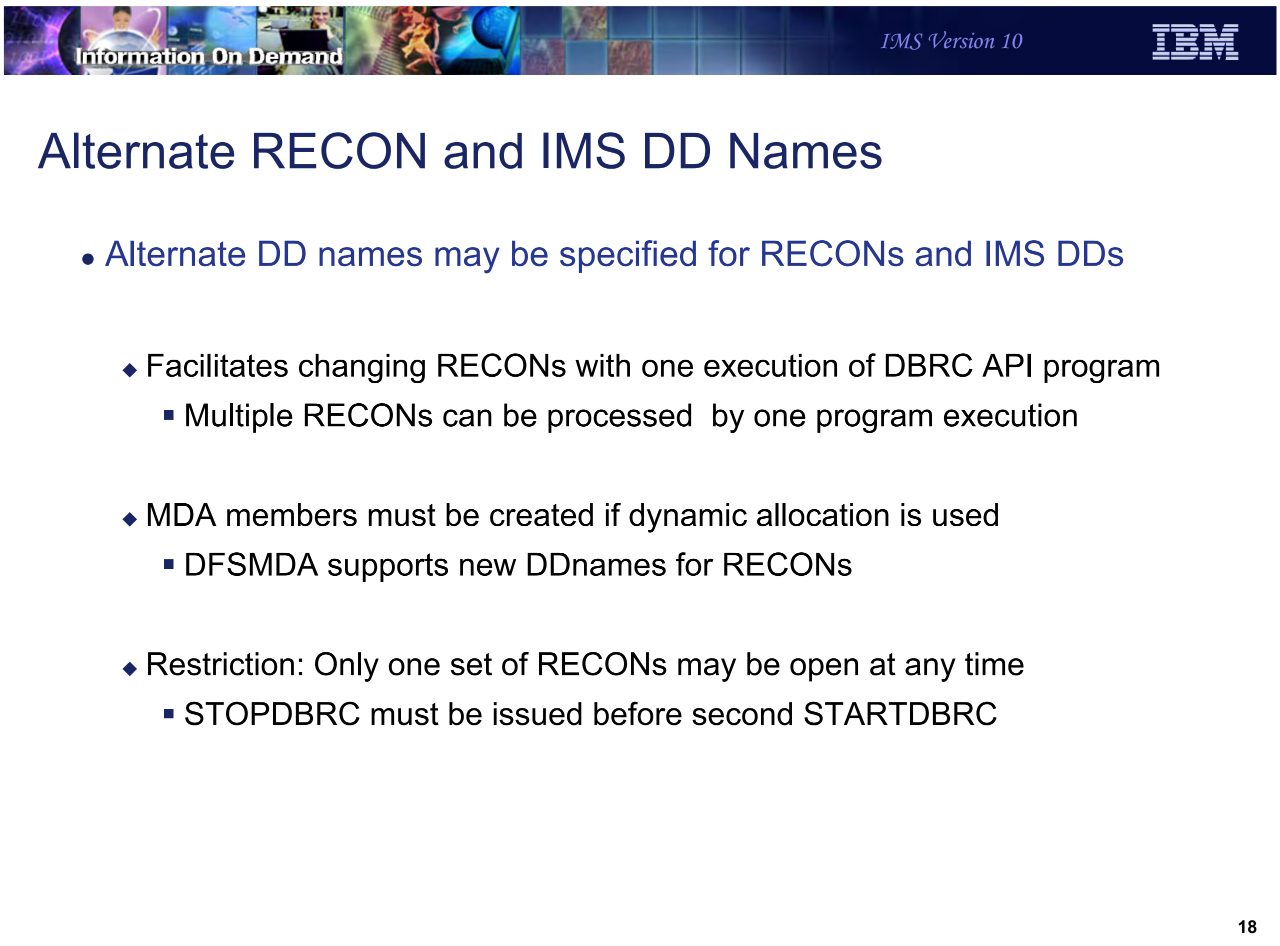
DBRC API Enhancements

- Alternate RECON and IMS DD names
- RECON update capability
  - Updates are made by using DBRC commands (INIT, CHANGE, NOTIFY)
- Register as subsystem
  - Allows application to authorize databases
- Database authorizations
  - Allows application to do utility functions
- QUERY enhancements
  - DBDS, Partition, Log, and wildcard support
- Security enhancements
  - Extension of DBRC command authorization
- Miscellaneous enhancements

Figure 14-16. DBRC API Enhancements CMA01.0

Notes:

The DBRC API is described in the IMS V10 *System Programming API Reference* manual. It contains details on the DSPAPI macro, its parameters and usage, and on the control blocks created by DBRC API requests.



Information On Demand
IMS Version 10
IBM

Alternate RECON and IMS DD Names

- Alternate DD names may be specified for RECONs and IMS DDs
  - Facilitates changing RECONs with one execution of DBRC API program
    - Multiple RECONs can be processed by one program execution
  - MDA members must be created if dynamic allocation is used
    - DFSMDA supports new DDnames for RECONs
  - Restriction: Only one set of RECONs may be open at any time
    - STOPDBRC must be issued before second STARTDBRC

Figure 14-17. Alternate RECON and IMS DD Names CMA01.0

Notes:

Programs using the DBRC API may use alternate names for the IMS (DBDLIB) and RECON DDNAMEs. This could make it easier to develop programs which access multiple sets of RECONs. In V9 these programs had to use dynamic allocation (SVC 99) to change the data sets for RECON1, RECON2, and RECON3 to access a different set of RECONs. In IMS V10 they may use JCL to allocate multiple sets or they may use DFSMDA to create dynamic allocation members for the different RECONs.

The DFSMDA macro has been enhanced to support the alternate DD names for RECONs. A new parameter, DDNAME=, has been added to the DFSMDA macro for TYPE=RECON.

Programs using the DBRC API are still restricted to accessing only one set of RECONs at a time. A FUNC=STOPDBRC request must be processed before a second set of RECONs may be opened with a second FUNC=STARTDBRC request.

As with previous releases, if you invoke the DBRC utility from your program you may use the DSPURXRT entry point. IMS V10 has added the capability to specify alternate DD names for the RECON data sets. This is done through the expansion of the list of DDNAMES passed to the entry point in the second word of the argument list.

DBRC API RECON Update Capability

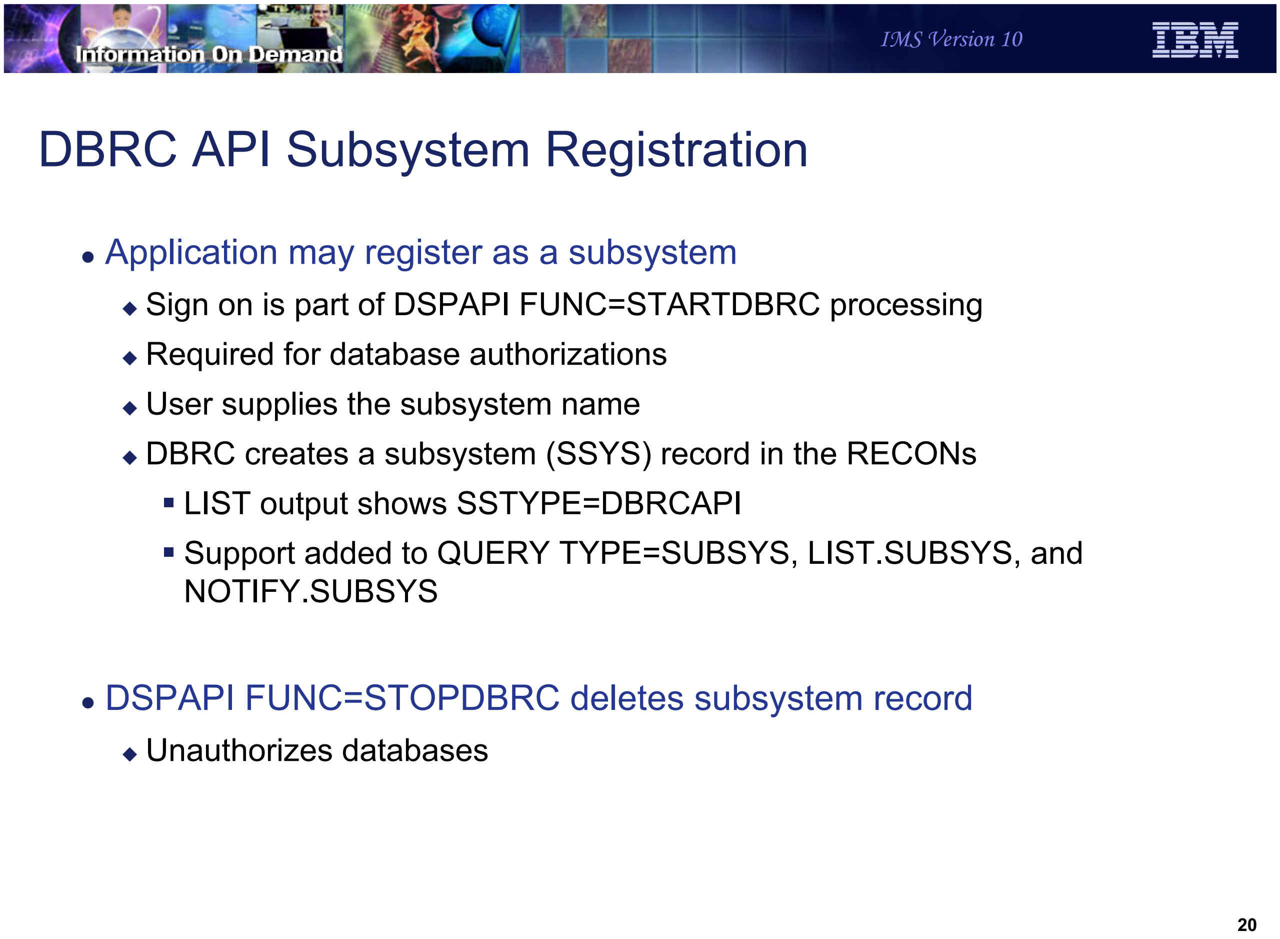
- Implemented with API command facility
  - Application program issues DBRC commands
    - INIT, CHANGE, DELETE, NOTIFY, GENJCL, BACKUP, and RESET
    - LIST commands are not valid
      - FUNC=QUERY should be used to list data from the RECONs
  - Uses FUNC=COMMAND on DSPAPI macro
  - JCL requirements are the same as DSPURX00
    - IMS DD statement may be required.
    - GENJCL commands require JCLPDS and JCLOUT DDs
      - Unless overridden by JCLPDS and JCLOUT command parameters
  - Output of command is returned within standard API output block

Figure 14-18. DBRC API RECON Update Capability CMA01.0

Notes:

The DBRC API has been extended. It was introduced in IMS V9. The original implementation of the DBRC API allowed you to query the data in the RECONs by issuing DSPAPI macros. The extensions to the API in V10 allow you to update the RECONs. Updates are done by allowing invoking DBRC commands. Commands such as INIT, CHANGE, DELETE, NOTIFY, and RESET may be used to update the RECONs. This interface may also be used to issue GENJCL and BACKUP commands which do not update the RECONs. LIST commands are not valid. The DSPAPI FUNC=QUERY function should be used to list data from the RECONs.

The output from commands is returned in an API output block. This block has the same format as the QUERY blocks that were introduced in V9. Details of the command blocks are shown later.


Information On Demand

IMS Version 10

DBRC API Subsystem Registration

- Application may register as a subsystem
  - Sign on is part of DSPAPI FUNC=STARTDBRC processing
  - Required for database authorizations
  - User supplies the subsystem name
  - DBRC creates a subsystem (SSYS) record in the RECONs
    - LIST output shows SSTYPE=DBRCAPI
    - Support added to QUERY TYPE=SUBSYS, LIST.SUBSYS, and NOTIFY.SUBSYS
- DSPAPI FUNC=STOPDBRC deletes subsystem record
  - Unauthorizes databases

20

Figure 14-19. DBRC API Subsystem Registration CMA01.0

Notes:

The STARTDBRC function is enhanced to support the creation of subsystem records. The user application may register with DBRC as a subsystem. This is required for authorizing databases. Database authorization support has also been added to the DBRC API. It will be explained later. Subsystem registration is done by adding an SSID= parameter to the DSPAPI FUNC=STARTDBRC macro. The meanings of the other parameters on the macro are unchanged.

A new type of subsystem has been added to the those that are stored in the SUBSYS record. This type is DBRCAPI. It is used when DSPAPI FUNC=STARTDBRC is used to create the SUBSYS record.

Support for this new type has been added to QUERY TYPE=SUBSYS, LIST.SUBSYS, and NOTIFY.SUBSYS.

QUERY TYPE=SUBSYS: This QUERY request has been enhanced to include SSTYPE=DBRCAPI to limit the returned information to only DBRC API subsystems. A new flag bit meaning has been added to the output block created by a QUERY

TYPE=SUBSYS. It indicates that the subsystem is a DBRC API subsystem. The bit is only set when VERSION=2.0 is specified on the QUERY.

LIST.SUBSYS and LIST.RECON: The output of the LIST.SUBSYS and LIST.RECON commands includes SSTYPE=DBRCAPI for DBRC API subsystems. You can list only DBRCAPI subsystems by using the DBRCAPI keyword on the LIST.SUBSYS command.

NOTIFY.SUBSYS: The NOTIFY.SUBSYS command has been enhanced. The new DBRCAPI keyword causes the command to create a DBRC API subsystem record.

The DBRCAPI subsystem records are deleted by DSPAPI FUNC=STOPDBRC macros. If any databases are authorized to the subsystem, they are unauthorized when the macro is processed.

Information On Demand
IMS Version 10
IBM

DBRC API Database Authorization

- DBRC API allows database authorizations and unauthorizations
  - Databases, partitions, and areas may be authorized
    - Authorizations control sharing of databases, partitions, and areas
  - Application must be registered as a subsystem
    - Cannot be READONLY
  - Access intents of RD, RO, and EX are allowed
    - UP is not allowed
  - Utility privileges may be granted
    - Allows authorization when some authorization flags are on
- Usage
  - May be used to provide DBRC interface for applications with utility functions

Figure 14-20. DBRC API Database Authorization CMA01.0

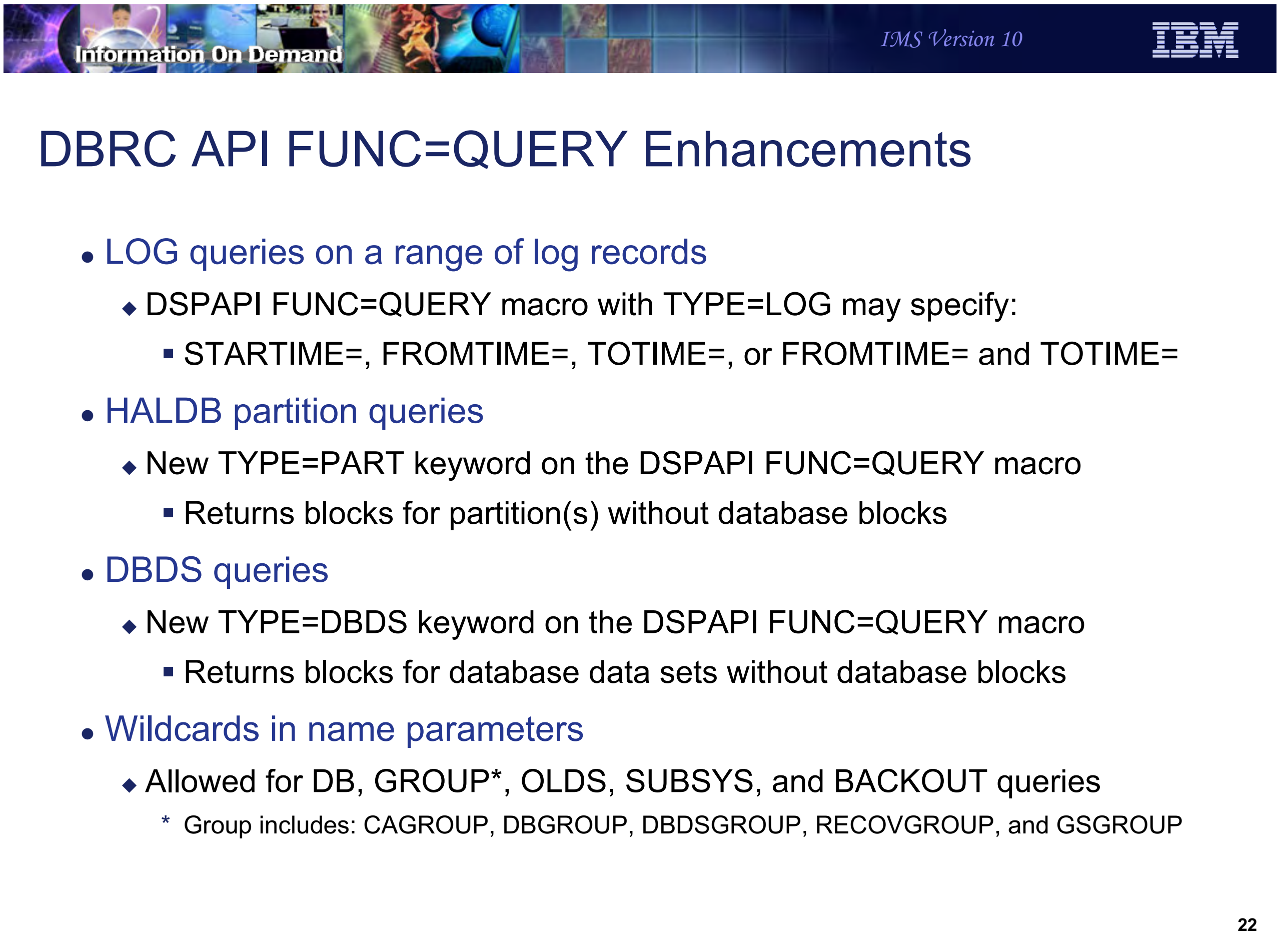
Notes:

Applications using the DBRC API may authorize and unauthorize databases, partitions, and areas. Databases are authorized to subsystems, so authorization requests come from applications that are registered as subsystems. Since authorizations update the RECONs, the application cannot be using READONLY.

When a database, partition, or area is authorized, its access intent may be specified. Block level data sharing is not supported for DBRC API users, therefore, UP is not allowed. If any changes are to be made to a database, partition, or area, EX access must be used.

Utility privileges may be granted with an authorization. This allows authorizations even though the 'Prohibit Further Authorization,' 'Image Copy Needed,' or 'Read Only' flags are on. This is explained further on the next page.

Authorizations may be needed when the DBRC API application is providing functions similar to IMS database utilities, such as back up, recovery, or reorganization.



Information On Demand
IMS Version 10
IBM

DBRC API FUNC=QUERY Enhancements

- LOG queries on a range of log records
  - DSPAPI FUNC=QUERY macro with TYPE=LOG may specify:
    - STARTIME=, FROMTIME=, TOTIME=, or FROMTIME= and TOTIME=
- HALDB partition queries
  - New TYPE=PART keyword on the DSPAPI FUNC=QUERY macro
    - Returns blocks for partition(s) without database blocks
- DBDS queries
  - New TYPE=DBDS keyword on the DSPAPI FUNC=QUERY macro
    - Returns blocks for database data sets without database blocks
- Wildcards in name parameters
  - Allowed for DB, GROUP*, OLDS, SUBSYS, and BACKOUT queries

* Group includes: CAGROUP, DBGROUP, DBDSGROUP, RECOVGROUP, and GSGROUP

Figure 14-21. DBRC API FUNC=QUERY Enhancements CMA01.0

Notes:

The DBRC API QUERY capability has been enhanced in several ways.

Queries for logs may request log records from a range of times. Queries for HALDB information may request data for partitions without requesting database data.

Database data set data may requested without requesting the data for the database in which the data set or data sets reside.

A wildcard capability has been added to several query types. This is the use of an asterisk (*) when specifying a name. The queries where this may be used are:

Query TYPE=Asterisk allowed with

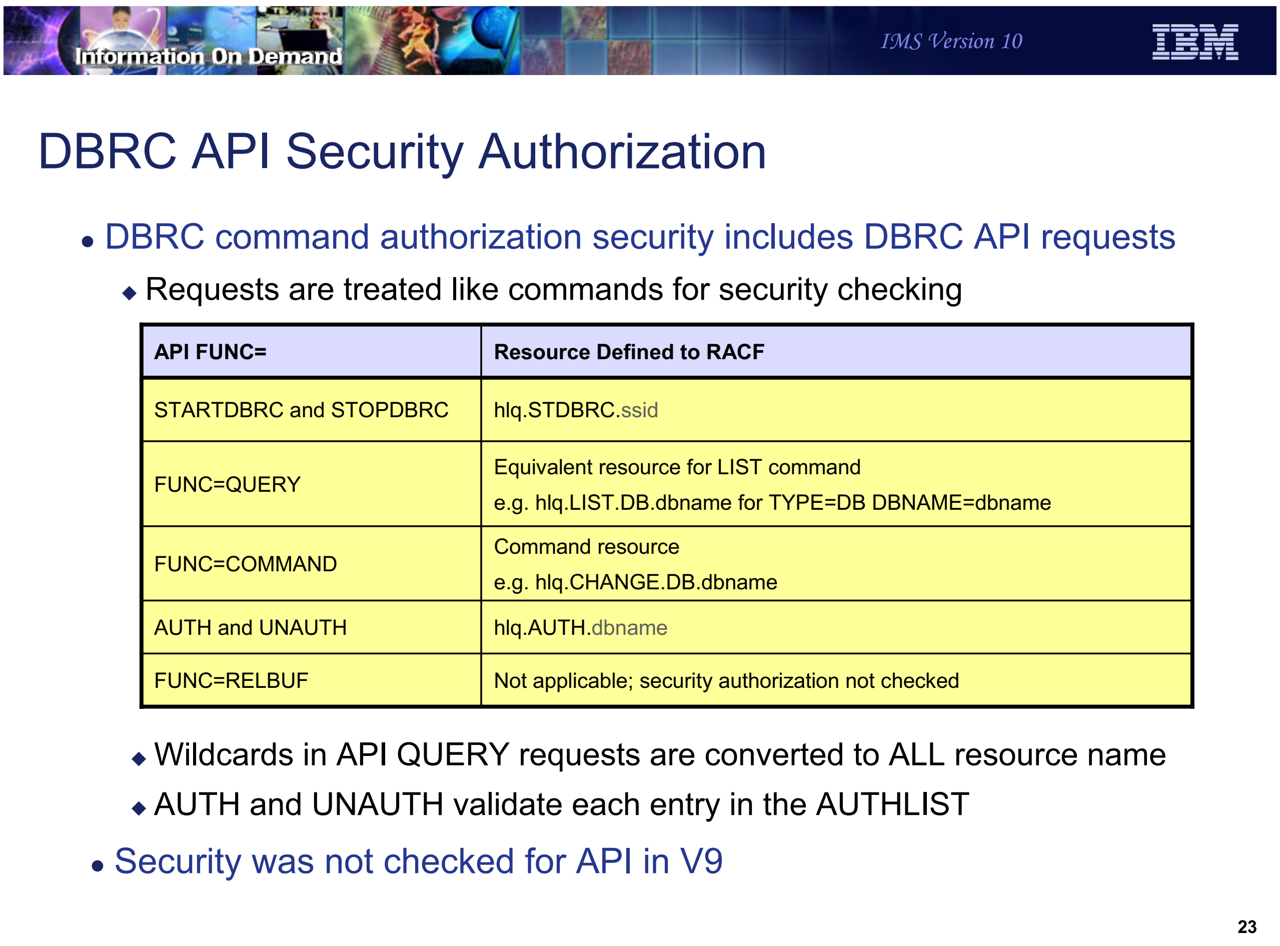
DBDBNAME=

*GROUPNAME=, GROUP=

OLDSSSID=

SUBSYSSSID=

BACKOUTSSID=


Information On Demand
IMS Version 10
IBM

DBRC API Security Authorization

- DBRC command authorization security includes DBRC API requests
  - Requests are treated like commands for security checking

API FUNC=	**Resource Defined to RACF**
STARTDBRC and STOPDBRC	hlq.STDBRC.ssid
FUNC=QUERY	Equivalent resource for LIST command e.g. hlq.LIST.DB.dbname for TYPE=DB DBNAME=dbname
FUNC=COMMAND	Command resource e.g. hlq.CHANGE.DB.dbname
AUTH and UNAUTH	hlq.AUTH.dbname
FUNC=RELBUF	Not applicable; security authorization not checked

  - Wildcards in API QUERY requests are converted to ALL resource name
  - AUTH and UNAUTH validate each entry in the AUTHLIST
- Security was not checked for API in V9

Figure 14-22. DBRC API Security Authorization CMA01.0

Notes:

DBRC command security was introduced in IMS V8. It may be used to invoke authorization checking for DBRC commands. RACF, or any SAF security product, may be used. Alternatively, an exit routine may be invoked or both the security product and the exit routine may be invoked. Commands are authorized by defining a resource representing the command. In RACF this is done with an RDEFINE statement.

This authorization is extended to the DBRC API in IMS V10. API requests invoke command authorization checking. Command authorization checking uses resources which are defined to secure specific commands or API requests. Resource names have the following form:

hlq.verb.resoucetype.resourcename

The high level qualifier (hlq) in the resource identifies a set of RECONs. “verb” identifies a DBRC command or API request type. “resourcetype” identifies a resource type on which the command or request operates. For example, a LIST.DB command operates on the database resource type. “resourcetype” is optional. “resourcename” identifies a specific resource instance. For example, a LIST.DB DBD(XYZ) command operates on the XYZ resource instance or name. “resourcename” is optional.

This scheme has been extended for API requests as shown in the table.

TYPE=STARTDBRC and TYPE=STOPDBRC requests are checked using resource of hlq.STDBRC.ssid. ssid specifies a subsystem and is optional. It restricts the use of the requests for a specific subsystem.

Security was not checked for any DBRC API requests in IMS V9. It is possible that a program which executed successfully in IMS V9 will fail security when executed in IMS V10.

Information On Demand
IMS Version 10
IBM

RECON Open

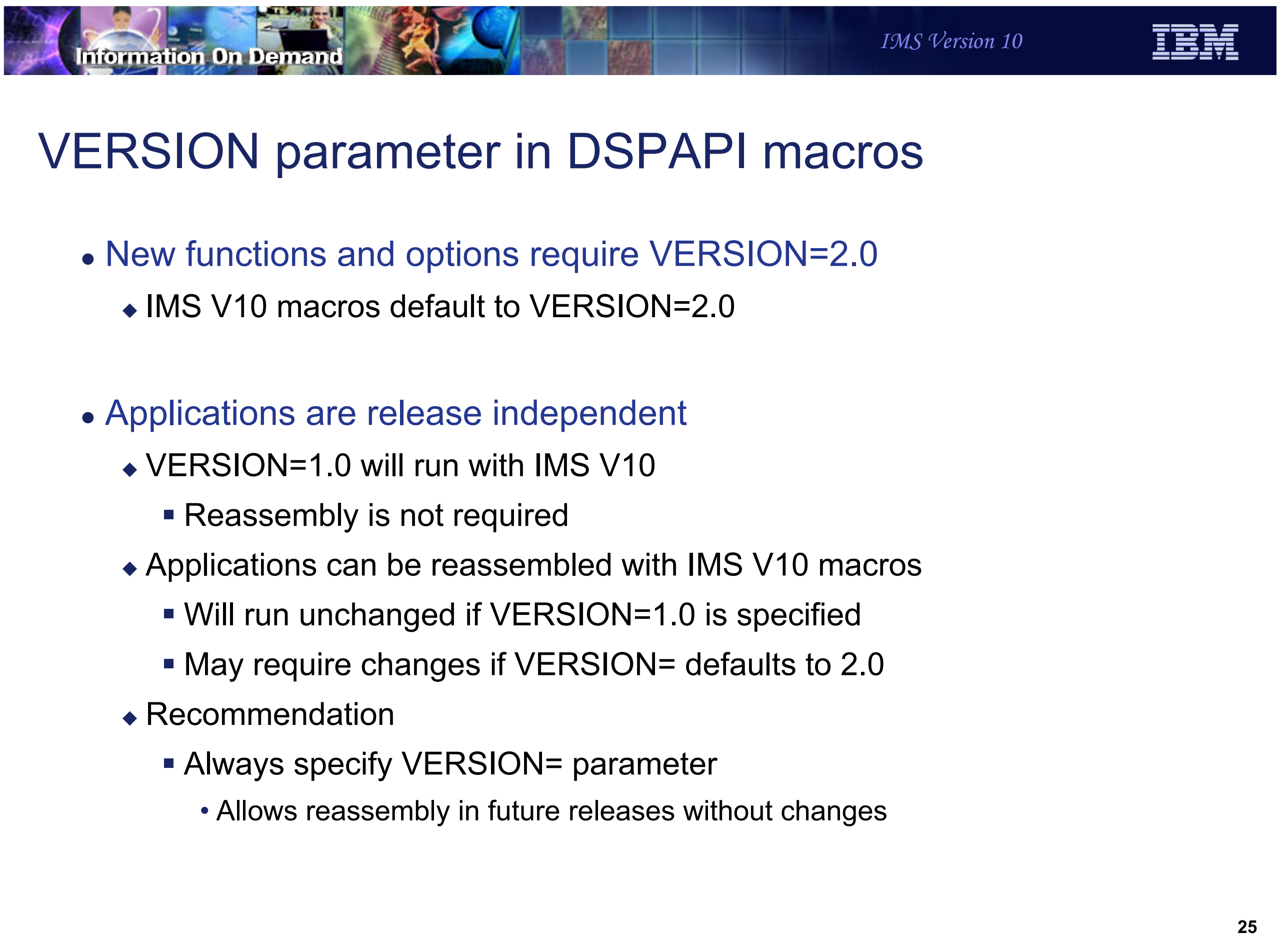
- RECONs may be initialized or upgraded using API
  - If VERSION=2.0 (V10) is specified on FUNC=STARTDBRC
    - STARTDBRC succeeds even with RECON not initialized or upgraded
      - RC=4 with reason code
      - Allows DBRC API program to INIT.RECON or UPGRADE
  - If VERSION=1.0 (V9) is specified on FUNC=STARTDBRC
    - STARTDBRC fails if RECON is not initialized or upgraded

Figure 14-23. RECON Open

CMA01.0

Notes:

The IMS V10 DBRC API has added the capability to open a set of RECONs which are not initialized or which are at a previous release level. This allows the application program to initialize the RECONs with an INIT.RECON command or to upgrade them with a CHANGE.RECON UPGRADE command. This could be useful for utility-like programs which create and upgrade RECONs.



Information On Demand
IMS Version 10
IBM

VERSION parameter in DSPAPI macros

- New functions and options require VERSION=2.0
  - IMS V10 macros default to VERSION=2.0
- Applications are release independent
  - VERSION=1.0 will run with IMS V10
    - Reassembly is not required
  - Applications can be reassembled with IMS V10 macros
    - Will run unchanged if VERSION=1.0 is specified
    - May require changes if VERSION= defaults to 2.0
  - Recommendation
    - Always specify VERSION= parameter
      - Allows reassembly in future releases without changes

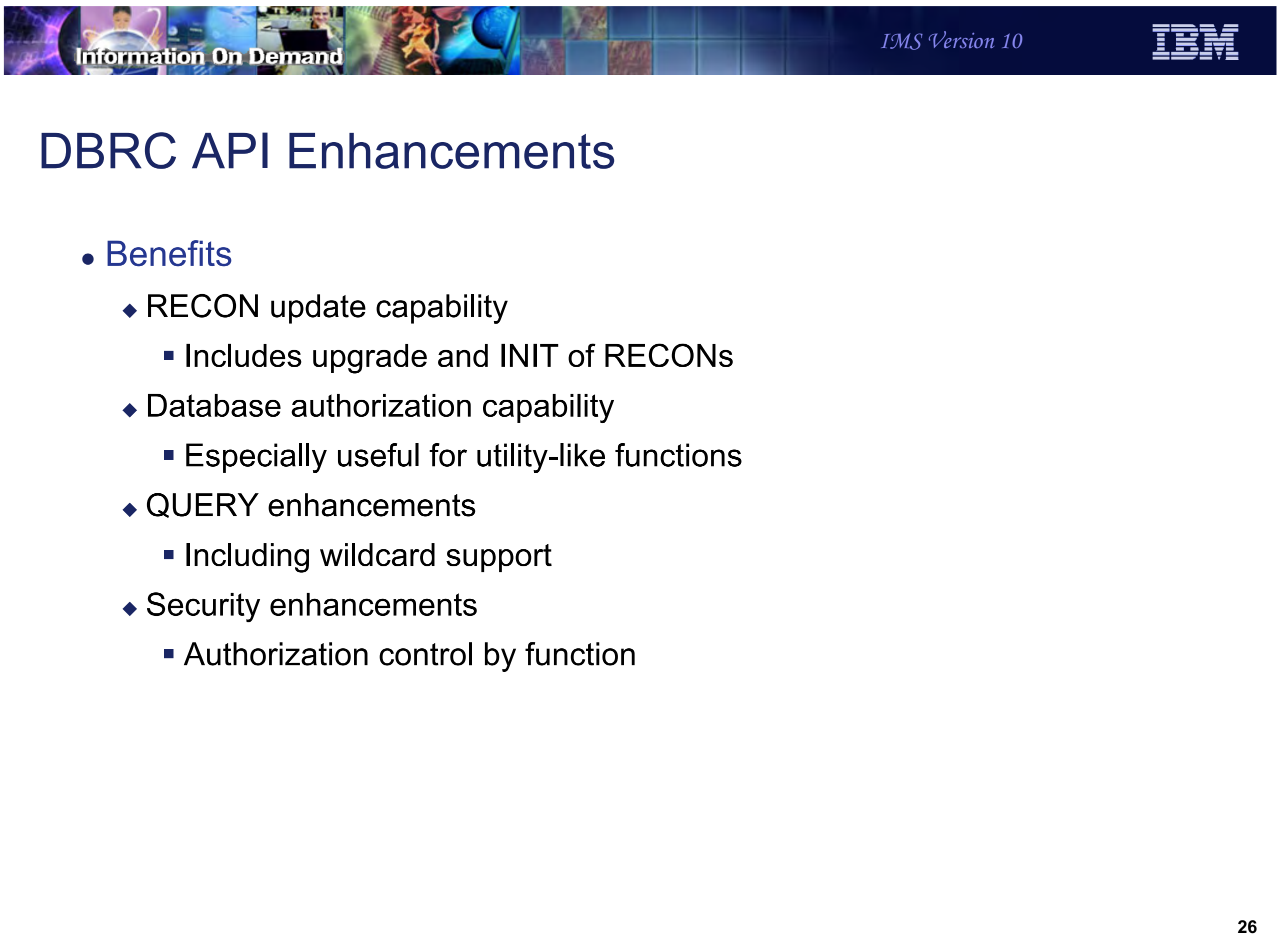
Figure 14-24. VERSION parameter in DSPAPI macros CMA01.0

Notes:

Each DBRC API macro includes the VERSION= parameter. New functions, such as AUTH, and new options, such as READONLY=YES, require VERSION=2.0. In IMS V9 the only valid value for VERSION was 1.0. It was also the default. In IMS V10 VERSION= defaults to 2.0.

Applications written in IMS V9 will continue to run in V10 without change. Reassembly is not required. In fact, reassembly could cause the program to change due to the change in the default for VERSION=. In some cases, the default of VERSION=2.0 may cause different results from the previous default of VERSION=1.0. This is not always the case. Some of the changes are only the use of previously reserved bytes in the control blocks that are produced. In any case, if you wrote a program for IMS V9 and reassemble it using an IMS V10 macro library, it is safest to specify VERSION=1.0 on the DFSAPI macros before the reassembly.

Since the VERSION= parameter defaults to the latest level of the macros and later levels may produce different results, it is safest to specify the VERSION= parameter value explicitly. This will ensure that future assemblies of DBRC API programs will produce the same results.



Information On Demand
IMS Version 10
IBM

DBRC API Enhancements

- Benefits
  - RECON update capability
    - Includes upgrade and INIT of RECONs
  - Database authorization capability
    - Especially useful for utility-like functions
  - QUERY enhancements
    - Including wildcard support
  - Security enhancements
    - Authorization control by function

Figure 14-25. DBRC API Enhancements CMA01.0

Notes:

The DBRC API is described in the IMS V10 *System Programming API Reference* manual. It contains details on the DSPAPI macro, its parameters and usage, and on the control blocks created by DBRC API requests.

Information On Demand

IMS Version 10

Parallel RECON Access

Figure 14-26. [No title] CMA01.0

Notes:

Information On Demand
IMS Version 10

Parallel RECON Access

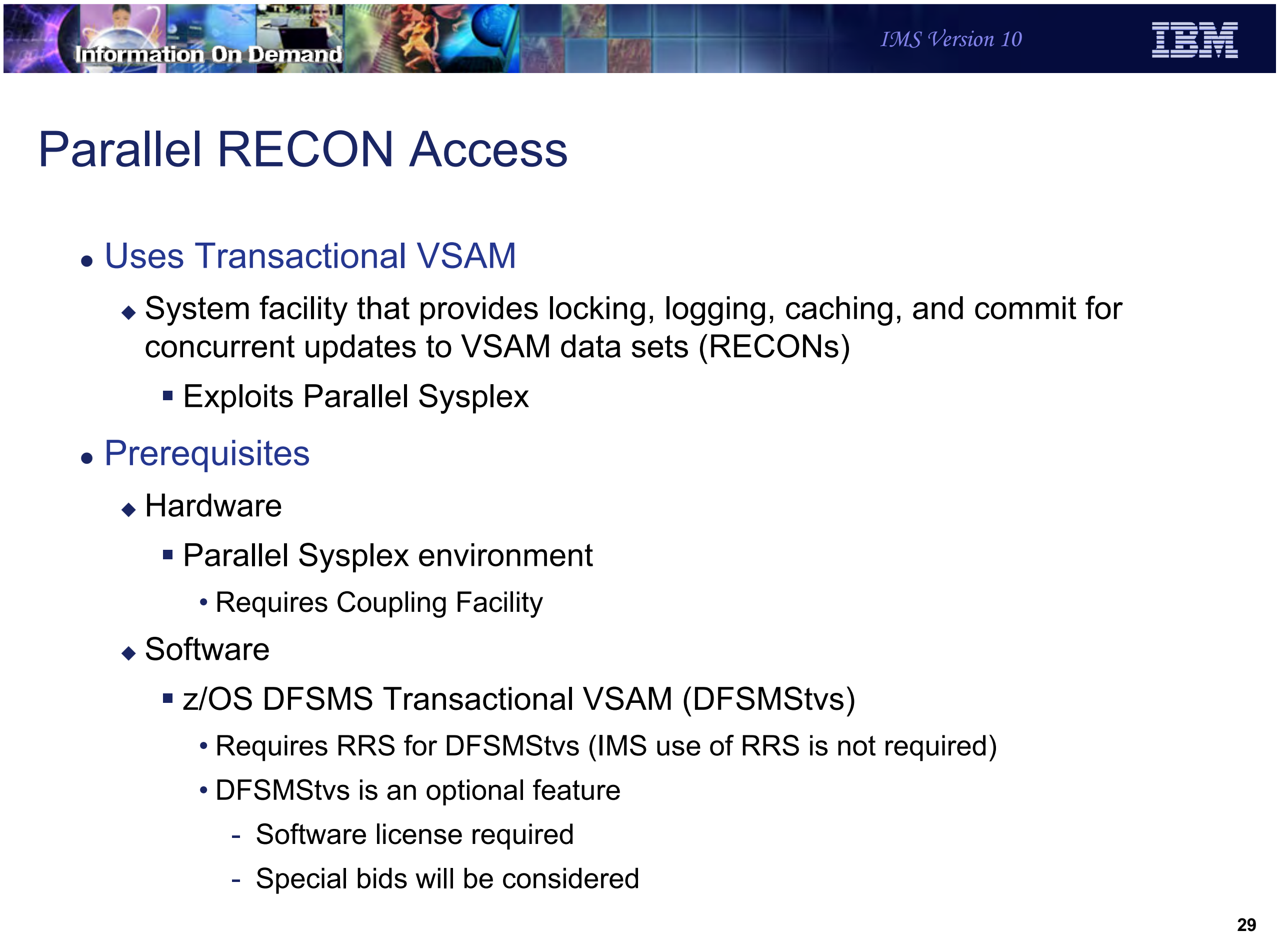
- Allows multiple DBRC instances to access the RECONs concurrently
  - DBRC instance: IMS Online subsystem, batch job, or utility
  - Parallel RECON Access is optional
- Eliminates serialization of accesses between DBRC instances
  - Data set RESERVE (or global enqueue) eliminated
- Reduces RECON contention
  - Could provide better responsiveness from IMS online and batch
  - Removes growth constraint

Figure 14-27. Parallel RECON Access CMA01.0

Notes:

Parallel RECON Access (PRA) is a new optional capability in IMS V10. It allows multiple DBRCs to access the RECONs at the same time.

Without PRA each DBRC instance must take its turn in accessing the RECONs. Multiple DBRCs may have the RECONs open at the same time, but only one may do I/O to the RECONs at any time. This restriction is eliminated with PRA. Serial access uses RESERVEs or global enqueues to serialize access to the RECONs. PRA eliminates this serialization. This reduces contention for the RECONs. It could provide better responsiveness, especially in situations where multiple online, batch, or utility executions of IMS are doing many I/Os to the RECONs.



Information On Demand
IMS Version 10
IBM

Parallel RECON Access

- Uses Transactional VSAM
  - System facility that provides locking, logging, caching, and commit for concurrent updates to VSAM data sets (RECONs)
    - Exploits Parallel Sysplex
- Prerequisites
  - Hardware
    - Parallel Sysplex environment
      - Requires Coupling Facility
  - Software
    - z/OS DFSMS Transactional VSAM (DFSMStvs)
      - Requires RRS for DFSMStvs (IMS use of RRS is not required)
      - DFSMStvs is an optional feature
        - Software license required
        - Special bids will be considered

Figure 14-28. Parallel RECON Access CMA01.0

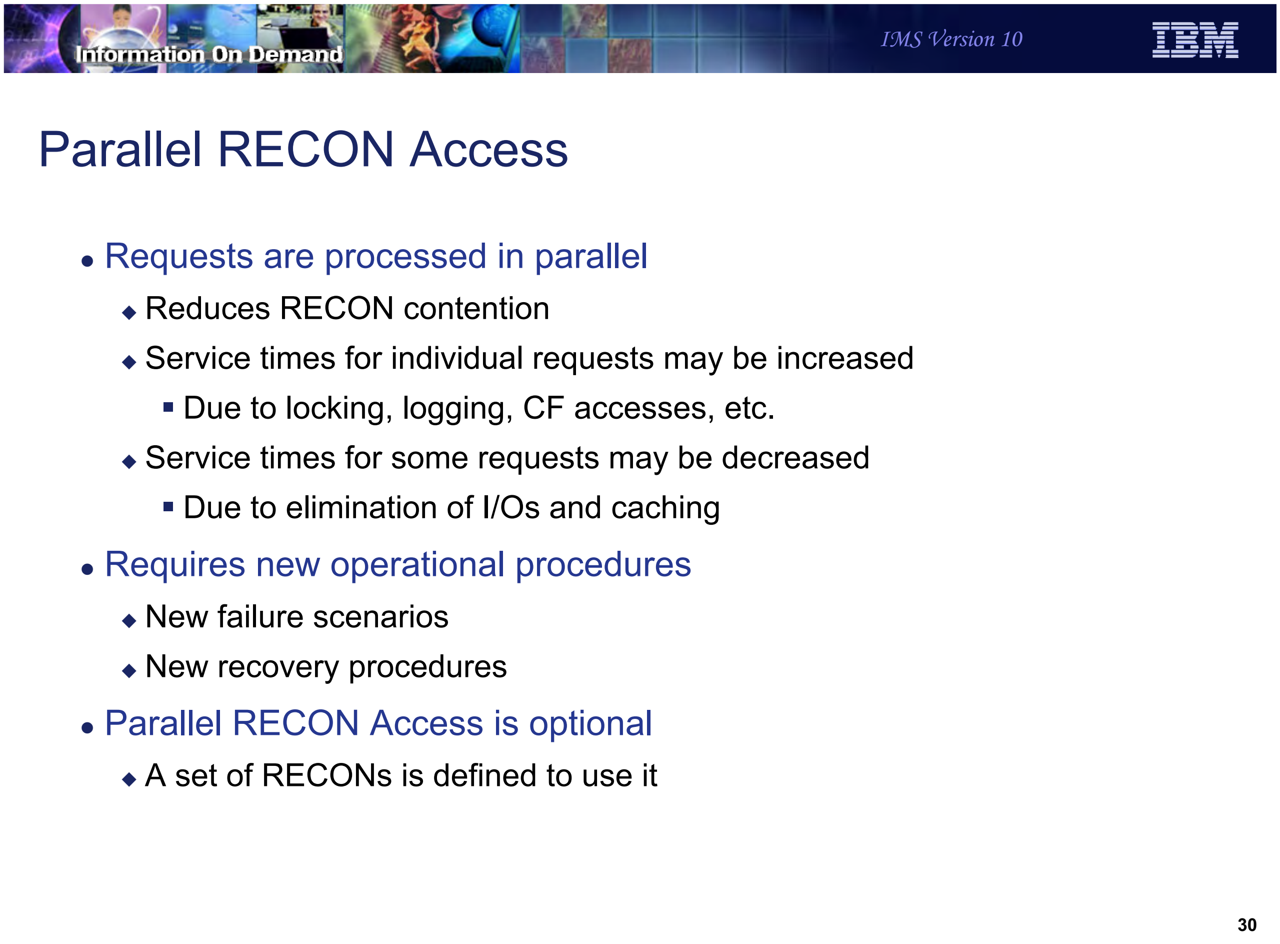
Notes:

PRA uses Transactional VSAM. This is a system facility that is provided by DFSMS. Transactional VSAM provides locking, logging, caching, and commit coordination for VSAM data sets such as the RECONs. It requires and exploits Parallel Sysplex.

PRA requires a Parallel Sysplex including a Coupling Facility. This is true even when all of the DBRC instances are running in only one z/OS system.

PRA requires DFSMStvs. This is an optional feature of DFSMS. There is a licensing fee associated with this feature. Special bids will be considered for IMS customers using the Parallel RECON Access function, who do not already have DFSMStvs, to acquire DFSMStvs for use restricted to IMS.

DFSMStvs uses Resource Recovery Services (RRS). RRS is used for commit coordination. IMS use of RRS is not required. That is, the RRS=Y IMS execution parameter is not required.



Information On Demand
IMS Version 10
IBM

Parallel RECON Access

- Requests are processed in parallel
  - Reduces RECON contention
  - Service times for individual requests may be increased
    - Due to locking, logging, CF accesses, etc.
  - Service times for some requests may be decreased
    - Due to elimination of I/Os and caching
- Requires new operational procedures
  - New failure scenarios
  - New recovery procedures
- Parallel RECON Access is optional
  - A set of RECONs is defined to use it

30

Figure 14-29. Parallel RECON Access CMA01.0

Notes:

Since PRA allows RECON requests to be processed in parallel, it potentially reduces RECON contention. On the other hand, service times for individual requests may increase due to the overhead of locking, logging, and Coupling Facility accesses that are required. Some services times may be decreased. This is due to the caching of RECON information in buffer pools and the CF.

PRA requires new operational procedures for recoveries. There are new possibilities for failures when the DFSMStvs environment is used.

PRA is optional in IMS V10. It is specified for a set of RECONs. There is a DBRC command for specifying PRA for a set or RECONs. Some RECON sets may use PRA while others continue to use serial access.

Information On Demand
IMS Version 10
IBM

Transactional VSAM (DFSMStvs) Overview

- DFSMStvs is an enhancement to VSAM RLS (record level sharing)
  - RLS uses locking and data caching in CFs
    - Typically used by CICS to allow concurrent updates to VSAM files by multiple CICSs
  - DFSMStvs adds logging, commit, and backout processing to RLS
    - Typically used to allow concurrent batch and CICS updates to VSAM files
    - Used by IMS V10 DBRC to provide Parallel RECON Access
- SMSVSAM address space
  - System address space which is started at z/OS IPL
  - Provides RLS and TVS services
  - One address space per LPAR

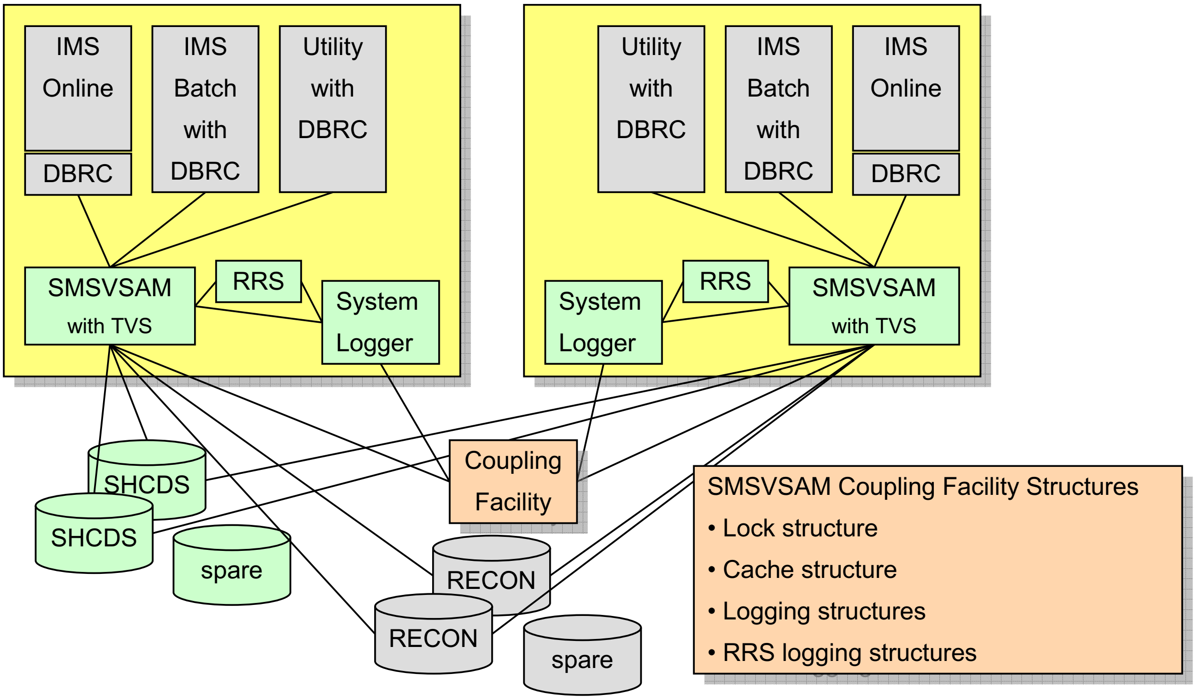
Figure 14-30. Transactional VSAM (DFSMStvs) Overview CMA01.0

Notes:

DFSMStvs (TVS) is based on VSAM Record Level Sharing (RLS). RLS uses locking and data caching to provide data sharing capabilities for VSAM files used by CICS systems. It allows multiple CICS systems to concurrently update VSAM files. RLS relies on the logging, commit coordination, and back out processing provided by CICS online systems. TVS adds its own logging, commit coordination, and back out support. This allows batch update jobs to share VSAM files between each other and CICS. PRA is using this same capability to allow multiple DBRC instances to do concurrent updates to the RECONs.

RLS and TVS execute in the SMSVSAM address space. This is a system address space that is typically started at z/OS initialization. There is only one SMSVSAM address space per LPAR. It provides RLS and TVS services to all users in the LPAR. This includes all instances of DBRC.

DBRC RECONs - DFSMStvs Environment


IMS Online
DBRC
IMS Batch with DBRC
Utility with DBRC
SMSVSAM with TVS
RRS
System Logger
Utility with DBRC
IMS Batch with DBRC
IMS Online
DBRC
RRS
SMSVSAM with TVS
System Logger
SHCDS
SHCDS
spare
Coupling Facility
RECON
RECON
spare

SMSVSAM Coupling Facility Structures

- Lock structure
- Cache structure
- Logging structures
- RRS logging structures

Figure 14-31. DBRC RECONs - DFSMStvs Environment CMA01.0

Notes:

This is an illustration of a DFSMStvs environment with multiple instances of DBRC using PRA. There is an SMSVSAM address space in both z/OS systems. This address space provides TVS services to the DBRCs in the LPAR. The illustration shows that there is an IMS online system, an IMS batch job (DLI or DBB), and a IMS utility in each system. TVS uses the system logger and RRS. RRS also uses the system logger. The system logger has structures in the coupling facility. The SMSVSAM address space also connects to the coupling facility for its own structures. These are cache structures and a lock structure. There are a pair of share control data sets (SHCDS) for the VSAM data sets. These data sets contain information about the use of VSAM data sets that are being shared. There is also a spare SHCDS. Of course, there are the RECON pair and spare.

Information On Demand
IMS Version 10

Transactional VSAM (DFSMStvs) Overview

- Recovery of failed users
  - Each DFSMStvs instance has an undo log
  - Used for backout after failures
- Recovery for failed SMSVSAM address space
  - Restarted automatically if it fails
    - Backs out in-flight work and releases retained locks
- Recovery for failed z/OS system
  - Peer recovery
    - Back outs done by another SMSVSAM address space
    - Locks released

Figure 14-32. Transactional VSAM (DFSMStvs) Overview CMA01.0

Notes:

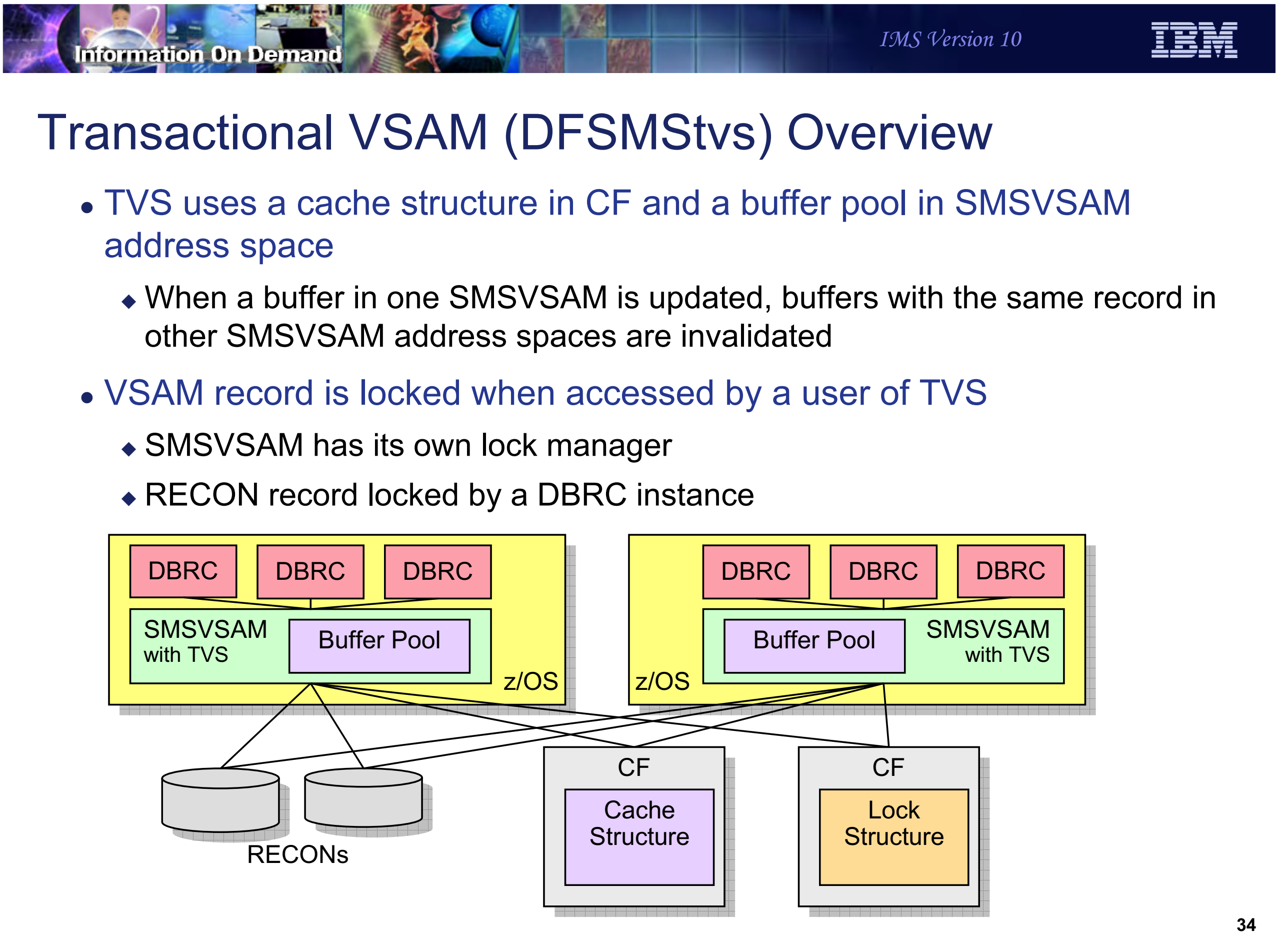
DFSMStvs has recovery capabilities to handle various kinds of failures.

Each DFSMStvs instance has its own undo log. These are log records which have the "before images" of records. If a user of DFSMStvs, such as DBRC fails, its uncommitted updates are backed out by DFSMStvs. The undo log is used for this purpose.

If DFSMStvs fails or the SMSVSAM address space fails, it is automatically restarted. When it restarts is backs out any in-flight work and releases any locks held by the in-flight transactions.

If a z/OS system fails, peer recovery is invoked. Back outs are done for the failed DFSMStvs by using another SMSVSAM address space in the Parallel Sysplex. The locks held for the failed work are released.

More information on these processes is provided later.


Information On Demand
IMS Version 10
IBM

Transactional VSAM (DFSMStvs) Overview

- TVS uses a cache structure in CF and a buffer pool in SMSVSAM address space
  - When a buffer in one SMSVSAM is updated, buffers with the same record in other SMSVSAM address spaces are invalidated
- VSAM record is locked when accessed by a user of TVS
  - SMSVSAM has its own lock manager
  - RECON record locked by a DBRC instance

DBRC
DBRC
DBRC
SMSVSAM with TVS
Buffer Pool
z/OS
DBRC
DBRC
DBRC
Buffer Pool
SMSVSAM with TVS
z/OS
RECONs
CF
Cache Structure
CF
Lock Structure

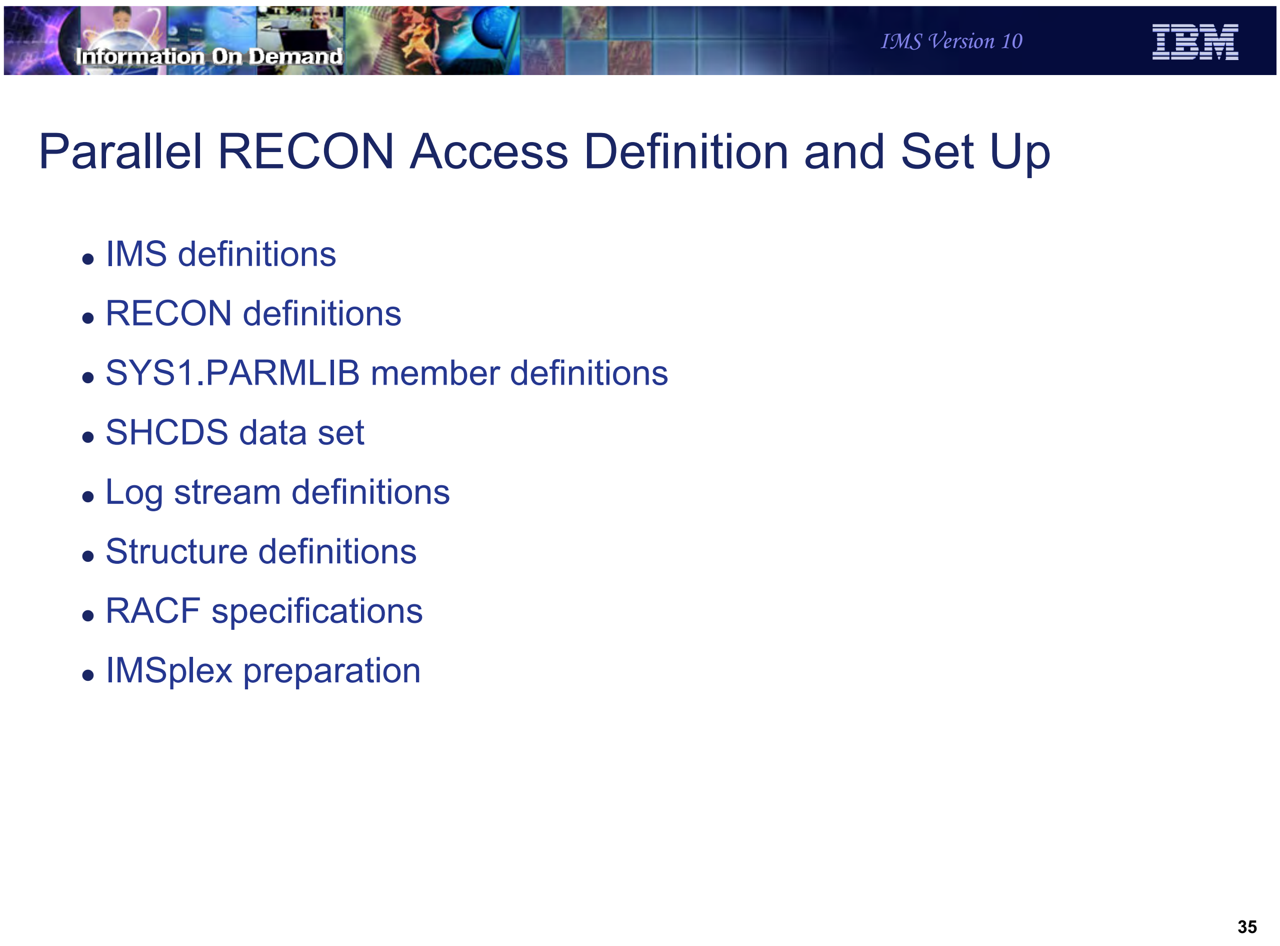
Figure 14-33. Transactional VSAM (DFSMStvs) Overview CMA01.0

Notes:

This illustrates a DFSMStvs environment with DBRC users. When a DBRC instance accesses a VSAM record in the RECONs, the request is processed by TVS in the SMSVSAM address space. The SMSVSAM address space has its own buffers. The buffer pools are maintained in data spaces owned by the SMSVSAM address space. When using TVS, DBRC does not have its own buffers for RECON processing. Instead, it uses the SMSVSAM buffers. These buffers are shared with all users of the SMSVSAM address space. These are all of the VSAM data sets in the z/OS system which are using RLS or TVS. SMSVSAM also uses cache structures in the Coupling Facilities. Each shared data set is assigned to a structure. When a VSAM record in an SMSVSAM buffer pool is updated, if it resides in buffers in other systems those buffers are invalidated. This is done using the cross invalidation capability associated with cache structures and Parallel Sysplex.

The processing of VSAM requests includes lock requests. SMSVSAM has its own lock manager. It does not use the IRLM, however, its lock manager provides functions similar to those provided by IRLM for IMS and DB2 databases. The owner of a lock request is the

DBRC instance. This is an IMS online system, an IMS batch job, or an IMS utility. Lock information is held in a lock structure in the CF.



Information On Demand
IMS Version 10
IBM

Parallel RECON Access Definition and Set Up

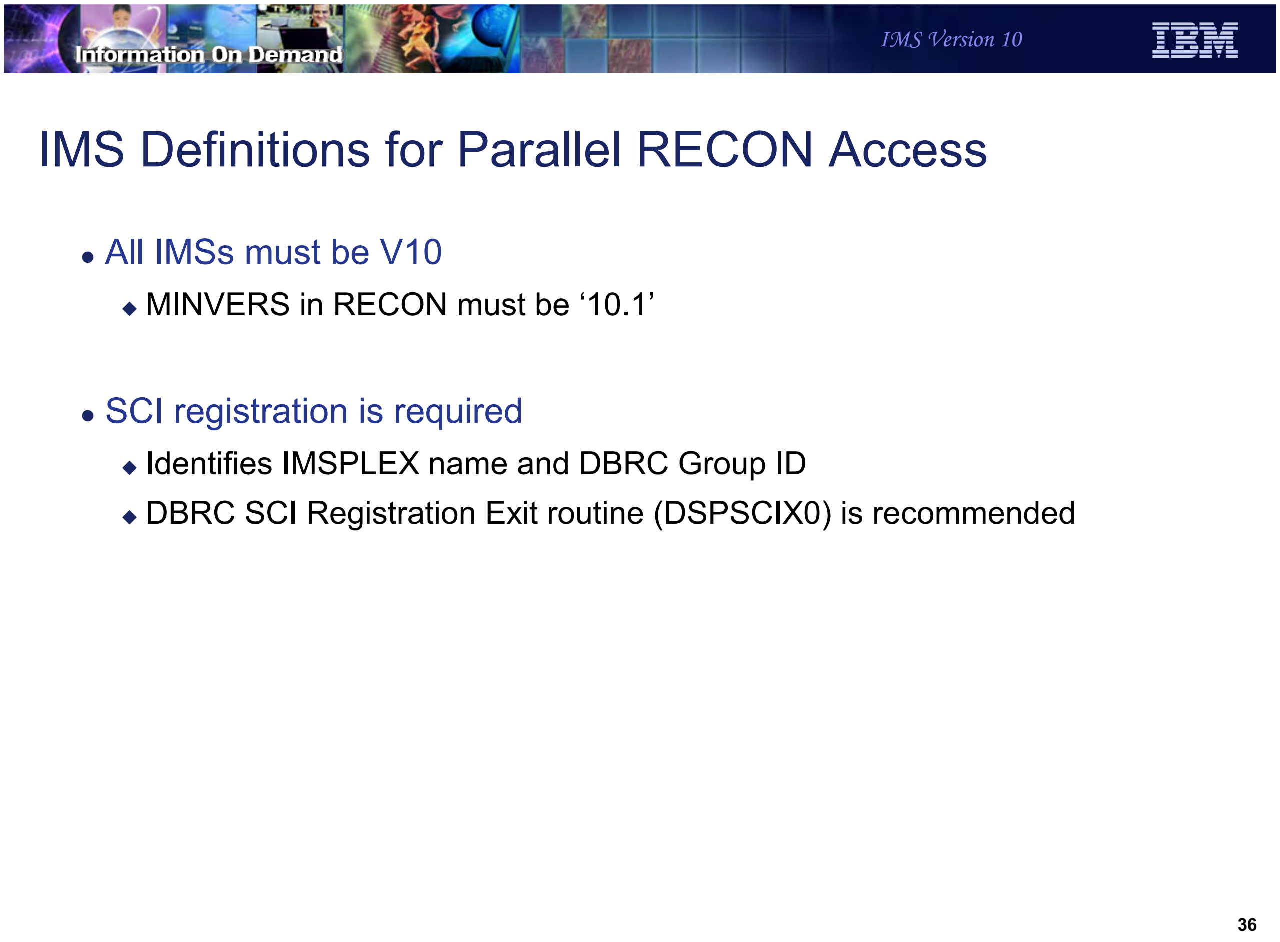
- IMS definitions
- RECON definitions
- SYS1.PARMLIB member definitions
- SHCDS data set
- Log stream definitions
- Structure definitions
- RACF specifications
- IMSplex preparation

35

Figure 14-34. Parallel RECON Access Definition and Set Up CMA01.0

Notes:

Defining Parallel RECON Access includes several specifications. These will be described on the following pages.


Information On Demand
IMS Version 10
IBM

IMS Definitions for Parallel RECON Access

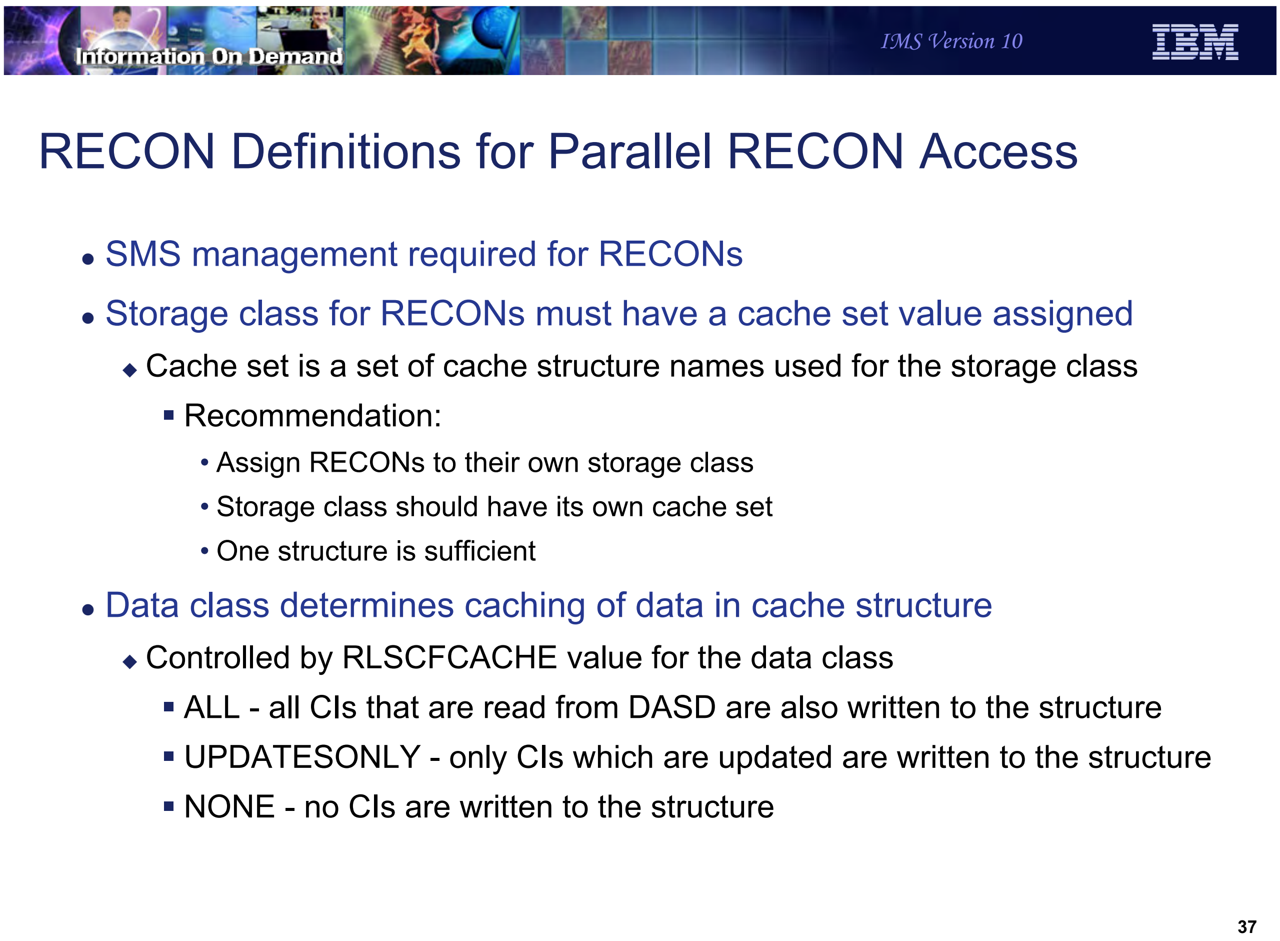
- All IMSs must be V10
  - MINVERS in RECON must be ‘10.1’
- SCI registration is required
  - Identifies IMSPLEX name and DBRC Group ID
  - DBRC SCI Registration Exit routine (DSPSCIX0) is recommended

Figure 14-35. IMS Definitions for Parallel RECON Access CMA01.0

Notes:

PRA requires IMS V10. To ensure that all users of a set of RECONs are using V10, a MINVERS value of '10.1' must be specified for the RECONs. This prevents any lower release level of IMS from using the RECONs.

SCI registration is required for PRA. This means that a CSL environment is required. Each system where an IMS using PRA will be run must have an SCI address space. Since SCI registration is required, the use of a DBRC SCI Registration Exit routine (DSPSCIX0) is recommended. This ensures that consistent registration is done by all DBRC users and it eliminates the need to add the IMSPLEX= parameter to the JCL for online systems, batch jobs, and utilities. In addition to the IMSPLEX value which is assigned by the IMSPLEX= parameter of the exit routine, IMS V10 has added the DBRC group ID. This ID may also be assigned by the exit routine.



Information On Demand

IMS Version 10

RECON Definitions for Parallel RECON Access

- SMS management required for RECONs
- Storage class for RECONs must have a cache set value assigned
  - Cache set is a set of cache structure names used for the storage class
    - Recommendation:
      - Assign RECONs to their own storage class
      - Storage class should have its own cache set
      - One structure is sufficient
- Data class determines caching of data in cache structure
  - Controlled by RLSCFCACHE value for the data class
    - ALL - all CIs that are read from DASD are also written to the structure
    - UPDATESONLY - only CIs which are updated are written to the structure
    - NONE - no CIs are written to the structure

Figure 14-36. RECON Definitions for Parallel RECON Access CMA01.0

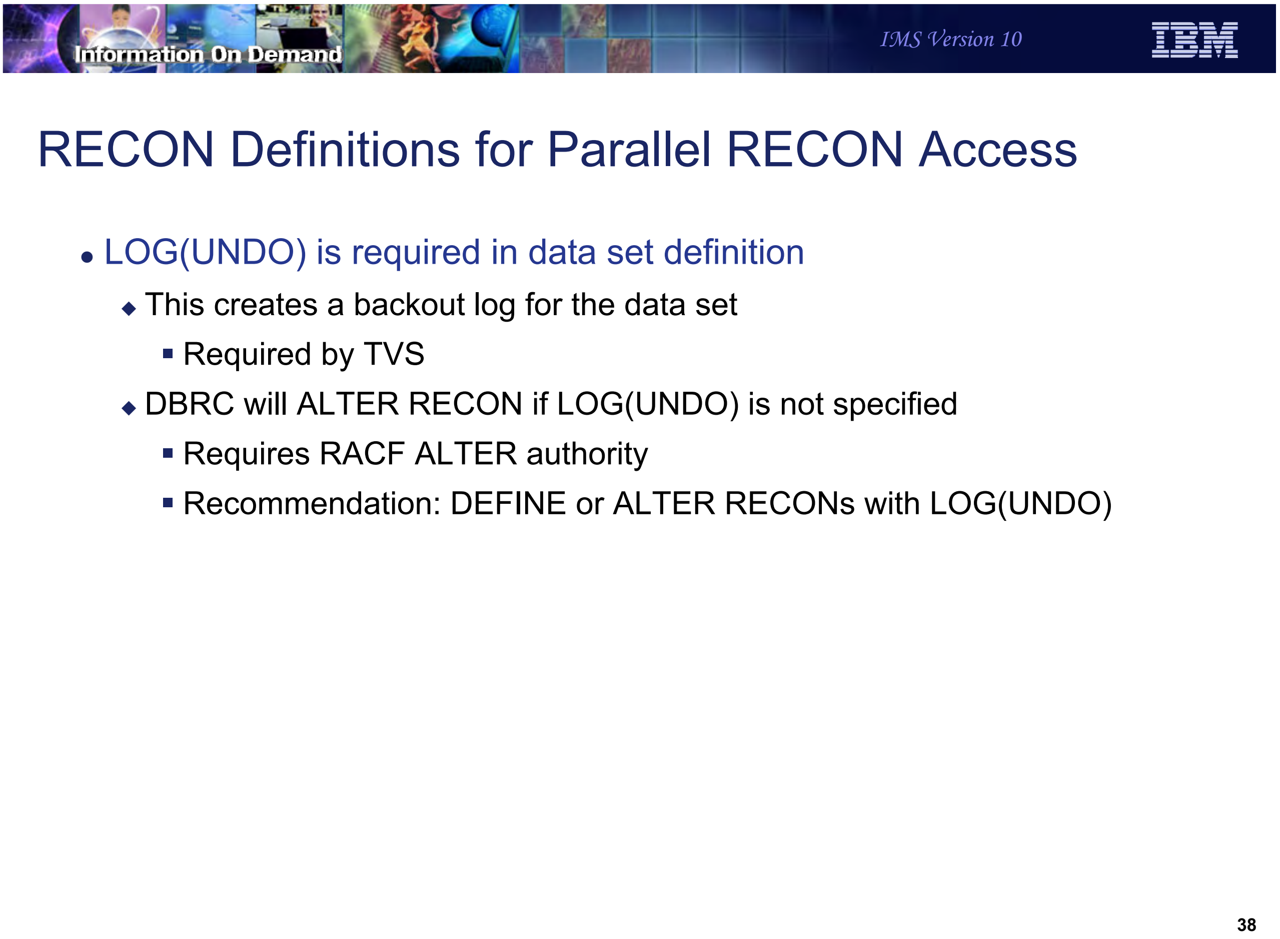
Notes:

When using PRA the RECONs must be under SMS management.

The RECONs must be defined with a storage class which has a cache set value. A cache set value assigns CF cache structures to the storage class. Many data sets (spheres) may have the same storage class. Multiple storage classes may use the same cache set. You can simplify the management and understanding of caching for the RECONs by assigning the RECON data sets to a storage class with it own cache set. The cache set needs only one structure defined to it. If the RECONs are the only RLS or TVS data sets in the storage class, they will have their own cache structure.

The data class for the RECONs determines the caching option. This is controlled by the RLSCFCACHE parameter for the data class. ALL specifies that all CIs read from DASD are also stored in the structure. UPDATESONLY specifies that CIs which are updated are written to the structure when they are written to DASD. NONE specifies that no CIs are stored in the structure. When NONE is specified, the structure is used only for buffer invalidation processing. There is another parameter which affects caching. This is the

RLS_MAXCFFEATURELEVEL parameter on the IGDSMSxx member of SYS1.PARMLIB. This will be explained when IGDSMSxx is covered later in this section of the class.



Information On Demand
IMS Version 10
IBM

RECON Definitions for Parallel RECON Access

- LOG(UNDO) is required in data set definition
  - This creates a backout log for the data set
    - Required by TVS
  - DBRC will ALTER RECON if LOG(UNDO) is not specified
    - Requires RACF ALTER authority
    - Recommendation: DEFINE or ALTER RECONs with LOG(UNDO)

38

Figure 14-37. RECON Definitions for Parallel RECON Access CMA01.0

Notes:

LOG(UNDO) is required in the definition of the RECONs for the use of Transactional VSAM. This creates a backout log for the data set when TVS is used. DBRC automatically sets this definition. If Parallel RECON Access is being used and LOG(UNDO) is not specified for the RECONs, DBRC will issue an ALTER to set the parameter to LOG(UNDO). If Parallel RECON Access is not being used, DBRC will ALTER RECON to LOG(NONE) if another value is defined. Of course, you can set LOG(UNDO) by issuing the DEFINE or ALTER with IDCAMS. In either case, the ALTER requires ALTER security from the security system, such as RACF.

Do not specify LOG(ALL) or a LOGSTREAMID parameter for the RECONs. LOG(ALL) creates a forward recovery log which is identified by the LOGSTREAMID. DBRC does not support forward recovery of the RECONs.

IFAPRDxx Member of SYS1.PARMLIB

- Product Enablement Policy
  - Used to enable separately priced features, such as DFSMStvs.
  - Example:

```
PRODUCT OWNER('IBM CORP')
   NAME('Z/OS')
   ID(5694-A01)
   VERSION(*) RELEASE(*) MOD(*)
   FEATURENAME(DFSMSTVS)
   STATE(ENABLED)
```

Figure 14-38. IFAPRDxx Member of SYS1.PARMLIB

Notes:

DFSMStvs is a separately priced feature of z/OS. It must be enabled with a product enablement policy which is defined in the IFAPRDxx member of SYS1.PARMLIB.

SHCDS Data Sets

- RLS and TVS require VSAM data sharing control data sets (SHCDS)
  - Contain name of lock structure, list of subsystems, status of subsystems, list of open spheres (data sets), etc.
  - Multiple actives and spare(s)
  - Name: SYS1.DFPSHCDS.qualifier.Vvolser
  - See *DFSMSdfp Storage Administration Reference* for definition information
  - Activated with commands:

```
V SMS,SHCDS(qualifier.Vvolser),NEW
V SMS,SHCDS(qualifier.Vvolser),NEWSPARE
```

Figure 14-39. SHCDS Data Sets CMA01.0

Notes:

If you have not implemented RLS, you must defined SHCDS data sets. If you already have implemented RLS, you already have these data sets defined. These data sets are used to hold the name of the lock structure, the list of subsystems, and a list of open VSAM spheres (data sets). It also contains information about the spheres such as which subsystems have them open.

There can be multiple SHCDS data sets. Two is typical. Spares may also be defined. Having a pair and a spare provides recovery capabilities similar to those for a pair and spare for the RECONs.

The data set name for an SHCDS is always of the form SYS1.DFPSHCDS.qualifier.Vvolser where qualifier is whatever you want it to be and volser is the volume serial for its volume. See the *DFSMSdfp Storage Administration Reference* manual for complete information on defining SHCDS data sets.

VARY commands are used to make a data set an active SHCDS or a spare.

SYS1.PARMLIB Members

- IEFSSNxx
  - Must include a statement for SMS
- IGDSMSxx parameters include:
  - `SYSNAME(name1,name2, …)`
    - Names associated with each TVS instance
  - `TVSNAME(n1,n2, …)`
    - Unique numbers associated with each TVS instance
  - `AKP(n1,n2,…)    AKP=(1000,1000,…) recommended`
    - Number of logging operations between activity key points. Used to delete log records no longer needed for backouts and shunt some records.
    - `AKP=(1000,1000,…)` recommended
  - `RLSINIT(YES|NO)`
    - Determines if SMSVSAM addr. space is started at system initialization

continued on next page

Figure 14-40. SYS1.PARMLIB Members CMA01.0

Notes:

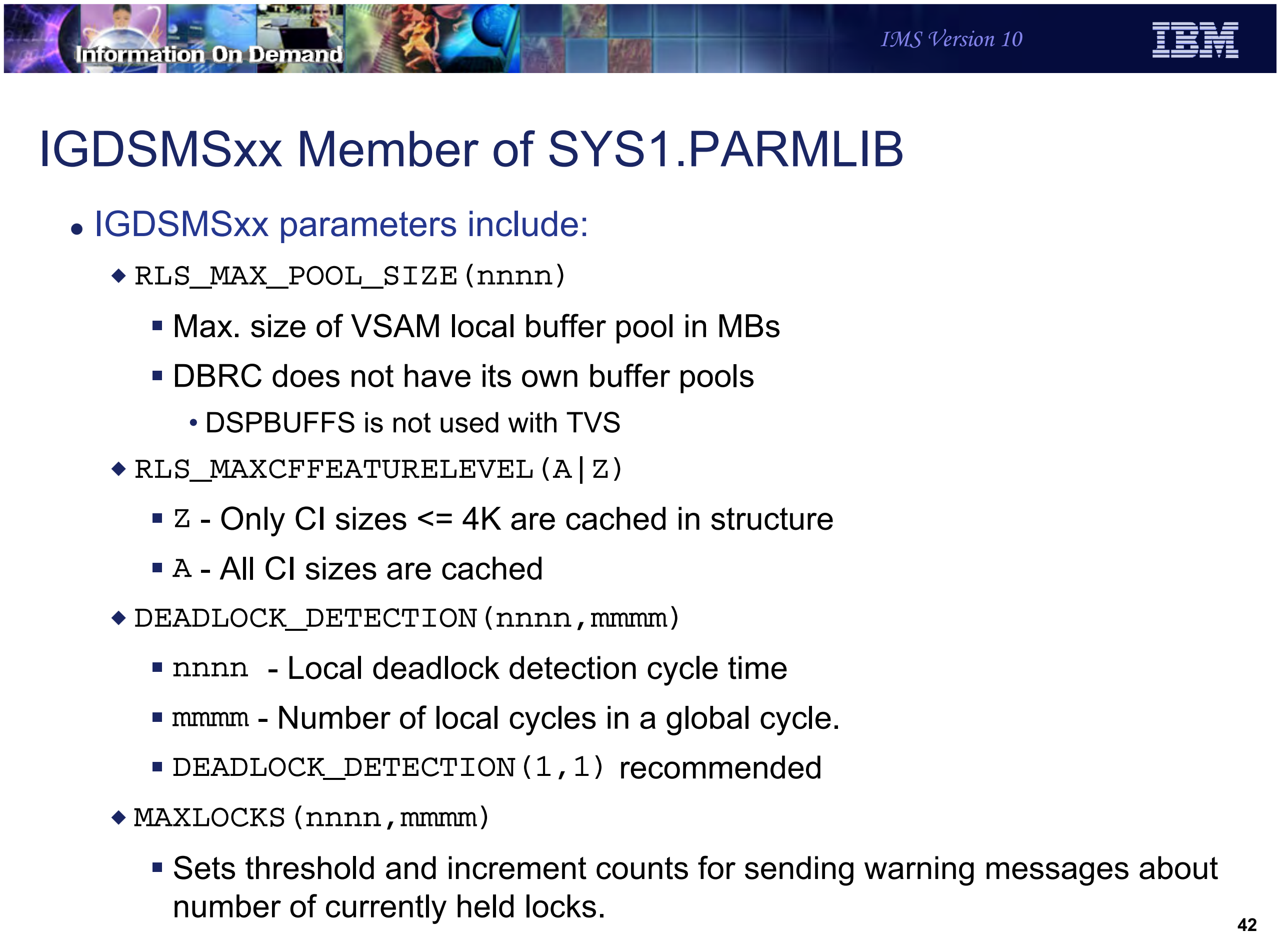
SMS must be defined to MVS as a subsystem. This is done by including a statement for SMS in the IEFSSNxx SYS1.PARMLIB member.

The names associated with each TVS instance are defined in the SYSNAME parameter. Similarly, each TVS instance is assigned a unique number with the TVSNAME parameter.

AKP controls the occurrence of activity key points. Separate numbers may be specified for the different systems. At these times log records which are no longer needed for backouts are deleted. Log records for units of recovery that have not logged in two AKPs are moved to the shunt log.

The IGDSMSxx member of SYS1.PARMLIB is described in the *z/OS MVS Initialization and Tuning Guide*. Some of the most important parameters for TVS users are shown here.

If RLSINIT(YES) is not specified, the SMSVSAM address space must be started with a V SMSVSAM,ACTIVE command.


Information On Demand
IMS Version 10
IBM

IGDSMSxx Member of SYS1.PARMLIB

- IGDSMSxx parameters include:
  - `RLS_MAX_POOL_SIZE(nnnn)`
    - Max. size of VSAM local buffer pool in MBs
    - DBRC does not have its own buffer pools
      - DSPBUFFS is not used with TVS
  - `RLS_MAXCFFEATURELEVEL(A|Z)`
    - `Z` - Only CI sizes <= 4K are cached in structure
    - `A` - All CI sizes are cached
  - `DEADLOCK_DETECTION(nnnn,mmmm)`
    - `nnnn` - Local deadlock detection cycle time
    - `mmmm` - Number of local cycles in a global cycle.
    - `DEADLOCK_DETECTION(1,1)` recommended
  - `MAXLOCKS(nnnn,mmmm)`
    - Sets threshold and increment counts for sending warning messages about number of currently held locks.

42

Figure 14-41. IGDSMSxx Member of SYS1.PARMLIB CMA01.0

Notes:

RLS_MAX_POOL_SIZE specifies in megabytes the maximum size of the VSAM local buffer pool in SMSVSAM. This pool is created in a data space. For serial access (not Parallel RECON Access) to the RECONs IMS uses the DSPBUFFS specification to determine the number of buffers used by a DBRC instance. This defaults to 60 index buffers and 120 data buffers. You may have specified a different value. As a starting point, you may make the SMSVSAM local buffer pool the size of the buffer pools for all of the concurrent DBRC instances that will be run. For example, if you took the default DSPBUFFS, have a RECON CI size of 16K, and have 10 concurrent DBRC instances, you could start with a buffer pool of 10 x (120 + 60) x 16K = 29M. Of course, a larger size might provide better performance by keeping more CIs in the pool and potentially avoiding some reads.

The RLS_MAXCFFEATURELEVEL parameter is used to specify if CIs larger than 4K will be cached in the SMSVSAM cache structure. 'Z' specifies that only 4K and smaller CIs will be cached. 'A' specifies that all may be cached. Since RECON CI sizes should be larger than 4K, you will need to specify 'A' to cache RECON CIs.

DEADLOCK_DETECTION(nnnn,nnnn) is used to specify the number of seconds in a local deadlock detection cycle and the number of local cycles in a global cycle. These default to 15 seconds and 4. Most installation will want faster deadlock detections. DEADLOCK_DETECTION(1,1) is recommended.

MAXLOCKS is used to control the issuing of warning messages about the number of locks held. These are messages IGW859I and IGW10074I. You might want to set these values to 50% and 80% of the number of record list entries in the lock structure.

RLSTMOUT may be specified but DBRC overrides this value. This is the time that a lock request will wait before it is timed out. DBRC always uses a value of 2 seconds.

CF_TIME is used to specify at what interval SMF type 42 records containing information and statistics on CF cache and lock structures are gathered.

Information On Demand

IMS Version 10

Log Streams

- Backout or undo log stream
  - Each instance of DFSMStvs has its own backout log stream
  - Name: IGWTVnnn.IGWLOG.SYSLOG
    - 'nnn' is TVSNAME from IGDSMSxx member
  - Used to hold back out records for updates
    - Updates to RECONs and other data sets using TVS
- Shunt log stream
  - Each instance of DFSMStvs has a shunt log stream
  - Name: IGWTVnnn.IGWSHUNT.SHUNTLOG
  - Used when backout requests fail and for long running units of recovery
    - Log data is moved from undo log to shunt log
  - Allows the undo log stream to be trimmed
    - Oldest records are removed from the undo log

Figure 14-42. Log Streams CMA01.0

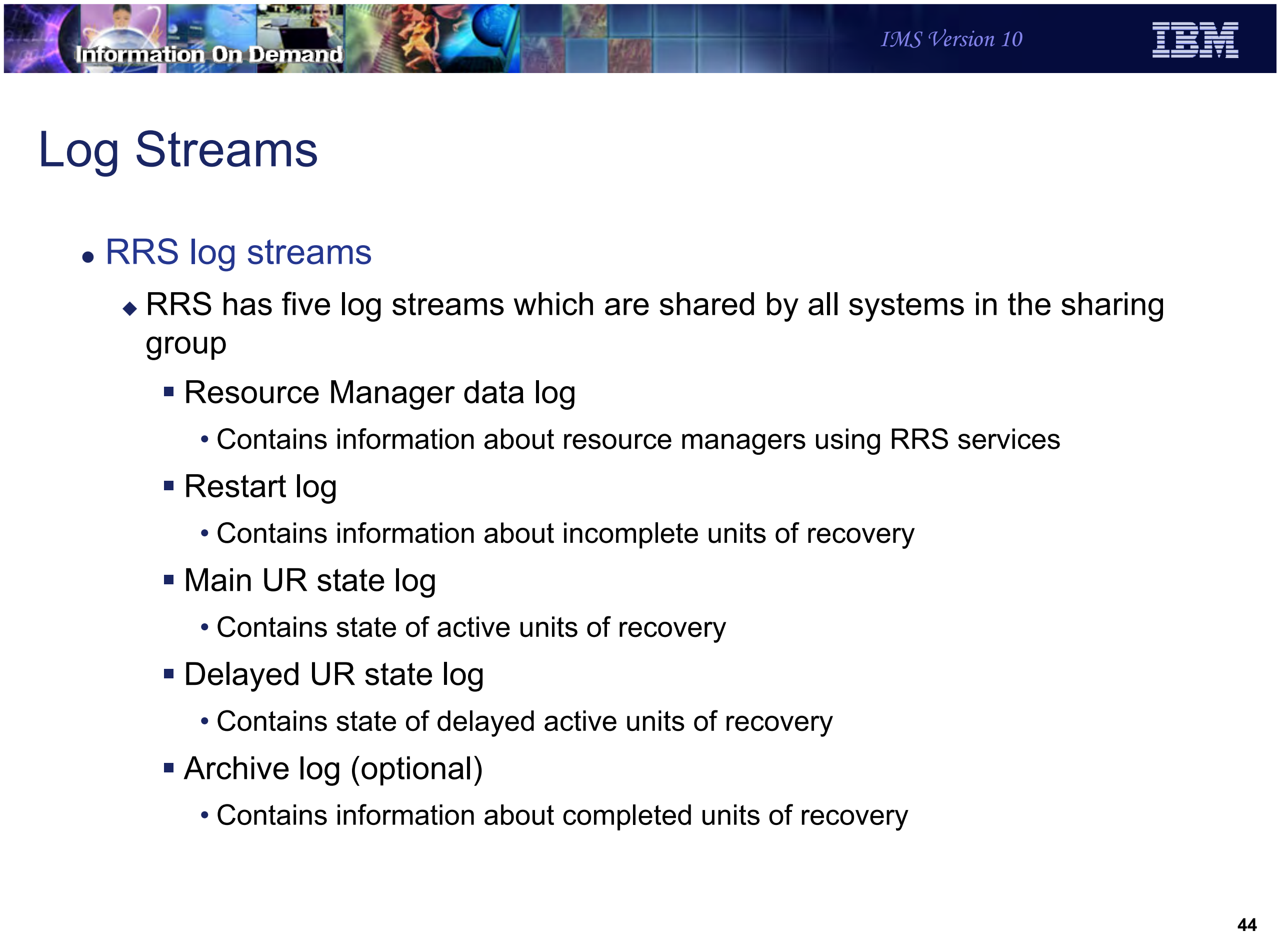
Notes:

DFSMStvs maintains an "undo" log stream for each instance of SMSVSAM. This means that there is one per z/OS system using DFSMStvs. The log stream contains "before" images of updates made by users of this DFSMStvs instance. It is used to back out any updates that have been done by units of recovery which fail or are victims in a deadlock. The log stream name is IGWTVnnn.IGWLOG.SYSLOB where "nnn" is the TVSNAME assigned in the IGDSMSxx member of SYS1.PROCLIB.

Typically, backout records do not need to be maintained for a long time. When the unit of recovery is committed, these log records may be deleted. On the other hand, there are occasions when the log records need to be kept for a long time. This would occur if a backout failed due to the inability to do the backout updates to the data set or if a unit of recovery was not committed for a long time. In these cases, log records may be moved to a shunt log stream. Each instance of DFSMStvs has a shunt log stream with the name IGWTVnnn.IGWSHUNT.SHUNTLOG where "nnn" is the TVSNAME.

Log records are moved from the undo log to the shunt log by activity keypoint processing. The frequency of this processing is determined by the AKP parameter in the IGDSMSxx member. If a unit of recovery has not logged for two AKPs, its log records are moved to the shunt log.

The use of the shunt log allows TVS to trim the undo log. TVS tracks the oldest log entry for any running unit of recovery. It can trim records that are older than this. This limits the size of the undo log and tends to keep all of its records in the log structure.


Information On Demand
IMS Version 10
IBM

Log Streams

- RRS log streams
  - RRS has five log streams which are shared by all systems in the sharing group
    - Resource Manager data log
      - Contains information about resource managers using RRS services
    - Restart log
      - Contains information about incomplete units of recovery
    - Main UR state log
      - Contains state of active units of recovery
    - Delayed UR state log
      - Contains state of delayed active units of recovery
    - Archive log (optional)
      - Contains information about completed units of recovery

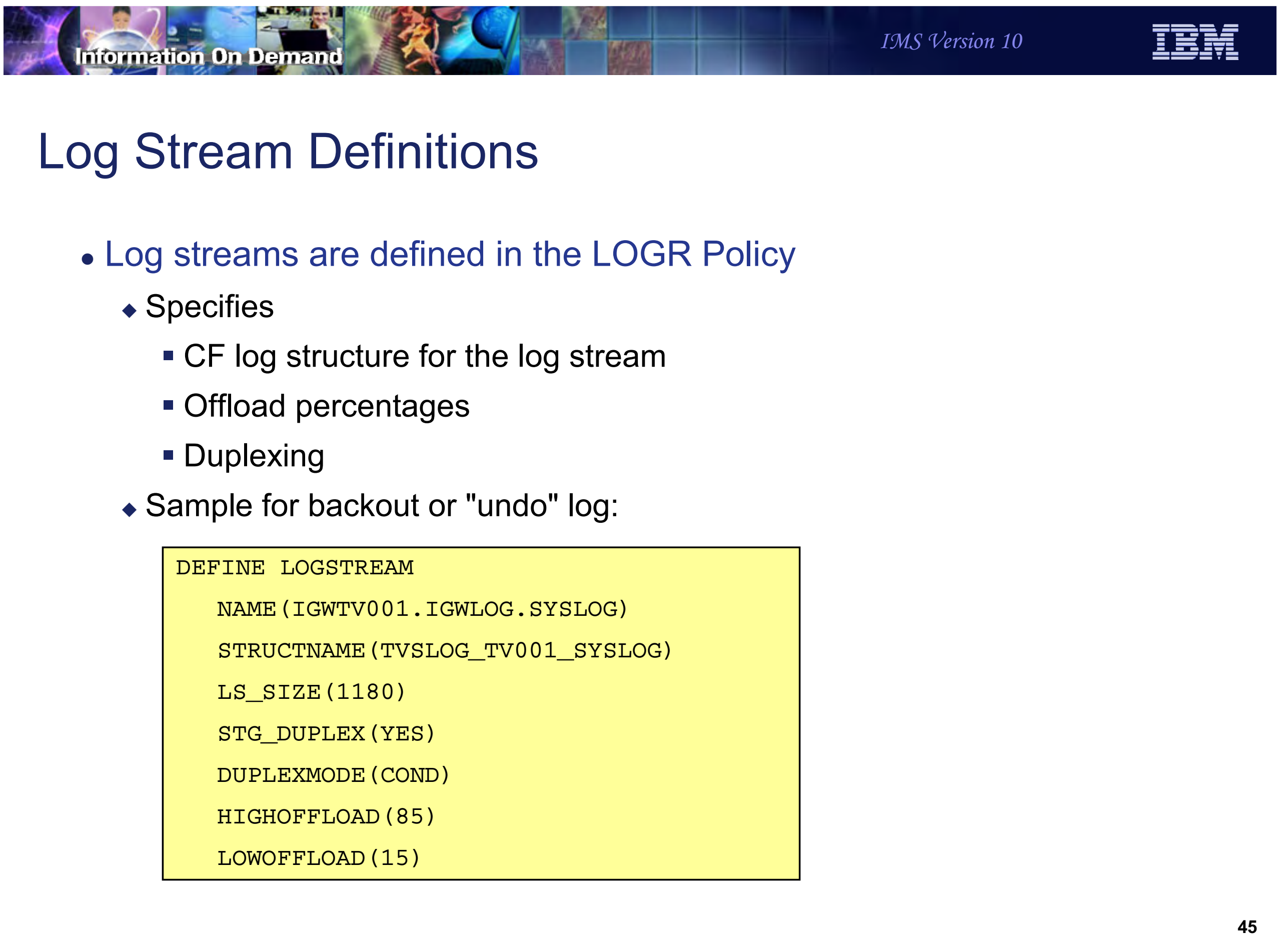
Figure 14-43. Log Streams CMA01.0

Notes:

Each request to DBRC is a unit of recovery which is tracked and logged by RRS.

RRS has multiple log streams. These log streams are used for various purposes. DFSMStvs is only one of several components which use RRS. Other IMS users of RRS include APPC, ODBA, and OTMA. Most installations will already have RRS log streams defined.

Many installations have already implemented RRS. For example, IMS requires RRS for ODBA and APPC protected conversations which use SYNCLVL=SYNPT.


Information On Demand
IMS Version 10

Log Stream Definitions

- Log streams are defined in the LOGR Policy
  - Specifies
    - CF log structure for the log stream
    - Offload percentages
    - Duplexing
  - Sample for backout or "undo" log:

```
DEFINE LOGSTREAM
   NAME(IGWTV001.IGWLOG.SYSLOG)
   STRUCTNAME(TVSLOG_TV001_SYSLOG)
   LS_SIZE(1180)
   STG_DUPLEX(YES)
   DUPLEXMODE(COND)
   HIGHOFFLOAD(85)
   LOWOFFLOAD(15)
```

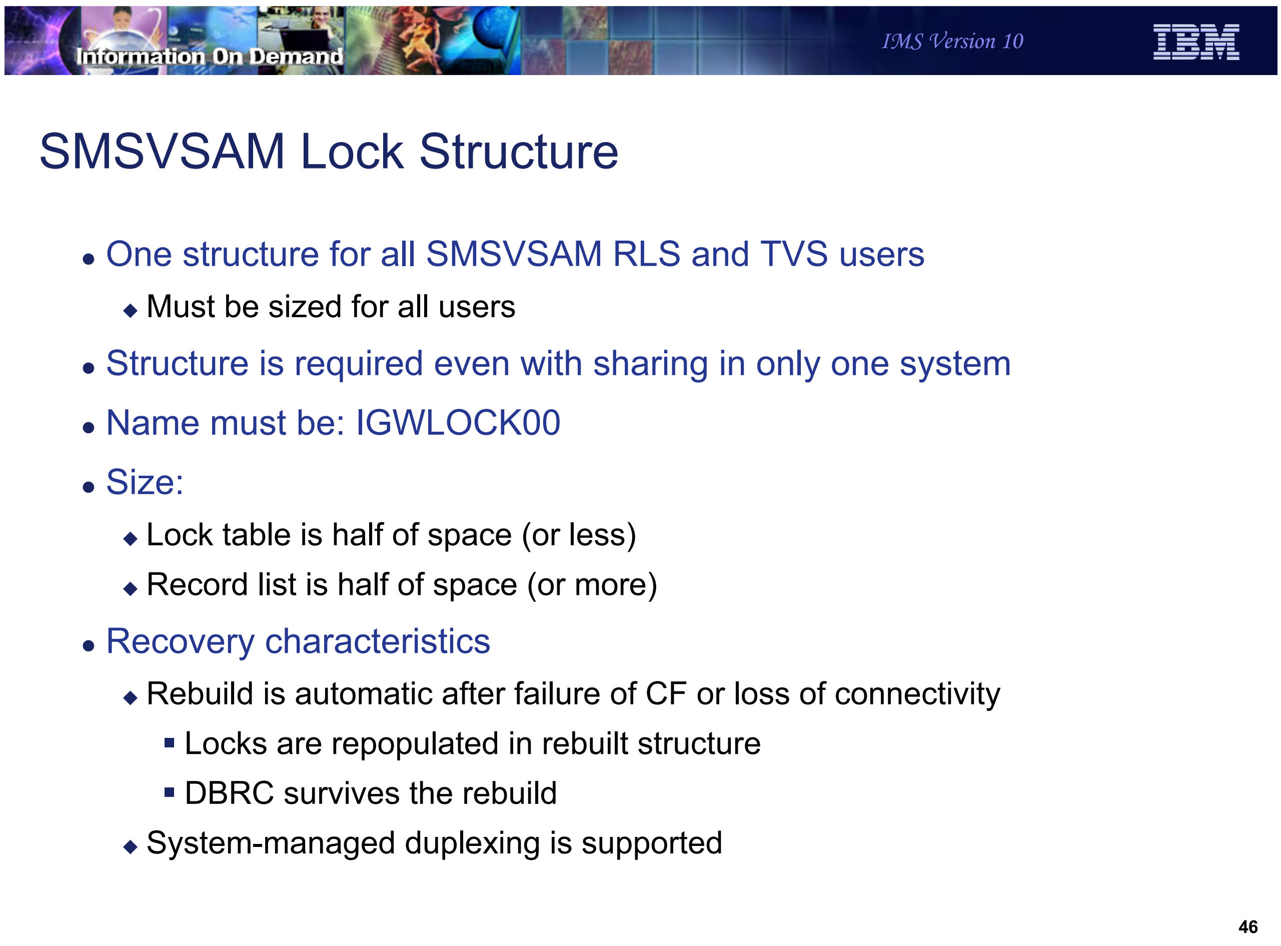
Figure 14-44. Log Stream Definitions CMA01.0

Notes:

Log streams must be defined in the LOGR policy for a sysplex. Each log stream is defined with a DEFINE LOGSTREAM statement. The policy assigns the log stream to a structure. The CFRM policy defines the size of the structure.

The sample definition shows some of the other parameters that you may define. The LS_SIZE parameter defines size of the data used for offloading the log stream from the structure. The size is specified in 4k blocks. STG_DUPLEX(YES) specifies that the log stream is duplexed in staging data sets when the condition defined in the DUPLEXMODE parameter is satisfied. DUPLEXMODE(COND) specifies that duplexing occurs when the CF structure is in a volatile CF or when the CF and the SMSVSAM address space reside in the same machine (CPC). The HIGHOFFLOAD parameter specifies the percentage of the structure that must be full before an offload process occurs. Offload moves log records from the structure to an offload data set. LOWOFFLOAD specifies the percentage of the log stream that may remain in the structure when an offload process completes. All parameters that you may specify in a LOGR policy are documented in the *z/OS MVS Setting Up a Sysplex* manual.

Since DBRC "transactions" that update the RECONs should be short-lived, undo logs for DBRC should be small. They are deleted by activity keypoint processing (AKP). In addition, the log records for long-lived transactions will be moved to the shunt log by activity keypoint processing (AKP).



Information On Demand

IMS Version 10

SMSVSAM Lock Structure

- One structure for all SMSVSAM RLS and TVS users
  - Must be sized for all users
- Structure is required even with sharing in only one system
- Name must be: IGWLOCK00
- Size:
  - Lock table is half of space (or less)
  - Record list is half of space (or more)
- Recovery characteristics
  - Rebuild is automatic after failure of CF or loss of connectivity
    - Locks are repopulated in rebuilt structure
    - DBRC survives the rebuild
  - System-managed duplexing is supported

Figure 14-45. SMSVSAM Lock Structure CMA01.0

Notes:

SMSVSAM uses a lock structure. It is always required, even when there is only one system using RLS or TVS. The lock structure holds locks for all of the data sets using RLS and TVS. This may include data sets other than RECONs. It must be sized to hold all of these locks.

The lock structure name is always IGWLOCK00. If the size of the lock structure is a power of 2, it is equally divided between the lock table and the record list. If it is not a power of two, the size of the lock table is the largest power of 2 that is smaller than half the structure size. The remaining space is used for the record list. Most DBRC actions do not hold many locks. These actions also typically hold locks for a short time. There is one exception. A LIST of the RECONs with the STATIC option gets a lock on every record. These locks are held until the LIST completes. A LIST with the STATIC option is generally not recommended for this reason.

The lock structure is automatically rebuilt after a failure of the lock structure, a failure of its CF, or loss of connectivity to its CF. If the SMSVSAM address spaces survive, the locks are repopulated in the rebuilt structure. DBRC will survive this recovery. This applies to all instances of DBRC including online systems, IMS batch jobs, and IMS utilities using DBRC.

System-managed duplexing may be used for the lock structure. Duplexing is not recommended when the structure is placed on a machine that does not contain an SMSVSAM address space connected to the structure. The overhead of duplexing is unlikely to be justified in this case since recovery is done without the loss of any users.

SMSVSAM Cache Structures

- Multiple structures may be used
  - Data sets are assigned to a storage class
  - Cache set name is defined for a storage class
  - Structure(s) are defined for each cache set
  - RECONs can have their own structure
    - Start with size equal to sum of the DSPBUFFs
- Store-through structure
  - All CIs are written to DASD when they are written to structure
- Recovery characteristics
  - Rebuild is automatic after failure of CF or loss of connectivity
    - Structure is empty after rebuild, buffers are invalidated in SMSVSAM
    - DBRC survives the rebuild
  - System-managed duplexing is not supported

Figure 14-46. SMSVSAM Cache Structures CMA01.0

Notes:

SMSVSAM uses one or multiple cache structures. Data sets are assigned to structures through their storage classes. A set of structures may be assigned to a storage class. The data sets in the storage class use the structures defined for the storage class. This allows you to set aside a structure for the exclusive use of the RECONs. By doing this you may size a structure to meet the needs of the RECONs. As a starting point you could make the structure large enough to hold all of the data that was previously in the DBRC buffers. These are defined with DSPBUFFS.

When storage class has multiple cache structures, data sets are assigned to a structure when they are opened. SMSVSAM attempts to balance the use of structures in a storage class.

The SMSVSAM cache structures are store-through structures. This means that all of the CIs that are written to the structure are also written to the RECON data sets. All committed updates are in the data sets.

If a Coupling Facility containing an SMSVSAM cache structure fails, the structure is automatically rebuilt on another CF. When the structure is rebuilt, it is not repopulated. All of the buffers containing CIs assigned to the structure are invalidated. This is similar to the actions that IMS takes when a full function cache structure is lost. If a structure cannot be rebuilt but the storage class has one or more other structures, the data sets using the failed structure are reassigned to another structure. DBRC survives the rebuilding of structures and the reassignment of data sets to other structures. These failures and recoveries are masked from DBRC by SMSVSAM.

System-managed duplexing is not supported for SMSVSAM cache structures. It is not needed since the loss of a structure is easily handled by the rebuild done by SMSVSAM.

Logger Structures

- TVS
  - Undo and shunt log streams for each TVS system
- RRS
  - RRS log streams
- Sizes
  - Sizes will depend on RECON activity and any other users of TVS and RRS
    - RECON activity should typically be small

Figure 14-47. Logger Structures CMA01.0

Notes:

Log structures are required for the DFSMStvs undo and shunt logs and for the RRS log stream.

PRA will typically not require much space in these structures. PRA "transactions" typically make few updates to the RECONs and do not wait for non-DBRC work. This tends to make its space requirements small. If a structure is not large enough, records are offloaded to logger data sets. This offloading depends on the size of the structure, the sizes of the log streams using the structure, and the HIGHOFFLOAD and LOWOFFLOAD parameters in the log stream definition in the LOGR policy.

Information On Demand
IMS Version 10
IBM

Security (RACF) Considerations

- UPDATE authority recommended for online DBRC address space
  - RECON data sets
    - and
  - STGADMIN.IGWSHCDS.REPAIR
  - Update authority is required for DBRC internal use of SHCDS commands
    - Lack of update authority could cause failures after RECON I/O errors
  - Read authority is required for all users
    - DBRC users without update authority could fail after a RECON I/O error

- ALTER authority required for RECONs to maintain LOG() parameter
  - ALTER authority required for job that does
    - CHANGE.RECON ACCESS(PARALLEL|SERIAL)

Figure 14-48. Security (RACF) Considerations CMA01.0

Notes:

DBRC issues some DFSMS commands for recovery situations. These commands require update authority to the RECONs and the STGADMIN.IGWSHCDS.REPAIR resource. Read authority is required for all users for the RECONs and the "REPAIR" resource. Update authority is required in some recovery situations. If a RECON I/O error occurs, it is likely that update authority will be required for the recovery to the spare RECON. A lack of update authority would result in the user failing. Obviously, it would be good for no users to fail, however, it is probably essential that online systems not fail.

DBRC ALTERs the RECONs when switching from serial to parallel access or from parallel to serial access. This should not be a typical operation. The ALTER is used to change from LOG(NONE) to LOG(UNDO) or from LOG(UNDO) to LOG(NONE). ALTER authority is required for the DBRC jobs that issue the command. DBRC also issues an ALTER when reconfiguring the RECONs if the spare has the wrong setting for the LOG parameter. If the spare has been defined with the correct LOG value, the ALTER is not required.

Parallel RECON Access Implementation

- PRA is turned on with a RECON setting

```
CHANGE.RECON ACCESS(SERIAL|PARALLEL)
INIT.RECON ACCESS(SERIAL|PARALLEL)
```

  - PARALLEL turns on PRA
    - Requires DFSMStvs environment
    - Requires "pair and spare" RECONs
  - SERIAL turns off PRA
- IMS does not have to be shut down to change access

Figure 14-49. Parallel RECON Access Implementation CMA01.0

Notes:

Parallel RECON Access is turned on by specifying ACCESS(PARALLEL) on the CHANGE.RECON or INIT.RECON command. It may be turned off with ACCESS(SERIAL). SERIAL is the default for the INIT.RECON command.

Since PRA requires DFSMStvs a complete DFSMStvs environment must be active when the command is issued. For example, the command must be issued on a z/OS system with an SMSVSAM address space and the lock and cache structures must be defined.

A good pair of active RECONs and a spare RECON must be available when the CHANGE.RECON ACCESS(PARALLEL) command is issued.

There cannot be an active RSR tracking subsystem when a change to parallel access is made.

LIST.xxx Command Options

- LIST.xxx CONCURR|STATIC [QUIESCE]
  - CONCURR - updates may occur during list process
    - Output may not be consistent
  - STATIC - produces output that is consistent as of a point in time
    - Locks held until list processing completes
  - STATIC QUIESCE - RECONs quiesced during list processing
    - No concurrent activity to the RECONs is allowed
      - Batch and utility jobs wait (even if not doing RECON activity)
    - Locking is not used in most cases
    - Consider using this option when static is required and many records will be listed (e.g. LIST.RECON)
- CHANGE.RECON LIST(STATIC|CONCURR)
  - Sets default for LIST commands

Figure 14-50. LIST.xxx Command Options CMA01.0

Notes:

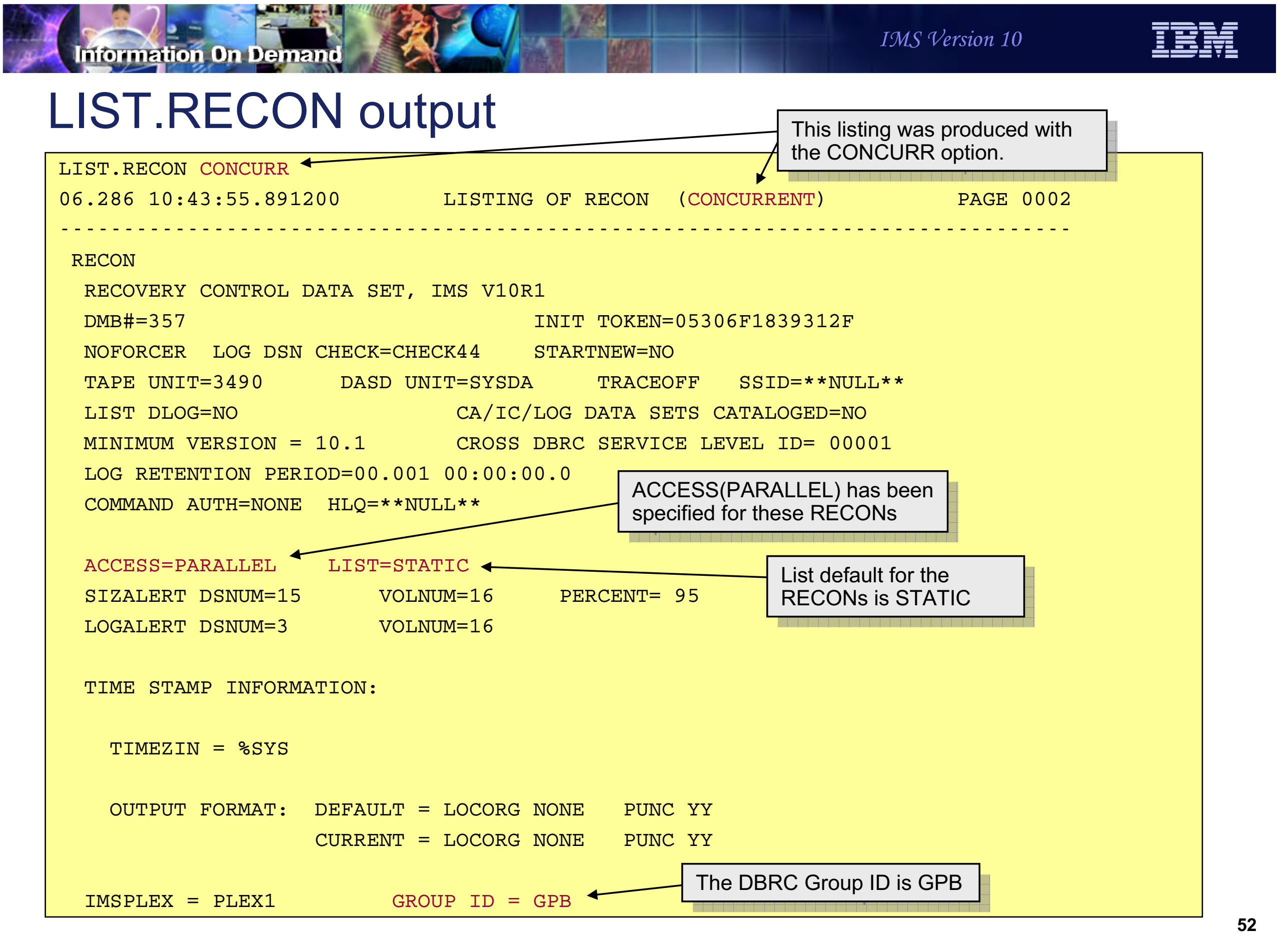
The LIST commands have new options with PRA. They are implemented with the CONCURR, STATIC, and QUIESCE keywords. These keywords are ignored when PRA is not being used.

If the CONCURR keyword is used, updates may occur to the RECONs while the list is being produced. This could make the output inconsistent since some of the records could be listed before a change and some after the change. For example, a database record might include an authorization to a subsystem but the subsystem record might not show the database as authorized to the subsystem. When CONCURR is used, a lock for a record is held only to read the record.

If the STATIC keyword is used, the listing is consistent. For example, if a database record listing includes an authorization to a subsystem, the subsystem record will show the database as authorized to the subsystem. When STATIC is used without the QUIESCE keyword, PRA locks each record when it reads it and holds the lock for the duration of the list processing.

The QUIESCE keyword may be used with STATIC. It quiesces all other activity to the RECONs during list processing. Other DBRC instances cannot access the RECONs during this time. Since all other activity is quiesced, locking is not needed to provide integrity or consistency unless there are retained locks. If there are retained locks from a failed DBRC for which recovery has not been done, a lock is held while DBRC is positioned on a record. QUIESCE reduces the overhead of the LIST command since it eliminates locking. You should consider using QUIESCE when a LIST command will read many RECON records. During the quiesce process batch jobs and utilities which use DBRC wait even if they are not attempting to access the RECONs. For example, batch jobs do no DL/I calls during the QUIESCE process.

The default of CONCURR or STATIC for the LIST command is set my the CHANGE.RECON LIST command. When RECONs are upgraded to V10, the default is set to STATIC. The QUIESCE keyword cannot be set by default.



LIST.RECON output

```
LIST.RECON CONCURR
06.286 10:43:55.891200        LISTING OF RECON  (CONCURRENT)          PAGE 0002
--------------------------------------------------------------------------------
 RECON
  RECOVERY CONTROL DATA SET, IMS V10R1
  DMB#=357                           INIT TOKEN=05306F1839312F
  NOFORCER  LOG DSN CHECK=CHECK44    STARTNEW=NO
  TAPE UNIT=3490      DASD UNIT=SYSDA     TRACEOFF   SSID=**NULL**
  LIST DLOG=NO                 CA/IC/LOG DATA SETS CATALOGED=NO
  MINIMUM VERSION = 10.1       CROSS DBRC SERVICE LEVEL ID= 00001
  LOG RETENTION PERIOD=00.001 00:00:00.0
  COMMAND AUTH=NONE  HLQ=**NULL**

  ACCESS=PARALLEL    LIST=STATIC
  SIZALERT DSNUM=15      VOLNUM=16      PERCENT= 95
  LOGALERT DSNUM=3       VOLNUM=16

  TIME STAMP INFORMATION:

    TIMEZIN = %SYS

    OUTPUT FORMAT:  DEFAULT = LOCORG NONE    PUNC YY
                    CURRENT = LOCORG NONE    PUNC YY

  IMSPLEX = PLEX1         GROUP ID = GPB
```

This listing was produced with the CONCURR option.

ACCESS(PARALLEL) has been specified for these RECONs

List default for the RECONs is STATIC

The DBRC Group ID is GPB

52

Figure 14-51. LIST.RECON output CMA01.0

Notes:

This is an example of some of the output of a LIST.RECON command. The line which includes, "LISTING OF RECON (CONCURRENT)" indicates that this listing was produced with the CONCURR option. In this case, the LIST command included the CONCURR keyword.

Near the middle of the example there is a line of "ACCESS=PARALLEL   LIST=STATIC". This indicates that PRA is active for this RECON and that the default of LIST commands in STATIC.

At the bottom of the example is "GROUP ID = GPB". GPB is the DBRC group ID used for this set of RECONs.

Locking of VSAM Records with PRA

- Locks are either shared or exclusive
  - Shared for get requests
  - Exclusive for 'get for update' requests
- Locks held during access or until a commit point
  - Locks for updates held until commit
  - Locks for read depends on options
    - LIST command with CONCURR
      - Lock released after record is read
    - LIST command with STATIC
      - Locks held until commit
    - Other reads hold locks until commit
      - Except reads used to browse multiple records looking for a specific record

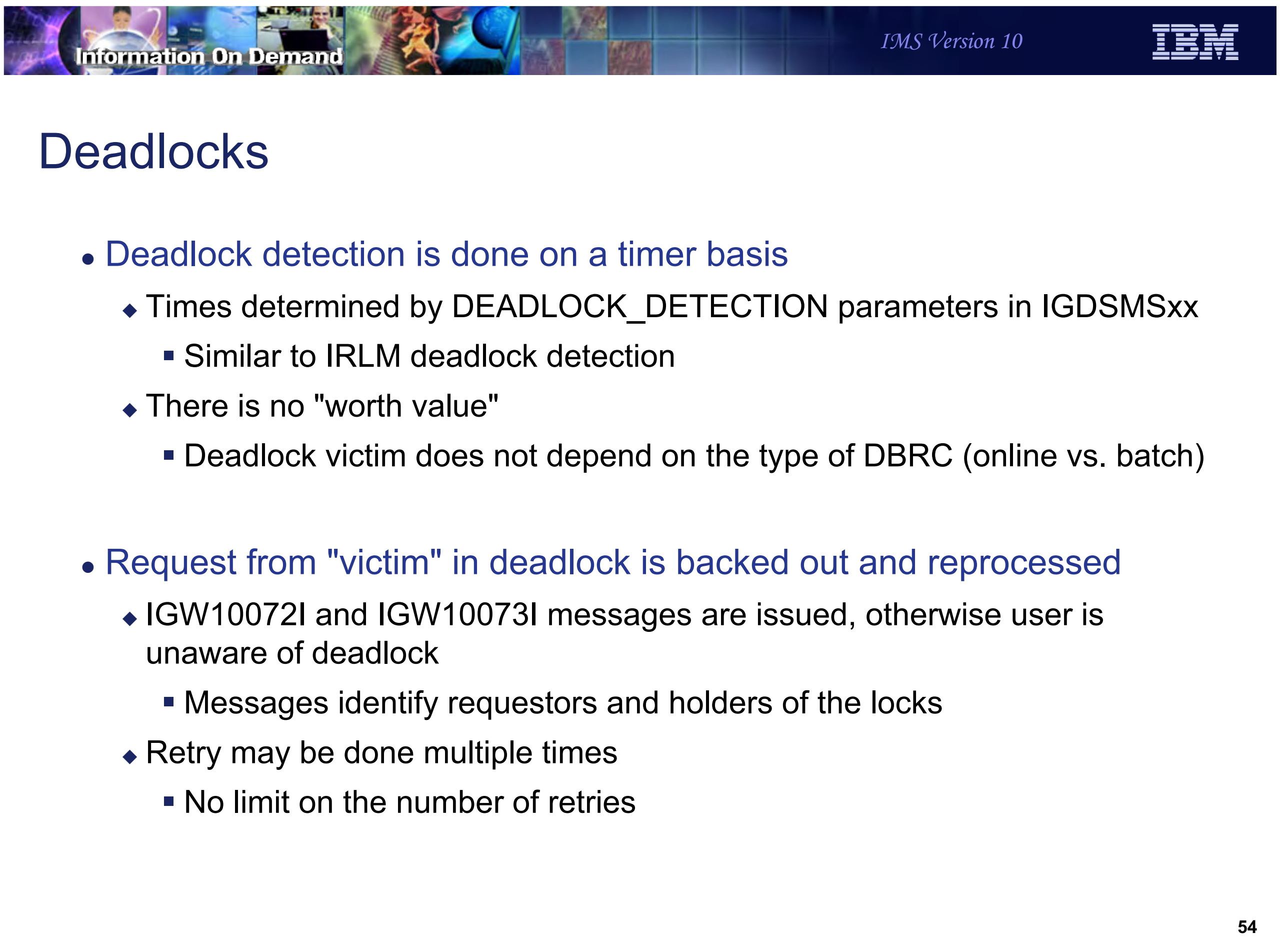
Figure 14-52. Locking of VSAM Records with PRA CMA01.0

Notes:

PRA locks RECON records when they are read. The locks are either at a share level or an exclusive level. As the names suggest, multiple DBRC instances may hold the lock for a record at the same time if they are all at the share level. There can be only one holder of a lock for a record at the exclusive level.

VSAM get requests use share level locks. VSAM get for update requests use exclusive level locks.

Commit occurs when the DBRC request from IMS or the DBRC command has completed. Exclusive level locks are held until commit. Share level locks may be released before commit. Typically, share level locks are held until commit. When the CONCURR option is used for a LIST command, the locks are released as part of the read process. When the STATIC is used for a LIST command, the locks are held until commit. Sometimes DBRC browses RECON records of a type looking for a specific record. When this occurs, locks are obtained and released as each record is read. When the specific record is found, the lock is obtained again and held until commit.



Information On Demand IMS Version 10 IBM

Deadlocks

- Deadlock detection is done on a timer basis
  - Times determined by DEADLOCK_DETECTION parameters in IGDSMSxx
    - Similar to IRLM deadlock detection
  - There is no "worth value"
    - Deadlock victim does not depend on the type of DBRC (online vs. batch)
- Request from "victim" in deadlock is backed out and reprocessed
  - IGW10072I and IGW10073I messages are issued, otherwise user is unaware of deadlock
    - Messages identify requestors and holders of the locks
  - Retry may be done multiple times
    - No limit on the number of retries

Figure 14-53. Deadlocks CMA01.0

Notes:

The deadlock detection process of SMSVSAM is similar to that done by IRLM. There is a local cycle and a global cycle. During the local cycle deadlocks within a SMSVSAM are detected. During the global cycle deadlocks between units of recovery using different SMSVSAMs are detected. The local cycle uses a waiters list in the detection process. At each cycle a list is built of those units of recovery which are waiting. If the same wait exists in two successive cycles, the locks are examined to see if a deadlock exists. When a deadlock is found, a victim is chosen. Unlike the IRLM implementation, there is no "worth" value associated with units of recovery. A victim is chosen without regard to the type of DBRC instance in which it is executing.

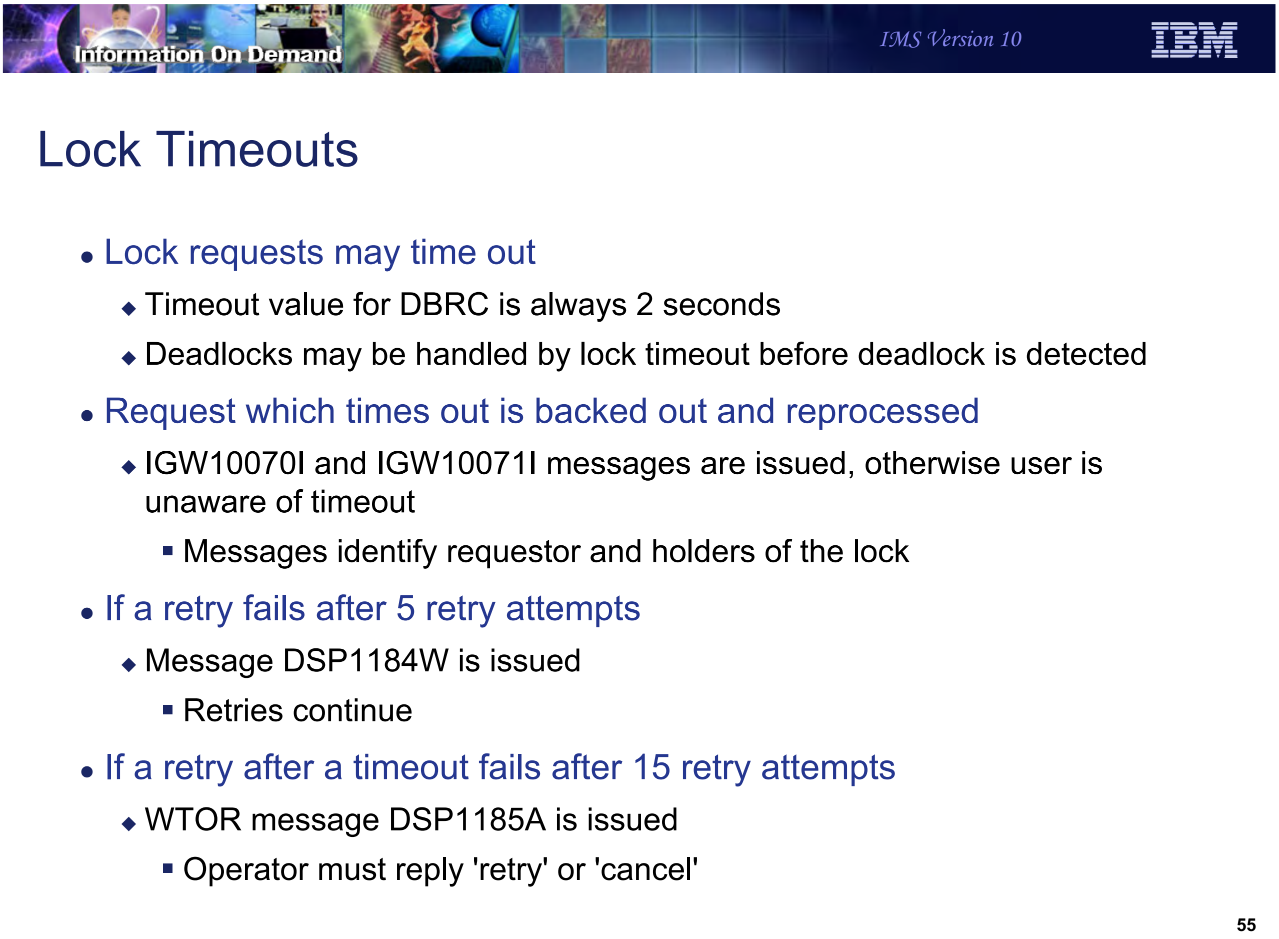
The UOR which is the victim in a deadlock is backed out and its request is reprocessed.

If a deadlock occurs an IGW10072I message is sent to the console. It is followed by IGW10073I messages. An IGW10073I message is issued once for each DBRC in the deadlock chain and includes the record that that DBRC holds and which record it is waiting for. The messages are:

IGW10072I *jobname stepname urid* VSAM RLS DETECTED A DEADLOCK. THERE ARE *nn* UNITS OF RECOVERY IN THE DEADLOCK CHAIN

IGW10073I {UNIT OF RECOVERY *urid* | SUBSYSTEM NAME *subsys* TRANSACTION ID *tranid*} RUNNING IN JOB *jobname* HOLDS {ADD TO END LOCK | EXCLUSIVE LOCK ON KEY | SHARED LOCK ON KEY} ON BASE CLUSTER NAME *cluster* [ALTERNATE INDEX NAME *altindex*] AND IS WAITING FOR {ADD TO END LOCK | EXCLUSIVE LOCK ON KEY | SHARED LOCK ON KEY} ON BASE CLUSTER NAME *cluster2* [ALTERNATE INDEX NAME *altindex2*] [HOLDING KEY VALUE = *key1* WAITING KEY VALUE = *key2*]

Deadlocks are retried indefinitely. This is no limit to the number of retries for them.



Information On Demand
IMS Version 10
IBM

Lock Timeouts

- Lock requests may time out
  - Timeout value for DBRC is always 2 seconds
  - Deadlocks may be handled by lock timeout before deadlock is detected
- Request which times out is backed out and reprocessed
  - IGW10070I and IGW10071I messages are issued, otherwise user is unaware of timeout
    - Messages identify requestor and holders of the lock
- If a retry fails after 5 retry attempts
  - Message DSP1184W is issued
    - Retries continue
- If a retry after a timeout fails after 15 retry attempts
  - WTOR message DSP1185A is issued
    - Operator must reply 'retry' or 'cancel'

Figure 14-54. Lock Timeouts CMA01.0

Notes:

If a time out occurs an IGW10070I message is sent to the console. It is followed by an IGW10071I message for each holder of the lock. The messages are:

> IGW10070I *jobname stepname urid* A REQUEST TIMED OUT WAITING FOR A LOCK. THERE ARE *nn* UNITS OF RECOVERY HOLDING THIS LOCK.
>
> IGW10071I {UNIT OF RECOVERY *urid* | SUBSYSTEM NAME *subsys* TRANSACTION ID *tranid*} RUNNING IN JOB *jobname* HOLDS {ADD TO END LOCK | EXCLUSIVE LOCK ON KEY | SHARED LOCK ON KEY} IN BASE CLUSTER NAME *cluster* [PATH NAME *path*] CAUSING {TRUE | FALSE} CONTENTION. [KEY VALUE = *key*]

A random delay is done before a retry after a timeout. This is done to resolve situations where two requestors wait on each other multiple times

Message DSP1184W is sent to indicate that five retries have been attempted. Retries continue. No action is required by the operator. The text of the message is: `DSP1184W VSAM ACCESS ERROR ENCOUNTERED  5 TIMES RC=0008 RSN=0022`

If a timeout for the same lock request occurs 15 times for a unit of recovery, the DSP1185A message is sent. It is sent after the backout of the UOR. The message is a WTOR. The operator must reply to the message. The operator may be able to determine which DBRC instance is holding the lock. If so, the RETRY reply may be issued after the holder of the lock has completed its work. The text of the message is `DSP1185A VSAM ACCESS ERROR ENCOUNTERED 15 TIMES RC=0008 RSN=0022 - REPLY ’RETRY’ OR ’CANCEL’`

The RSN=0022 in the DSP1184W and DSP1185A messages indicate the reason for the error is a timeout for a lock request.

Retries After Lock Timeouts and Deadlocks

- Timeout or Deadlock results in backout and retry of the unit of recovery
  - Backout only applies to RECON updates
    - Does not apply to other data sets
  - This could produce duplicate output for GENJCL, LIST, and other commands
    - Output may have already been written to SYSPRINT and/or JCLOUT
  - Message issued to inform user
    - DSP1186I DBRC COMMAND RETRY DRIVEN
    - Output produced before this message should be ignored or discarded

Figure 14-55. Retries After Lock Timeouts and Deadlocks CMA01.0

Notes:

As explained before, a lock timeout or a deadlock results in the backout of the unit of recovery and the reprocessing of the DBRC request. The backout applies to the updates to the RECONs. It does not apply to any other output data sets, such as SYSPRINT or the JCLOUT data set used for GENJCL commands. Timeouts and deadlocks may produce duplicate outputs to these data sets. DBRC recognizes the possibility of these occurrences. It issues the DSP1186I message when a command is retried. This warns the user that there may be duplicate output from the command. Duplicate output tends to be a problem with GENJCL and LIST commands. It could occur for other commands, but the consequences are less serious. For example, the SYSPRINT output of a CHANGE.DB command could contain repeated lines.

The explanation of the DSP1186I message is:

> An error that could be retried was detected. DBRC is attempting to reprocess a command that might have produced external output. DBRC retries the processing, which encountered errors, that it considers capable of being retried (for example, deadlock or timeout). In this instance, data might have been written to a data set (for

example, SYSPRINT) or the JCLOUT data sets. Since command processing is redriven, output produced prior to this message should be ignored. Output to a JCLOUT data set that is submitted directly to an internal reader may produce duplicate JCL or JCL that does not run.

Monitoring of Locks

- The D SMS,CFLS command provides locking information

```
-D SMS,CFLS
IEE932I 236
IGW320I 14:52:52 Display SMS,CFLS
PRIMARY STRUCTURE:IGWLOCK00 VERSION:B8FC73F0F2A13441 SIZE:16384K
RECORD TABLE ENTRIES:48525 USED:4
System   Interval   LockRate   ContRate  FContRate   WaitQLen
14:52:52 Display SMS,CFLS
SC64     1 Minute        3.3     25.969      0.000       0.33
SC64      1 Hour        84.0      4.797      0.416       0.10
SC64      8 Hour        17.8      1.004      0.092       0.00
SC64       1 Day         7.7      0.354      0.044       0.00
  (03)   1 Minute        2.2     17.714      0.000       0.19
  (03)    1 Hour        56.0      3.220      0.282       0.05
  (03)    8 Hour        11.4      0.671      0.059       0.00
  (03)     1 Day         4.5      0.236      0.024       0.00
***************** LEGEND *****************
LockRate = number of lock requests per second
CONTRATE = % of lock requests globally managed
FCONTRATE = % of lock requests falsely globally managed
WaitQLen = Average number of requests waiting for locks
```

Figure 14-56. Monitoring of Locks CMA01.0

Notes:

The D SMS,CFLS command displays locking information from SMSVSAM. In this example the lock structure size is 16384K or 16M. This implies that the lock table size is 8M which is half of 16M. There are 48252 record table (or record list) entries. Only 4 of these record table entries are being used at the time of the display.

The following lines show the locking rate, the contention rate, the false contention rate, and the average number of requests waiting for locks during the last minute, the last hour, the last 8 hours, and the last day. The data returned by the command is for the system where the command is issued and an average of all of the systems. The response shows the lock and contention rates for system SC64. The lines beginning with "(03)" are averages for all of the systems. The "(03)" indicates that there are three systems.

Information On Demand

IMS Version 10

SMSVSAM Messages About Excessive Locks

IGW326W * Warning *** DFSMS SMSVSAM RECORD TABLE IN IGWLOCK00 IS *percent* % FULL.**

- The record table in lock structure IGWLOCK00 is *percent* % full. This message will appear when the record table is more than 80% full.

IGW859I JOB *jobname* UNIT OF RECOVERY *urid* HAS REQUESTED *nnn* LOCKS

- This message is based on the first MAXLOCKS parameter in IGDSMS*xx*

IGW10074I JOB *jobname* UNIT OF RECOVERY *urid* HAS REQUESTED *nnn* LOCKS. SYSTEM MAXIMUM IS *mmm*

- This message is based on the increment value in the MAXLOCKS parameter in the IGDSMS*xx*

Figure 14-57. SMSVSAM Messages About Excessive Locks CMA01.0

Notes:

The IGW326W message is issued when the record table (record list) in the lock structure is 80% full. If the record table becomes full, new locks for updates cannot be granted.

The IGW859I message is issued when a unit of recovery first exceeds the first MAXLOCKS subparameter in the IGDSMSxx member.

The IGW10074I message is issued after the IGW859I message when the unit of recovery requests more locks. Its issuance is controlled by the second MAXLOCKS subparameter. This subparameter specifies an incremental number of locks. The message is issued each time the incremental number of additional locks is requested.

RMF Information

- Structure Activity Reports
  - Cache structure
    - Reports size, service times, reads from cache, cross invalidations
  - Lock structure
    - Reports size, service times, contention, false contention
- RMF III
  - VSAM RLS Activity report
    - Reports by storage class or by data set
      - Read rate
      - % reads from SMSVSAM pool, cache structure, and DASD
      - % reads for which SMSVSAM buffer was invalid
  - VSAM LRU Overview report
    - May be used to get sizing information for the SMSVSAM buffer pools

Figure 14-58. RMF Information CMA01.0

Notes:

Lock and cache structure usage information is also available from the RMF Structure Activity Report. IMS data sharing users are familiar with these reports for the IRLM lock structure and OSAM and VSAM cache structures. The same type of report is available for the SMSVSAM structures.

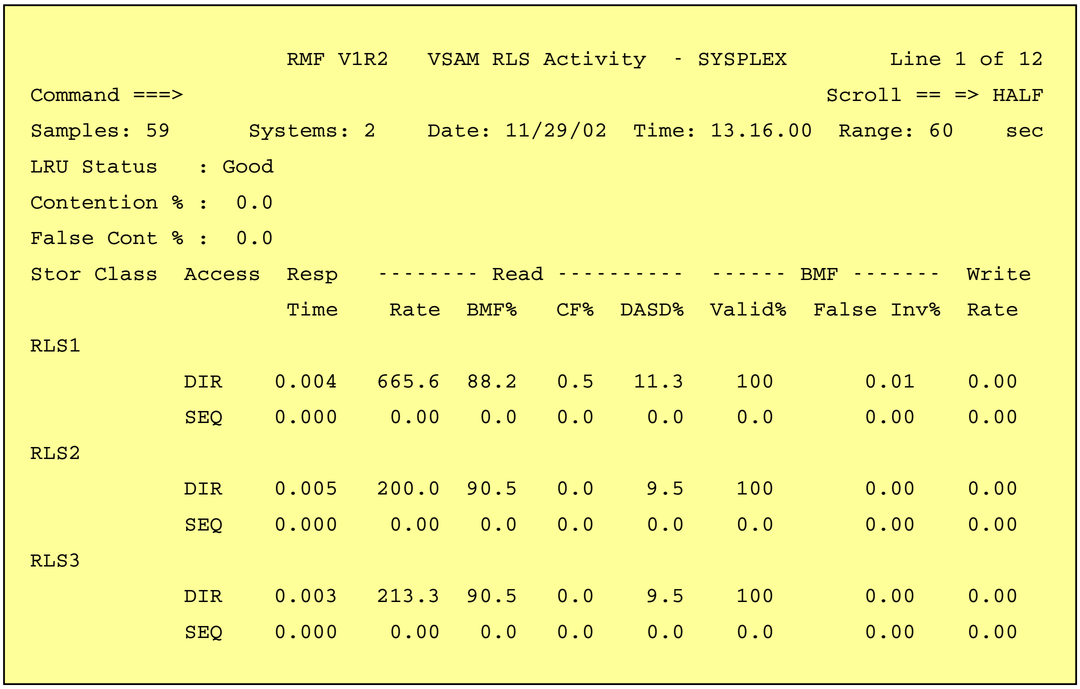
RMF III has a VSAM RLS Activity report. An example of this report is shown on the next page.

The RMF III VSAM LRU Overview report provides information about the size of the SMSVSAM buffer pool. The size of this pool is dynamically adjusted by SMSVSAM. The RLS_MAX_POOL_SIZE parameter sets the limit for the pool size, although it might be temporarily exceeded. During each LRU cycle, SMSVSAM determines whether the system is over the goal. If it is, adjustments are made. The report shows the percentage of time when the buffer pool exceeded the desired limit. It also reports on percentages of requests which were satisfied from the buffer pool, the cache structures, and DASD.

Information On Demand

IMS Version 10

RMF III - VSAM RLS Activity report

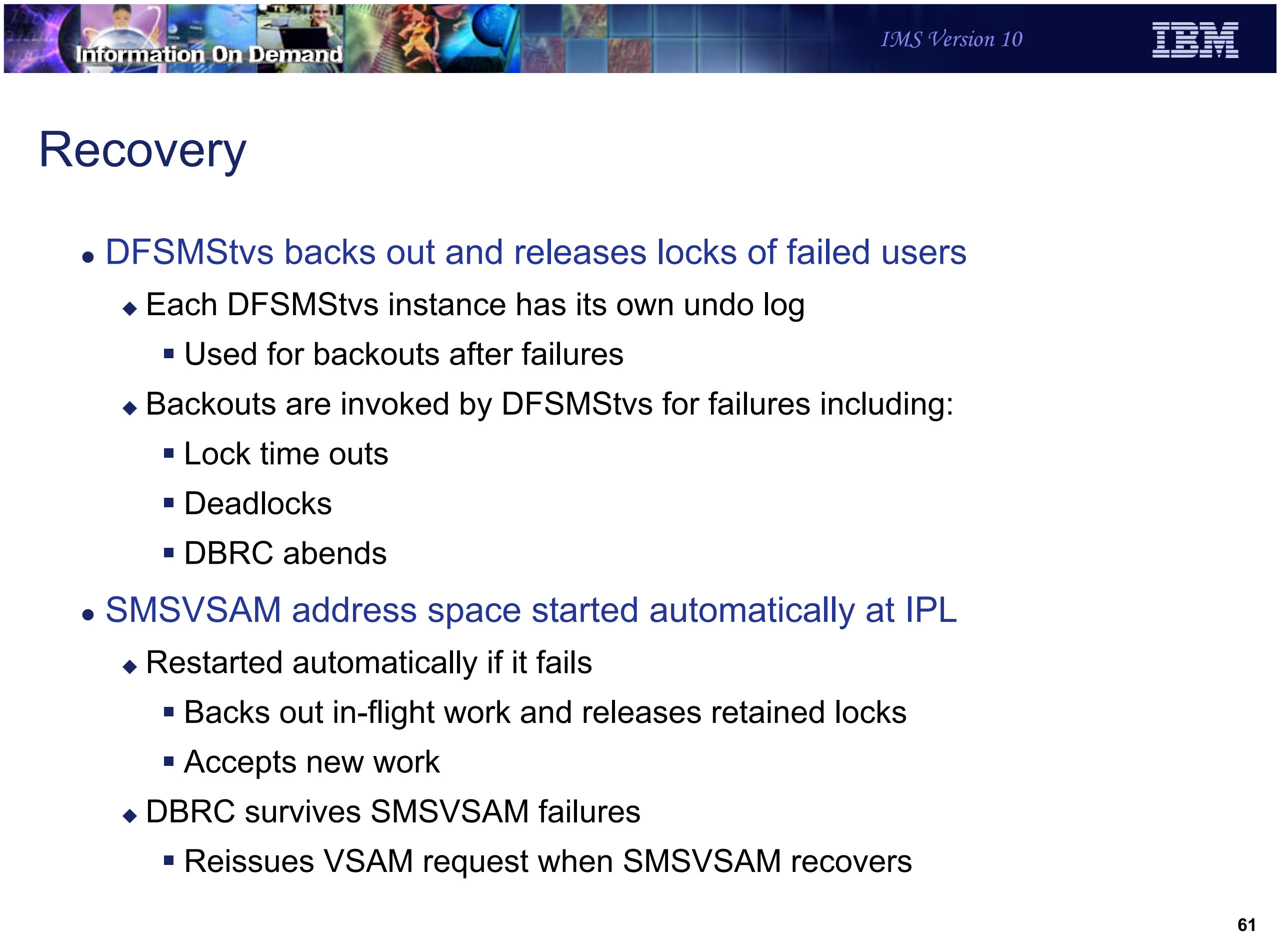


```
                    RMF V1R2   VSAM RLS Activity  - SYSPLEX       Line 1 of 12
Command ===>                                                    Scroll == => HALF
Samples: 59      Systems: 2    Date: 11/29/02  Time: 13.16.00  Range: 60    sec
LRU Status    : Good
Contention % :  0.0
False Cont % :  0.0
Stor Class  Access  Resp   -------- Read ---------  ------ BMF -------  Write
                    Time    Rate  BMF%   CF%  DASD%  Valid%  False Inv%  Rate
RLS1
            DIR    0.004   665.6  88.2   0.5   11.3    100        0.01   0.00
            SEQ    0.000    0.00   0.0   0.0    0.0    0.0        0.00   0.00
RLS2
            DIR    0.005   200.0  90.5   0.0    9.5    100        0.00   0.00
            SEQ    0.000    0.00   0.0   0.0    0.0    0.0        0.00   0.00
RLS3
            DIR    0.003   213.3  90.5   0.0    9.5    100        0.00   0.00
            SEQ    0.000    0.00   0.0   0.0    0.0    0.0        0.00   0.00
```

Figure 14-59. RMF III - VSAM RLS Activity report CMA01.0

Notes:

This is an example of the RMF III VSAM RLS Activity report. It reports on three storage classes used by RLS. These storage classes are RLS1, RLS2, and RLS3. For each class it reports direct and sequential access activities. Under the "Read" columns it reports the percentage of requests that are satisfied from the SMSVSAM buffers (BMF%), from the coupling facility structures (CF%), and from DASD (DASD%). Under "BMF" is shows the percentage of reads which found the CI in the SMSVSAM buffer pool but the buffer had been invalidated by an update to the CI in another system.


Information On Demand
IMS Version 10
IBM

Recovery

- DFSMStvs backs out and releases locks of failed users
  - Each DFSMStvs instance has its own undo log
    - Used for backouts after failures
  - Backouts are invoked by DFSMStvs for failures including:
    - Lock time outs
    - Deadlocks
    - DBRC abends
- SMSVSAM address space started automatically at IPL
  - Restarted automatically if it fails
    - Backs out in-flight work and releases retained locks
    - Accepts new work
  - DBRC survives SMSVSAM failures
    - Reissues VSAM request when SMSVSAM recovers

Figure 14-60. Recovery CMA01.0

Notes:

Each DFSMStvs instance has its own undo log and associated shunt log. The logs are shared by all users of the DFSMStvs instance, but they are not shared by DFSMStvs instances on different z/OS systems. The undo and shunt logs are used for backouts when there is a failure or deadlock by a user of DFSMStvs services, such as a DBRC instance.

When you IPL a z/OS system the SMSVSAM address space is automatically started if it has been defined for the system. If the address space fails, it is automatically restarted by z/OS. When it is restarted, it backs out any work that was in-flight at the time of the failure. It also releases the locks held for the work. After this recovery, it accepts new work.

Recovery

- Peer recovery for z/OS system failures
  - In case of system failures, peer instance of DFSMStvs may be started on another z/OS
    - This z/OS must have a primary instance of DFSMStvs on it
    - ARM is highly recommended
  - Peer uses log to backout in-flight work and release retained locks
  - Peer does not accept new work

Figure 14-61. Recovery CMA01.0

Notes:

Peer recovery is the process of completing the work that was left in an incomplete state due to the failure of an instance of DFSMStvs. Peer recovery is done by another instance of DFSMStvs. Peer recovery occurs only in cases of system failure, not merely when DFSMStvs fails. When a z/OS system fails, a peer recovery instance of DFSMStvs is used to back out in-flight work and release locks. The z/OS system where peer recovery is run must have a primary instance of DFSMStvs running on it.

Peer recovery is not automatically invoked unless Automatic Restart Management (ARM) is used to invoke it. Without ARM, an operator command is required to start peer recovery. ARM information is available in the *z/OS DFSMStvs Planning and Operating Guide* and in *z/OS MVS Setting Up a Sysplex*.

Peer recovery backs out the in-flight work of the failed system and releases the locks for this work. The peer does not accept new work.

The peer recovery instance of DFSMStvs does the following:

- It registers with VSAM RLS and RRS as the failed instance of DFSMStvs and indicates to RRS that it is beginning restart processing as the failed instance.
- It reads the failed instance's undo log to retrieve information about in-progress units of recovery.
- It invokes RRS to retrieve information about unit of recovery status.
- It processes the log data and the information returned by RRS to determine whether to commit or back out units of recovery. It backs out in-flight work and releases locks for this work.
- When all outstanding units of recovery have been processed, it unregisters with RRS and VSAM RLS.

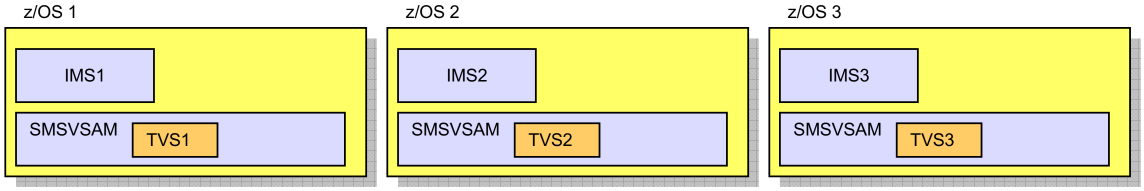
The peer recovery instance of DFSMStvs registers as the failed instance in order to gain access to the failed instance's resources. This registration persists until peer recovery is complete. As a result, should the failed instance restart, it will be unable to initialize because its attempt to register will fail. It may be necessary to restart DFSMStvs manually once peer recovery is complete.

Information On Demand

IMS Version 10

Peer Recovery Example

- Before failure:


z/OS 1
IMS1
SMSVSAM
TVS1
z/OS 2
IMS2
SMSVSAM
TVS2
z/OS 3
IMS3
SMSVSAM
TVS3

- After failure:
  - TVS1 peer recovery may be done on z/OS2 or z/OS3
    - It does not require that IMS1 be restarted
  - IMS1 may be restarted on a different z/OS

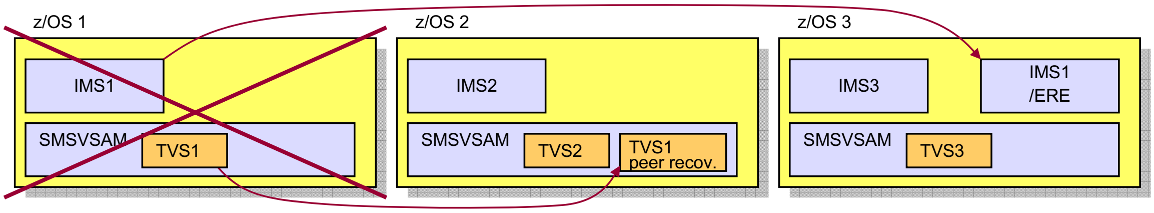
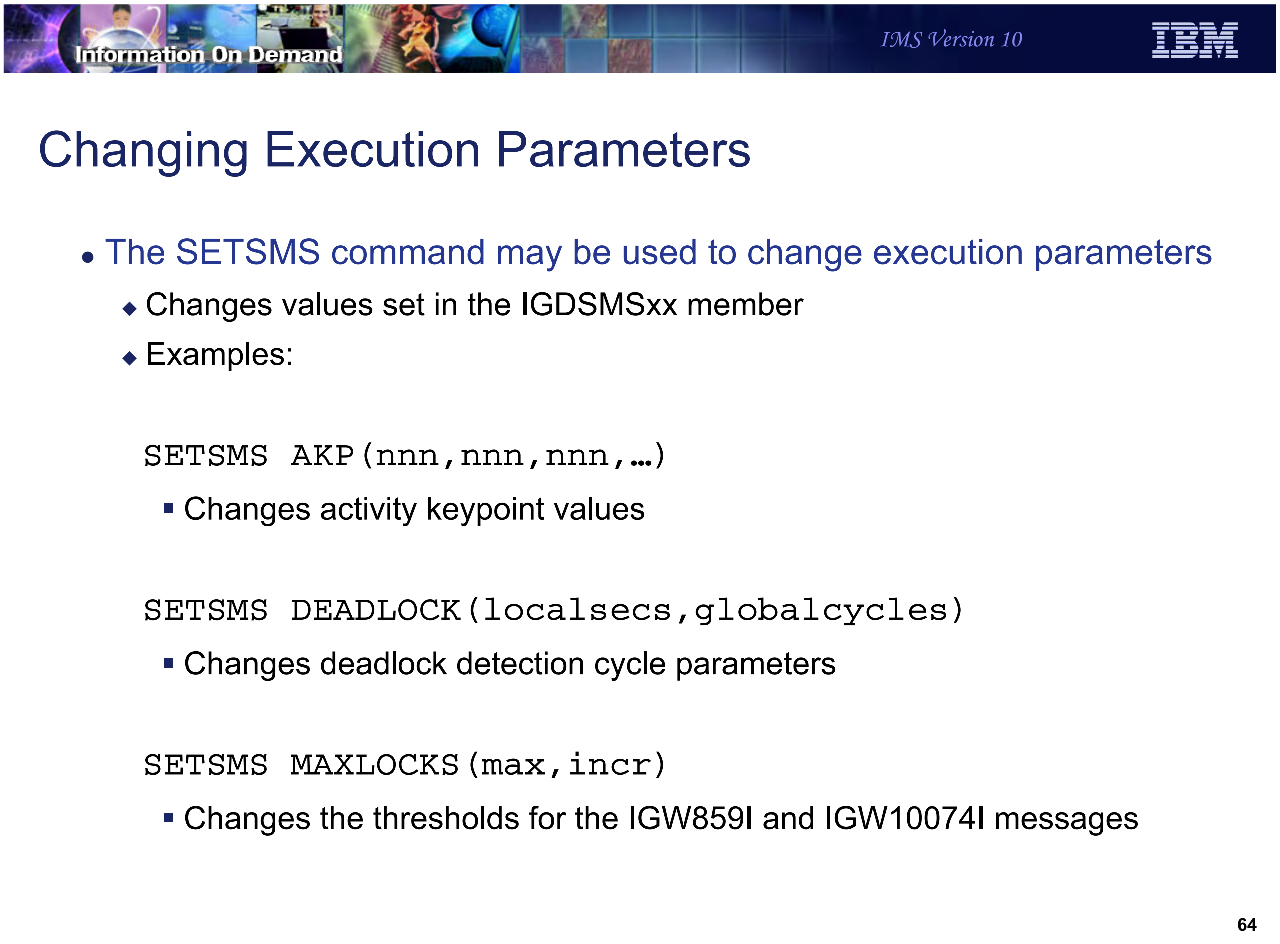

z/OS 1
IMS1
SMSVSAM
TVS1
z/OS 2
IMS2
SMSVSAM
TVS2
TVS1
peer recov.
z/OS 3
IMS3
IMS1
/ERE
SMSVSAM
TVS3

Figure 14-62. Peer Recovery Example CMA01.0

Notes:

This example illustrates that peer recovery may be done on any z/OS system which has a SMSVSAM address space with a current primary instance of DFSMStvs. Peer recovery does not require that the IMSs using it be restarted. Of course, you would probably want to restart any failed IMS online system. The emergency restart of IMS1 could be done on any z/OS which has a DFSMStvs instance. It does not have to be done on the same z/OS system where peer recovery is being done for the failed instance of DFSMStvs.


Information On Demand
IMS Version 10
IBM
64

Changing Execution Parameters

- The SETSMS command may be used to change execution parameters
  - Changes values set in the IGDSMSxx member
  - Examples:

```
SETSMS AKP(nnn,nnn,nnn,…)
```

- Changes activity keypoint values

```
SETSMS DEADLOCK(localsecs,globalcycles)
```

- Changes deadlock detection cycle parameters

```
SETSMS MAXLOCKS(max,incr)
```

- Changes the thresholds for the IGW859I and IGW10074I messages

Figure 14-63. Changing Execution Parameters CMA01.0

Notes:

The SETSMS command may be used to change parameters that were specified in the IGDSMSxx member. Examples of the kinds of parameters that may be changed are shown here.

Information On Demand
IMS Version 10
IBM

RECON Header/Header Extension Processing

- Serial access
  - RECON header contains DBRC options and "multiple update" (MUP) records
    - Options include:
      - Trace options, time format options, …
    - MUP record is used by DBRC requests which update multiple RECON records
      - It's a backout log
    - Every DBRC request reads the header
      - Options may have changed or backout may need to be done
  - RECON header extension has information about COPY1, COPY2, and spare
    - Every DBRC request reads the header extension
      - Reconfigures if the pair and spare have changed

Figure 14-64. RECON Header/Header Extension Processing CMA01.0

Notes:

This explains how the RECON header and header extension records are used when Parallel RECON Access is NOT used.

With serial access, the RECON header and header extension are read by each DBRC request.

The header is read for two reasons. It is read to determine the current RECON options. These include things such as trace options and time format options. Second, the header is read to determine is there is a multiple update (MUP) record. A MUP record is written before DBRC makes updates to multiple RECON records as part of one DBRC request. This is used to ensure that all of the updates are made, or if they are not made, that they may be backed out. MUP records contain information sufficient for backing out updates to the RECONs. After the MUP record is written, DBRC makes the updates to the indicated RECON records. When these are completed, the MUP record is deleted. Each DBRC request first checks for a MUP record. If one exists, it indicates that another DBRC instance has failed in the middle of making multiple updates. The DBRC instance which reads the MUP record, backs out the partial updates and deletes the MUP record. This mechanism provides integrity with serial access to the RECONs.

The header extension is read to determine the current RECON configuration. This is the current COPY1, COPY2 and spare data sets. They could have been changed by another DBRC instance since this DBRC last accessed the RECONs.

This processing differs with Parallel RECON Access.

Information On Demand
IMS Version 10
IBM

RECON Header/Header Extension Processing

- Parallel RECON Access
  - RECON header contains DBRC options
    - RECON Header is locked with share (CR) lock
      - Only waits for updaters of the header
      - Avoids almost all possible contentions
    - MUP records are not used
      - Incomplete units of recovery are backed out by DFSMStvs
  - RECON header extension has information about COPY1, COPY2, and spare
    - Each DBRC request does not read the header ext.
      - Changes in the configuration are propagated by SCI
      - Reduces RECON I/Os
    - Configuration information is read from the header ext. when a DBRC instance is initialized

Figure 14-65. RECON Header/Header Extension Processing CMA01.0

Notes:

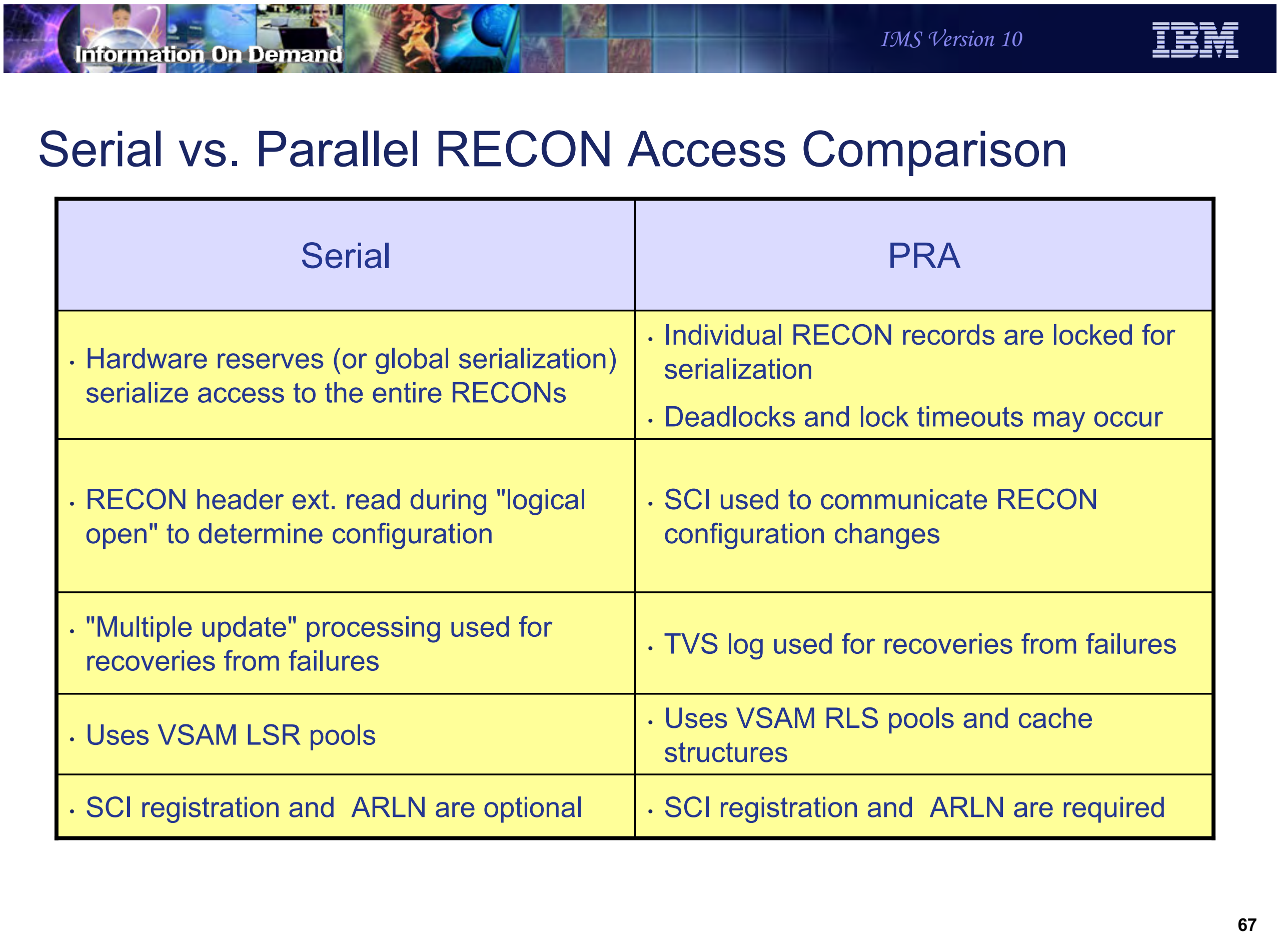
The header record is read by DBRC "logical open" but is locked with a shared lock except when updates are done. It is read to check on the current RECON options. Updates are rare. They are typically done by a DBRC CHANGE.RECON command.

Parallel RECON Access eliminates the need to read the RECON header extension for every DBRC request.

The RECON header extension still contains the current RECON configuration information, but DBRC does not need to read the header for every DBRC request. If a DBRC instance reconfigures the RECONs, it uses SCI to communicate the new configuration to the other instances. The configuration information is obtained from the header extension when a new DBRC instance is initialized. The header extension does not have to be read for every subsequent request of the DBRC instance.

MUP records are not used with Parallel RECON Access. Backouts of incomplete updates are done by DFSMStvs using its undo log stream. This does not eliminate the need to read the header for each DBRC request. The continues to be read to find any changes to RECON options. (e.g. if a CHANGE.RECON TRACEON has been issued).

These changes in processing of the header extension record with Parallel RECON Access eliminate a reason for possible contention on the header extension record. It also reduces the number of I/Os to the RECONs required for most DBRC requests.


Information On Demand
IMS Version 10
IBM

Serial vs. Parallel RECON Access Comparison

Serial	PRA
Hardware reserves (or global serialization) serialize access to the entire RECONs	Individual RECON records are locked for serialization Deadlocks and lock timeouts may occur
RECON header ext. read during "logical open" to determine configuration	SCI used to communicate RECON configuration changes
"Multiple update" processing used for recoveries from failures	TVS log used for recoveries from failures
Uses VSAM LSR pools	Uses VSAM RLS pools and cache structures
SCI registration and ARLN are optional	SCI registration and ARLN are required

Figure 14-66. Serial vs. Parallel RECON Access Comparison CMA01.0

Notes:

Serial processing uses hardware reserves or global data set serialization to serialize access to the RECONs. Only one DBRC instance may be accessing the RECONs at any time. PRA allows multiple DBRC instances to access the RECONs at the same time. It locks individual RECON records to provide integrity. Since it uses locking, there is the possibility of deadlocks or timeouts for lock requests.

Serial processing maintains the status of the RECONs in the header/header extension records. Every access to the RECONs first has to check the header/header extension for the current status. PRA uses SCI communication to notify other DBRC instances of any change in the RECON configuration. PRA does not access the header/header extension with every access to the RECONs.

Serial processing has special handling for requests that require updates to multiple RECON records. This is needed to maintain integrity for multiple updates. It writes a MUP record to the header/header extension before doing its intended multiple updates and then deletes the MUP information after the intended multiple updates are done. PRA does not have this processing. It uses TVS to maintain integrity for multiple updates. If one or more

RECON records are updated, but the other records in the request cannot be updated, the TVS log is used to back out the updates that were made.

Serial processing uses VSAM LSR (local shared resources) pools. Each DBRC instance has its own pools. These pools are invalidated at the beginning of each request to DBRC. This is necessary because another DBRC instance may have modified the CIs that are in the pool. PRA does not use LSR pools. It uses the SMSVSAM RLS pool in its LPAR. Parallel Sysplex buffer invalidations are used to invalidate only the buffers that have been updated by DBRC instances running in other LPARs. Cache structures are used in the invalidation process. These cache structures also maintain copies of RECON CIs so that some reads to the RECON data sets may be eliminated.

SCI registration and Automatic RECON Loss Notification are optional with serial access. They are required with PRA.

PRA Migration and Coexistence

- Parallel RECON Access cannot coexist with serial access
- PRA requires MINVERS('10.1')
- Vendor or user code which accesses the RECONs directly must be modified
  - DBRC API highly recommended
    - Supports PRA
  - Otherwise,
    - Must use RRS commit/backout
    - Invoke read integrity options on RPLs
    - Must handle new RPL error codes
    - …
    - However, there is no "quiesce" interface for vendor or user code
- RECON I/O exit routine (DSPCEXT0) must be modified

Figure 14-67. PRA Migration and Coexistence CMA01.0

Notes:

This is a summary of the migration and coexistence considerations. Most of these have been covered previously.

Products and application programs which access the RECONs natively must be adapted for DFSMStvs. The recommended way of handling this is the use of the DBRC API. The DBRC API supports both Parallel RECON Access and serial access. If the DBRC API is not used, the programs must support the DFSMStvs interfaces. These include the use of RRS for commit and backout, opening the RECONs for DFSMStvs access, and the use of RPLs which invoke DFSMStvs access. Even if they successfully implement these interfaces, full support cannot be done. For example, there is no interface to quiesce user code for the LIST command with the STATIC QUIESCE option.

PRA Migration and Coexistence

- PRA is invoked by CHANGE.RECON ACCESS(PARALLEL)
  - ALTERs RECON definition
    - LOG(UNDO)
- PRA Requires DFSMStvs environment
  - IGDSMSxx parameters
  - Structure and log stream definitions
  - SHCDS data sets
  - RACF authority
  - RRS
  - SMSVSAM address space
  - Updated operation and recovery procedures

Figure 14-68. PRA Migration and Coexistence CMA01.0

Notes:

Parallel RECON Access is invoked with the CHANGE.RECON ACCESS(PARALLEL) command. The command internally issues an ALTER for the RECON data sets changing the LOG parameter to LOG(UNDO). This requires that the DFSMStvs environment be present. Of course, operations procedures and recovery procedures should be updated for DFSMStvs.

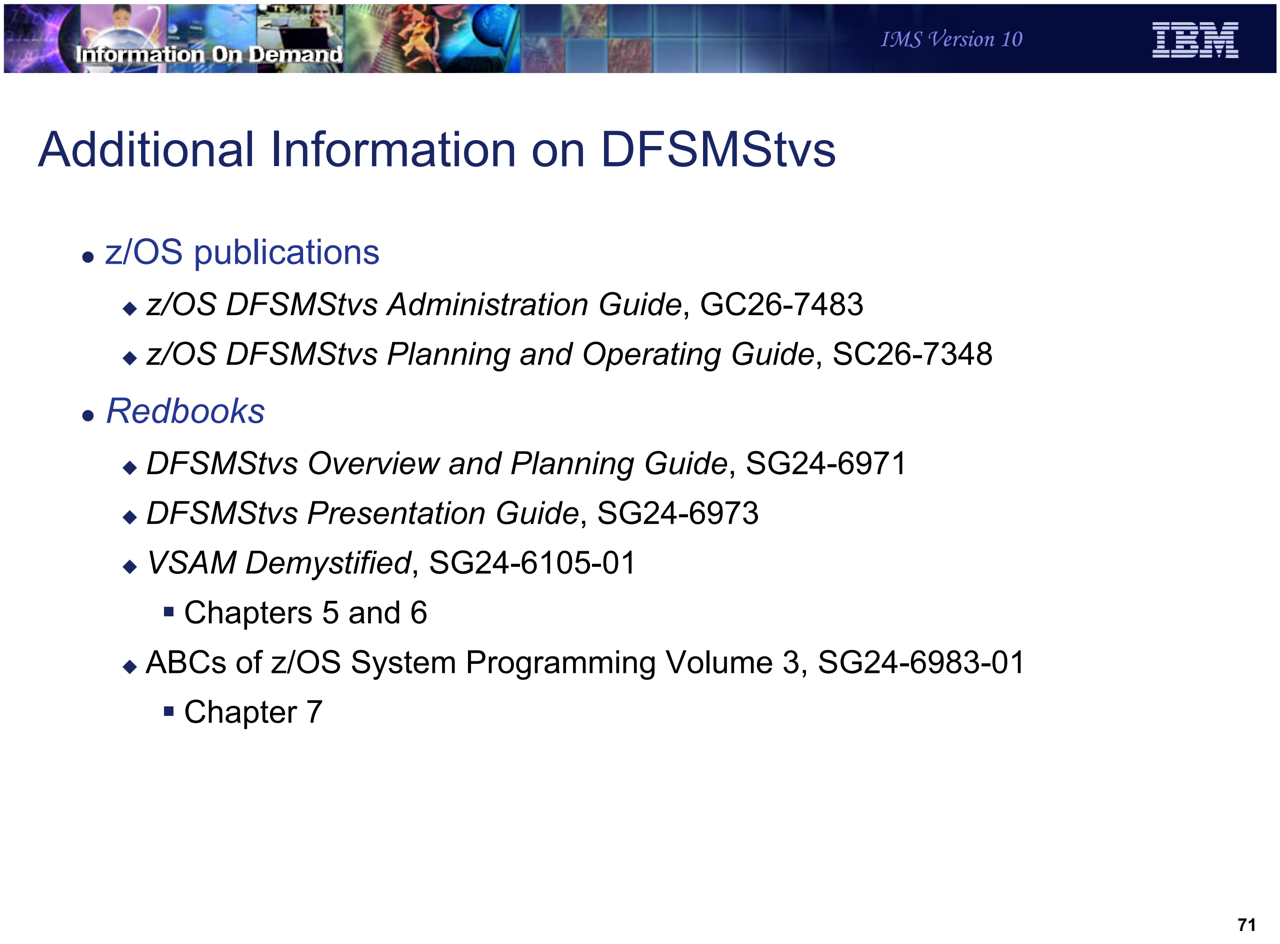
PRA Summary and Benefits

- Parallel RECON Access
  - Exploitation of Transactional VSAM
  - Concurrent RECON activity by multiple DBRC instances
- Benefits
  - Reduction of RECON contention
  - Increased throughput
  - Reduction of interference with online systems from batch jobs and utilities
  - Removal of growth constraint

Figure 14-69. PRA Summary and Benefits CMA01.0

Notes:

PRA exploits Transactional VSAM for the RECON data sets. This allows multiple DBRC instances, which are online systems, IMS batch jobs, and IMS utilities, to access the RECON data sets concurrently. Volume or global data set serialization is not required. This reduces RECON contention. The reduced contention provides for greater growth potential and increased throughput possibilities. The primary recipient of the benefits for most installations will be online systems since they will suffer less interference from batch jobs and utilities which are accessing the RECONs.


Information On Demand
IMS Version 10
IBM

Additional Information on DFSMStvs

- z/OS publications
  - *z/OS DFSMStvs Administration Guide*, GC26-7483
  - *z/OS DFSMStvs Planning and Operating Guide*, SC26-7348
- *Redbooks*
  - *DFSMStvs Overview and Planning Guide*, SG24-6971
  - *DFSMStvs Presentation Guide*, SG24-6973
  - *VSAM Demystified*, SG24-6105-01
    - Chapters 5 and 6
  - ABCs of z/OS System Programming Volume 3, SG24-6983-01
    - Chapter 7

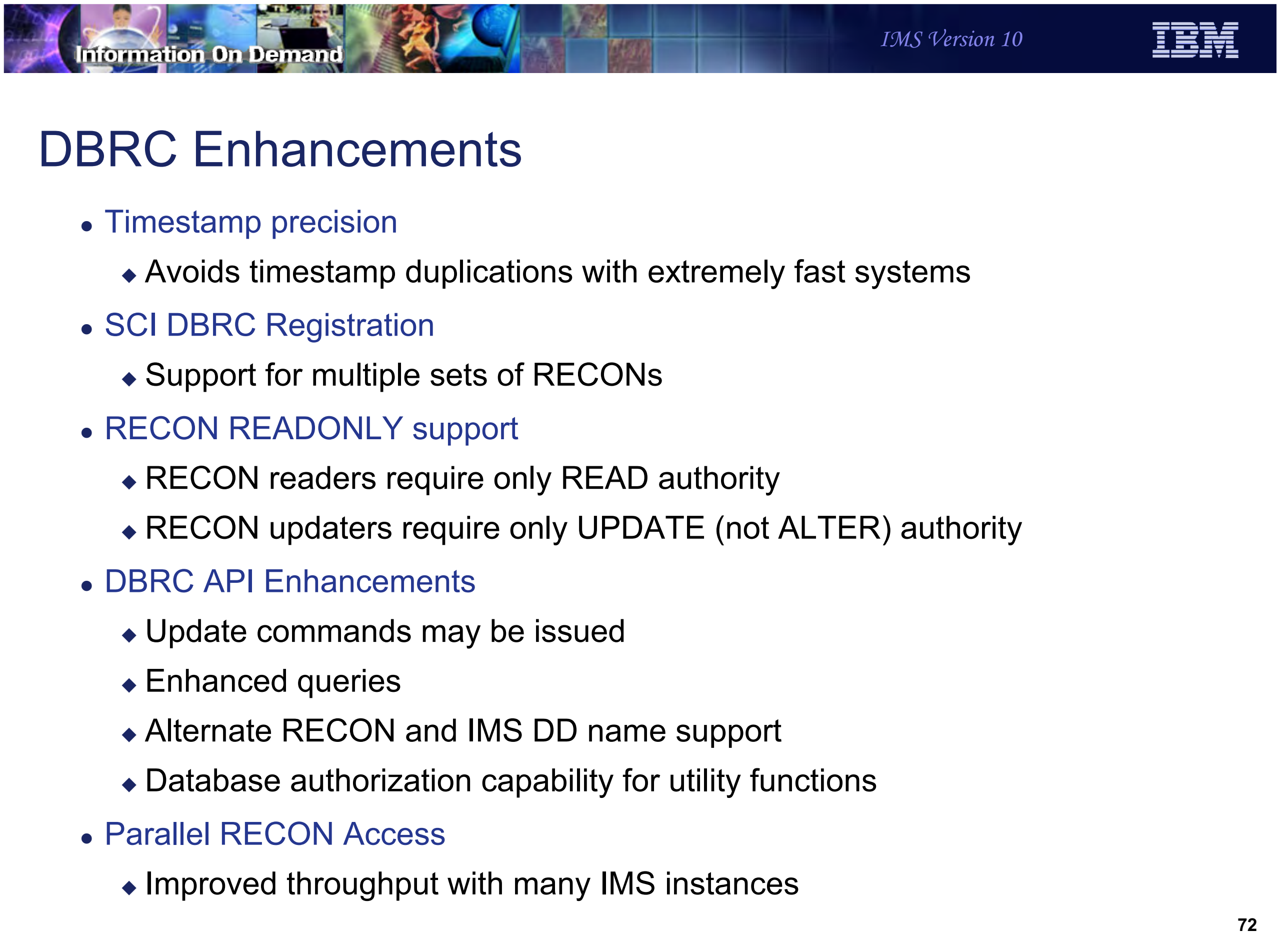
Figure 14-70. Additional Information on DFSMStvs CMA01.0

Notes:

These books provide information on DFSMStvs. Much of the information assumes that DFSMStvs is being used to support concurrent access to VSAM files from CICS and batch programs. When reading these publications keep in mind that DBRC use of DFSMStvs has the following characteristics:

- Forward recovery logs are not used for the RECONs.
- Each DBRC request is a "transaction". Requests include:
  - DBRC commands or subsets of them. Some commands are processed as multiple requests.
  - Subsystem signons and signoffs
  - Database authorizations or unauthorizations
  - Recordings of log
  - Recordings of image copies
  - Recordings of reorganizations

- Recordings of database recoveries
- Creation of allocation records (first update to a data set by a subsystem)
- and many others


Information On Demand
IMS Version 10

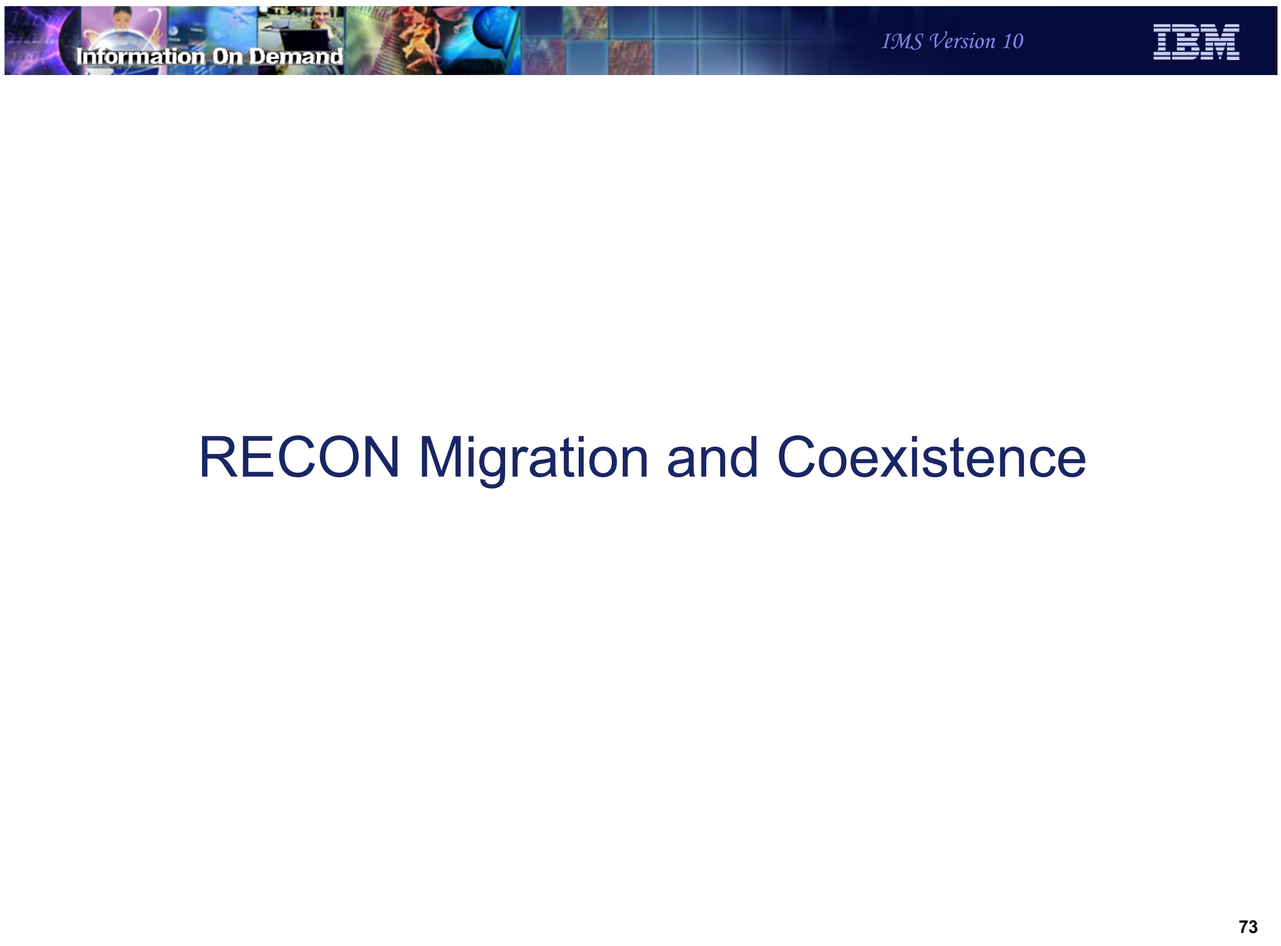
DBRC Enhancements

- Timestamp precision
    - Avoids timestamp duplications with extremely fast systems
- SCI DBRC Registration
    - Support for multiple sets of RECONs
- RECON READONLY support
    - RECON readers require only READ authority
    - RECON updaters require only UPDATE (not ALTER) authority
- DBRC API Enhancements
    - Update commands may be issued
    - Enhanced queries
    - Alternate RECON and IMS DD name support
    - Database authorization capability for utility functions
- Parallel RECON Access
    - Improved throughput with many IMS instances

Figure 14-71. DBRC Enhancements CMA01.0

Notes:

This summarizes the DBRC enhancements in IMS V10.



Information On Demand
IMS Version 10
IBM

RECON Migration and Coexistence

Figure 14-72. [No title] CMA01.0

Notes:

Supported Migrations and Coexistence

- IMS V8 to IMS V10
  - Apply DBRC coexistence SPE APAR PK06145 to IMS V8
  - Upgrade RECONs from IMS V8 to IMS V10

- IMS V9 to IMS V10
  - Apply DBRC coexistence SPE APAR PK06147 to IMS V9
  - Upgrade RECONs from IMS V9 to IMS V10

Figure 14-73. Supported Migrations and Coexistence CMA01.0

Notes:

IMS V8 RECONs may be upgraded directly to IMS V10. Similarly, IMS V9 RECONs may be upgraded to IMS V10. There is no support to upgrade RECONs from previous releases directly to IMS V10.

PK06145 is an IMS V8 SPE (Small Programming Enhancement) APAR. It allows IMS V8 to use RECONs which have been upgraded to IMS V10.

PK06147 is an IMS V9 SPE APAR. It allows IMS V9 to use RECONs which have been upgraded to IMS V10.

These APARs should be applied to IMS V8 or IMS V9 before its RECONs are upgraded to IMS V10.

RECON Upgrade

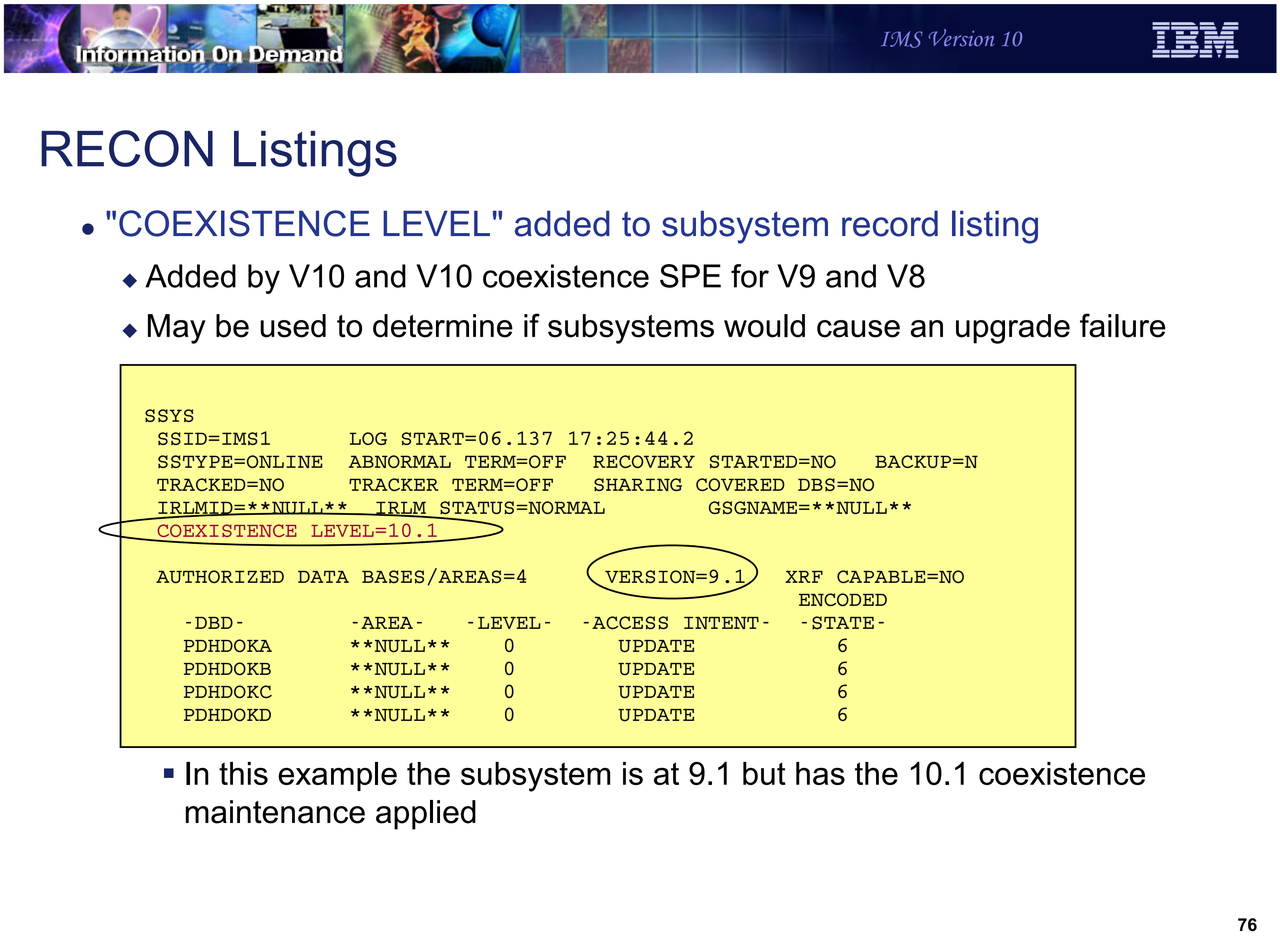
- CHANGE.RECON UPGRADE
  - Using IMS V10 DBRC Utility (DSPURX00)
  - May be executed while subsystems are running
    - Upgrade fails if there is subsystem record for an IMS V8 or V9 subsystem without the DBRC coexistence DBRC SPE
      - Some utilities do not create subsystem records
        - They are not protected by the check for subsystem records
        - If they are running without the SPE, unpredictable results may occur
        - Examples: Change Accumulation, Log Archive, DSPURX00, HALDB Partition Definition Utility, V9 DBRC API applications

Figure 14-74. RECON Upgrade CMA01.0

Notes:

RECONs are upgraded to IMS V10 by using the DBRC CHANGE.RECON UPGRADE command with the IMS V10 DBRC utility (DSPURX00).

The upgrade may be run while the RECONs are allocated to and being used by IMS V8 or IMS V9, Of course, these systems must be able to use IMS V10 RECONs. The upgrade checks the RECONs to ensure that any subsystems using the RECONs are capable of using IMS V10 RECONs. It does this by examining the SUBSYS records in the RECONs. Some IMS utilities do not create SUBSYS records. Thus, the upgrade cannot determine if they are running. Users must ensure that any IMS utility which is running at the time of the upgrade has the appropriate maintenance (PK06145 or PK06147) which allows it to read IMS V10 RECONs.



RECON Listings

- "COEXISTENCE LEVEL" added to subsystem record listing
  - Added by V10 and V10 coexistence SPE for V9 and V8
  - May be used to determine if subsystems would cause an upgrade failure

```
SSYS
 SSID=IMS1      LOG START=06.137 17:25:44.2
 SSTYPE=ONLINE  ABNORMAL TERM=OFF  RECOVERY STARTED=NO   BACKUP=N
 TRACKED=NO     TRACKER TERM=OFF   SHARING COVERED DBS=NO
 IRLMID=**NULL**  IRLM STATUS=NORMAL        GSGNAME=**NULL**
 COEXISTENCE LEVEL=10.1

 AUTHORIZED DATA BASES/AREAS=4      VERSION=9.1   XRF CAPABLE=NO
                                                   ENCODED
   -DBD-        -AREA-   -LEVEL-  -ACCESS INTENT-  -STATE-
   PDHDOKA      **NULL**    0        UPDATE           6
   PDHDOKB      **NULL**    0        UPDATE           6
   PDHDOKC      **NULL**    0        UPDATE           6
   PDHDOKD      **NULL**    0        UPDATE           6
```

    - In this example the subsystem is at 9.1 but has the 10.1 coexistence maintenance applied

Figure 14-75. RECON Listings CMA01.0

Notes:

IMS V10 has added the coexistence level to the RECON listing of subsystem records. This has also been added to IMS V8 and V9 by the IMS V10 coexistence SPEs for these releases. The VERSION= field has existed in previous releases. It indicates the IMS release level of the subsystem. The COEXISTENCE LEVEL= field indicates if the coexistence maintenance for a later release has been applied. In this example, the IMS V10 DBRC coexistence maintenance has been applied to the IMS V9 system used by this subsystem. This listing could have been produced by an IMS V9 DBRC utility with the IMS V10 coexistence SPE applied or it could have been produced by the IMS V10 DBRC utility.

RECON Upgrade

- Upgrade from IMS V8 or V9 to IMS V10
  - Reads SSYS records to check for DBRC SPE
  - Updates RECON header record
    - Sets version indicator, sets MINVERS value, resets CDSLID value, sets ACCESS to SERIAL, and sets LIST to STATIC
  - Updates RECON header extension record
    - Sets version indicator, moves PLEXNAME, and adds GROUPID
  - Updates Image Copy records
    - IC type field expanded and moved
  - Updates Subsystem records
    - API flag added
  - Quick!

Figure 14-76. RECON Upgrade CMA01.0

Notes:

The upgrade of the RECONs includes the reading of the subsystem (SSYS) records to ensure that these subsystems are running with the DBRC coexistence SPE. If not, the subsystem could not use the RECONs and the upgrade fails.

The upgrade changes a few records in the RECONs. The header and header extension records are changed to include the correct version indicator and to values for the parameters shown. The Cross DBRC Service Level ID (CDSLID) is set to the higher of the value in the RECONs before the upgrade and "1". The Image Copy records are rewritten to accommodate the larger IC type field. This field was expanded to include new image copy types for FlashCopy and fuzzy user image copies. The subsystem records are rewritten to accommodate the DBRC API flag for subsystems using the DBRC API.

Since only a few records are updated, a typical upgrade will be quick.

MINVERS

- IMS V10 MINVERS valid values
  - '8.1', '9.1', and '10.1'

- Upgrade of RECONs
  - MINVERS('7.1') changed to MINVERS('8.1')
  - MINVERS('8.1') remains MINVERS('8.1')
  - MINVERS('9.1') remains MINVERS('9.1')

Figure 14-77. MINVERS CMA01.0

Notes:

MINVERS is the parameter on the INIT.RECON and CHANGE.RECON commands which controls the minimum level of IMS which may use the RECONs. The minimum level of IMS which can use V10 RECONs is V8. If the previous MINVERS value was for '7.1', it is changed to '8.1' by the upgrade. Otherwise, upgrades do not change the MINVERS value.

MINVERS values of 81 and 91 (specified without the decimal point or the single quotes) are accepted by IMS V10 for compatibility, however, using the single quotes and the decimal point are required for specifying V10 ('10.1') and recommended for specifying V8 ('8.1') and V9 ('9.1').

IMSplex Name and DBRC Group ID

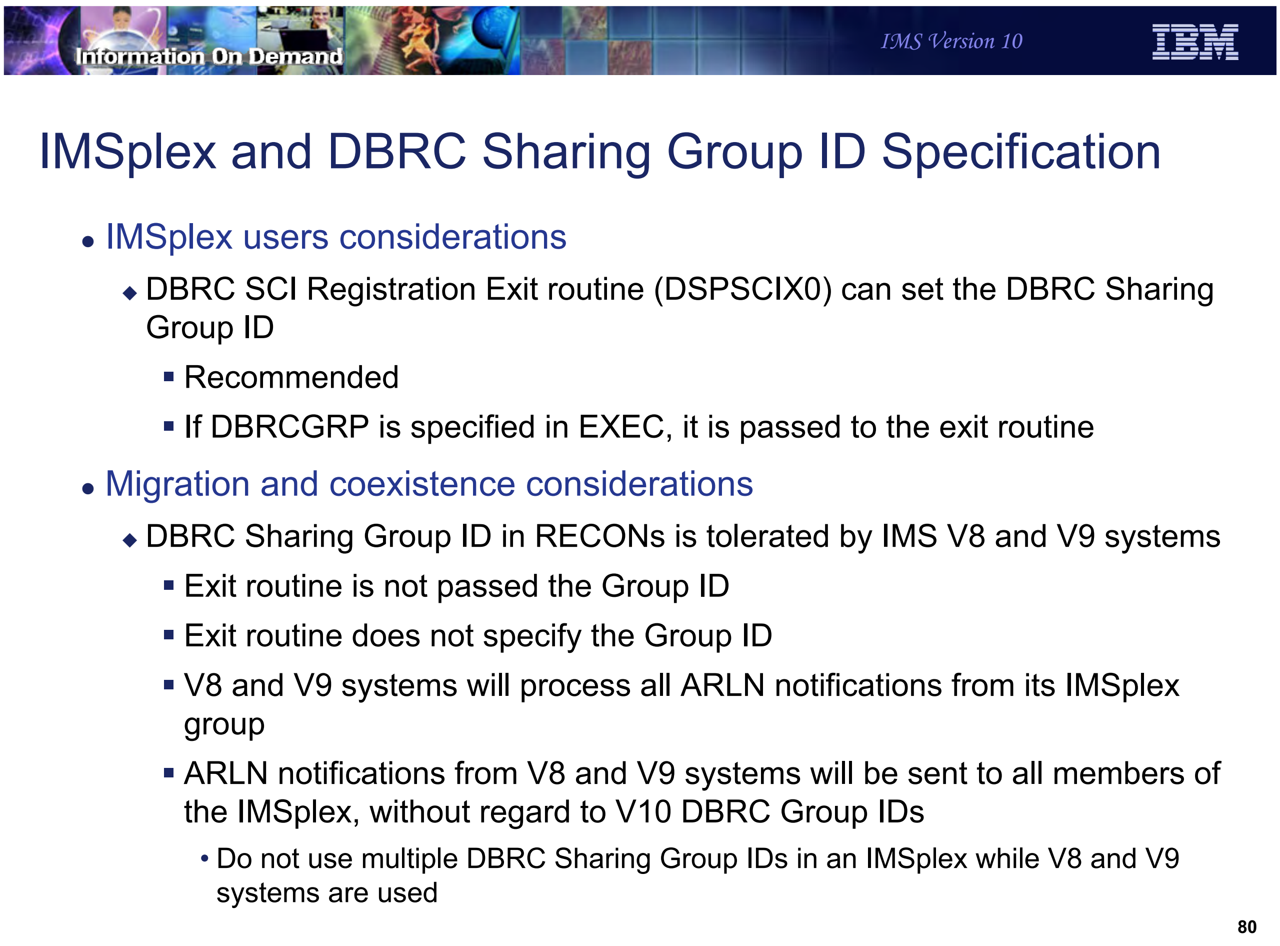
- IMSplex name is specified with 5 characters (xxxxx)
  - IMS V8 and V9 store 'CSLxxxxx' in the RECON header extension record
  - IMS V10 stores 'xxxxxyyy' in the RECON header extension record
    - 'yyy' is the DBRC Group ID
      - DBRC Group ID defaults to '001'
      - Upgrade uses the default
- Example
  - IMS V9: IMSplex name is MYPLX
    - Contents in RECON Header Extension record: 'CSLMYPLX'
    - RECON Listing: `IMSPLEX=MYPLX`
  - IMS V10 after upgrade of RECONs with IMSplex name of MYPLX
    - Contents in RECON Header Extension record: 'MYPLX001'
    - RECON Listing: `IMSPLEX=MYPLX      GROUP ID=001`

Figure 14-78. IMSplex Name and DBRC Group ID CMA01.0

Notes:

The IMSplex name is optional. It is required for Automatic RECON Loss Notification and Parallel RECON Access. The IMSplex name is specified with up to 5 characters. It is specified either in the IMSPLEX= execution parameter or by the DBRC SCI Registration exit routine. When first specified, it is stored in the RECONs. IMS V8 and V9 store the IMSplex name as 'CSLxxxxx' where 'xxxxx' is the value specified in the IMSPLEX= parameter or in the exit routine. When the RECONs are upgraded to V10, the value stored is 'xxxxxyyy' where 'yyy' is the DBRC Group ID. The upgrade sets the DBRC Group ID to '001' which is the default value.

IMS V8, V9, and V10 list only the 5 characters of the IMSplex name in listings of the RECON header. These listing include a line with IMSPLEX=xxxxx when an IMSplex name has been stored in the RECONs. If there is no value stored, the line includes IMSPLEX=**NONE**. IMS V10 listings also include the DBRC Group ID on this line. If there is no IMSplex name the Group ID is listed as GROUP=**NONE**. If there is an IMSplex name, the Group ID is listed as GROUP=yyy where yyy is the Group ID.


Information On Demand
IMS Version 10

IMSplex and DBRC Sharing Group ID Specification

- IMSplex users considerations
  - DBRC SCI Registration Exit routine (DSPSCIX0) can set the DBRC Sharing Group ID
    - Recommended
    - If DBRCGRP is specified in EXEC, it is passed to the exit routine
- Migration and coexistence considerations
  - DBRC Sharing Group ID in RECONs is tolerated by IMS V8 and V9 systems
    - Exit routine is not passed the Group ID
    - Exit routine does not specify the Group ID
    - V8 and V9 systems will process all ARLN notifications from its IMSplex group
    - ARLN notifications from V8 and V9 systems will be sent to all members of the IMSplex, without regard to V10 DBRC Group IDs
      - Do not use multiple DBRC Sharing Group IDs in an IMSplex while V8 and V9 systems are used

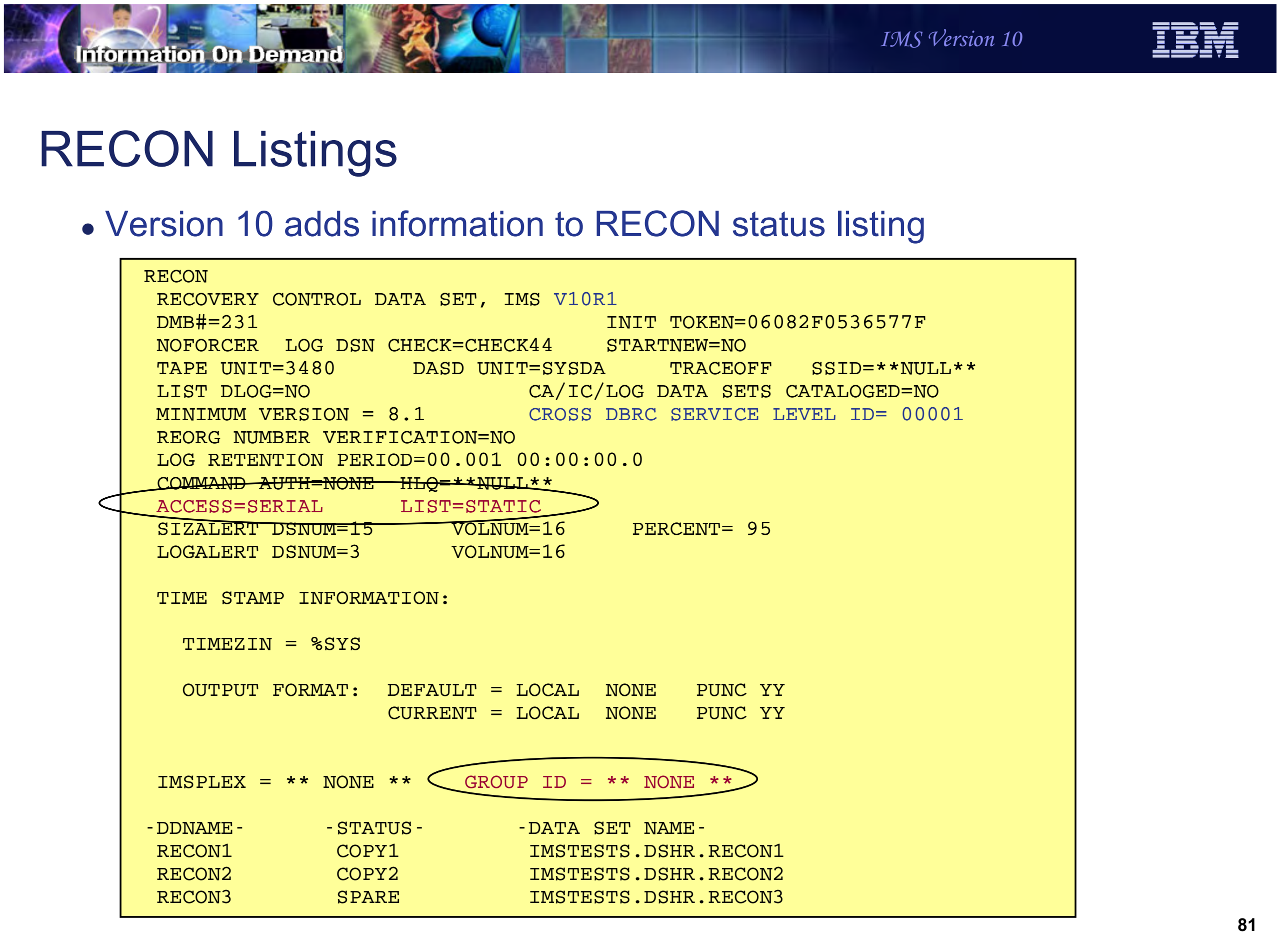
80

Figure 14-79. IMSplex and DBRC Sharing Group ID Specification CMA01.0

Notes:

The DBRC SCI Registration exit routine (DSPSCIX0) may be used to specify the IMSplex name, as in previous releases, and the new DBRC Group ID. The use of the exit routine is recommended for users of IMSplex. It removes the requirement to specify IMSPLEX= for the execution of all IMS jobs which use DBRC. This includes batch jobs and utilities. With IMS V10 the exit also may specify the DBRC Sharing Group ID. This removes the requirement to specify DBRCGRP= for IMS executions.

IMS V8 and V9 systems can tolerate the specification of the DBRC Group ID in the RECONs. DBRCGRP= is not a valid parameter on the EXEC statement for V8 and V9. When the exit routine is invoked in a V8 or V9 environment, the DBRC Group ID is not passed to it. The exit routine cannot specify the DBRC Group ID. Even though a V8 or V9 instance cannot specify the DBRC Group ID, it can join an IMSplex where V10 instances are using DBRC Group IDs. The V8 or V9 instance will be passed all ARLN notifications from the IMSplex group. If a V8 or V9 system reconfigures its RECONs, its ARLN notification will be processed by all members of the IMSplex. This will include all V10 systems. If there are multiple DBRC Groups, all members of all groups will process the notification. For these reasons, you should not use multiple DBRC Group IDs in an IMSplex while you are still using IMS V8 or V9 systems.



Information On Demand
IMS Version 10
IBM

RECON Listings

- Version 10 adds information to RECON status listing

```
RECON
 RECOVERY CONTROL DATA SET, IMS V10R1
 DMB#=231                           INIT TOKEN=06082F0536577F
 NOFORCER  LOG DSN CHECK=CHECK44    STARTNEW=NO
 TAPE UNIT=3480      DASD UNIT=SYSDA     TRACEOFF   SSID=**NULL**
 LIST DLOG=NO                 CA/IC/LOG DATA SETS CATALOGED=NO
 MINIMUM VERSION = 8.1        CROSS DBRC SERVICE LEVEL ID= 00001
 REORG NUMBER VERIFICATION=NO
 LOG RETENTION PERIOD=00.001 00:00:00.0
 COMMAND AUTH=NONE  HLQ=**NULL**
 ACCESS=SERIAL      LIST=STATIC
 SIZALERT DSNUM=15      VOLNUM=16     PERCENT= 95
 LOGALERT DSNUM=3       VOLNUM=16

 TIME STAMP INFORMATION:

   TIMEZIN = %SYS

   OUTPUT FORMAT:  DEFAULT = LOCAL  NONE   PUNC YY
                   CURRENT = LOCAL  NONE   PUNC YY


 IMSPLEX = ** NONE **    GROUP ID = ** NONE **

-DDNAME-      -STATUS-       -DATA SET NAME-
 RECON1        COPY1          IMSTESTS.DSHR.RECON1
 RECON2        COPY2          IMSTESTS.DSHR.RECON2
 RECON3        SPARE          IMSTESTS.DSHR.RECON3
```

Figure 14-80. RECON Listings CMA01.0

Notes:

The RECON status or header listing has some added and changed information.

Of course, the IMS version is listed as "V10R1". This means that the RECONs have been upgraded to IMS V10.

There is a new line which lists the type of RECON access, either SERIAL or PARALLEL. On the same line the default for the DBRC LIST command, either STATIC or CONCURR, is shown.

On the line where the IMSPLEX value is shown, the DBRC Group ID value is also shown. In this example, these parameters have no values so "** NONE **" is listed.

The sample listing shown here includes the "CROSS DBRC SERVICE LEVEL ID". This also appears on IMS V9 RECON listings when the maintenance for APARs PQ98655 and PK01097 is applied and on IMS V8 RECON listings when the maintenance for APARs PQ98654 and PK01096 is applied. The service level ID is used to invoke functions which require a consistent level of maintenance on all IMS systems using the RECONs.

Information On Demand
IMS Version 10
IBM

DBRC Migration Steps

1. Install IMS V8 or V9 DBRC Migration/Coexistence SPEs
2. Install IMS V10 DBRC Type 4 SVC
    - The V10 Type 4 SVC may be used with V8 and V9
3. Upgrade RECONs using the IMS V10 SDFSRESL library
4. Begin using IMS V10

5. Discontinue all use of IMS V8 and V9
6. CHANGE.RECON MINVERS('10.1')
    - Full precision will be used in timestamps

Figure 14-81. DBRC Migration Steps CMA01.0

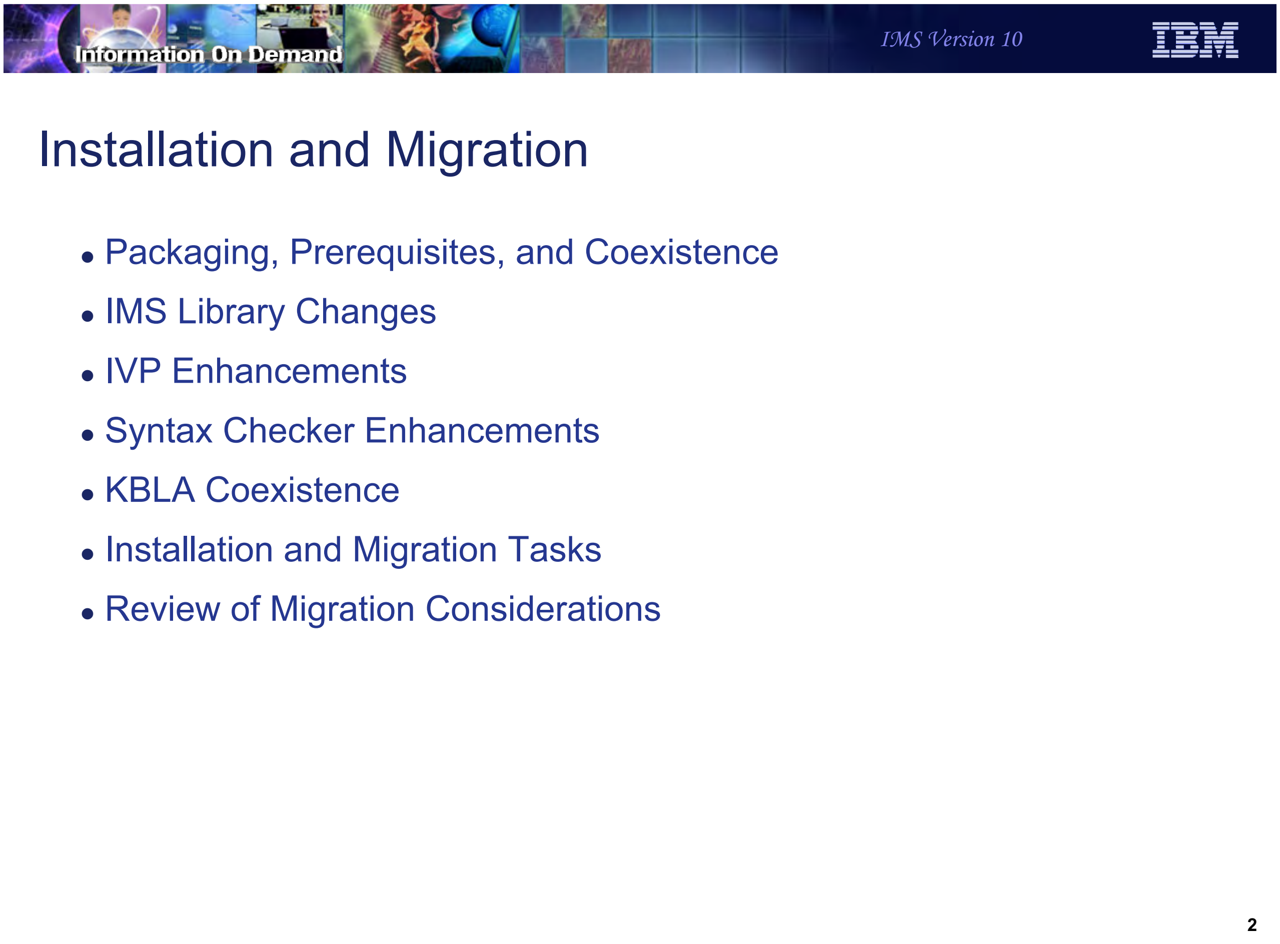
Notes:

This shows the DBRC steps for migration to IMS V10.

The first set of steps allows you to begin using IMS V10. The migration/coexistence SPE must be installed on the old release before you upgrade the RECONs to V10. The V10 DBRC Type 4 SVC must be installed before you may use IMS V10. The upgrade of the RECONs to IMS V10 requires that you use the SDFSRESL library created by the installation of IMS V10. The upgrade using this library will be to the IMS V10 format. Once the RECONs have been upgraded, you may begin using IMS V10. You may also continue to use IMS V8 or V9.

Once you have discontinued all use of IMS V8 and V9, you can change the MINVERS value to '10.1'. This causes IMS to begin using the increased precision timestamp. Before changing MINVERS to '10.1', you must ensure that the IMS utility control statements that you use specify full precision in their timestamps. The control statements generated by GENJCL statements will always generate control statements with the correct timestamps. Remember that the position of the timestamp in the control statements for the IMS V10 Change Accumulation and Database Recovery utilities does not depend on the MINVERS value, however, if MINVERS is not '10.1' the low order part of the timestamp does not matter since these positions in timestamps are not recorded in the RECONs unless MINVERS('10.1') is specified.

Unit 15.Installation and Migration



Information On Demand
IMS Version 10
IBM

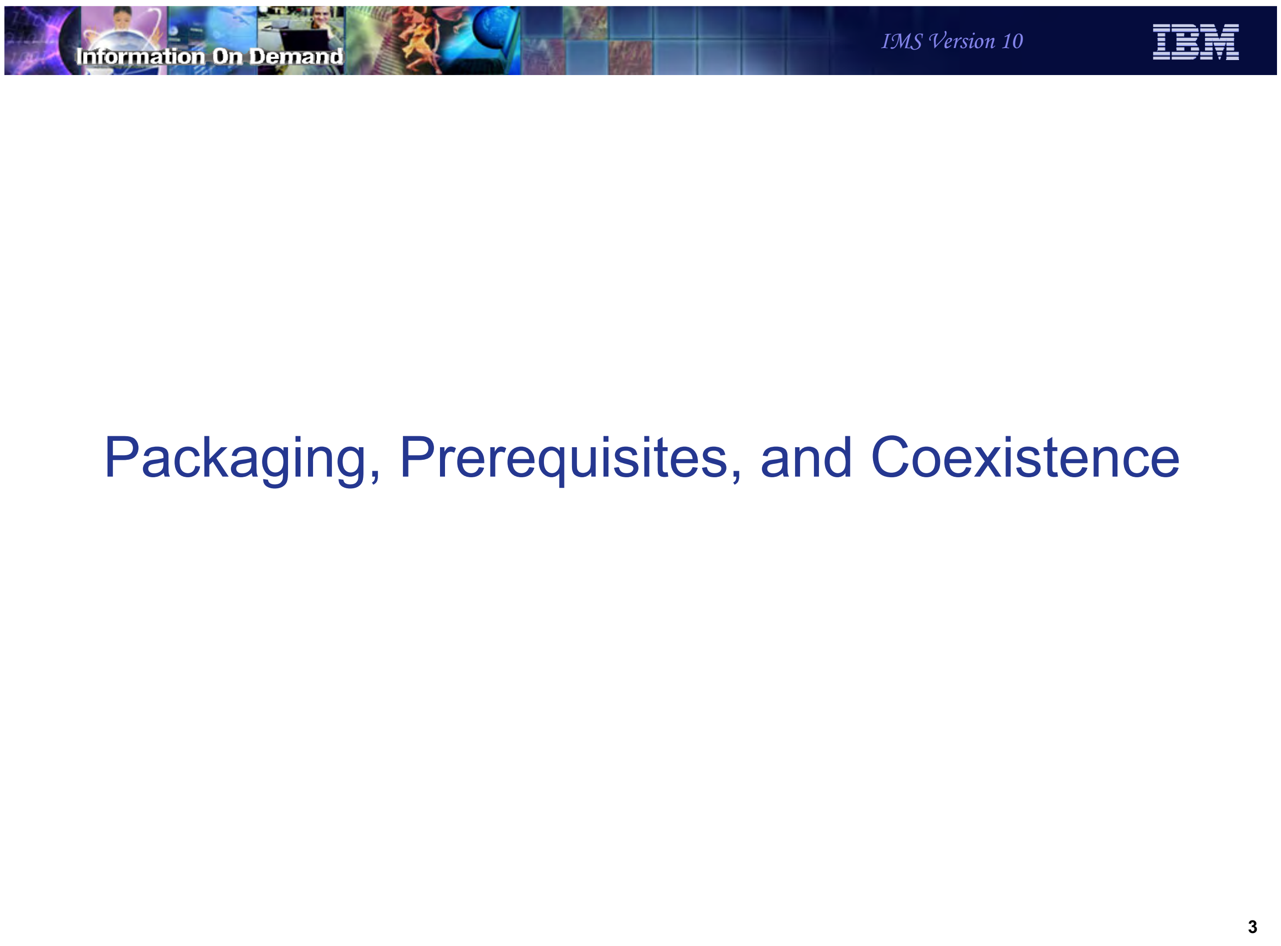
Installation and Migration

- Packaging, Prerequisites, and Coexistence
- IMS Library Changes
- IVP Enhancements
- Syntax Checker Enhancements
- KBLA Coexistence
- Installation and Migration Tasks
- Review of Migration Considerations

2

Figure 15-1. Installation and Migration CMA01.0

Notes:



Information On Demand

IMS Version 10

Packaging, Prerequisites, and Coexistence

Figure 15-2. [No title] CMA01.0

Notes:

Packaging

- IMS V10 product number: 5635-A01

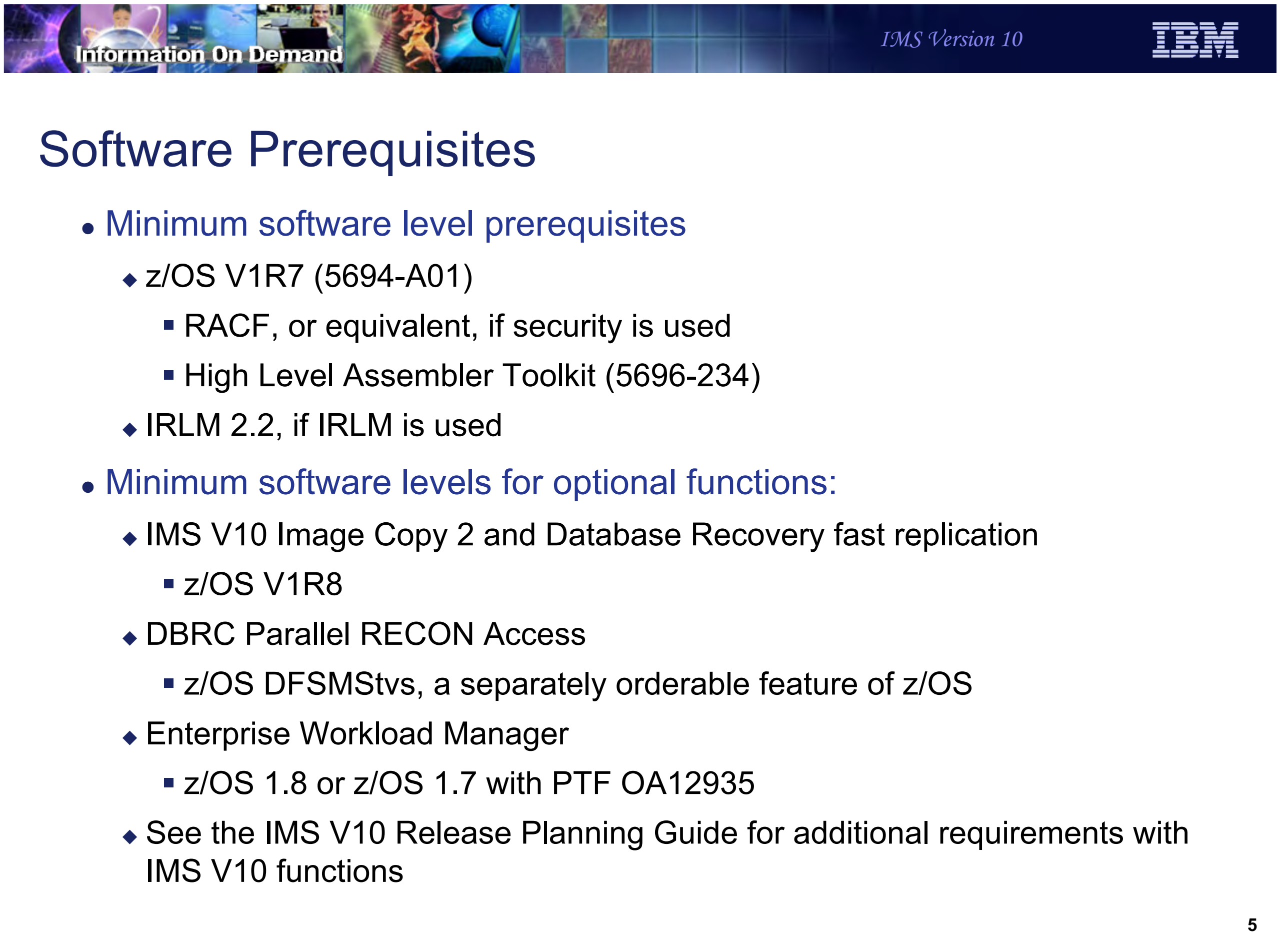
FMID	Feature Description
HMK1010	System Services
JMK1011	Database Manager
JMK1012	Transaction Manager
JMK1013	Extended Terminal Option (ETO)
JMK1014	Recovery Level Tracking (RSR)
JMK1015	Database Level Tracking (RSR)
JMK1016	IMS Java
HIR2220	IRLM 2.2

Figure 15-3. Packaging CMA01.0

Notes:

IMS V10 packaging is the same as that for IMS V9. Transaction Manager is a prerequisite for ETO. Recovery Level Tracking RSR is a prerequisite for Database Level Tracking RSR.

IRLM 2.2 is the only IRLM shipped and supported with IMS V10.


Information On Demand
IMS Version 10
IBM

Software Prerequisites

- Minimum software level prerequisites
    - z/OS V1R7 (5694-A01)
        - RACF, or equivalent, if security is used
        - High Level Assembler Toolkit (5696-234)
    - IRLM 2.2, if IRLM is used
- Minimum software levels for optional functions:
    - IMS V10 Image Copy 2 and Database Recovery fast replication
        - z/OS V1R8
    - DBRC Parallel RECON Access
        - z/OS DFSMStvs, a separately orderable feature of z/OS
    - Enterprise Workload Manager
        - z/OS 1.8 or z/OS 1.7 with PTF OA12935
    - See the IMS V10 Release Planning Guide for additional requirements with IMS V10 functions

Figure 15-4. Software Prerequisites CMA01.0

Notes:

The minimum level of z/OS for IMS V10 is z/OS V1R7. In addition to z/OS the user must install RACF, or an equivalent security product, in order to use security with IMS V10. As with previous IMS releases, the High Level Assembler Toolkit is required to provide assembler macros that IMS uses. If the IRLM is used, IRLM 2.2 is required. Program Isolation (PI) is also supported with IMS V10. IRLM is required for block level data sharing.

z/OS V1.7 runs on the following IBM System z9 and zSeries servers or equivalents: z9-109, z900, z990 , z800, and z890 .

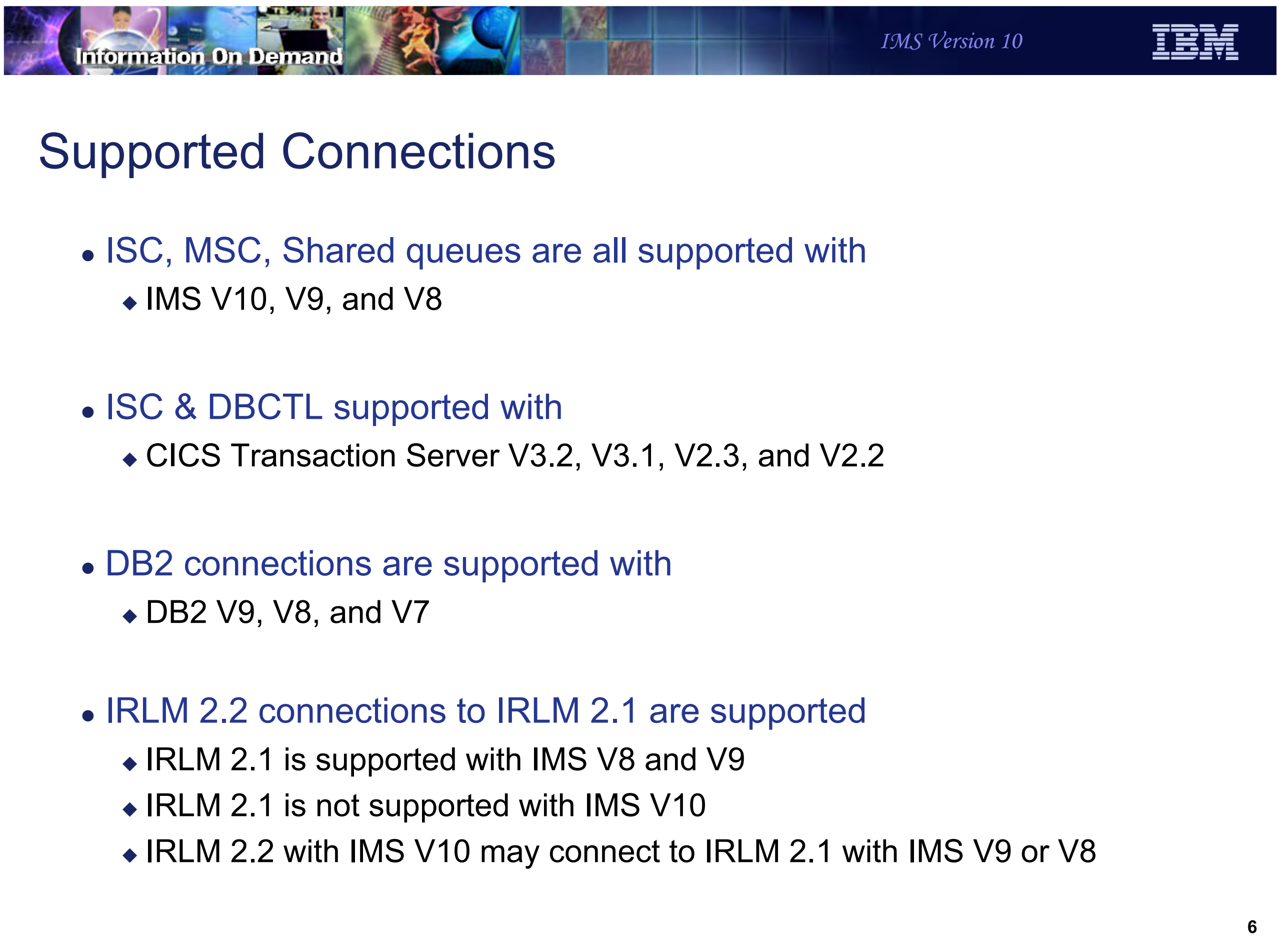
z/OS V1R8 is required if the fast replication function of Image Copy 2 and the Database Recovery utility are used.

DBRC Parallel RECON Access requires z/OS DFSMStvs which is a separately orderable feature of z/OS V1R7 or z/OS V1R8. Special bids will be considered for IMS customers using the Parallel RECON Access function, who do not already have DFSMStvs, to acquire DFSMStvs for use restricted to IMS.

DFSMS APAR OA11468/PTF UA18949 is a prerequisite for Fast Path customers running at z/OS 1.7 or above.

EWLM support is available in z/OS 1.8 or via APAR OA12935 for z/OS 1.7. For both releases of z/OS the EWLM, Virtualization Engine, and Java SDK FMIDs must be installed in the z/OS system.

The IMS V10 Release Planning Guide has additional information about requirements when using particular functions in IMS V10.


Information On Demand
IMS Version 10

Supported Connections

- ISC, MSC, Shared queues are all supported with
  - IMS V10, V9, and V8
- ISC & DBCTL supported with
  - CICS Transaction Server V3.2, V3.1, V2.3, and V2.2
- DB2 connections are supported with
  - DB2 V9, V8, and V7
- IRLM 2.2 connections to IRLM 2.1 are supported
  - IRLM 2.1 is supported with IMS V8 and V9
  - IRLM 2.1 is not supported with IMS V10
  - IRLM 2.2 with IMS V10 may connect to IRLM 2.1 with IMS V9 or V8

Figure 15-5. Supported Connections CMA01.0

Notes:

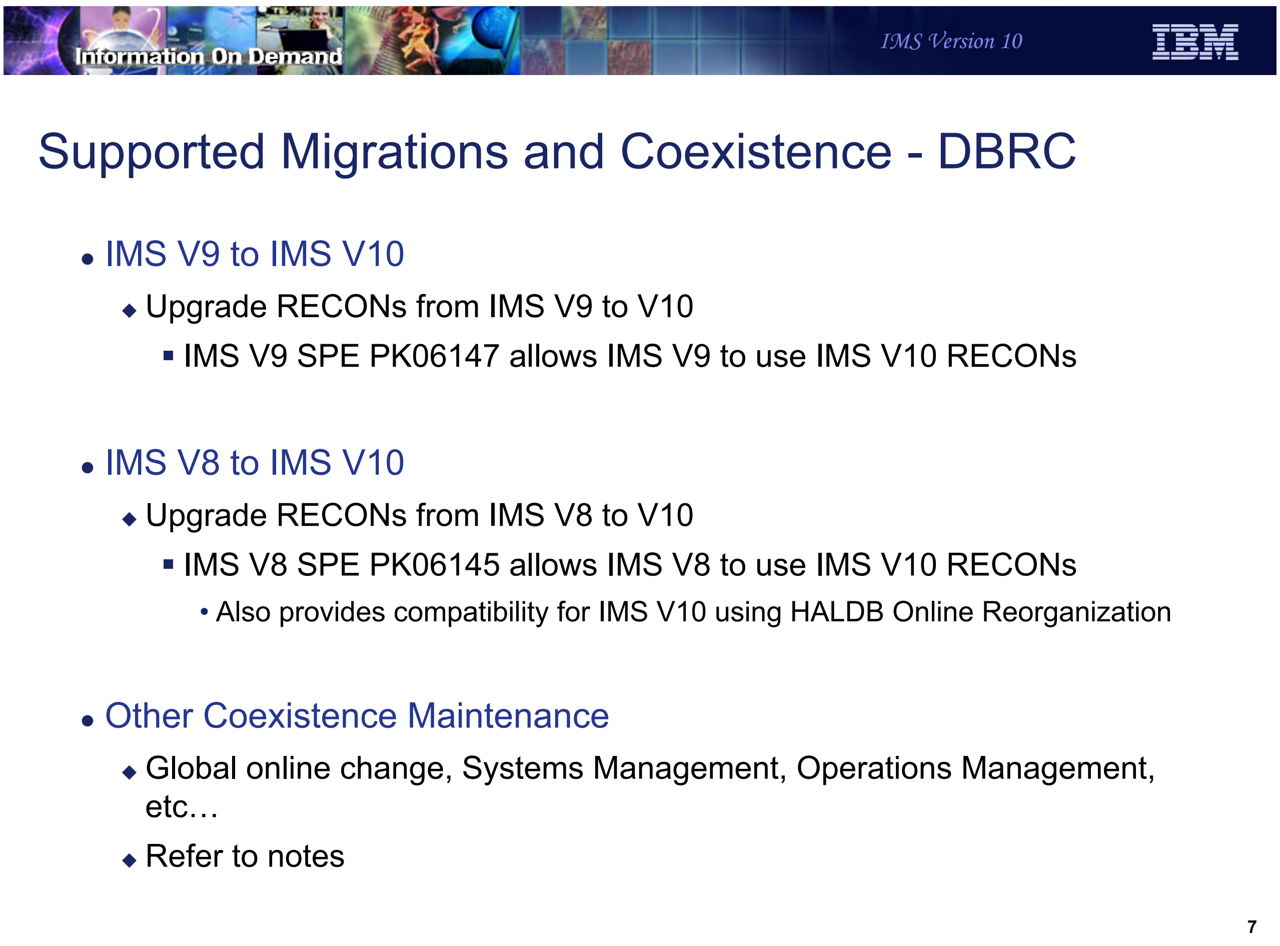
All currently supported releases of IMS and CICS are supported for ISC connectivity to IMS V10.

All currently supported releases of IMS are supported for MSC connectivity to IMS V10.

All currently supported releases of IMS are supported for shared queues with IMS V10 Transaction Manager.

All currently supported releases of DB2 on z/OS are supported for connections from IMS V10.

All currently supported releases of CICS are supported for DBCTL connectivity to IMS V10.



Information On Demand
IMS Version 10
IBM

Supported Migrations and Coexistence - DBRC

- IMS V9 to IMS V10
  - Upgrade RECONs from IMS V9 to V10
    - IMS V9 SPE PK06147 allows IMS V9 to use IMS V10 RECONs
- IMS V8 to IMS V10
  - Upgrade RECONs from IMS V8 to V10
    - IMS V8 SPE PK06145 allows IMS V8 to use IMS V10 RECONs
      - Also provides compatibility for IMS V10 using HALDB Online Reorganization
- Other Coexistence Maintenance
  - Global online change, Systems Management, Operations Management, etc…
  - Refer to notes

Figure 15-6. Supported Migrations and Coexistence - DBRC CMA01.0

Notes:

IMS V8 and V9 RECONs may be upgraded to IMS V10 by executing the DBRC utility (DSPURX00) and using the CHANGE.RECON UPGRADE command with an IMS V10 SDFSRESL library. Before doing the upgrade you should apply the Small Programming Enhancement to your IMS V8 or V9 system. This allows the V8 or V9 systems to use the RECONs after they have been upgraded to V10.

SPE PK06145 for IMS V8 also supplies compatibility support that allows IMS V10 to invoke HALDB Online Reorganizations (OLR) for partitions that are accessed by IMS V8. IMS V8 cannot invoke HALDB OLR, but when this maintenance is applied it may access partitions for which OLR is used.

Other Coexistence Maintenance:

- Global Online Change Coexistence APARs

  V8 - PK23401; V9 - PK23402

The Global Online Change coexistence SPEs allow lower level IMS systems to share an OLCSTAT data set with IMS V10. IMS V10 makes changes to the header record. The SPEs allow IMS V8 and V9 to implement and tolerate these changes.

- System Management Enhancement Coexistence SPEs

  V8 - PK30188; V9 - PK30189

  The System Management Enhancement coexistence SPEs allow lower level IMS systems to coexist with IMS V10. Lower level IMS systems which process transaction submitted from the OM API will receive an AD status code if they reply to the IOPCB.

- Resource Consistency Checking Coexistence SPEs

  V8 - PK32969; V9 - PK32970

  Resource Consistency Checking is not done in IMS V10

  IMS V10 does not support the resource consistency checking function of the Resource Manager. The Resource Consistency Checking SPEs allow lower level IMS systems to use this function among themselves while in an IMSplex with IMS V10 systems. The elimination of resource consistency checking is explained later in this section of the class.

- Operations Management Coexistence SPEs

  V8 - PK27279; V9 - PK27280

  The Operations Management coexistence SPEs allow OM address spaces and IMSs at lower level of IMS to coexist in a CSL environment with IMS V10 OM.

Coexistence - IMS Utilities

- Batch Backout, Log Archive, and Log Recovery
  - Use the utility from the IMS release that created the log
- IMS V10 Database Recovery utility
  - Accepts Image Copies produced by IMS V10, V9, and V8
  - Accepts HISAM Unloads produced by IMS V10, V9, and V8
  - Accepts Change Accum data sets produced by IMS V10, V9, and V8
  - Accepts logs produced by IMS V10, V9, and V8
- IMS V10 Change Accumulation utility
  - Accepts logs produced by IMS V10, V9, and V8
  - Accepts Change Accum data sets produced by IMS V10, V9, and V8

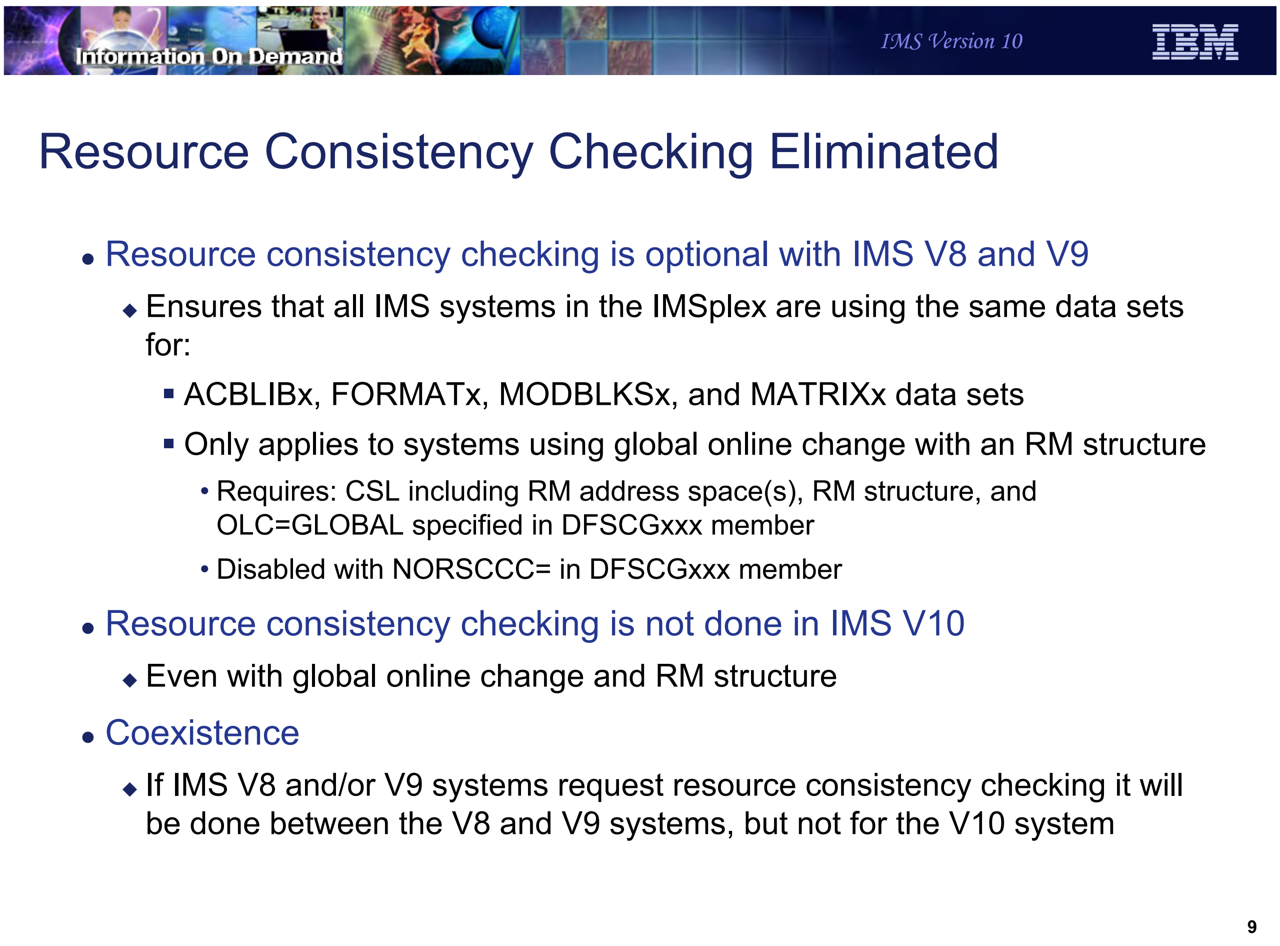
Figure 15-7. Coexistence - IMS Utilities CMA01.0

Notes:

The Batch Backout, Log Archive, and Log Recovery utilities access one log. The release level of the utility must match the IMS release that was used to create the log.

The IMS V10 Database Recovery utility (DFSURDB0) accepts Image Copies, HISAM Unload data sets, Change Accumulation data sets, and IMS logs as inputs. These inputs may be created by any currently supported release of IMS.

The Change Accumulation utility accepts IMS logs and Change Accumulation data sets as inputs. These inputs may be created by any currently supported release of IMS.



Information On Demand
IMS Version 10
IBM

Resource Consistency Checking Eliminated

- Resource consistency checking is optional with IMS V8 and V9
  - Ensures that all IMS systems in the IMSplex are using the same data sets for:
    - ACBLIBx, FORMATx, MODBLKSx, and MATRIXx data sets
    - Only applies to systems using global online change with an RM structure
      - Requires: CSL including RM address space(s), RM structure, and OLC=GLOBAL specified in DFSCGxxx member
      - Disabled with NORSCCC= in DFSCGxxx member
- Resource consistency checking is not done in IMS V10
  - Even with global online change and RM structure
- Coexistence
  - If IMS V8 and/or V9 systems request resource consistency checking it will be done between the V8 and V9 systems, but not for the V10 system

9

Figure 15-8. Resource Consistency Checking Eliminated CMA01.0

Notes:

Resource consistency checking for global online change is not done by IMS V10. This function was available in IMS V8 and V9. It checks to ensure that the data set names used for the active ACBLIB, FORMAT, MODBLKS, and MATRIX data sets are the same for all of the IMS systems. It is only done when global online change is be used (OLC=GLOBAL is specified in the DFSCGxxx member) and an RM structure is used. It can be disabled in V8 and V9 by specifying values for the NORSCCC= parameter in the DFSCGxxx member. Users were likely to disable resource consistency checking since the use of resource consistency checking created a single point of failure. For example, the loss of the ACBLIB data set would effect all systems since all would be using the same data set.

IMS V10 does not do resource consistency checking. The NORSCCC= parameter in the DFSCGxxx member may be specified for compatibility, but it is ignored by V10.

If your IMSplex has a mixture of IMS V10 and V9 or V8 systems, resource consistency checking will apply only to the V8 and V9 systems. When resource consistency checking is used, only the V8 and V9 systems must use the same data set(s). The Resource Consistency Checking SPEs, PK32969 for V8 and PK32970 for V9, must be applied for the

IMS V8 or V9 systems to use resource consistency checking when an IMS V10 system is part of the IMSplex.

Remote Site Recovery (RSR) Migration/Coexistence

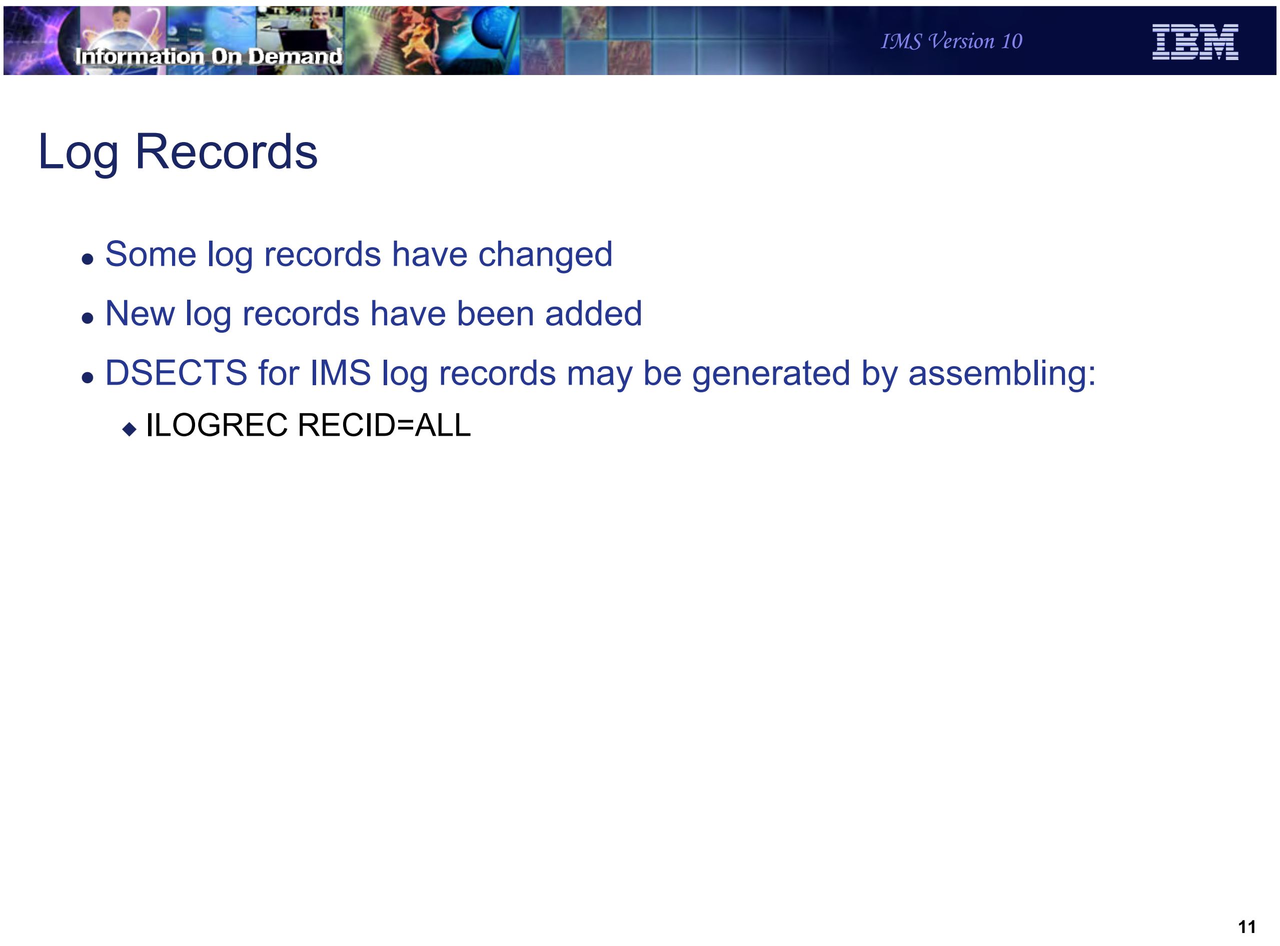
- V10 RSR tracking system can process logs created by V10. V9, or V8
- V10 RSR Isolated Log Sender can send logs created by V10, V9, or V8
- Logs created by V10 cannot be processed by V8 or V9 tracking system or Isolated Log Sender

- Migration steps
  - Upgrade the RSR tracking system RECONs to V10
  - Migrate RSR tracking system to V10
  - Upgrade the active system RECONs to V10
  - Migrate active system Transport Manager Subsystem (TMS) running Isolated Log Sender to V10
  - Migrate active IMS to V10

Figure 15-9. Remote Site Recovery (RSR) Migration/Coexistence CMA01.0

Notes:

The migration of systems using RSR is similar to migrations for previous releases. IMS V10 tracking systems can process logs produced by lower releases. The IMS V10 Isolated Log Sender (ILS) function of the Transport Manager System (TMS) can process logs created by lower releases. On the other hand, IMS V8 and V9 tracking systems cannot accept logs produced by IMS V10 and the IMS V8 and V9 ILSs cannot accept logs produced by IMS V10. Of course, you could migrate all of the RSR components at the same time. You would more likely prefer to migrate them in stages. The restrictions mentioned above imply that the order of migration of the components is as shown on the slide. The tracking system must be migrated before or at the same time as the ILS at the active site. The ILS at the active site must be migrated before or at the same time as the active IMS system. The RECONs must be upgraded to IMS V10 before the systems that use them are migrated to IMS V10.


Information On Demand
IMS Version 10
IBM

Log Records

- Some log records have changed
- New log records have been added
- DSECTS for IMS log records may be generated by assembling:
  - ILOGREC RECID=ALL

Figure 15-10. Log Records CMA01.0

Notes:

Among the log records which have been changed for IMS V10 are the following types:

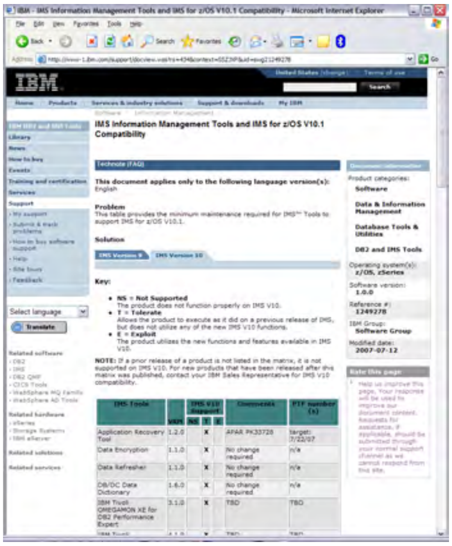
> 07, 08, 22, 4001, 4004, 4006, 4007, 4012, 4083, 45, 56FA, 5950, 5951, 5953, 5954, 5955, 5956, 5957, 5958, 56FA, and 7205.

This is NOT an exhaustive list of changed log records.

If you have application programs which process IMS log records, you should examine to see if they are affected by the changes to the log records. You can assemble DSECTs for IMS log records by using the ILOGREC macro.

IMS Tools Migration/Coexistence

- Some products may require updates
  - Contact your vendor for information on requirements
  - IBM has a web site with consolidated information about requirements for IBM IMS tools with IMS V10, refer to:
    - www.ibm.com//software/data/db2imstools/
    - Click on "Support"
    - Click on "IMS Tools and IMS Compatibility"
    - Click on "IMS Version 10“
- IMS APARs for fallback from IMS V10 to IMS V8 or V9 using QCF
  - Required so that IMS V9 and V8 can recognize V10 messages
    - PK29667 (V9) and PK29666 (V8)
      - Not required for migration from V8 or V9 to V10


IBM - IMS Information Management Tools and IMS for z/OS V10.1 Compatibility - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address Go
IBM
Search
Home Products Services & industry solutions Support & downloads My IBM
Library
News
How to buy
Events
Training and certification
Services
Support
Select language
Translate
Related software
Related hardware
Related solutions
Related services
Rate this page

IMS Information Management Tools and IMS for z/OS V10.1 Compatibility

Technote (FAQ)

This document applies only to the following language version(s):
English

Problem
This table provides the minimum maintenance required for IMS™ Tools to support IMS for z/OS V10.1.

Solution

IMS Version 9 | IMS Version 10

Key:

- **NS = Not Supported**
  The product does not function properly on IMS V10.
- **T = Tolerate**
  Allows the product to execute as it did on a previous release of IMS, but does not utilize any of the new IMS V10 functions.
- **E = Exploit**
  The product utilizes the new functions and features available in IMS V10.

NOTE: If a prior release of a product is not listed in the matrix, it is not supported on IMS V10. For new products that have been released after this matrix was published, contact your IBM Sales Representative for IMS V10 compatibility.

IMS Tools	VRM	IMS V10 Support			Comments	PTF number (s)
		NS	T	E		
Application Recovery Tool	1.2.0		X		APAR PK33728	target: 7/22/07
Data Encryption	1.1.0		X		No change required	n/a
Data Refresher	1.1.0		X		No change required	n/a
DB/DC Data Dictionary	1.6.0		X		No change required	n/a
IBM Tivoli OMEGAMON XE for DB2 Performance Expert	3.1.0		X		TBD	TBD

Document information
Product categories:
Software
Data & Information Management
Database Tools & Utilities
DB2 and IMS Tools
Operating system(s):
z/OS, zSeries
Software version:
1.0.0
Reference #:
1249278
IBM Group:
Software Group
Modified date:
2007-07-12

Figure 15-11. IMS Tools Migration/Coexistence CMA01.0

Notes:

There are APARs for IMS V9 and V8 to support fallback from IMS V10 to these releases. The fixes for these APARs are needed if messages created under IMS V10 need to be requeued to an IMS V9 or V8 system after a fallback to these releases. These fixes are not required to take messages from an IMS V8 or V9 system and requeue them on an IMS V10 system.

QCF 2.1 requires the fix for APAR P28050 for IMS V10 support.

There are APARs for IMS V9 and V8 to support fallback from IMS V10 using the IMS Queue Control Facility (QCF). The fixes for these APARs are needed if messages created under IMS V10 need to be requeued to an IMS V9 or V8 system after a fallback to these releases. These fixes are not required to take messages from an IMS V8 or V9 system and requeue them on an IMS V10 system.

Information On Demand
IMS Version 10

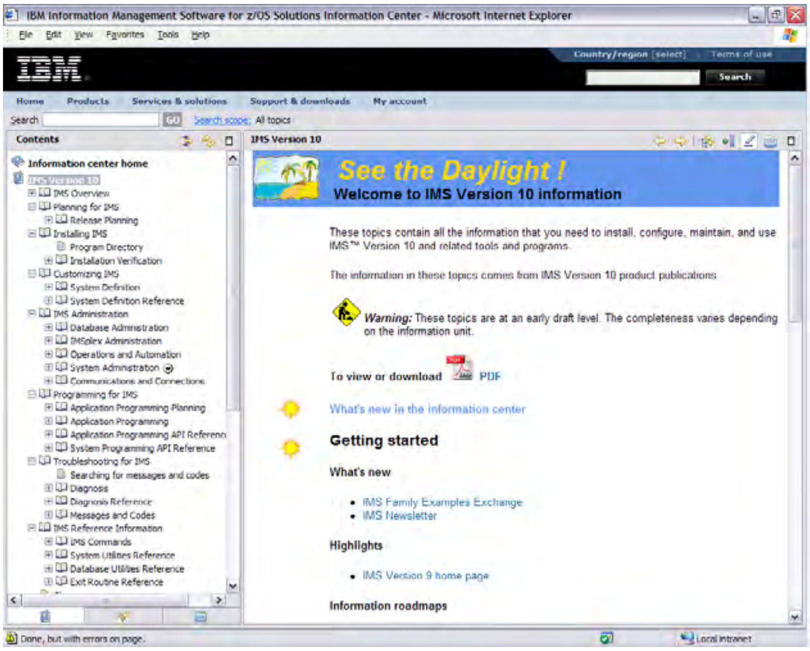
IMS Library Reorganization

Figure 15-12. [No title] CMA01.0

Notes:

IMS Library

- Information Center contains information on IMS V10


IBM Information Management Software for z/OS Solutions Information Center - Microsoft Internet Explorer
File Edit View Favorites Tools Help
IBM
Country/region [select] | Terms of use
Search
Home Products Services & solutions Support & downloads My account
Search GO Search scope: All topics
Contents
IMS Version 10
Done, but with errors on page.
Local intranet

Information center home
IMS Version 10
- IMS Overview
- Planning for IMS
  - Release Planning
- Installing IMS
  - Program Directory
  - Installation Verification
- Customizing IMS
  - System Definition
  - System Definition Reference
- IMS Administration
  - Database Administration
  - IMSplex Administration
  - Operations and Automation
  - System Administration
  - Communications and Connections
- Programming for IMS
  - Application Programming Planning
  - Application Programming
  - Application Programming API Referenc
  - System Programming API Reference
- Troubleshooting for IMS
  - Searching for messages and codes
  - Diagnosis
  - Diagnosis Reference
  - Messages and Codes
- IMS Reference Information
  - IMS Commands
  - System Utilities Reference
  - Database Utilities Reference
  - Exit Routine Reference

See the Daylight !

Welcome to IMS Version 10 information

These topics contain all the information that you need to install, configure, maintain, and use IMS™ Version 10 and related tools and programs.

The information in these topics comes from IMS Version 10 product publications.

Warning: These topics are at an early draft level. The completeness varies depending on the information unit.

To view or download PDF

What's new in the information center

Getting started

What's new

- IMS Family Examples Exchange
- IMS Newsletter

Highlights

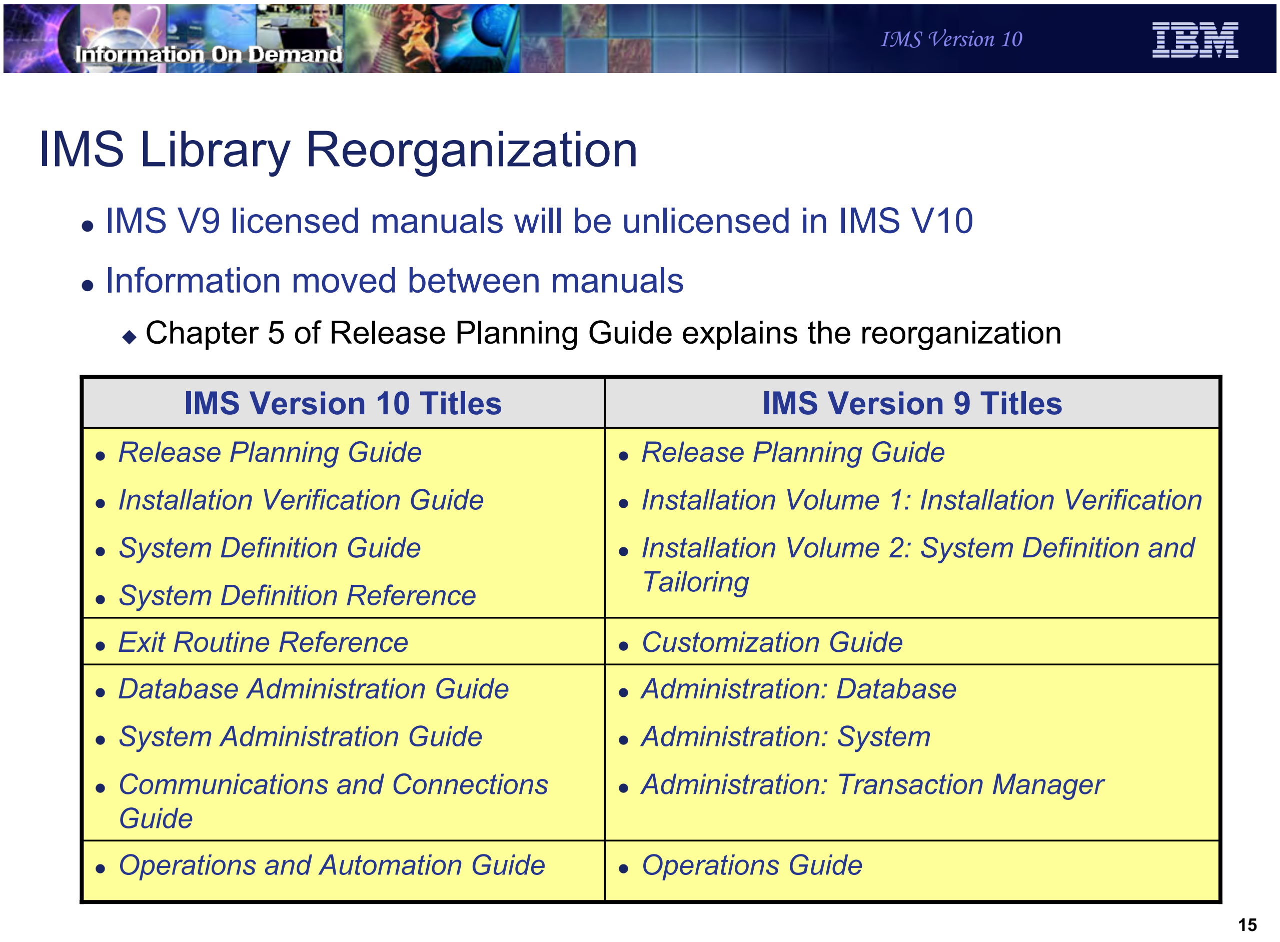
- IMS Version 9 home page

Information roadmaps

Figure 15-13. IMS Library CMA01.0

Notes:

The Information Center has been updated to include information on IMS V10. The topics on the left side of the screen map to the titles of the IMS V10 manuals.


Information On Demand
IMS Version 10
IBM

IMS Library Reorganization

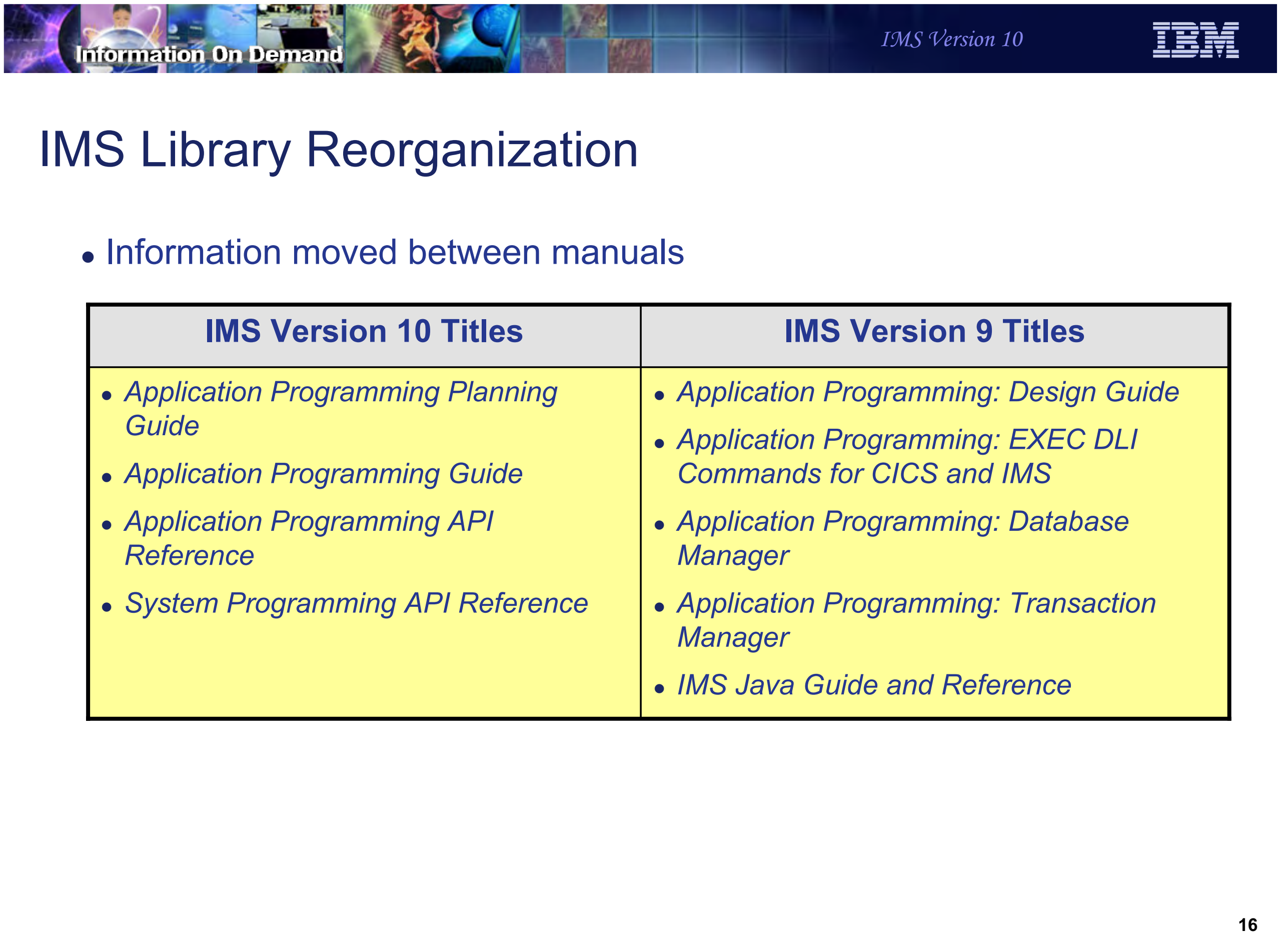
- IMS V9 licensed manuals will be unlicensed in IMS V10
- Information moved between manuals
  - Chapter 5 of Release Planning Guide explains the reorganization

IMS Version 10 Titles	IMS Version 9 Titles
Release Planning Guide *Installation Verification Guide* *System Definition Guide* *System Definition Reference*	*Release Planning Guide* *Installation Volume 1: Installation Verification* *Installation Volume 2: System Definition and Tailoring*
Exit Routine Reference	*Customization Guide*
Database Administration Guide *System Administration Guide* *Communications and Connections Guide*	*Administration: Database* *Administration: System* *Administration: Transaction Manager*
Operations and Automation Guide	*Operations Guide*

Figure 15-14. IMS Library Reorganization CMA01.0

Notes:

The following table summarizes the source of the content of these IMS V10 manuals.



IMS Library Reorganization

- Information moved between manuals

IMS Version 10 Titles	IMS Version 9 Titles
• *Application Programming Planning Guide* • *Application Programming Guide* • *Application Programming API Reference* • *System Programming API Reference*	• *Application Programming: Design Guide* • *Application Programming: EXEC DLI Commands for CICS and IMS* • *Application Programming: Database Manager* • *Application Programming: Transaction Manager* • *IMS Java Guide and Reference*

Figure 15-15. IMS Library Reorganization CMA01.0

Notes:

The following table summarizes the source of the content of these IMS V10 manuals.

Information On Demand
IMS Version 10
IBM

IMS Library Reorganization

- Information moved between manuals

IMS Version 10 Titles	IMS Version 9 Titles
- *Command Reference, Volume 1*	- *Command Reference Volume 1*
- *Command Reference, Volume 2*	- *Command Reference Volume 2*
- *Command Reference, Volume 3*	
- *Database Utilities Reference*	- *Utilities: Database and Transaction Manager*
- *System Utilities Reference*	- *Utilities: System*

Figure 15-16. IMS Library Reorganization CMA01.0

Notes:

The following table summarizes the source of the content of these IMS V10 manuals.

Information On Demand
IMS Version 10
IBM

IMS Library Reorganization

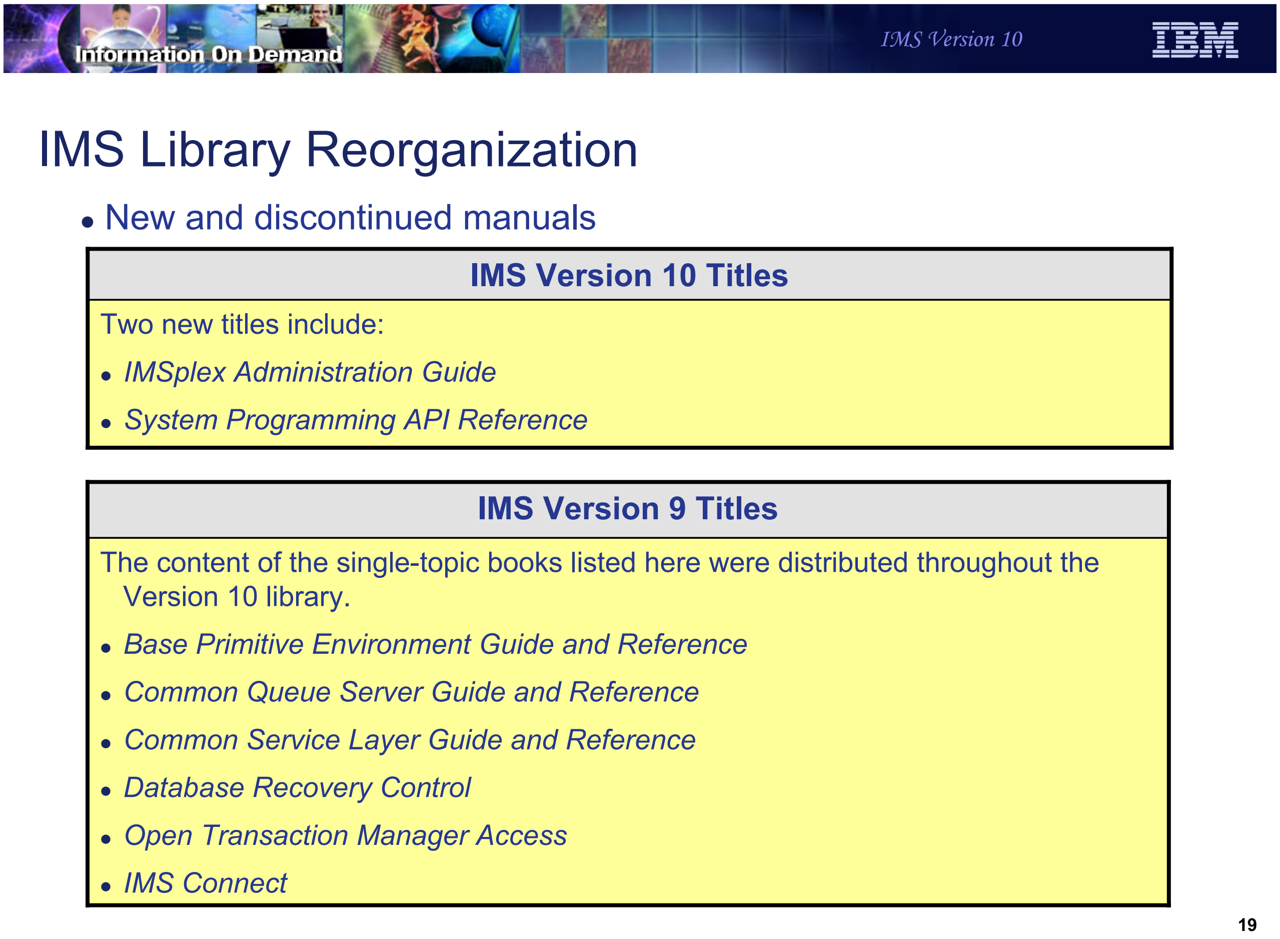
- Information moved between manuals

IMS Version 10 Titles	IMS Version 9 Titles
• *Diagnosis Guide* • *Diagnosis Reference* • *Messages and Codes Reference, Volume 1: DFS Messages* • *Messages and Codes Reference, Volume 2: Non-DFS Messages* • *Messages and Codes Reference, Volume 3: IMS Abend Codes* • *Messages and Codes Reference, Volume 4: IMS Component Codes*	• *Diagnosis Guide and Reference (licensed)* • *Failure Analysis Structure Tables (licensed)* • *Messages and Codes Volume 1: Non-DFS Messages* • *Messages and Codes Volume 2: DFS Messages*

Figure 15-17. IMS Library Reorganization CMA01.0

Notes:

The following table summarizes the source of the content of these IMS V10 manuals.



Information On Demand
IMS Version 10

IMS Library Reorganization

- New and discontinued manuals

IMS Version 10 Titles
Two new titles include:
• *IMSplex Administration Guide*
• *System Programming API Reference*

IMS Version 9 Titles
The content of the single-topic books listed here were distributed throughout the Version 10 library.
• *Base Primitive Environment Guide and Reference*
• *Common Queue Server Guide and Reference*
• *Common Service Layer Guide and Reference*
• *Database Recovery Control*
• *Open Transaction Manager Access*
• *IMS Connect*

Figure 15-18. IMS Library Reorganization CMA01.0

Notes:

The following table summarizes the source of the content of these IMS V10 manuals.

Information On Demand

IMS Version 10

IBM

IVP Enhancements

Figure 15-19. [No title] CMA01.0

Notes:

Information On Demand
IMS Version 10
IBM

Highlights

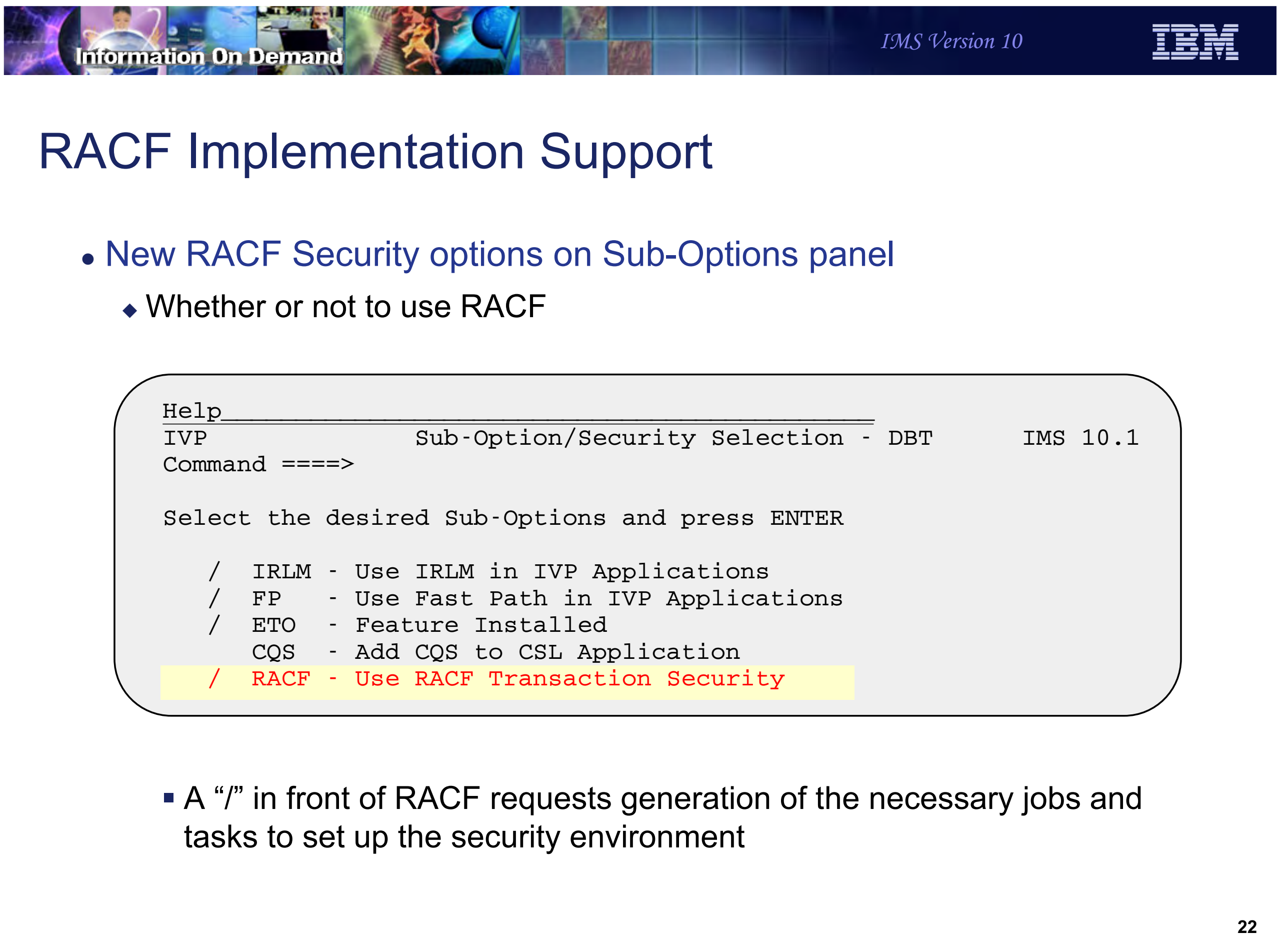
- IVP enhancements for existing functions
  - Support of RACF implementation and SMU removal
  - IMS Java sample application
  - IMS Connect sample application
- IVP support for new functions
  - Operations Manager (OM) Subscribe/Unsubscribe sample
  - Dynamic Resource Definition User Interface (DRD UI)
- Continues to be invoked as in previous releases
  - ISPF Option 6
    - EXEC ‘*hlq.SDFSCLST(DFSIXC01)*’ ‘HLQ(*imshlq*)’

      Where *imshlq* is the High Level Qualifier of the IMS data sets
  - IMS Application Menu

Figure 15-20. Highlights CMA01.0

Notes:

The IMS V10 Installation Verification Program provides new sample applications and examples to clarify existing functionality in IMS. Additionally, the IVP includes support for new functions that are introduced in IMS V10.



RACF Implementation Support

- New RACF Security options on Sub-Options panel
  - Whether or not to use RACF

```
Help_______________________________________________
IVP                 Sub-Option/Security Selection - DBT      IMS 10.1
Command ====>

Select the desired Sub-Options and press ENTER

   /  IRLM - Use IRLM in IVP Applications
   /  FP   - Use Fast Path in IVP Applications
   /  ETO  - Feature Installed
      CQS  - Add CQS to CSL Application
   /  RACF - Use RACF Transaction Security
```

  - A “/” in front of RACF requests generation of the necessary jobs and tasks to set up the security environment

Figure 15-21. RACF Implementation Support CMA01.0

Notes:

A new RACF Security section is added to the Sub-options panel which requires the user to specify whether or not RACF Security is to be used.

The default is not to use this sub-option (no slash).

If the sub-option is selected, the IVP builds the necessary jobs and tasks to define resources to RACF and to set up the use of several IMS security user exit routines. The user can modify the sample RACF resource definition task. The sample user exit routines always authorize the user to the resources. This sub-option is not available for DB batch.

New IVP steps that implement RACF security:

- IV_D218T which includes actions that replace the IV_E316J actions in previous releases. The new step establishes IMS Security and provides an example of defining the IVP security definitions to RACF.
- IV_C105J which provides a new step to replace the IV_E315J actions in previous releases. The new step assembles and binds the RACF security exits in

IMS.SDFSISRC into IMS.SDFSRESL and replaces the uses of DFSISIS0 with DFSRAS00.

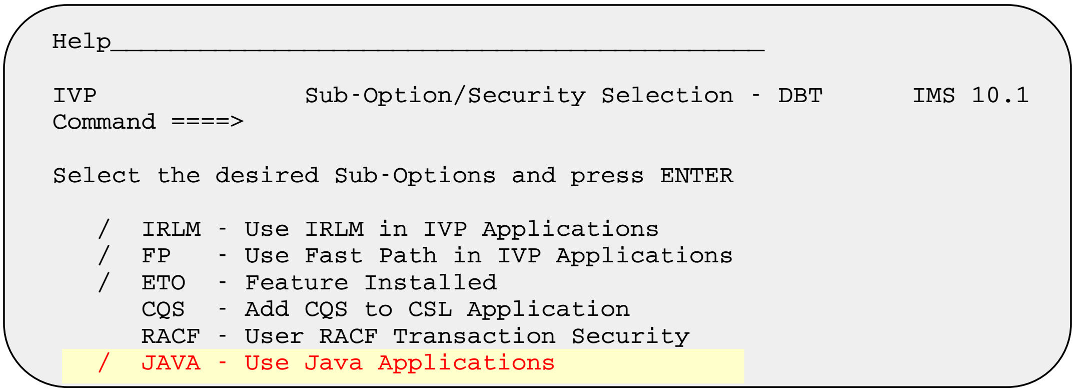
Modified IVP steps for RACF security:

- IV_C210T which browses the Stage1 source deck. Depending on the user security choice of using RACF or not, the SECURITY definition macro is also changed to specify whether or not the application resource access security authorization using RACF and exit routines are to be performed by IMS at execution time.
- If the RACF Security option was checked on the Sub-option panel, then the definition using RACF and exit routines is created as follows: SECURITY TYPE=(RAS,SIGNEXIT,TRANEXIT,RACFCOM),SECLVL=(SIGNON,TRANAUTH)
- If the security definition is to be done without using RACF then the definition is: SECURITY TYPE=(NORAS), SECLVL=(NOSIGN,NOTRAN)

Information On Demand
IMS Version 10

IMS Java Sample

- New Java Application options on Sub-Options panel



```
Help______________________________________________

IVP                Sub-Option/Security Selection - DBT      IMS 10.1
Command ====>

Select the desired Sub-Options and press ENTER

   /  IRLM - Use IRLM in IVP Applications
   /  FP   - Use Fast Path in IVP Applications
   /  ETO  - Feature Installed
      CQS  - Add CQS to CSL Application
      RACF - User RACF Transaction Security
   /  JAVA - Use Java Applications
```

  - Adds necessary jobs and tasks for the IMS JDBC application sample
  - The Java sub-option is not available for DB batch or DCCTL environments

Figure 15-22. IMS Java Sample CMA01.0

Notes:

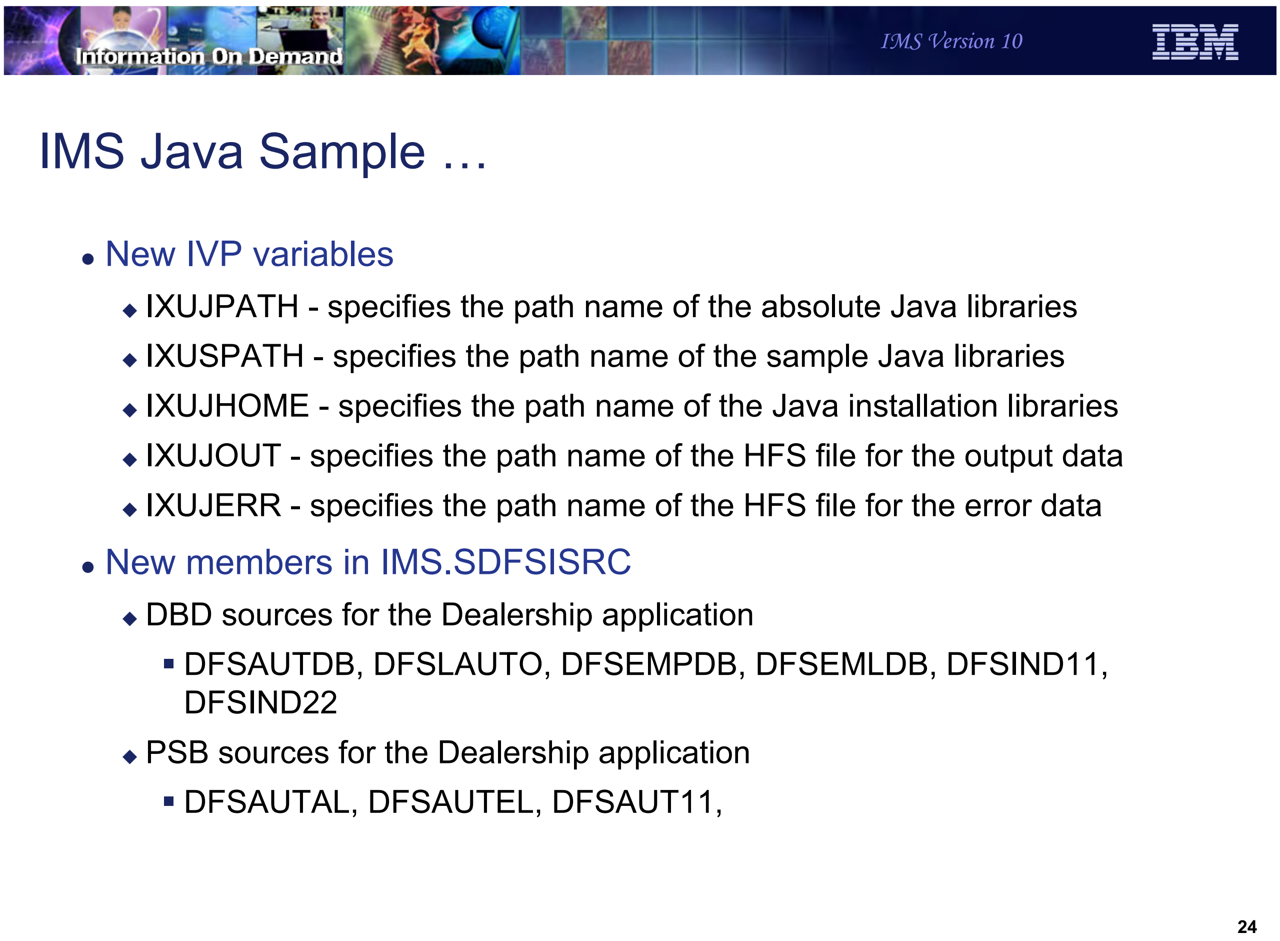
A new Java section is added to the Sub-options panel. If the sub-option is selected, the IVP adds the necessary jobs and tasks for the IMS JDBC application execution.

The Java Sample verifies that a Java workload can be run in an IMS Java Dependent Region environment. The changes in the IVP that support this capability are included in the "C", "E", "G", and "H" steps.

The IVP provides the steps that:

- Add new database, application, and transaction definition statements to the IVP stage 1 source input.
- Add control statement members to the IMS.PROCLIB for Java virtual machine (JVM) initialization.
- Allocate the data set for the auto dealership database.
- Start an IMS control region.
- Start a JMP dependent region.

- Run a transaction in the JMP dependent region.
- Run a batch job in the JBP region to access databases.



Information On Demand
IMS Version 10
IBM

IMS Java Sample …

- New IVP variables
  - IXUJPATH - specifies the path name of the absolute Java libraries
  - IXUSPATH - specifies the path name of the sample Java libraries
  - IXUJHOME - specifies the path name of the Java installation libraries
  - IXUJOUT - specifies the path name of the HFS file for the output data
  - IXUJERR - specifies the path name of the HFS file for the error data
- New members in IMS.SDFSISRC
  - DBD sources for the Dealership application
    - DFSAUTDB, DFSLAUTO, DFSEMPDB, DFSEMLDB, DFSIND11, DFSIND22
  - PSB sources for the Dealership application
    - DFSAUTAL, DFSAUTEL, DFSAUT11,

Figure 15-23. IMS Java Sample … CMA01.0

Notes:

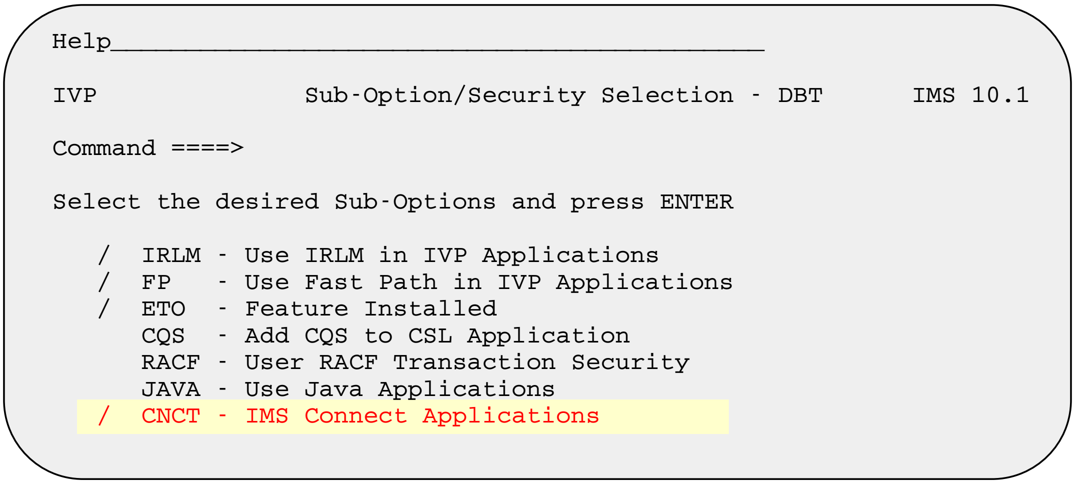
Several new IVP variables have also been added to support the Java sample:

- IXUJPATH - specifies the path name of the absolute Java libraries. The default value is /usr/lpp/ims/imsjava10. This substitutes a value for string “ImsjavaPath” in the IMS DFSJVMMS and DFSJVMEV proclib members.
- IXUSPATH - specifies the path name of the sample Java libraries. The default value is /usr/lpp/ims/imsjava10. This substitutes a value for string “SamplesPath” in the IMS DFSJVMMS and DFSJVMEV proclib members.
- IXUJHOME - specifies the path name of the Java installation libraries. The default value is /usr/lpp/java150/J1.5/bin. This substitutes a value for string “JavaHome” in the IMS DFSJVMMS and DFSJVMEV proclib members.
- IXUJOUT - specifies the path name of the HFS file for the output data. The default value is /tmp
- IXUJERR - specifies the path name of the HFS file for the error data. The default value is /tmp

To support the IVP, new DBD and PSB sources have been added into the IMS.SDFSISRC library.

IMS Connect Sample Application

- New IMS Connect sample application option



```
Help_____________________________________________

IVP                 Sub-Option/Security Selection - DBT      IMS 10.1

Command ====>

Select the desired Sub-Options and press ENTER

   /  IRLM - Use IRLM in IVP Applications
   /  FP   - Use Fast Path in IVP Applications
   /  ETO  - Feature Installed
      CQS  - Add CQS to CSL Application
      RACF - User RACF Transaction Security
      JAVA - Use Java Applications
   /  CNCT - IMS Connect Applications
```

Figure 15-24. IMS Connect Sample Application

Notes:

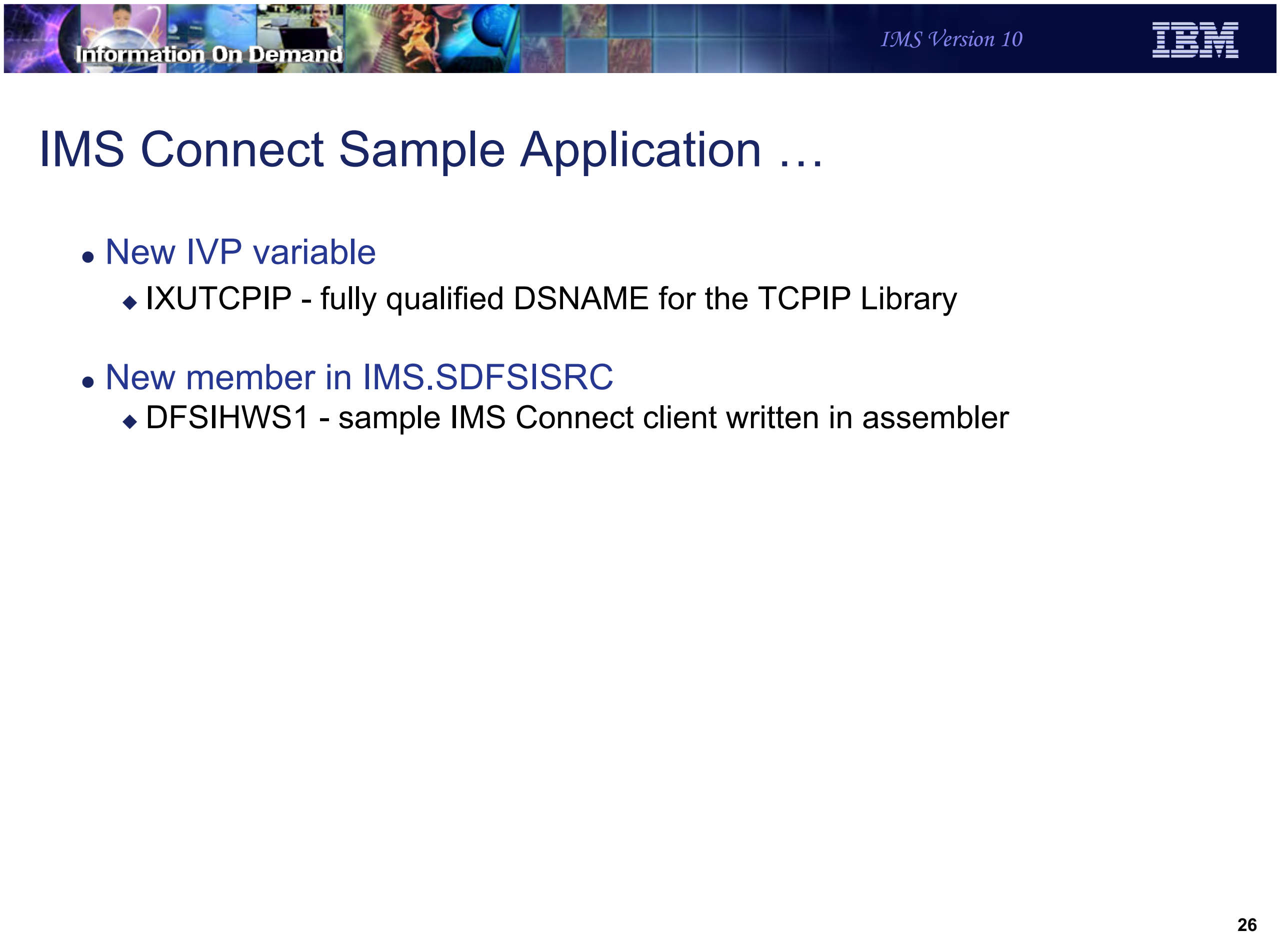
A new IMS Connect section is added to the Sub-options panel. The default is not to use this sub-option (no slash). It is available for DB/DC and XRF environments. If this sub-option is selected, the IVP adds the necessary jobs and tasks for IMS Connect application execution

DFSIHWS1 is an assembler language IMS Connect application program that sends the PART transaction to IMS Connect and receives a response.

The IVP provides the steps that:

- Assemble and bind the sample application program and user exits.
- Add a configuration member in the PROCLIB data set to define the IMS Connect environment.
- Allocate the Recorder data set HWSRCDR for output data capture.
- Start an IMS control region with OTMA enabled.
- Start the IMS Connect address space.

- Send a transaction message to IMS via IMS Connect and receive a response.



Information On Demand
IMS Version 10
IBM

IMS Connect Sample Application …

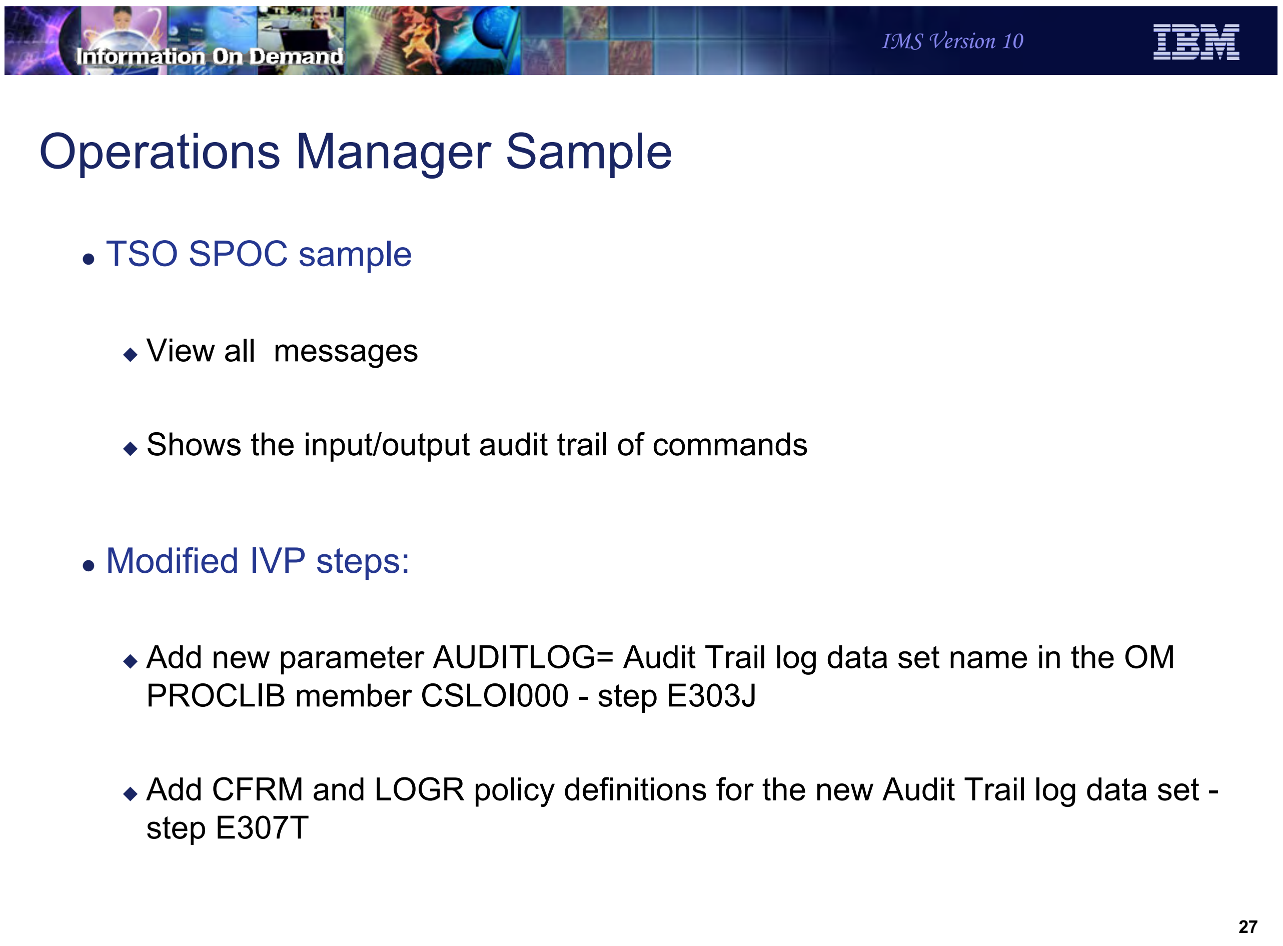
- New IVP variable
  - IXUTCPIP - fully qualified DSNAME for the TCPIP Library
- New member in IMS.SDFSISRC
  - DFSIHWS1 - sample IMS Connect client written in assembler

26

Figure 15-25. IMS Connect Sample Application … CMA01.0

Notes:

To support the IMS Connect IVP, a new IVP variable and a new member in IMS.SDFSISRC have been added.



Information On Demand
IMS Version 10
IBM

Operations Manager Sample

- TSO SPOC sample
  - View all messages
  - Shows the input/output audit trail of commands
- Modified IVP steps:
  - Add new parameter AUDITLOG= Audit Trail log data set name in the OM PROCLIB member CSLOI000 - step E303J
  - Add CFRM and LOGR policy definitions for the new Audit Trail log data set - step E307T

27

Figure 15-26. Operations Manager Sample CMA01.0

Notes:

IMS V10 provides a TSO SPOC sample to view all messages and displays information from the audit trail.

The changes are delivered as enhancements to the CSL “E” steps of the IVP jobs and tasks.

Manage Resources

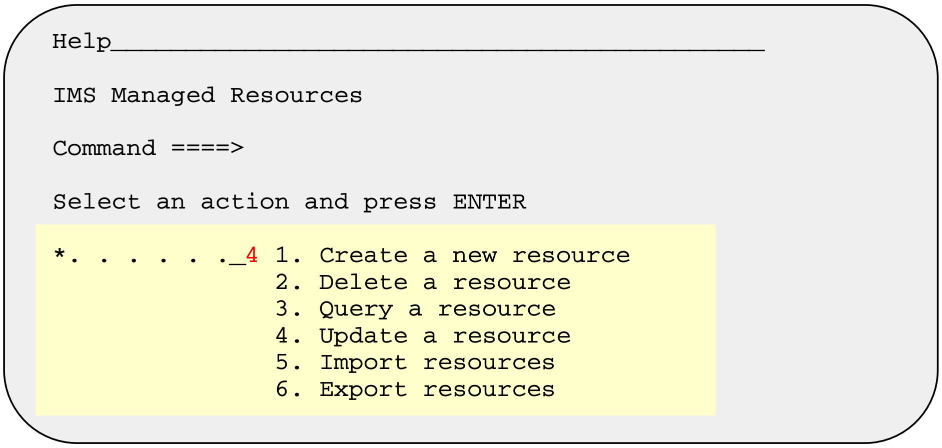
- Dynamic Resource Definition is a new capability in IMS 10
  - Manage Resources is a general purpose interface that can be used to issue various resource definition and operational commands
- The IVP for Manage Resources provides
  - A set of ISPF panels that provide support for DRD commands
  - An IVP example of updating a program.
  - An IVP example of updating a transaction.

Figure 15-27. Manage Resources CMA01.0

Notes:

Manage Resources is a new capability in IMS 10. The IVP provides samples, using Manage Resources, that update both a program and a transaction.

DRD Example



```
Help______________________________________________

IMS Managed Resources

Command ====>

Select an action and press ENTER

*. . . . . ._4 1. Create a new resource
               2. Delete a resource
               3. Query a resource
               4. Update a resource
               5. Import resources
               6. Export resources
```

Figure 15-28. DRD Example CMA01.0

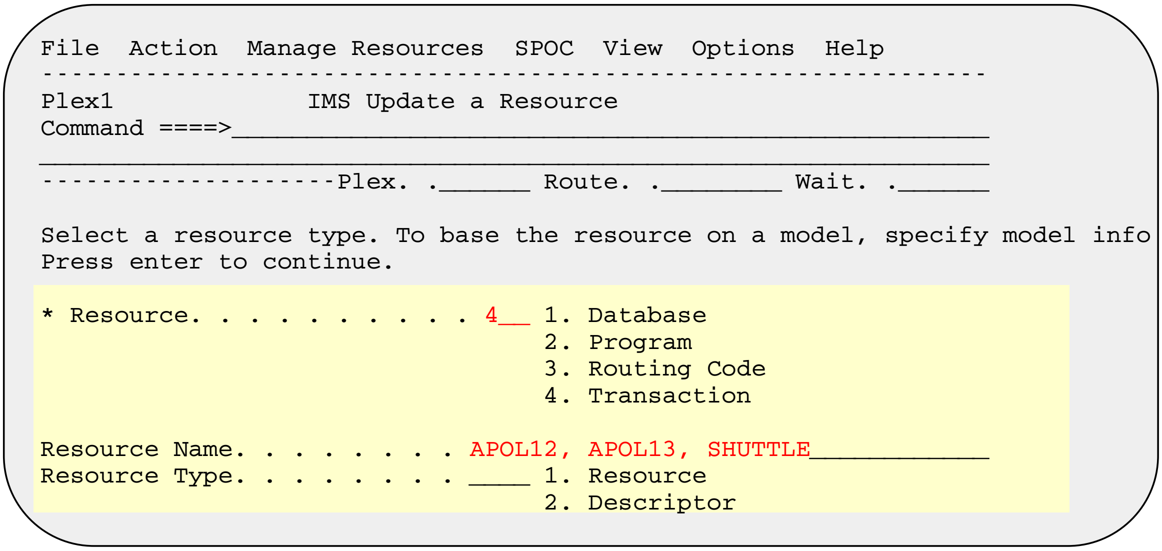
Notes:

A new panel provides the user with options on actions that can be performed against resources. In this example a request is made to update a resource by entering option 4.

Information On Demand

IMS Version 10

DRD Example …



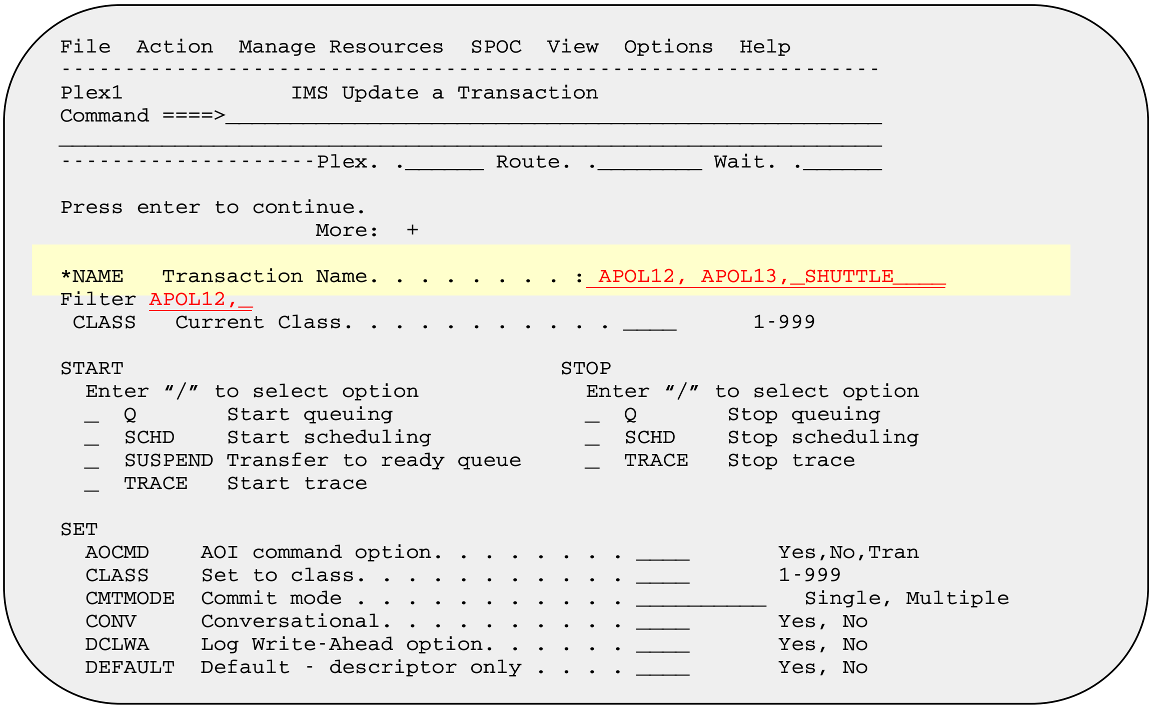
```
File   Action  Manage Resources  SPOC  View  Options  Help
----------------------------------------------------------------
Plex1                IMS Update a Resource
Command ====>____________________________________________________
_________________________________________________________________
--------------------Plex. ._____ Route. ._______ Wait. ._____

Select a resource type. To base the resource on a model, specify model info
Press enter to continue.

* Resource. . . . . . . . . . . 4__ 1. Database
                                    2. Program
                                    3. Routing Code
                                    4. Transaction

Resource Name. . . . . . . . APOL12, APOL13, SHUTTLE____________
Resource Type. . . . . . . . ____ 1. Resource
                                  2. Descriptor
```

Figure 15-29. DRD Example … CMA01.0

Notes:

The next panel that displays is the panel that allows a user to select which resource is to be updated. The example on this visual shows a request to update a transaction.

DRD Example …



```
File  Action  Manage Resources  SPOC  View  Options  Help
----------------------------------------------------------------
Plex1                IMS Update a Transaction
Command ====>___________________________________________________
________________________________________________________________
--------------------Plex. .______ Route. .________ Wait. .______

Press enter to continue.
                    More:  +

*NAME   Transaction Name. . . . . . . . : APOL12, APOL13,_SHUTTLE____
Filter APOL12,_
 CLASS   Current Class. . . . . . . . . . ____      1-999

START                                    STOP
  Enter "/" to select option               Enter "/" to select option
  _  Q        Start queuing                _  Q        Stop queuing
  _  SCHD     Start scheduling             _  SCHD     Stop scheduling
  _  SUSPEND Transfer to ready queue       _  TRACE    Stop trace
  _  TRACE    Start trace

SET
  AOCMD    AOI command option. . . . . . . . ____       Yes,No,Tran
  CLASS    Set to class. . . . . . . . . . . ____       1-999
  CMTMODE  Commit mode . . . . . . . . . . . __________   Single, Multiple
  CONV     Conversational. . . . . . . . . . ____       Yes, No
  DCLWA    Log Write-Ahead option. . . . . . ____       Yes, No
  DEFAULT  Default - descriptor only . . . . ____       Yes, No
```

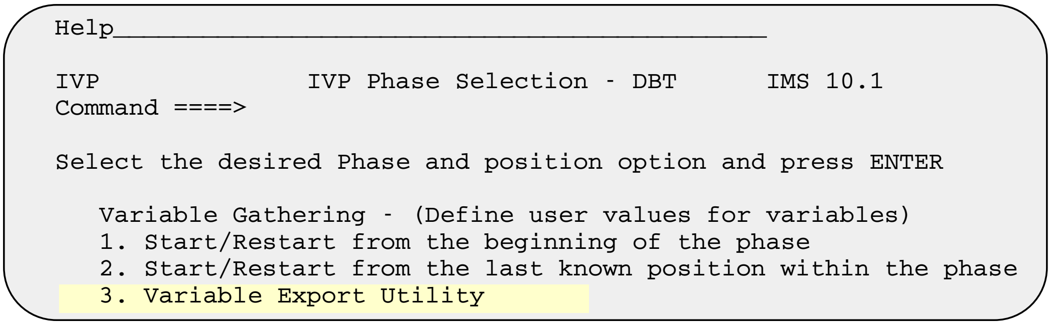
Figure 15-30. DRD Example … CMA01.0

Notes:

The previous request results in the panel that defines the parameter values applicable to a transaction. The user can now provide the desired changes.

Miscellaneous

- Support for IXUUPROC variable as optional rather than required
  - User JES2 PROCLIB ddname or JES3 DDNAME suffix
    - If not specified, /*JOBPARM or //*MAIN statement not generated
    - JCLLIB ORDER statement can be used to specify PROCLIB
- New option to access the IVP Variable Export Utility from IVP panel



```
Help____________________________________________

IVP               IVP Phase Selection - DBT      IMS 10.1
Command ====>

Select the desired Phase and position option and press ENTER

   Variable Gathering - (Define user values for variables)
   1. Start/Restart from the beginning of the phase
   2. Start/Restart from the last known position within the phase
   3. Variable Export Utility
```

Figure 15-31. Miscellaneous CMA01.0

Notes:

The IVP variable IXUUPROC (User JES2 PROCLIB DDNAME or JES3 DDNAME suffix) was a required value in prior releases of IMS. In the IMS V10 IVP, it is made optional. If the variable is not specified, the JES2 “/*JOBPARM PROCLIB=xxxxxxxx” JCL statement or the JES3 “//*MAIN PROC=xx” is not generated in the JCL. This allows the user to specify the JES PROCLIBs using the JCLLIB ORDER statement in one of the JESx variables IXUJESC1 to IXUJESC5.

The Variable Export Utility can be directly accessed as an option from the IVP Phase Selection panel.

Miscellaneous …

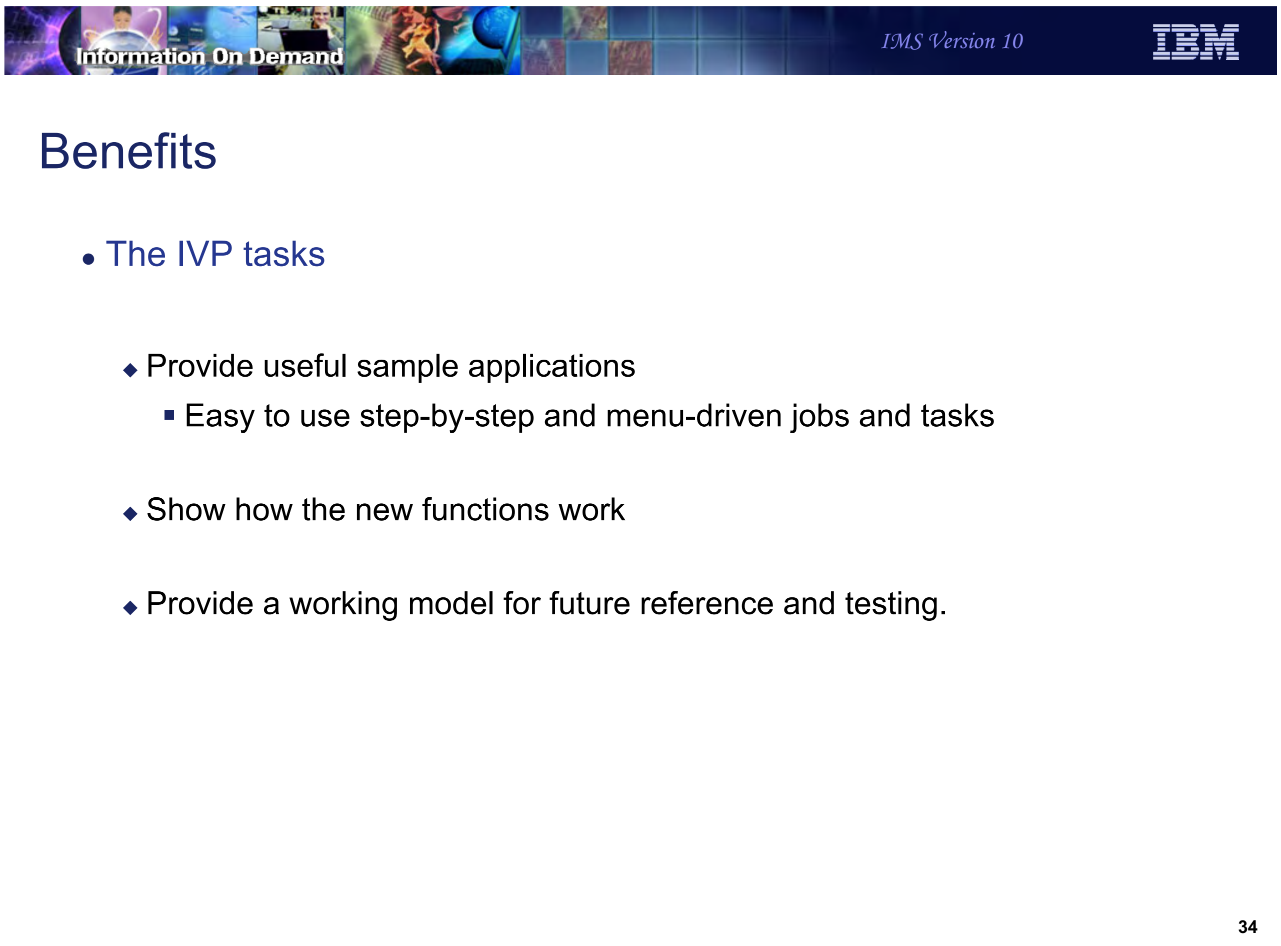
- More useful MFS device definition
  - Default of DEV TYPE=(3270,2),FEAT=IGNORE
    - Rather than DEV TYPE=3270-A02,FEAT=IGNORE
- ALL IVP JCL has been changed for UNIT type
  - SYSALLDA
    - From SYSDA

Figure 15-32. Miscellaneous … CMA01.0

Notes:

The IMS IVP job IV_E204J has been changed to provide a DEV statement default of 'DEV TYPE=(3270,2),FEAT=IGNORE' rather than 'DEV TYPE=3270-A02,FEAT=IGNORE'. This is used for all terminals defined to IMS with 'DEV TYPE=3270-An' and SIZE=(x,80) where x is greater than or equal to 24 (in addition to using it for terminals specifically defined as 'DEV TYPE=(3270,2)'. The change makes the IVP easier to use for many customers because it eliminates the need to have to edit the MFS source to match their systems.

The hard coded unit type SYSDA is changed to SYSALLDA in all the IVP JCL. SYSDA is not a required unit type. SYSALLDA is a required unit type.



Information On Demand

IMS Version 10

Benefits

- The IVP tasks
  - Provide useful sample applications
    - Easy to use step-by-step and menu-driven jobs and tasks
  - Show how the new functions work
  - Provide a working model for future reference and testing.

34

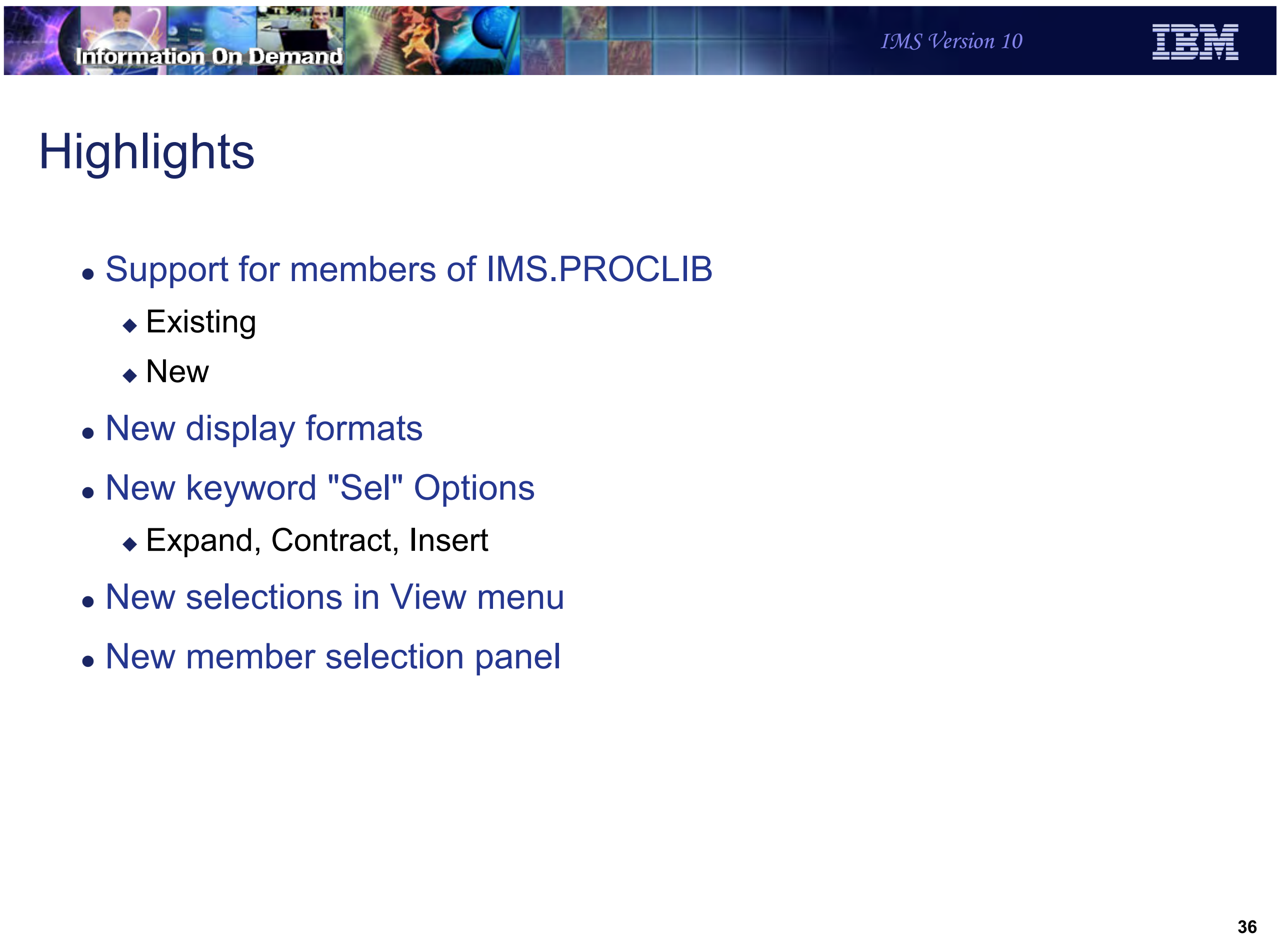
Figure 15-33. Benefits CMA01.0

Notes:

Syntax Checker Enhancements

Figure 15-34. [No title] CMA01.0

Notes:



Information On Demand

IMS Version 10

Highlights

- Support for members of IMS.PROCLIB
  - Existing
  - New
- New display formats
- New keyword "Sel" Options
  - Expand, Contract, Insert
- New selections in View menu
- New member selection panel

36

Figure 15-35. Highlights CMA01.0

Notes:

The Syntax Checker is an IMS capability that can define, verify and validate the parameters and their values in the members of IMS.PROCLIB. IMS V10 provides several enhancements to the Syntax Checker as listed on this visual.

Invocation

- IMS 10 Syntax Checker
  - Continues to be invoked as in previous releases
    - ISPF Option 6
    - EXEC '*hlq.SDFSEXEC(DFSSCSRT*)' 'HLQ(*hlq*)'

      Where *hlq* is the High Level Qualifier of the IMS data sets
  - Coexists on the same z/OS along with previous releases of IMS
    - Can be used to check parameters in IMS 10 and previous IMS releases

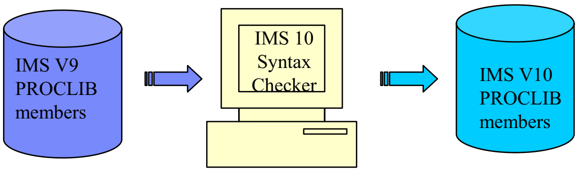
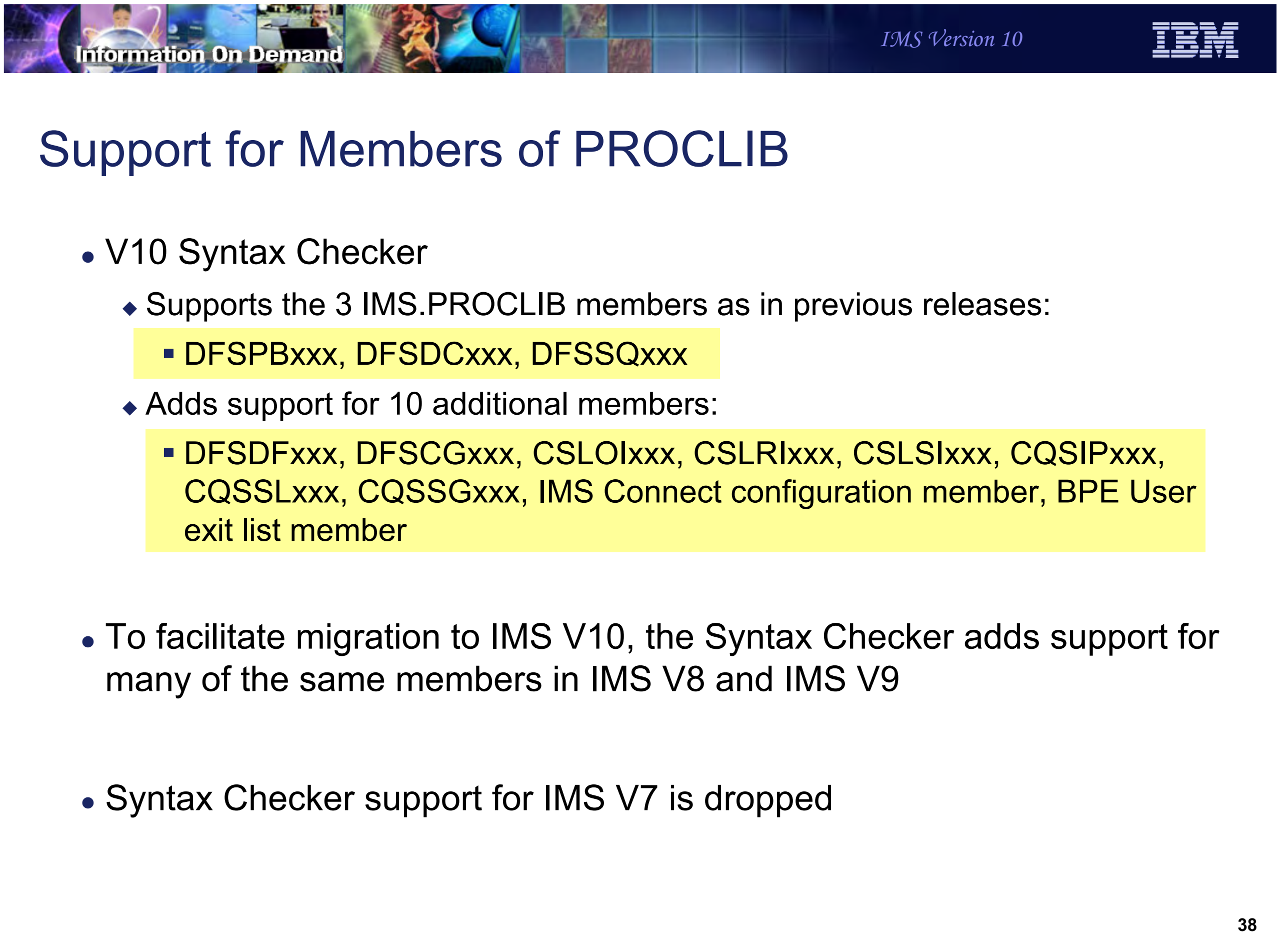


Figure 15-36. Invocation CMA01.0

Notes:

The invocation of the Syntax Checker in IMS V10 is no different than in previous releases. It is a stand alone, offline product that can coexist on the same z/OS environment as previous releases of IMS. When used to check parameters of previous releases, make sure that the release number shown is correct.


Information On Demand
IMS Version 10
IBM

Support for Members of PROCLIB

- V10 Syntax Checker
  - Supports the 3 IMS.PROCLIB members as in previous releases:
    - DFSPBxxx, DFSDCxxx, DFSSQxxx
  - Adds support for 10 additional members:
    - DFSDFxxx, DFSCGxxx, CSLOIxxx, CSLRIxxx, CSLSIxxx, CQSIPxxx, CQSSLxxx, CQSSGxxx, IMS Connect configuration member, BPE User exit list member
- To facilitate migration to IMS V10, the Syntax Checker adds support for many of the same members in IMS V8 and IMS V9
- Syntax Checker support for IMS V7 is dropped

Figure 15-37. Support for Members of PROCLIB CMA01.0

Notes:

In IMS V10, the Syntax Checker continues to support the following members: DFSPBxxx, DFSDCxxx and the DFSSQxxx. In addition, support for 10 other members of IMS.PROCLIB has been added:

DFSDFxxx - Dynamic Resource Definition, Common Service Layer, Shared Queues, and Diagnostics and Statistics parameters. (Support for V10 only)

DFSCGxxx - Common Service Layer Parameters

CSLOIxxx - OM Initialization Parameters

CSLRIxxx - RM Initialization Parameters

CSLSIxxx - SCI Initialization Parameters

CQSIPxxx - CQS Initialization Parameters

CQSSLxxx - CQS Local Structure Definition Proclib Member

CQSSGxxx - CQS Global Structure Definition Proclib Member

IMS Connect Configuration Member. (Note: These members do not have a specific member name prefix. The user can select any name. Support for V9 and V10 only).

BPE User Exit List member (Note: These members do not have a specific member name prefix. The user can select any name).

To facilitate migration to IMS V10, the Syntax Checker's support for the currently processed members, DFSPBxxx, DFSDCxxx and the DFSSQxxx includes any new parameters in V8 and V9 that have been added since GA of those releases. Support of IMS V7 PROCLIB members is dropped. Syntax Checker only supports the three latest releases of IMS.

Support for Members of PROCLIB …

- DFSDFxxx member
  - New in IMS V10
    - Supports Dynamic Resource Definition, Common Service Layer and Shared Queues parameters
      - Special section tag
        - <SECTION=DYNAMIC_RESOURCES>
        - <SECTION=SHARED_QUEUES>
        - <SECTION=COMMON_SERVICE_LAYER>
        - <SECTION=DIAGNOSTICS_STATISTICS>
- Member names that have no pre-defined prefix
  - Set by the System Programmer
    - IMS Connect configuration member
    - BPE user exit list member

Figure 15-38. Support for Members of PROCLIB … CMA01.0

Notes:

IMS V10 introduces a new member in IMS.PROCLIB. DFSDFxxx supports the parameters for Dynamic Resource Definition, Common Service Layer, Shared Queues, and Diagnostics and Statistics . The Syntax Checker recognizes these parameters and provides validation support.

Note that the DFSDFxxx member is new to IMS V10. Additionally, the IMS Connect configuration member and BPE user exit list member all have names that can be tailored to the user environment. As a result, the names are not specifically listed here but can be defined to the Syntax Checker during execution.

Support for Comments

- V10 Syntax Checker recognizes comments for several members
  - “*” in column 1 for multiple lines
  - “/*…..*/ ” in the same line as the parameter

Example

```
*-----------------------------------------------------------------*
* IMS COMMON SERVICE LAYER PROCLIB MEMBER                         *
*
*-----------------------------------------------------------------*
CMDSEC=N,                   /* NO CMD AUTHORIZATION CHECKING */
```

Figure 15-39. Support for Comments CMA01.0

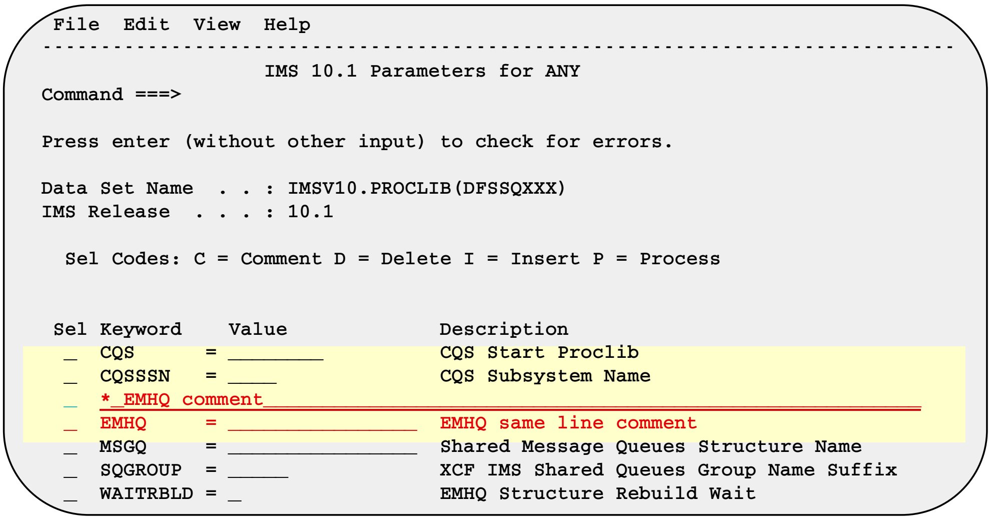
Notes:

The IMS V10 Syntax Checker recognizes comments in several of the members of Proclib. Comments can be in two forms:

- When an “*” is encountered in column 1, the entire line is considered a comment. A member can have multiple consecutive comment lines.
- On the same line as a a parameter, the text between a paired “/*” and “*/” is considered a comment.

Support for Members of PROCLIB …

- Changed DFSSQxxx member
  - IMS 10 supports comments

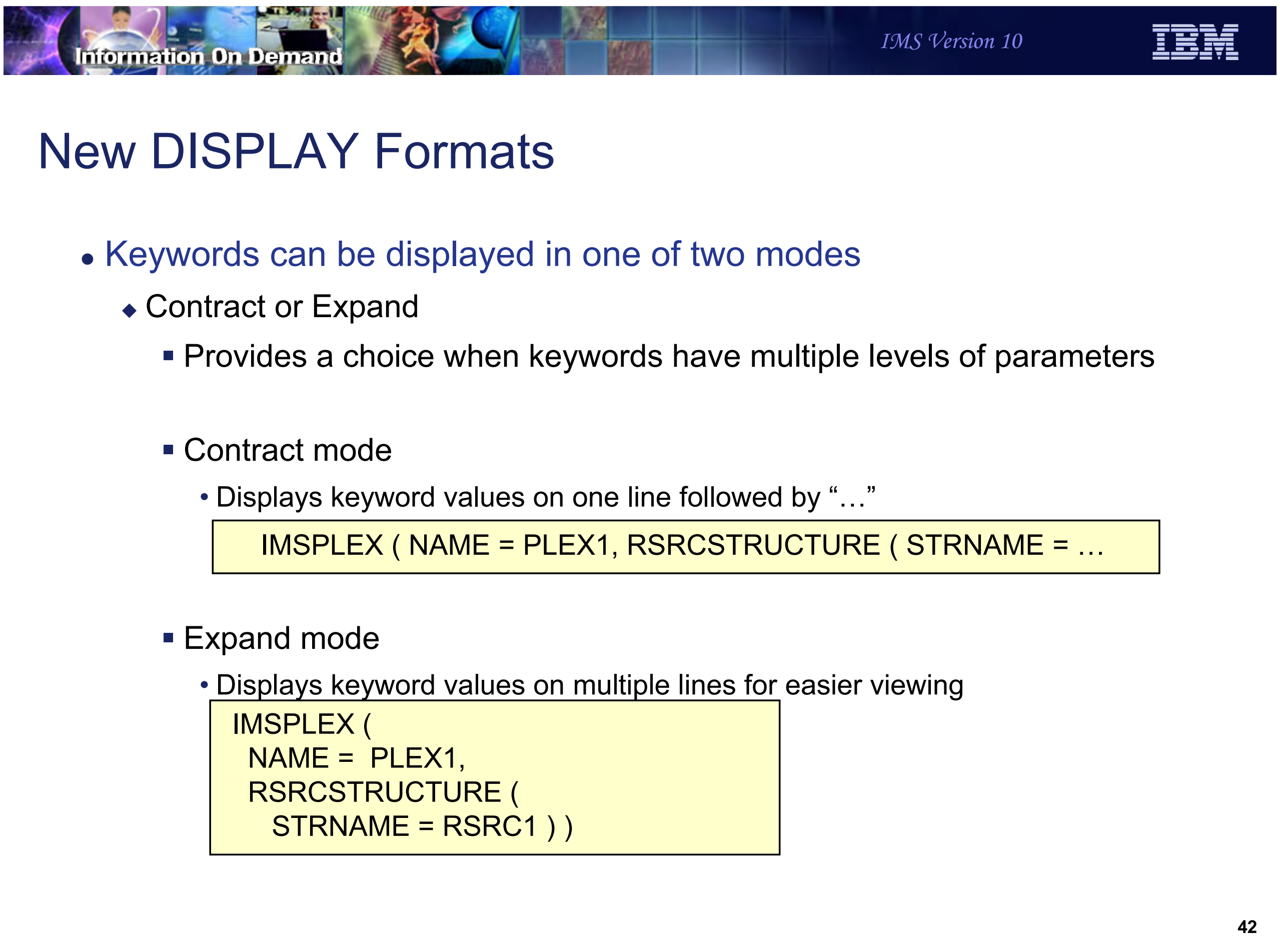


```
 File  Edit  View  Help
------------------------------------------------------------------------------
                   IMS 10.1 Parameters for ANY
Command ===>

Press enter (without other input) to check for errors.

Data Set Name  . . : IMSV10.PROCLIB(DFSSQXXX)
IMS Release  . . . : 10.1

  Sel Codes: C = Comment D = Delete I = Insert P = Process


 Sel Keyword    Value                Description
  _  CQS      = ________             CQS Start Proclib
  _  CQSSSN   = ____                 CQS Subsystem Name
  _  *_EMHQ comment_____________________________________________________________
  _  EMHQ     = ________________     EMHQ same line comment
  _  MSGQ     = ________________     Shared Message Queues Structure Name
  _  SQGROUP  = _____                XCF IMS Shared Queues Group Name Suffix
  _  WAITRBLD = _                    EMHQ Structure Rebuild Wait
```

Figure 15-40. Support for Members of PROCLIB … CMA01.0

Notes:

The panel for the Shared Queues member, DFSSQxxx, now supports the addition of comments. Note that this is added only for IMS V10 and does not affect IMS V8 or IMS V9.


Information On Demand
IMS Version 10
IBM

New DISPLAY Formats

- Keywords can be displayed in one of two modes
  - Contract or Expand
    - Provides a choice when keywords have multiple levels of parameters
    - Contract mode
      - Displays keyword values on one line followed by "…"

```
IMSPLEX ( NAME = PLEX1, RSRCSTRUCTURE ( STRNAME = …
```

    - Expand mode
      - Displays keyword values on multiple lines for easier viewing

```
IMSPLEX (
  NAME =  PLEX1,
  RSRCSTRUCTURE (
     STRNAME = RSRC1 ) )
```

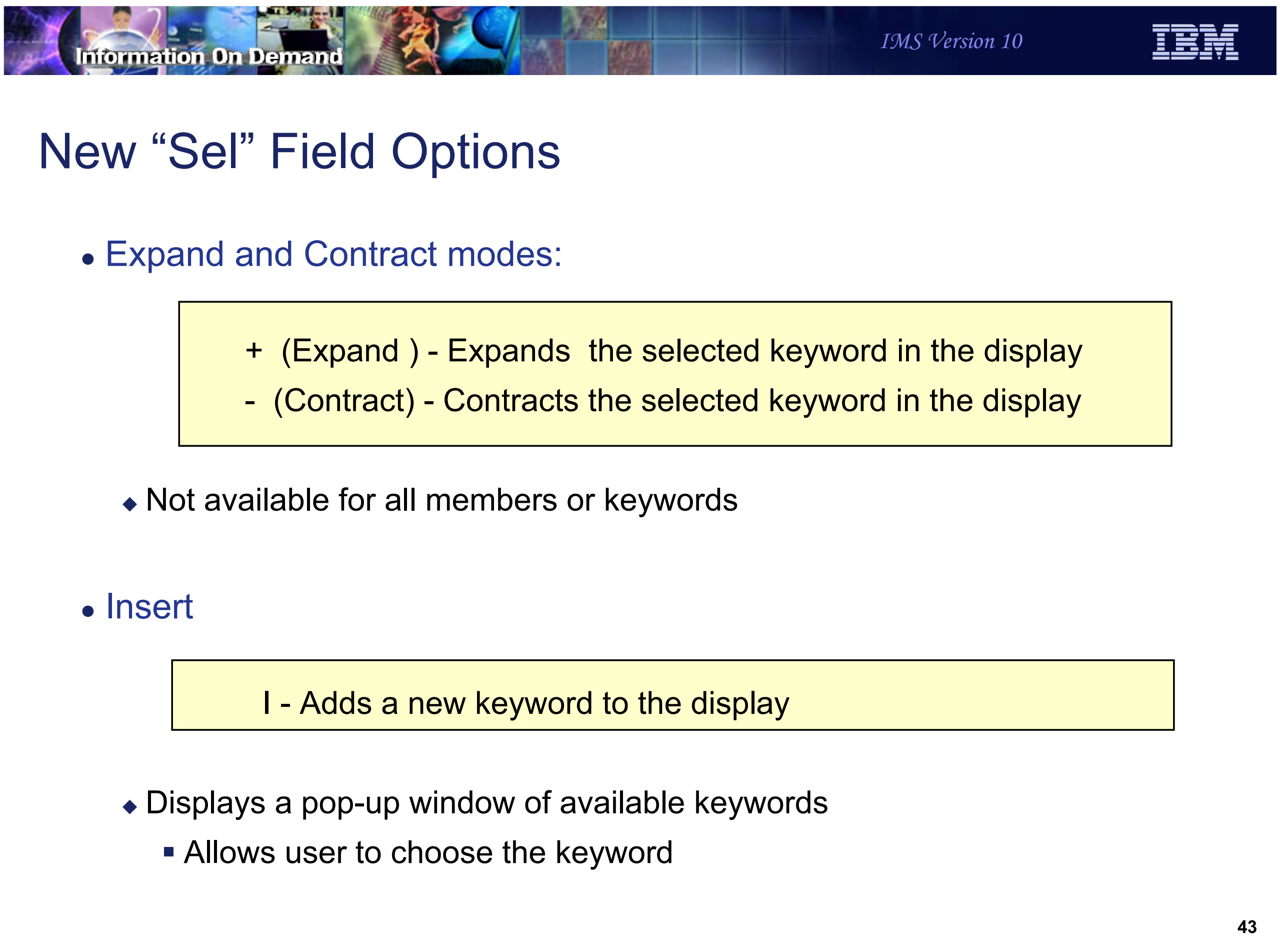
Figure 15-41. New DISPLAY Formats CMA01.0

Notes:

To facilitate viewing keyword values when sub-keywords and sub-keyword lists are available, two modes are provided.

The Contract mode displays the keyword values on one line. If the values associated with the keyword extend beyond the line, they are not shown but a "…" is displayed to give the viewer an indication that there is more to the keyword that is not displayed.

The Expand mode uses multiple lines, and as many as are needed, to display all the keywords, sub-keywords and sub-keyword lists that are part of the definition.


Information On Demand
IMS Version 10
IBM

New “Sel” Field Options

- Expand and Contract modes:

+ (Expand) - Expands the selected keyword in the display
- (Contract) - Contracts the selected keyword in the display

  - Not available for all members or keywords

- Insert

I - Adds a new keyword to the display

  - Displays a pop-up window of available keywords
    - Allows user to choose the keyword

Figure 15-42. New “Sel” Field Options CMA01.0

Notes:

The “Sel” field in the Parameters panel provides new options to invoke the Expand and Contract modes:

- The “Expand” “Sel” option, “+” (plus), displays the keyword on multiple lines. As described on previous visuals, when a keyword is expanded, the sub-parameters are displayed in a predefined order.
- The “Contract” “Sel” option, “-” (minus sign), displays the keywords on one line. As much of the keyword as will fit on one line is displayed followed by “...”.

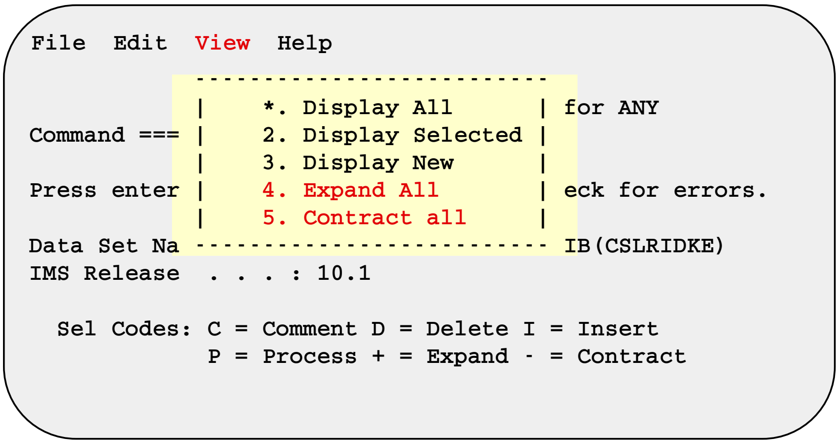
The options are not available for all members or keywords.

The “Sel” field also provides the option to Insert a new keyword. When selected, a pop-up window of available keywords is displayed to allow the user to choose the desired keyword.

- I (Insert) adds a new keyword to the display. Note that insertion of a comment requires the use of Sel code “C”.

New “Pull Down” Options

- The “Pull Down” options that are displayed when a View is requested have been enhanced to include
  - Expand All - expands all the keywords in the display
  - Contract All - contracts all the keywords in the display
  - Not available for all members



```
File  Edit  View  Help
            -------------------------
            |    *. Display All       | for ANY
Command === |    2. Display Selected |
            |    3. Display New       |
Press enter |    4. Expand All        | eck for errors.
            |    5. Contract all      |
Data Set Na -------------------------  IB(CSLRIDKE)
IMS Release  . . . : 10.1

  Sel Codes: C = Comment D = Delete I = Insert
             P = Process + = Expand - = Contract
```

Figure 15-43. New “Pull Down” Options CMA01.0

Notes:

New “Pull Down” options are displayed when View is selected from the top of the screen. The Expand All and Contract ALL options are new.

Information On Demand
IMS Version 10

New Member Selection Panel

- Some new members names have no fixed prefix
  - Defined for an environment and loaded when required
    - BPE Exit List Member
    - IMS Connect Configuration Member

- Once Syntax Checker detects unknown member name
  - Member Selection list is displayed.

Figure 15-44. New Member Selection Panel CMA01.0

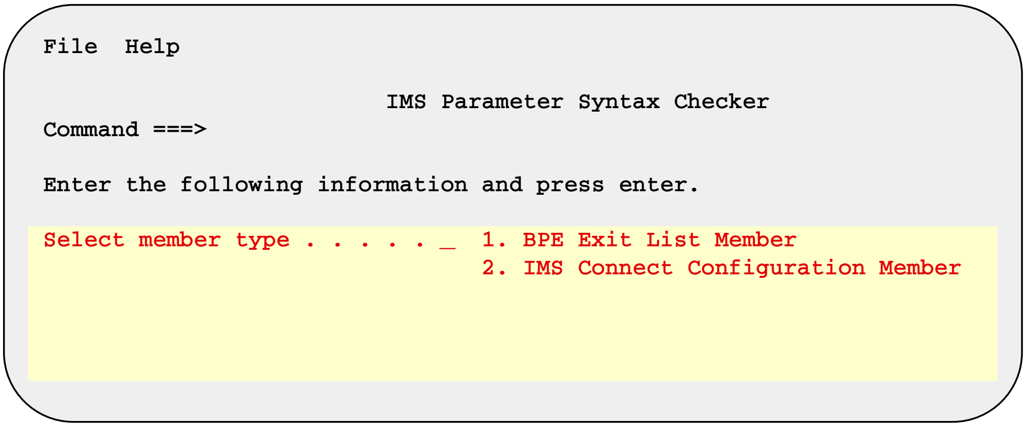
Notes:

Two of the new PROCLIB member names that can be processed by the Syntax Checker do not have required names or prefixes. They are defined by the user requesting the function and then specified as part of the startup procedure. These members include the BPE Exit List Member and the IMS Connect Configuration Member.

When the Syntax Checker detects an unknown member name, a new Member Selection list is displayed to assist in the identification of the member.

New Member Selection Panel …

- Example



```
File  Help

                         IMS Parameter Syntax Checker
Command ===>

Enter the following information and press enter.

Select member type . . . . . . _  1. BPE Exit List Member
                                  2. IMS Connect Configuration Member
```

Figure 15-45. New Member Selection Panel … CMA01.0

Notes:

This visual shows the panel that is displayed to assist the user in identifying the member type.

Information On Demand
IMS Version 10
IBM

Benefits

- IMS V10 Syntax Checker
  - Assists with IMS release to release migrations when converting PROCLIB members
    - Provides the ability to define, verify and validate parameters
    - Supports new parameters
    - Supports 13 members of IMS.PROCLIB

Figure 15-46. Benefits CMA01.0

Notes:

KBLA Coexistence

Figure 15-47. [No title] CMA01.0

Notes:

Knowledge-Base Log Analysis Highlights

- KBLA Enhancements
  - Support for IMS 10 log record changes
  - Support for concurrent KBLA-supported levels of IMS (V9 and V10)
    - Ensures correct log record DSECTs are used for the corresponding IMS version of the log records
    - Generates JCL appropriate to the IMS release
  - Ease of migration to IMS 10
    - Ensures that prior release parameters are not invalidated or superseded by the installation of IMS 10
      - New panel to facilitate migration
      - No reconfiguration required for setup

Figure 15-48. Knowledge-Base Log Analysis Highlights CMA01.0

Notes:

IMS V9 introduced the ISPF menu-based utility, KBLA (Knowledge-Based Log Analysis), for processing log records.

IMS V10 enhances this capability to support the log record changes introduced in this release. Additionally, enhancements to support multiple concurrent releases of IMS ensure that the correct log record DSECTs associated with an IMS release are used and that the parameters which were defined for the prior release are not invalidated or superseded by the installation of a new release.

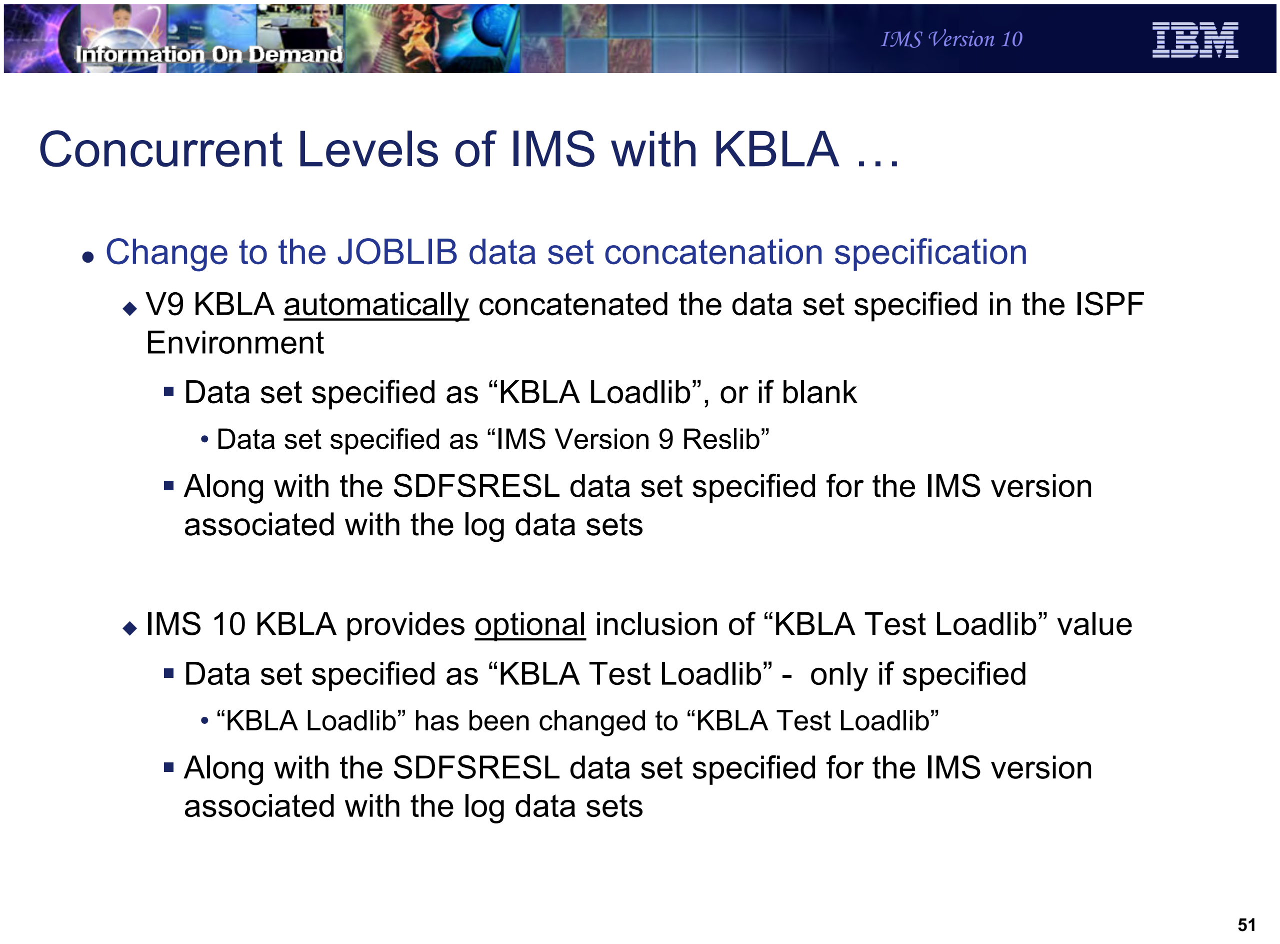
Concurrent Levels of IMS with KBLA

- SDFSRESL contains the KBLA utility modules
- Support for selection of IMS level-related SDFSRESL data set
    - Default Parameter Maintenance Panel (0.1) allows entry of four data sets
        - Entries specifying version 9 or version 10 can be selected for processing
            - Every KBLA processing panel has a parameter where the user can specify the desired IMS level

Figure 15-49. Concurrent Levels of IMS with KBLA CMA01.0

Notes:

Users specify the appropriate SDFSRESL library for an IMS version in the “Default Parameter Maintenance” panel (Option 0.1). An audit in KBLA restricts the specifiable version number to either 9 or 10. Different SDFSRESL libraries for a version can be specified such as 9, 91, 10, 101, etc. KBLA attempts to find the best match of the IMS version specified on a processing panel to the SDFSRESL data set supplied on the Option 0.1 panel.


Information On Demand
IMS Version 10
IBM

Concurrent Levels of IMS with KBLA …

- Change to the JOBLIB data set concatenation specification
  - V9 KBLA automatically concatenated the data set specified in the ISPF Environment
    - Data set specified as “KBLA Loadlib”, or if blank
      - Data set specified as “IMS Version 9 Reslib”
    - Along with the SDFSRESL data set specified for the IMS version associated with the log data sets
  - IMS 10 KBLA provides optional inclusion of “KBLA Test Loadlib” value
    - Data set specified as “KBLA Test Loadlib” - only if specified
      - “KBLA Loadlib” has been changed to “KBLA Test Loadlib”
    - Along with the SDFSRESL data set specified for the IMS version associated with the log data sets

Figure 15-50. Concurrent Levels of IMS with KBLA … CMA01.0

Notes:

With IMS V9, if a value was not specified for the 'KBLA Loadlib‘ parameter in the KBLA ISPF Option 0.1 panel (“IMS K.B.L.A - Define KBLA Environment), KBLA defaulted to the specification associated with IMS V9 resource library. KBLA always included this data set as the first DSN in the JOBLIB DD concatenation sequence.

V10 renames the variable from “KBLA Loadlib” which previously implied that a value was required, to “KBLA Test Loadlib” which is intended to imply that a value is optional and perhaps only used for testing purposes. If specified, the associated data set will be included as the first DSN in the JOBLIB concatenation sequence.

Migration to V10 with KBLA

- Migration Panel
  - Useful when migrating from IMS V9 KBLA
  - Automatically invoked only on initial entry to the IMS V10 Level KBLA ISPF environment after migration
  - Purpose
    - Ensures that the IMS V10 SDFSRESL data set is defined as a variable for the KBLA ISPF environment
    - Establishes the environment

Figure 15-51. Migration to V10 with KBLA CMA01.0

Notes:

In order to facilitate migration from prior IMS releases, a migration panel is provided. This panel will automatically be invoked if the user has not explicitly supplied a V10 SDFSRESL, which is identified as such on the Option 0.1 panel.

Existing IMS V9 KBLA users should note that their previous user-set parameter values are retained during the migration and any new parameters that have been added to the IMS V10 KBLA environment are implemented as ‘optional’ parameters.

Migration to V10 with KBLA …

- Migration Panel Function
  - Sets the SDFSRESL data set name to *hlq*.SDFSRESL
    - Provides ability to specify actual name to be used
  - Performs maintenance on the “KBLA Test Loadlib” variable:
    - Automatically nullifies the value if it matches the V9 SDFSRESL data set name
    - By default, retains the parameter value if different from the V9 SDFSRESL data set name
    - Allows the user to manually
      - Nullify the parameter, or
      - Change it to any desired value

Figure 15-52. Migration to V10 with KBLA … CMA01.0

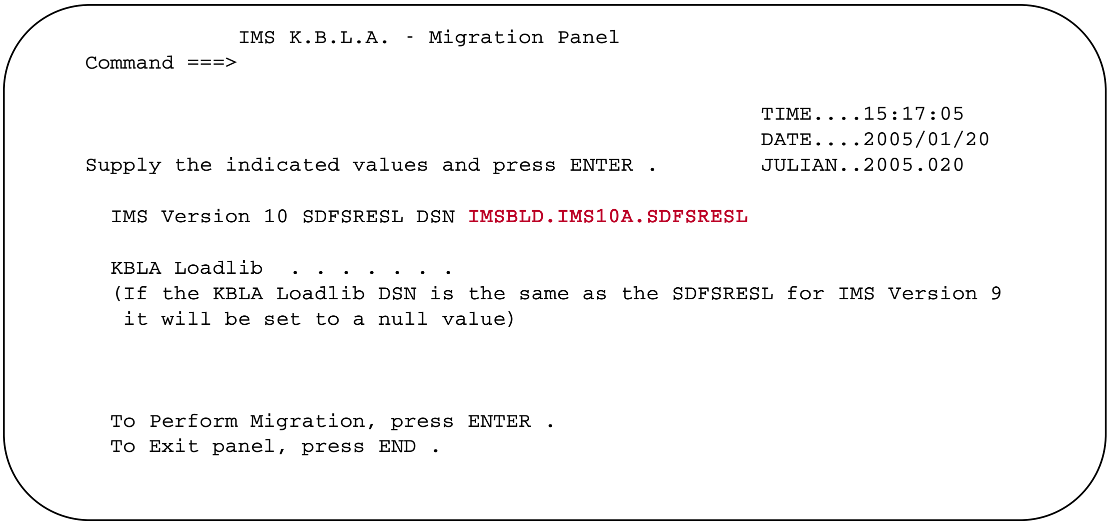
Notes:

If not specified, the V10 SDFSRESL name is set to *hlq*.SDFSRESL where *hlq* is the high level qualifier for the environment.

Additionally, the panel automatically propagates the IMS V9 level “KBLA Loadlib” value into “KBLA Test Loadlib”. The user can nullify or set the value to a different name. If the panel determines that the ‘KBLA Loadlib’ data set is the same data set as the existing IMS V9 data set, this value will be removed. The panel will provide an opportunity to enter another data set name as the ‘KBLA Loadlib’, or this variable may be left blank. When the JCL for KBLA utilities is generated, the JOBLIB will contain only the SDFSRESL data set name for the selected IMS version if the ‘KBLA Loadlib’ variable contains blanks.

An Example - Migration …

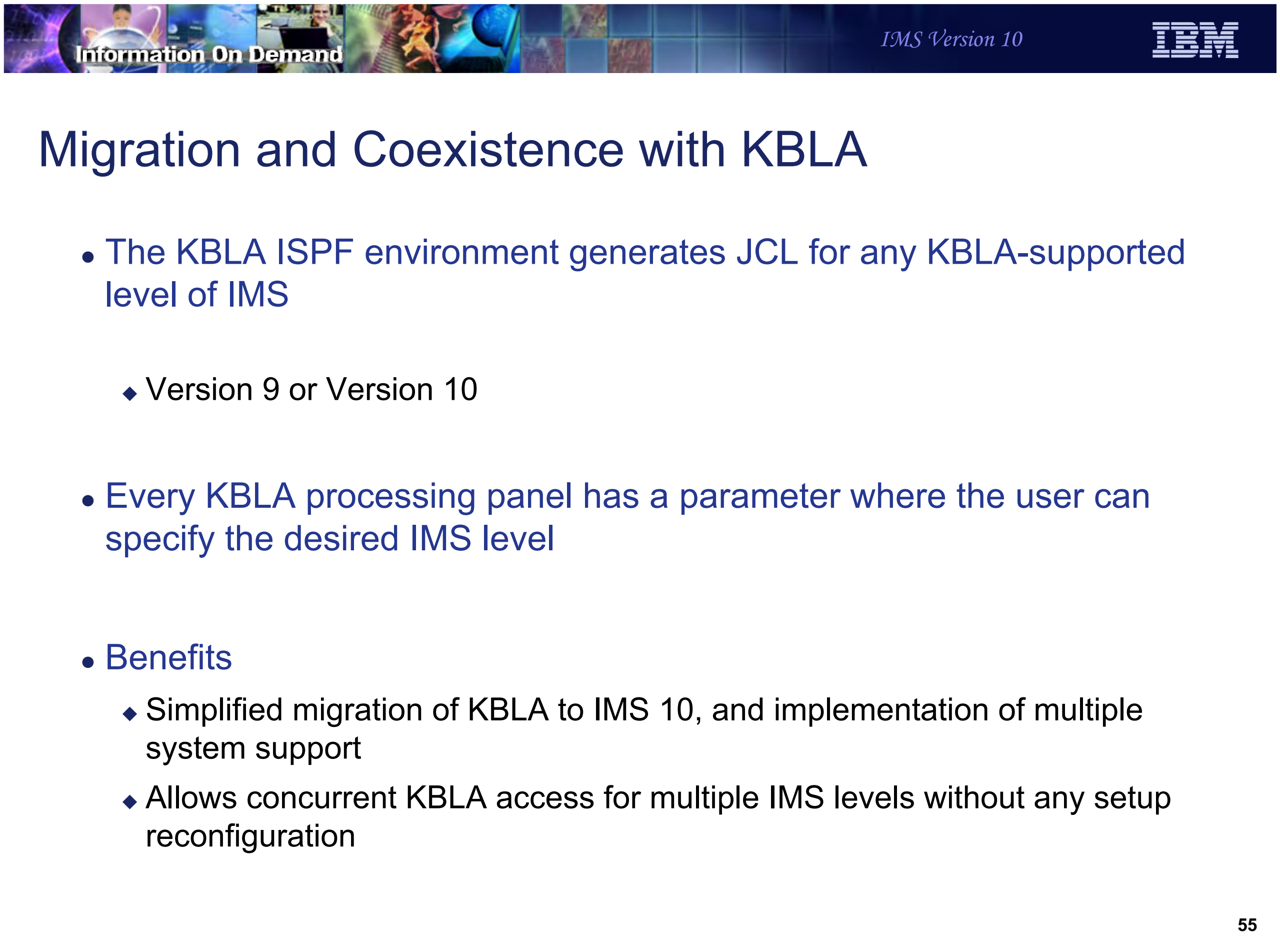
- Migration Panel



```
            IMS K.B.L.A. - Migration Panel
Command ===>


                                                     TIME....15:17:05
                                                     DATE....2005/01/20
Supply the indicated values and press ENTER .        JULIAN..2005.020

  IMS Version 10 SDFSRESL DSN IMSBLD.IMS10A.SDFSRESL

  KBLA Loadlib  . . . . . . .
  (If the KBLA Loadlib DSN is the same as the SDFSRESL for IMS Version 9
   it will be set to a null value)



  To Perform Migration, press ENTER .
  To Exit panel, press END .
```

Figure 15-53. An Example - Migration … CMA01.0

Notes:


Information On Demand
IMS Version 10
IBM

Migration and Coexistence with KBLA

- The KBLA ISPF environment generates JCL for any KBLA-supported level of IMS
  - Version 9 or Version 10
- Every KBLA processing panel has a parameter where the user can specify the desired IMS level
- Benefits
  - Simplified migration of KBLA to IMS 10, and implementation of multiple system support
  - Allows concurrent KBLA access for multiple IMS levels without any setup reconfiguration

Figure 15-54. Migration and Coexistence with KBLA CMA01.0

Notes:

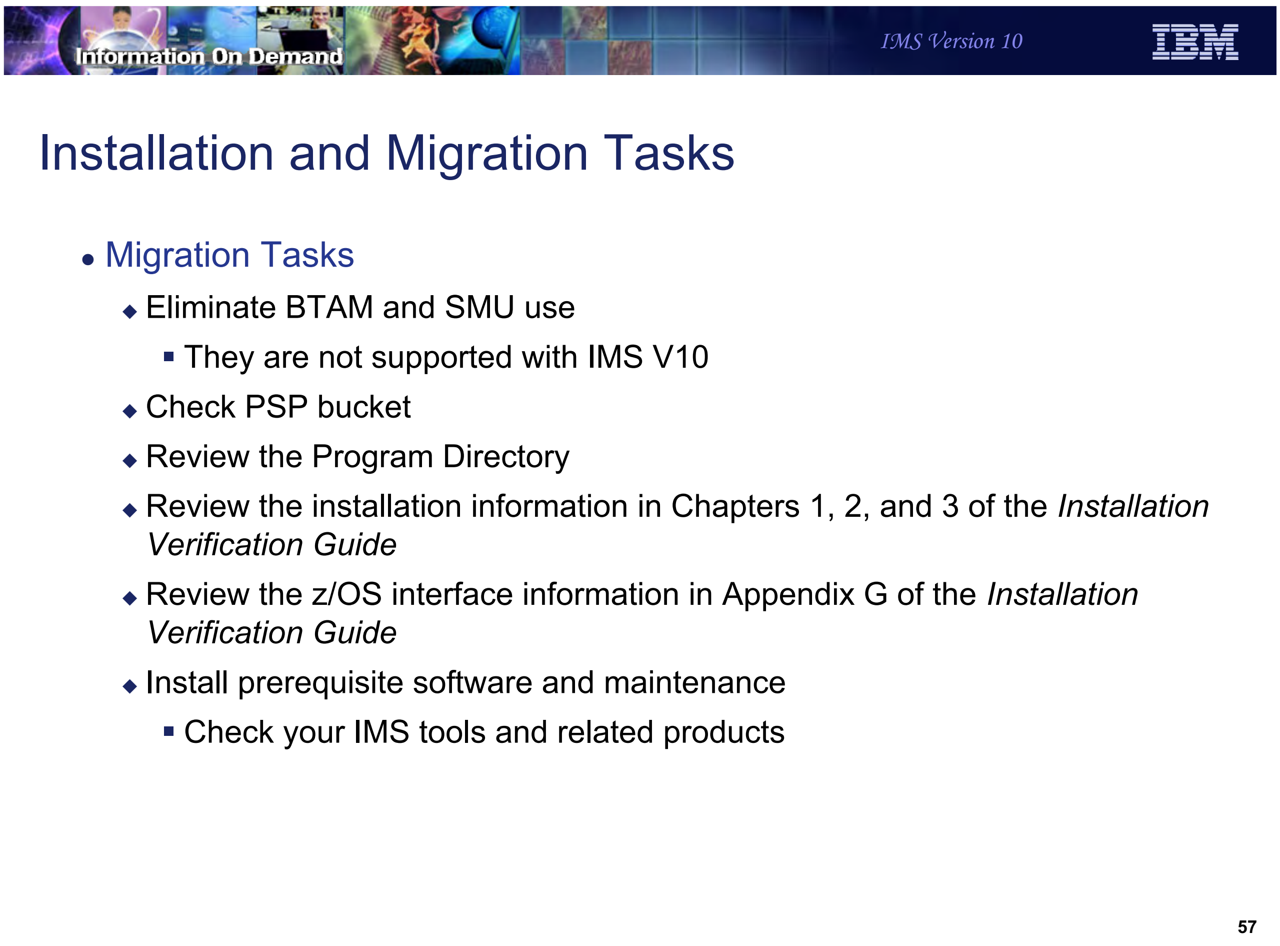
KBLA uses the IMS version information on the processing panel to create the JCL appropriate to the log data sets that are being processed.

Information On Demand
IMS Version 10
IBM

Installation and Migration Tasks

Figure 15-55. [No title] CMA01.0

Notes:


Information On Demand
IMS Version 10
IBM

Installation and Migration Tasks

- Migration Tasks
  - Eliminate BTAM and SMU use
    - They are not supported with IMS V10
  - Check PSP bucket
  - Review the Program Directory
  - Review the installation information in Chapters 1, 2, and 3 of the *Installation Verification Guide*
  - Review the z/OS interface information in Appendix G of the *Installation Verification Guide*
  - Install prerequisite software and maintenance
    - Check your IMS tools and related products

Figure 15-56. Installation and Migration Tasks CMA01.0

Notes:

This is an overview of the tasks for migration to IMS V10.

Remember that IMS V10 is the first release of IMS not to support BTAM or the Security Management Utility (SMU). Their use should be discontinued before migrating to IMS V10.

As for all installations of new products the Preventive Service Planning (PSP) bucket and the Program Directory for the product should be reviewed before beginning the migration.

The IMS V10 *Installation Verification Guide* has content similar to the *Installation Volume 1: Installation Verification* manual for IMS V9. You should read this Guide before beginning the migration process. Chapters 1, 2, and 3 and Appendix G should be reviewed for installation information.

Other products may need to be upgraded for use with IMS V10. They could require maintenance or new releases.

Information On Demand
IMS Version 10
IBM

Installation and Migration Tasks

- Migration Tasks (continued)
  - Apply coexistence maintenance to other IMS systems
    - DBRC coexistence SPEs
    - System Management Enhancements coexistence SPEs
    - Global Online Change coexistence SPEs
    - Operations Management coexistence SPEs
  - Evaluate and update IMS exit routines
    - RECON I/O Exit Routine (DSPCEXT0)
  - Install IMS V10 using SMP/E installation process
    - CBPDO or ServerPac may be used
  - Run the IVP

Figure 15-57. Installation and Migration Tasks CMA01.0

Notes:

You should apply DBRC coexistence SPEs to your IMS V8 or V9 systems before upgrading your RECONs to IMS V10. This is required for the IMS V8 or V9 systems to be able to use the RECONs after the upgrade.

Similarly, you should apply the other coexistence SPEs to your lower level IMS systems.

The only user exit routine that must be updated for use with IMS V10 is the RECON I/O Exit Routine (DSPCEXT0). If you use this routine, you should examine it for required changes due to the change in RECON records.

CBPDO is Custom-Built Product Delivery Offering. The CBPDO product package consists of one logical tape (multiple volumes). A CBPDO package that includes IMS can also include other products in the same System Release (SREL). CBPDO also provides service for the products included with the product order. The service includes all PTFs available within one week of order fulfillment. All PTFs are identified by one or more SOURCEIDs, including PUTyymm, RSUyymm, SMCREC, and SMCCOR.

ServerPac is an entitled software delivery package. It consists of products and service for which IBM has performed the SMP/E installation steps and some of the post-SMP/E installation steps. To install the package on your system and complete the installation of the software it includes, use the CustomPac Installation Dialog, which is the same dialog used for all CustomPac offerings, including SystemPac® (dump-by-data-set format), ProductPac®, and RefreshPac. For IMS, ServerPac allocates, catalogs, and loads all the data sets; sets up the SMP/E environment; supplies a job to update PARMLIB (IEFSSNxx, PROGxx, IEASVCxx, and SCHEDxx) ; and directs you to start the IVP

Running the IVP is optional, but recommended. All required installations tasks are done outside of the IVP. The IVP verifies that the installation is correct.

Installation and Migration Tasks

- Migration Tasks (continued)
  - System definition
  - Install the Type 2 and Type 4 SVCs
  - Review IMS Resource Cleanup Module considerations
    - IMS V10 uses dynamic resource cleanup module
    - Do not delete the IMS V8 module until IMS V8 use is discontinued
  - Upgrade RECONs
  - ACBGEN

Figure 15-58. Installation and Migration Tasks CMA01.0

Notes:

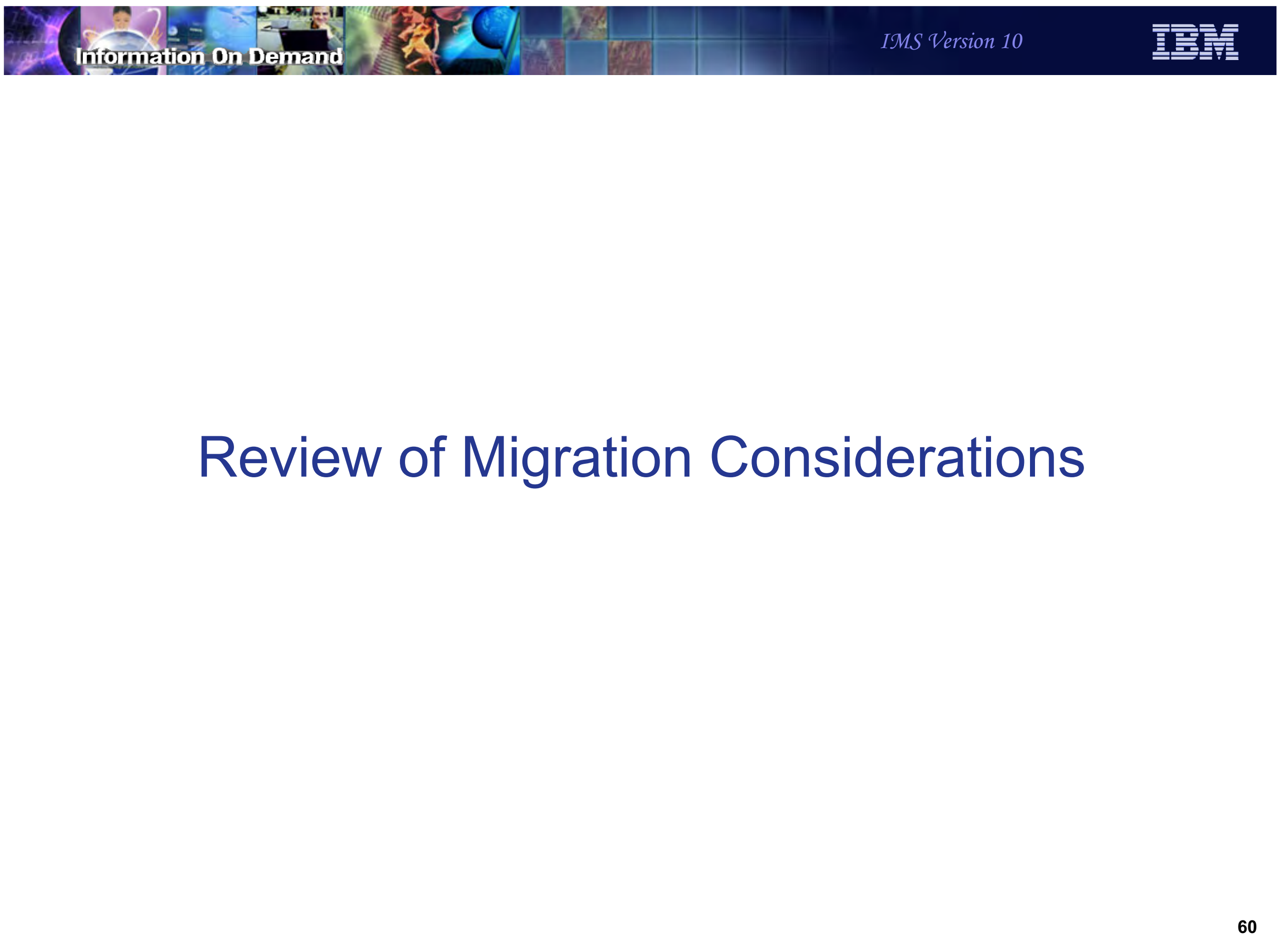
System definition is required as with previous IMS releases. Most system definition statements from previous IMS releases are compatible with IMS V10. BTAM definitions are no longer allowed.

System definition creates the Type 2 and Type 4 SVC modules which must be installed in the z/OS system. A z/OS IPL is not required. They may be installed by running DFSUSVC0 and specifying SVCTYPE=(2,4).

In IMS Versions 9 and 10 use a dynamic resource cleanup module (DFSMRC20). No user setup is required; you do not need to install the static resource cleanup module (DFSMRCL0) on the host z/OS system for IMS V9 or V10. IMS V8 uses the static resource cleanup module (DFSMRCL0). Do not delete it from your z/OS system while there is still a possibility that you will need to run IMS V8.

Upgrade the RECONs by using the CHANGE.RECON UPGRADE command using the IMS V10 release of the DBRC utility.

An ACBGEN is required for use with the online system or any batch DBB jobs.


Information On Demand
IMS Version 10
IBM

Review of Migration Considerations

Figure 15-59. [No title] CMA01.0

Notes:

Information On Demand IMS Version 10 IBM

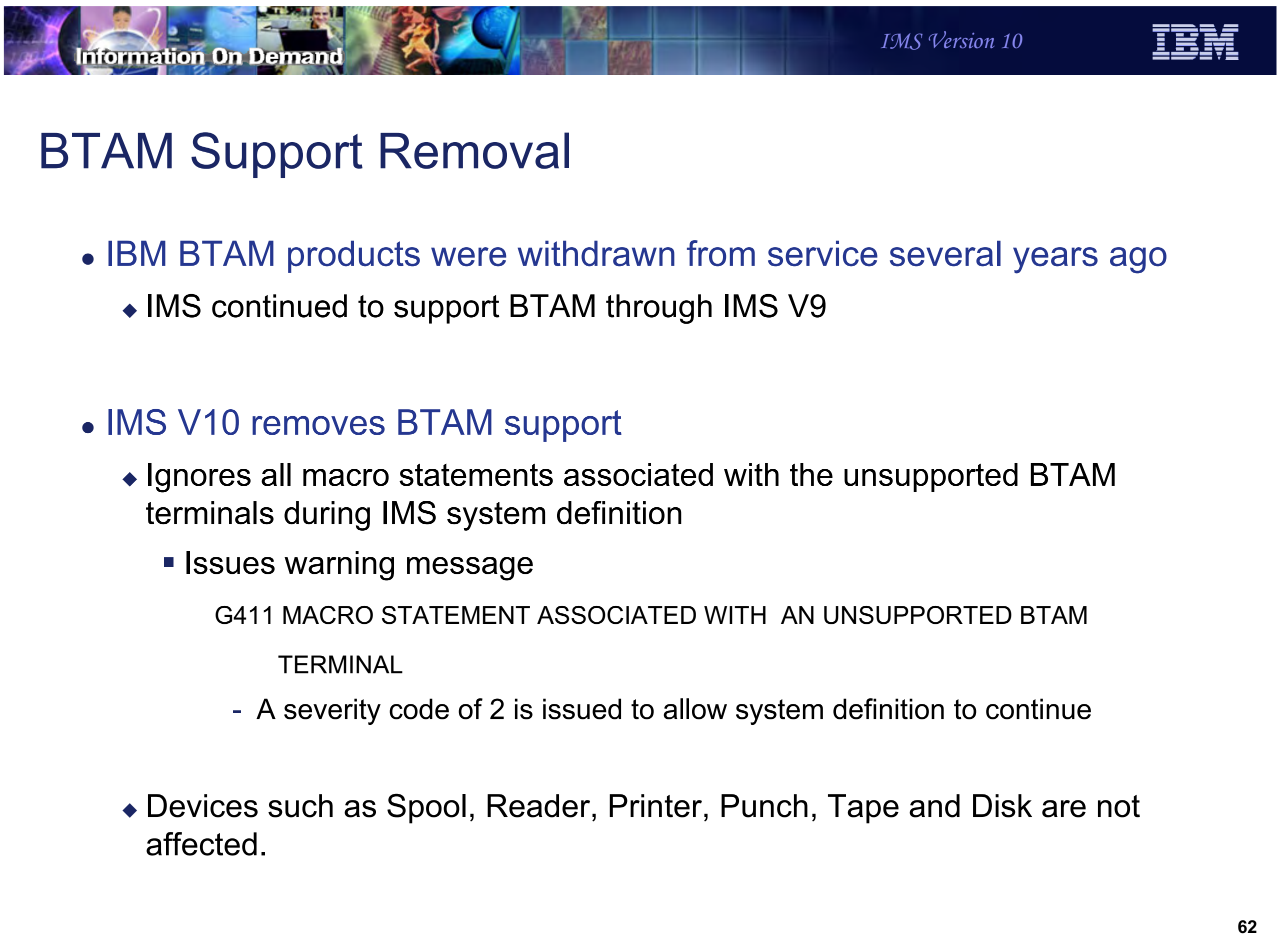
Migration Considerations

- This section repeats information previously covered in the class
  - These are changes that might affect your migration
    - Changed messages
    - Changed responses
    - Changed parameters
    - Changed control statements
    - …

Figure 15-60. Migration Considerations CMA01.0

Notes:

The following pages repeat information that appears elsewhere in this class. This is a collection of information about migration that was previously presented in the class. The information concerns changes in IMS V10 that might require adjustments when you migrate without taking advantage of new optional capabilities.



BTAM Support Removal

- IBM BTAM products were withdrawn from service several years ago
  - IMS continued to support BTAM through IMS V9
- IMS V10 removes BTAM support
  - Ignores all macro statements associated with the unsupported BTAM terminals during IMS system definition
    - Issues warning message

      G411 MACRO STATEMENT ASSOCIATED WITH AN UNSUPPORTED BTAM TERMINAL

      - A severity code of 2 is issued to allow system definition to continue
  - Devices such as Spool, Reader, Printer, Punch, Tape and Disk are not affected.

Figure 15-61. BTAM Support Removal CMA01.0

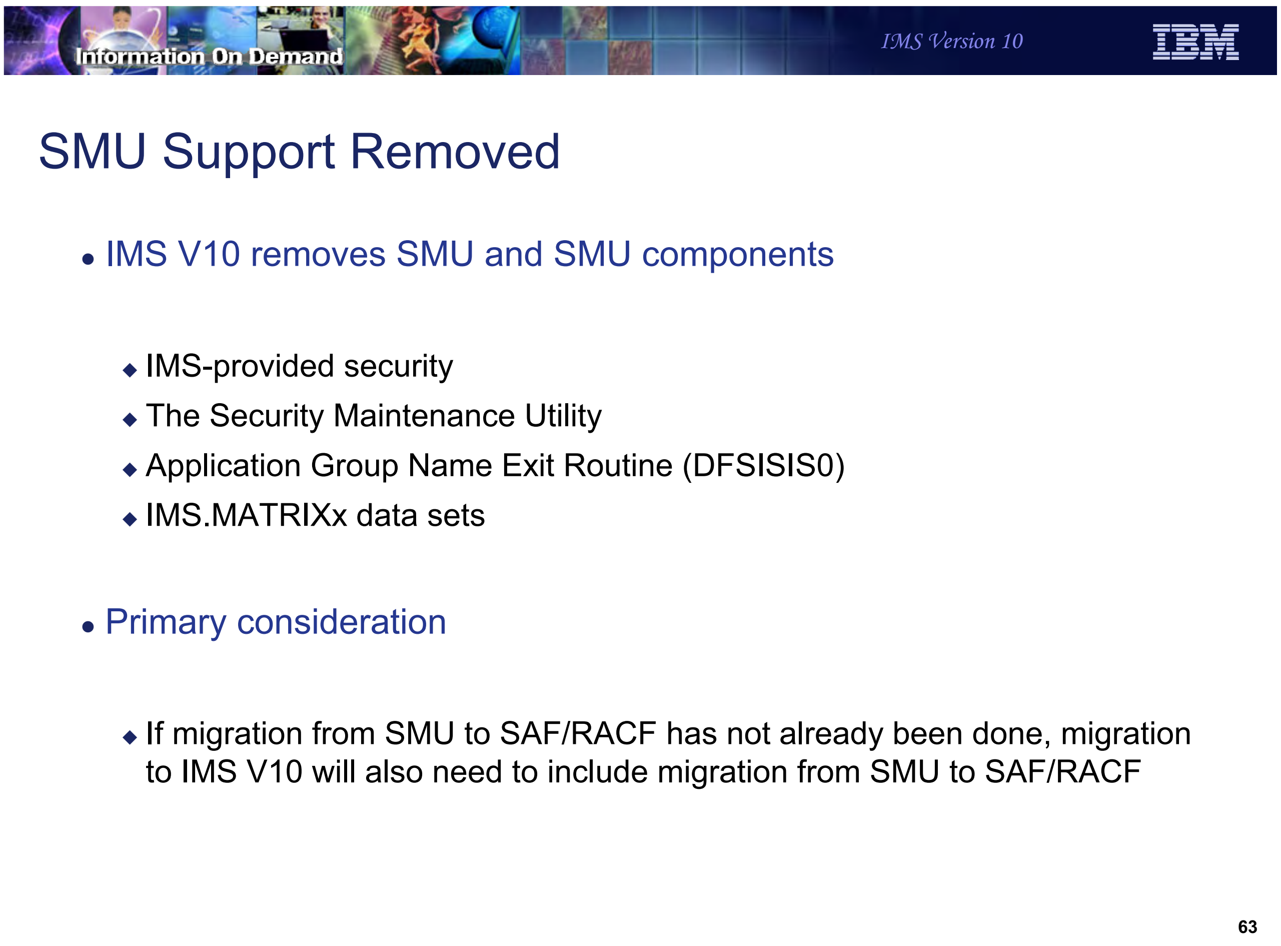
Notes:

Although IBM withdrew marketing and service of BTAM products several years ago, IMS continued to support the BTAM macros through IMS V9. IMS V10 removes this support. The following list shows the device types affected:

BTAM Device Type	Comments or Other Specifications
1050	Switched Terminal
2740	Non-Station-Control
2740	Non-switched, model 1
2740	Non-switched, model 2
2740	Switched Terminal, model 1
2741	Non-switched
2741	Switched Terminal

2260	Local
2780	
3270	Remote, Non-switched
3270	Local
3270	Switched Terminal
3275	Switched Terminal
3741	Switched Terminal
SYSTEM/3	
SYSTEM/7	BSC, BSC and Contention
SYSTEM/7	Start/Stop, Start/Stop and Contention

Warning message G411 will be issued if the macro statement operand has an unsupported BTAM terminal specification during the IMS STAGE 1 system definition process. In addition, a severity code of 2 will be issued to allow system definition to continue. This warning message and severity code will be documented in the IMS V10 Messages and Codes manual.



Information On Demand
IMS Version 10
IBM

SMU Support Removed

- IMS V10 removes SMU and SMU components
  - IMS-provided security
  - The Security Maintenance Utility
  - Application Group Name Exit Routine (DFSISIS0)
  - IMS.MATRIXx data sets
- Primary consideration
  - If migration from SMU to SAF/RACF has not already been done, migration to IMS V10 will also need to include migration from SMU to SAF/RACF

Figure 15-62. SMU Support Removed CMA01.0

Notes:

IMS V10 no longer supports the Security Maintenance Utility (SMU) capability. As a result, SMU components have been removed including the SMU Utility, the AGN Exit Routine (DFSISIS0) and the MATRIX data sets. Note that the IMS.MATRIXx data sets have been deleted and removed from all IMS procedures and logic.

A primary consideration in this area includes migration. IMS systems that use SMU must migrate to the SAF interface at the time that the upgrate to IMS V10 occurs.

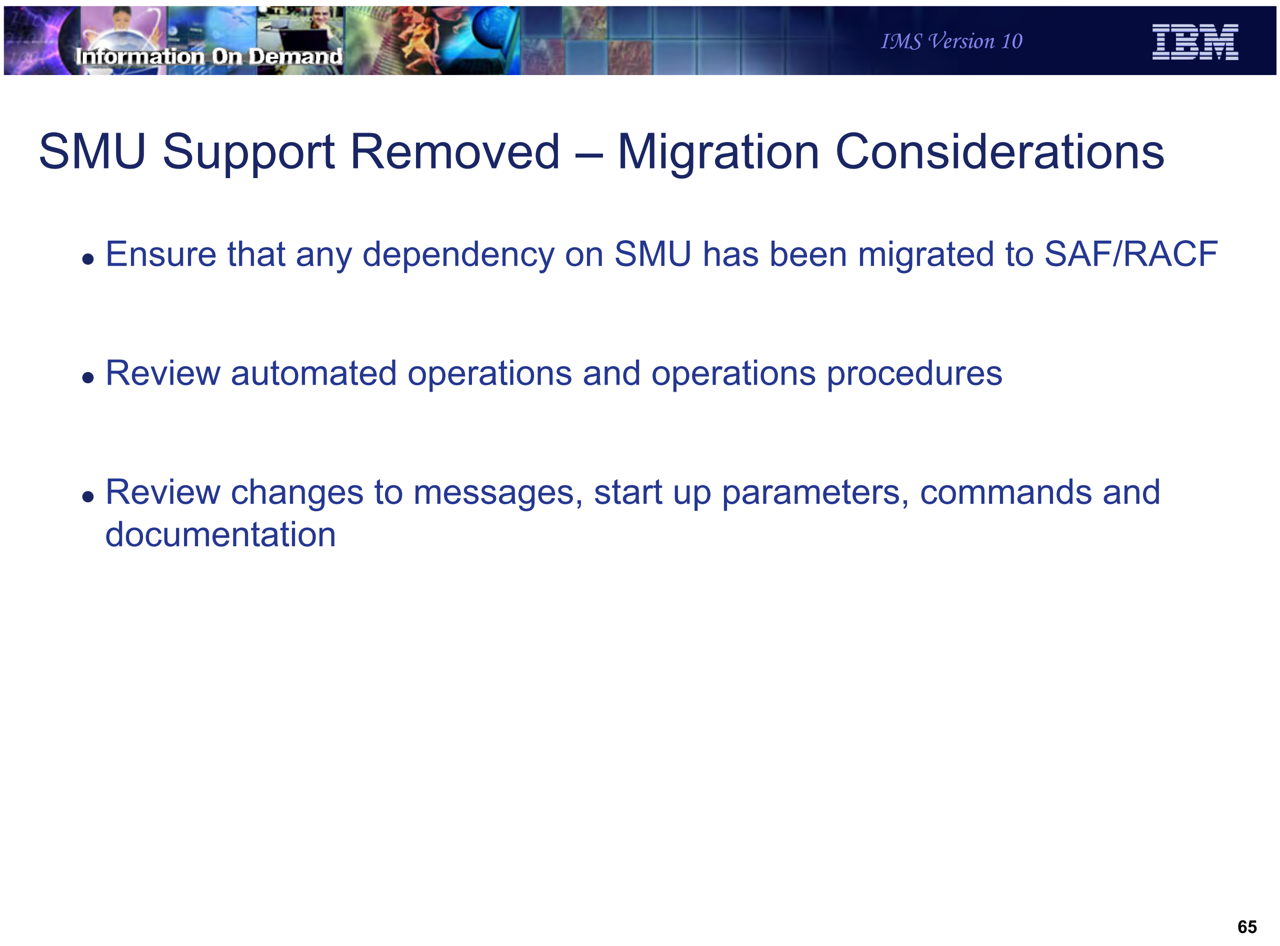
SMU to SAF/RACF Migration

- Migration to SAF (RACF or equivalent) can begin in IMS V8 or IMS V9
  - V9 includes RACF enhancements for SMU functions which previously had no alternatives
    - Enhancements include areas of AGN, AOI, TCO, MSC link receive, signon verification, and /LOCK and /UNLOCK
    - V9 is a transition release where SMU is still available
  - Migration from SMU to SAF and from IMS V8 to IMVS V10 is a valid option
    - Begin SMU to SAF migration in IMS V8
    - Migration using V9 enhancements will have to occur during the V10 migration
- Migration references
  - Chapter 6, IMS V9 Implementation Guide redbook (SG24-6398)
  - Chapter 4, IMS V9 Administration Guide: System (SC18-7807)
  - Chapter 3 and 4, IMS V9 Release Planning Guide (GC17-7831)

Figure 15-63. SMU to SAF/RACF Migration CMA01.0

Notes:

Both IMS V8 and V9 provide migration paths to the use of SAF/RACF security. IMS V9 provides migration capabilities that are not available with IMS V8.


Information On Demand
IMS Version 10

SMU Support Removed – Migration Considerations

- Ensure that any dependency on SMU has been migrated to SAF/RACF
- Review automated operations and operations procedures
- Review changes to messages, start up parameters, commands and documentation

Figure 15-64. SMU Support Removed – Migration Considerations CMA01.0

Notes:

Transaction Scheduling

- SCHD= parameter on system definition TRANSACT macro is ignored
- Only one scheduling option is used in IMS V10
  - SCHD=3 option is always used
    - When the transaction defined on the TRANSACT macro cannot be scheduled for "internal reasons", any transaction in the selected class may be scheduled. If no more transactions for this class exist, IMS schedules transaction in next eligible class.
      - "Internal reasons" are database intent or no more space in PSB pool or DMB pool to bring in needed blocks
  - SCHD=1 was the default in previous releases
    - When the transaction defined on the TRANSACT macro cannot be scheduled for internal reasons, only transaction of equal or higher priority in the selected class may be scheduled. If five intent failures occur within a class, transactions in the next class are attempted

Figure 15-65. Transaction Scheduling CMA01.0

Notes:

In previous release of IMS, the SCHD= parameter on the system definition TRANSACT macro specified the scheduling option used when the transaction defined on the TRANSACT macro could not be scheduled for internal reasons (database intent or no more space in PSB pool or DMB pool to bring in needed blocks). IMS V10 always uses the SCHD=3 option. The default in previous releases was SCHD=1.

SCHD=1 specified that only transactions of equal or higher priority in the selected class would be scheduled. Five consecutive intent conflicts are allowed within a class before IMS starts scheduling the next eligible class.

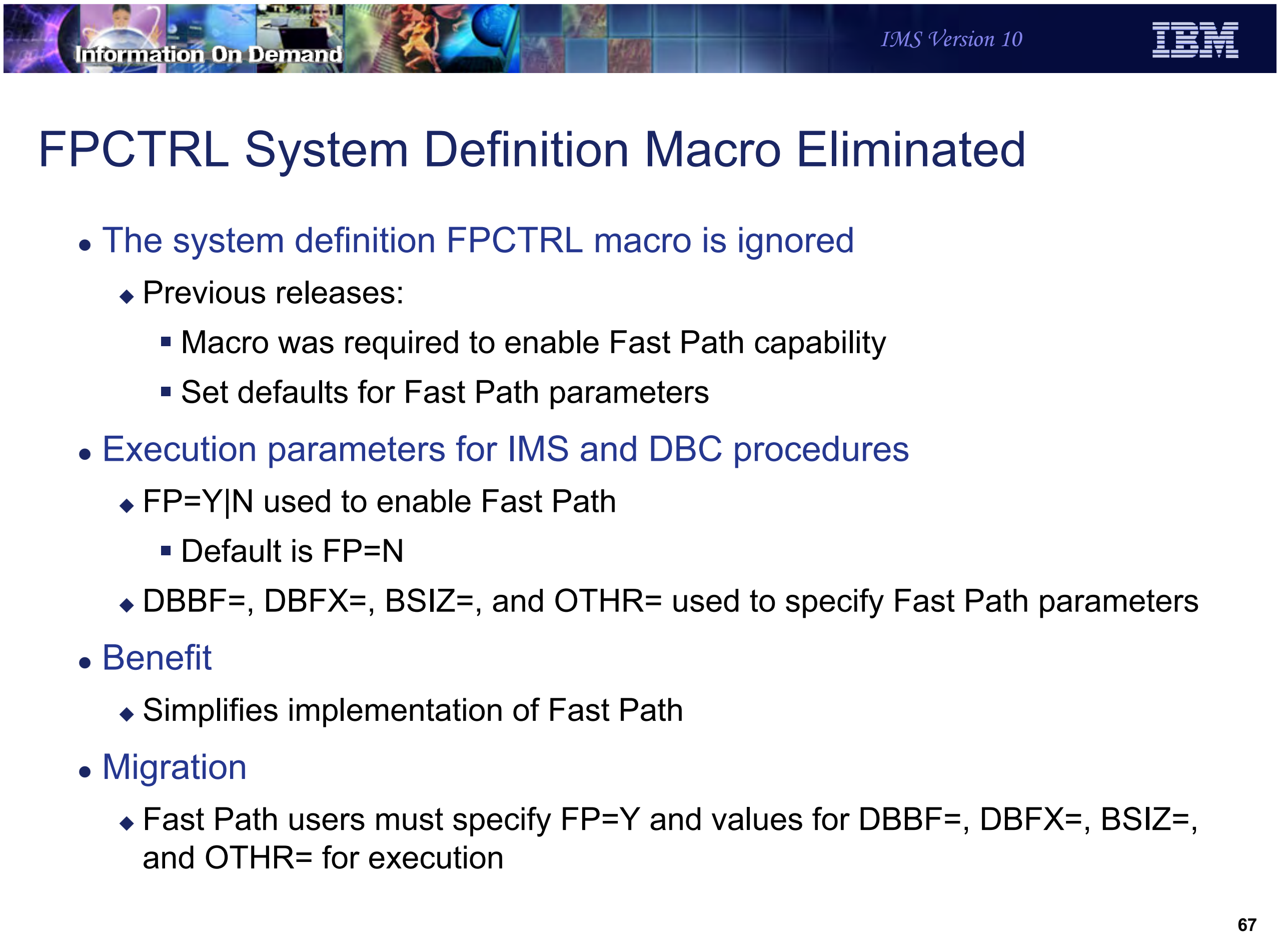
SCHD=3 specified that any transaction in the selected class could be scheduled. IMS starts scheduling the next eligible class after attempting to schedule all the transactions in the current class

Two other options were available in previous releases.

SCHD=2 specified that only higher-priority transactions in the selected class could be scheduled.

SCHD=4 specified that IMS should skip to the next class and attempt to schedule the highest-priority transaction in that class.

Scheduling failures for the "internal reasons" should be rare. Intent conflicts only occur when PROCOPTs with the E (exclusive) option are used. The PSB and DMB pools should always be large enough to hold the currently required PSBs and DMBs.



Information On Demand
IMS Version 10
IBM

FPCTRL System Definition Macro Eliminated

- The system definition FPCTRL macro is ignored
  - Previous releases:
    - Macro was required to enable Fast Path capability
    - Set defaults for Fast Path parameters
- Execution parameters for IMS and DBC procedures
  - FP=Y|N used to enable Fast Path
    - Default is FP=N
  - DBBF=, DBFX=, BSIZ=, and OTHR= used to specify Fast Path parameters
- Benefit
  - Simplifies implementation of Fast Path
- Migration
  - Fast Path users must specify FP=Y and values for DBBF=, DBFX=, BSIZ=, and OTHR= for execution

67

Figure 15-66. FPCTRL System Definition Macro Eliminated CMA01.0

Notes:

IMS V10 has simplified the implementation of Fast Path. It does not have to specified at system definition time. The FPCTRL system definition macro is ignored. In previous releases it was required to enable Fast Path capabilities. It also was used to specify the default values for some Fast Path parameters. Fast Path capabilities are always generated for DB/DC and DBCTL systems. They are enabled by a parameter at execution time.

You must specify FP=Y at execution time for an online system to enable Fast Path capabilities in IMS V10. The default for the parameter is FP=N. Since defaults for Fast Path parameters cannot be specified at system definition time, they should be specified at execution time. These parameters are DBBF=, DBFX=, BSIZ=, and OTHR=. The default values for these parameters are the same as the default values for the FPCTRL macro in previous releases. These are:

DBBF=10

DBFX=4

BSIZ=2048

OTHR=2

These values are highly unlikely to be appropriate for most Fast Path users.

Information On Demand
IMS Version 10
IBM

DSP1929I System Startup Message

- Changes in DFS1929I IMS system startup message
  - New header information
    - IMS version and Control Region type
  - Parameters which do not apply to the system are eliminated
  - Sent two times
    - First: specified parameters
    - Second: used parameters
      - After reading the log for restart

Figure 15-67. DSP1929I System Startup Message CMA01.0

Notes:

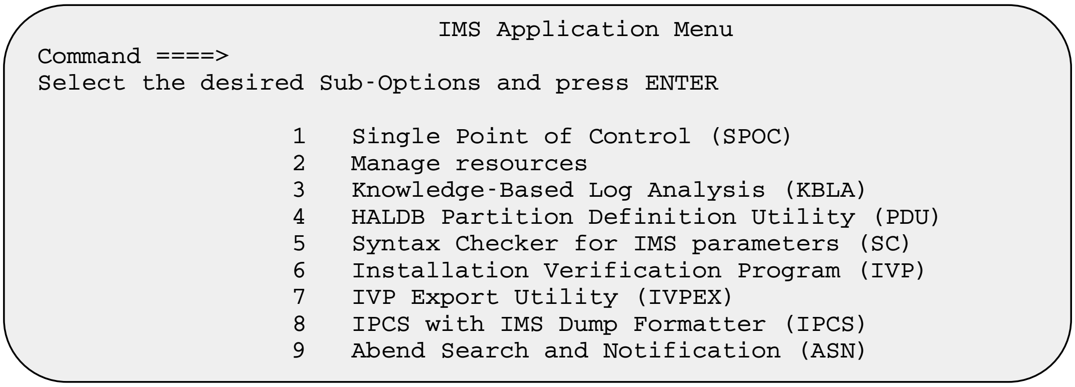
When migrating to IMS V10, user-written routines or procedures that look for the DFS1929I message should recognize the header format change to include the IMS version and Control Region type. Additionally, these routines should be modified to recognize that there are two versions of the message.

Information On Demand

IMS Version 10

IMS Application Menu

- Two options have been added to the IMS Application Menu
  - 2 Manage resources
  - 9 Abend Search and Notification
- Some options have moved in the list



```
                              IMS Application Menu
Command ====>
Select the desired Sub-Options and press ENTER

                  1   Single Point of Control (SPOC)
                  2   Manage resources
                  3   Knowledge-Based Log Analysis (KBLA)
                  4   HALDB Partition Definition Utility (PDU)
                  5   Syntax Checker for IMS parameters (SC)
                  6   Installation Verification Program (IVP)
                  7   IVP Export Utility (IVPEX)
                  8   IPCS with IMS Dump Formatter (IPCS)
                  9   Abend Search and Notification (ASN)
```

Figure 15-68. IMS Application Menu CMA01.0

Notes:

The IMS Application Menu has been modified to accommodate two new options. "Manage Resources" is option 2. This changed the option numbers for the items that follow it. "Abend Search and Notification" has been added as option 9.

Information On Demand
IMS Version 10
IBM

PSB Sizes Increase

- PSB PCB sizes increase by 16 bytes in IMS V10
  - May require larger PSB pools
    - CSAPSB and DLIPSB pools
  - Resident PSBs will require more space
  - Estimated size increase: about 5% for CSA, about 1% for SAS

Figure 15-69. PSB Sizes Increase CMA01.0

Notes:

ACBGEN Default Change for BUILD DBD=

- Previous IMS versions
    - ACBGEN with BUILD DBD= did not always rebuild PSBs referencing the DBD
        - Only rebuilt PSBs if DBD physical structure changed
- IMS Version 10
    - ACBGEN with BUILD DBD= rebuilds PSBs referencing the DBD by default
        - Done for support of ACBLIB member online change
        - May elongate ACBGEN execution time
    - BLDPSB=NO overrides default
        - PSBs are not rebuilt unless DBD structure is changed
        - May be used for ACBGENs not used for ACBLIB member online change

```
BUILD DBD=(CUSTOMER,ORDER),BLDPSB=NO
```

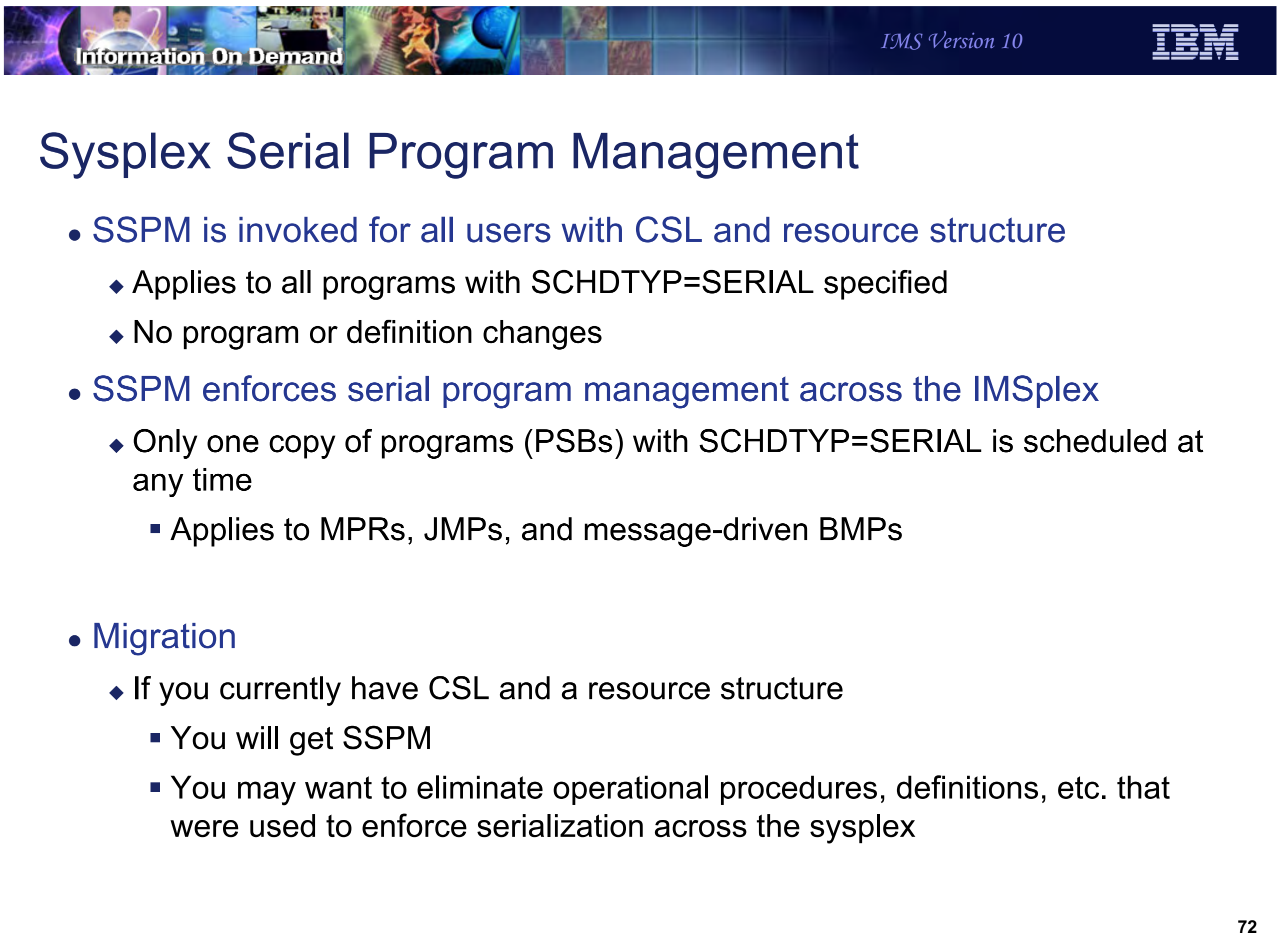
Figure 15-70. ACBGEN Default Change for BUILD DBD= CMA01.0

Notes:

The BLDPSB=YES option is new with IMS V10 and ACBLIB member online change. It indicates that ACBGEN rebuilds all PSBs that reference the changed DBD on the BUILD DBD=dbdname statement into the staging ACBLIB. Member online change requires that all PSBs be rebuilt (flag set in PDIR) or it will fail. For performance reasons, these relationships are found during the execution of ACBGEN in batch versus having to discover these relationships online when ACBLIB member online change is invoked. However, this means there may be additional processing done at ACBGEN time and the ACBGEN utility may run longer. BLDPSB=YES is the default.

In previous IMS versions, ACBGEN worked as if BLDPSB=NO was specified. BLDPSB=NO indicates that ACBGEN does not rebuild PSBs that reference the changed DBD if the changed DBD does not change the physical structure of the DBD. The DBD is rebuilt but PSBs are not necessarily rebuilt (ex. exit/randomizer change). You can still specify BLDPSB=NO if you are not planning to use ACBLIB member online change.

BLDPSB=NO is not the default, so this operates differently than in previous versions. It must be specified if you want this capability.


Information On Demand
IMS Version 10
IBM

Sysplex Serial Program Management

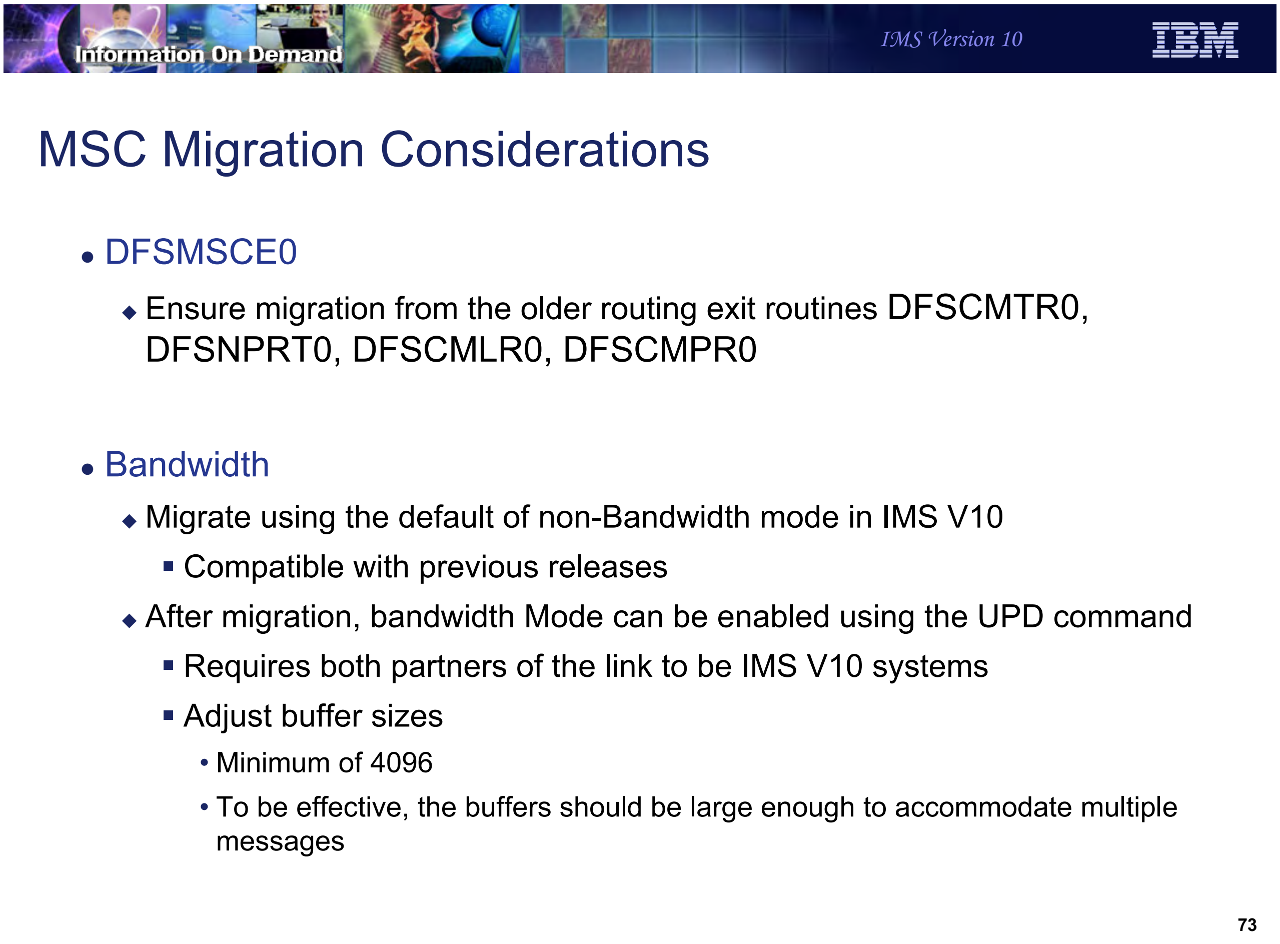
- SSPM is invoked for all users with CSL and resource structure
    - Applies to all programs with SCHDTYP=SERIAL specified
    - No program or definition changes
- SSPM enforces serial program management across the IMSplex
    - Only one copy of programs (PSBs) with SCHDTYP=SERIAL is scheduled at any time
        - Applies to MPRs, JMPs, and message-driven BMPs
- Migration
    - If you currently have CSL and a resource structure
        - You will get SSPM
        - You may want to eliminate operational procedures, definitions, etc. that were used to enforce serialization across the sysplex

Figure 15-71. Sysplex Serial Program Management CMA01.0

Notes:

If you have an IMSplex with the common service layer (SCI, OM, and RM address spaces) with a resource structure, serial program management will automatically be enforced across the IMSplex with IMS V10. There are no program or definitions changes required for this. Only one copy of a program with SCHDTYP=SERIAL will be scheduled in the IMSplex at any time. This applies to MPRs, JMPs, and message-driven BMPs. It does not apply to non-message driven BMPs, JBPs, CCTL (CICS), or ODBA applications such as DB2 Stored Procedures.

If you have implemented the Common Service Layer and defined a resource structure with previous releases of IMS, you will automatically get the SSPM function when you migrate your IMS systems to V10. You probably created some operational procedures and/or definitions to enforce serialization in your previous release. For example, you may have started a dependent region with the class for the transaction on only one IMS system. This is OK with V10, but it is not required for the enforcement of serialization.


Information On Demand
IMS Version 10
IBM

MSC Migration Considerations

- DFSMSCE0
  - Ensure migration from the older routing exit routines DFSCMTR0, DFSNPRT0, DFSCMLR0, DFSCMPR0
- Bandwidth
  - Migrate using the default of non-Bandwidth mode in IMS V10
    - Compatible with previous releases
  - After migration, bandwidth Mode can be enabled using the UPD command
    - Requires both partners of the link to be IMS V10 systems
    - Adjust buffer sizes
      - Minimum of 4096
      - To be effective, the buffers should be large enough to accommodate multiple messages

Figure 15-72. MSC Migration Considerations CMA01.0

Notes:

When migrating to IMS V10, ensure that the older routing exit routines, if any, have been converted to DFSMSCE0. This migration can be accomplished in any of the supported previous releases (V7, V8, V9) prior to the actual IMS V10 migration. If the exit routines still exist while the V10 migration is in progress then the V10 migration tasks need to include the upgrade of the previous releases to DFSMSCE0.

A like-for-like migration from a previous release to IMS V10 allows MSC to be initialized in non-bandwidth mode. This mode is compatible with MSC operations in previous releases.

The ability to turn on bandwidth mode is provided via the UPD command and requires both partners of the link to be at an IMS V10 level. In this mode, the minimum size of the buffers should be 4096. To be more effective, the buffer sizes should be defined to accommodate multiple messages.

MSC Migration Considerations …

- Buffer size
  - MSPLINK buffer size ranges have changed
    - Minimum is1024, maximum is 65,536
      - Increase over IMS V8/V9 definitions
      - VTAM sizes for IMS V10 can be any size and no longer follow an algorithm
    - Validate that partner IMS systems have matching buffer specifications
      - If the IMSV8/V9 systems have link buffer sizes less than 1024, they must be modified to a minimum of 1024

Figure 15-73. MSC Migration Considerations … CMA01.0

Notes:

Note that the buffer size minimum and maximum ranges have changed in IMS V10. If the link is a VTAM link, the buffer size can now be any size defined in the range and no longer requires a size that fits into the previous formula of X times 2 to the power of Y, where X had to be a value from 8 to 15, and Y has to be a value from 3 to 13. . The MSPLINK macro description in the System Definition Guide still documents a table but only for compatibility purposes.

If the partner of the MSC link is a V8 or V9 IMS system then those definitions must be at a minimum of 1024.

OTMA Migration Considerations

- Message flood control
  - Default limit of 5000 is set an initialization
    - To deactivate the function, define the INPUT parameter in the DFSYDTx descriptor or issue the /STA TMEMBER INPUT command
- Time-out
  - Default is set to 5 seconds
    - To deactivate the function, define the T/O parameter in the DFSYDTx descriptor or issue the /STA TMEMBER TIMEOUT command
- /DIS OTMA and /DIS TMEMBER command enhancements
  - Single line output has been expanded to two-line output in order to include DRU exit name info and time-out info.
  - The first column info of /DISPLAY TMEMBER TPIPE is now the input message count instead of output enqueue count.

Figure 15-74. OTMA Migration Considerations CMA01.0

Notes:

The considerations listed on this visual address issues that should be considered when migrating from a previous release of IMS. The assumption is that migration to IMS V10 is based on existing functionality without adding any new capabilities during the migration process.

The message flood control enhancement in IMS V10 is automatically enabled with a default limit of 5000 messages. To provide compatibility with previous releases and deactivate the support, either specify an INPUT value of 0 in a descriptor or issue the /STA TMEMBER INPUT command.

Likewise, the timeout support for synchronous CM1 message is automatically enabled with a default value of 5 minutes. To provide compatibility with previous releases and deactivate the support, specify T/O value of 0 in a descriptor or issuing the /STA TMEMBER TIMEOUT command.

Note that the /DISPLAY command output associated with OTMA and TMEMBER requests has been expanded to two output lines and includes new information. As a migration consideration, this is a key issue for automated operations.

Information On Demand
IMS Version 10
IBM

Log Transaction Analysis, Log Merge, and Statistical Analysis utilities

- Migration
  - Old Log Transaction Analysis utility JCL continues to work
    - Specification of `NOLOG` or `NOREPORT` on EXEC statement ignored
      - These actions are now controlled by the presence or absence of DD statements
  - Log Merge utility continues to work but is not required
    - Multiple logs may be specified as input to the other utilities
  - Old Statistical Analysis utility JCL must be modified
    - Move SYSIN from step 6 to the first step
    - Delete the LOGOUT DD statement from first step
    - Delete the last five steps
  - Logs from previous releases are not valid input to these V10 utilities

Figure 15-75. Log Transaction Analysis, Log Merge, and Statistical Analysis utilities CMA01.0

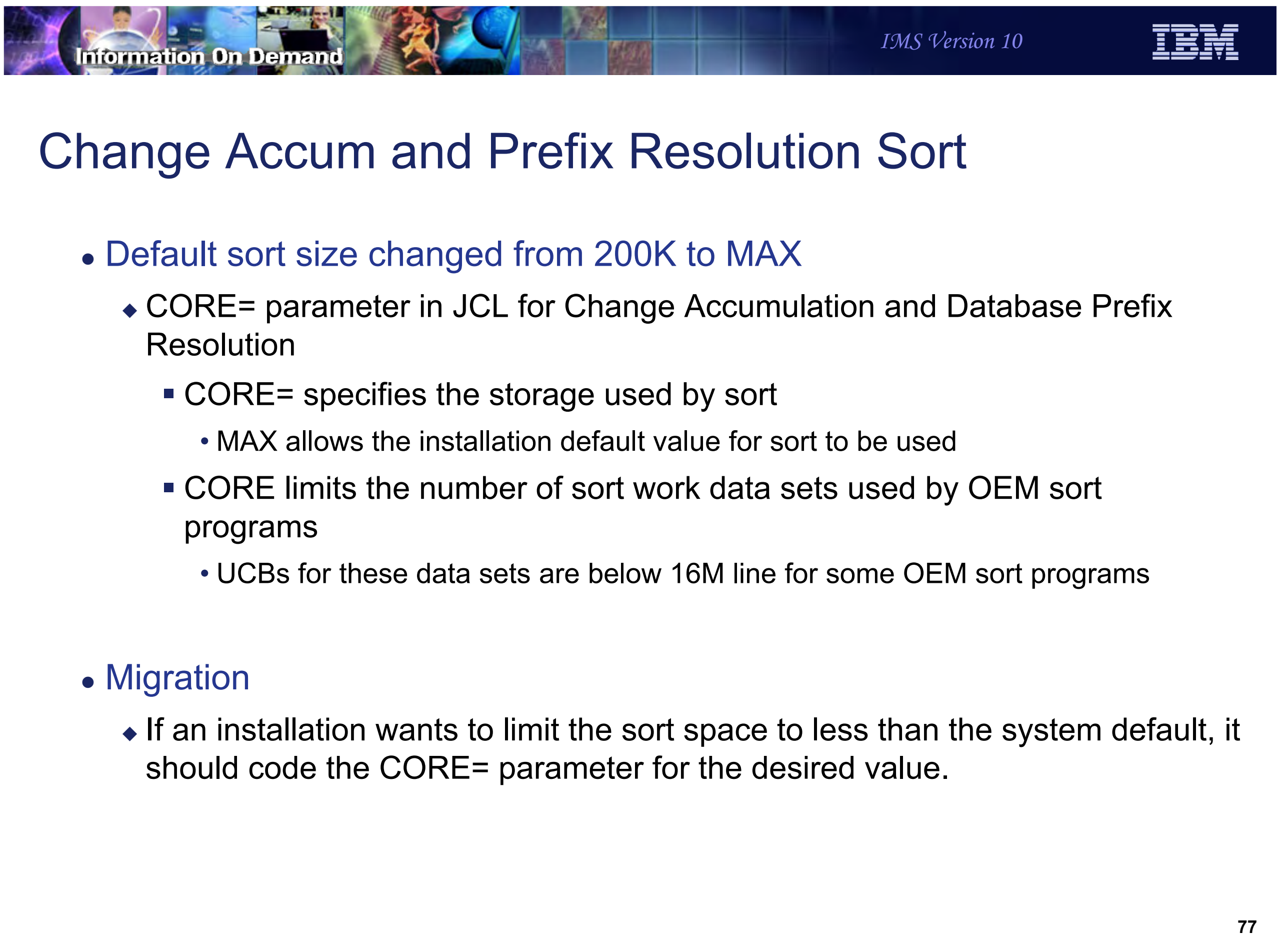
Notes:

The JCL used in previous releases for the Log Transaction Utility can be used with IMS V10. There is one slight incompatibility. Previously you could specify an OUT= parameter on the EXEC statement. OUT=NOLOG specified that an output report should not be written to the LOGOUT DD data set. OUT=NOREPORT specified that an output report should not be written to the PRINTER DD data set. In IMS V10 if you do not want either of these data sets do not include the DD statement for it. IMS V10 ignores the OUT= parameter on the EXEC statement.

The Log Merge utility was used in previous releases to merge logs from different systems. The output of Log Merge was used as input to the Log Transaction Analysis or Statistical Analysis utilities. The Log Merge utility can be used with IMS V10, but it is not required. Both analysis utilities can merge multiple logs as part of their processing.

The JCL used for the Statistical Analysis utilities in previous releases must be changed for IMS V10. The actions needed to change the JCL are listed on the slide. The control statements were specified in the SYSIN data set of the sixth step. The SYSIN data set is now part of the first (and only) step. The LOGOUT DD in the old first step was used to pass data to the following steps. Since only one step is used in IMS V10, the LOGOUT DD is no longer used. The last five steps are no longer used, so their JCL should be eliminated.

Logs produced by IMS V8 and IMS V9 are not valid input to the IMS V10 Log Transaction Analysis utility and the IMS V10 Statistical Analysis utility.


Information On Demand
IMS Version 10
IBM

Change Accum and Prefix Resolution Sort

- Default sort size changed from 200K to MAX
  - CORE= parameter in JCL for Change Accumulation and Database Prefix Resolution
    - CORE= specifies the storage used by sort
      - MAX allows the installation default value for sort to be used
    - CORE limits the number of sort work data sets used by OEM sort programs
      - UCBs for these data sets are below 16M line for some OEM sort programs
- Migration
  - If an installation wants to limit the sort space to less than the system default, it should code the CORE= parameter for the desired value.

Figure 15-76. Change Accum and Prefix Resolution Sort CMA01.0

Notes:

The default for the CORE= parameter on the Database Change Accumulation and Database Prefix Resolution utilities is changed in IMS V10 from 200K to MAX. When MAX is used, the IMS utility does not limit the amount of space used by sort. It is limited by the default value specified for the sort product. The sort space is particularly important for some non-IBM sort programs which create UCBs for sort work data sets below the 16M line. The sort space may limit the number of these data sets that may be used. By increasing the default space, the number of sort work data sets may be increased. This may shorten the elapsed execution time for these utilities.

A new message has been added to the Database Change Accumulation and Database Prefix Resolution utilities. It is issued when the CORE= parameter is not specified. The message is:

```
DFS3007I SORT CORE SIZE WILL DEFAULT TO CORE=MAX
```

If an installation wants to limit the sort space to less than the system default, it should code the CORE= parameter for the desired value.

Information On Demand
IMS Version 10

DBRC Timestamp Precision

- Migration Considerations
  - Full precision used after MINVERS('10.1') specified
    - Fallback to MINVERS < 10.1 requires deletion of RECON log records
  - Utility control statement changes are unlikely to affect users
    - GENJCL.CA and GENJCL.RECOV create correct statements
      - Used by almost all users
  - RECON listings and message outputs include full precision timestamps when MINVERS('10.1')
  - Abbreviated timestamps supported for most uses
  - No changes for DBRC API
    - Timestamps continue to be 12 byte internal timestamp format
  - Timestamp precision of GENJCL.USER output could change
    - Default changes when MINVERS('10.1') is set unless overridden in skeletal JCL

Figure 15-77. DBRC Timestamp Precision CMA01.0

Notes:

Full precision timestamps are not implemented unless the RECONs have MINVERS('10.1') specified. Even when MINVERS is set to a lower value, the IMS V10 Change Accumulation and Database Recovery utility control statements require new formats which accommodate full precision timestamps. Nevertheless, this is unlikely to be a concern to users since GENJCL.CA and GENJCL.RECOV in IMS V10 always produce control statements with the V10 formats.

When MINVERS('10.1') is set log records are created with the full precision timestamps. If you fallback to MINVERS('9.1') or MINVERS('8.1'), the log records created with full precision timestamps must be deleted before the CHANGE.RECON MINVERS(…) command is issued. The records that must be deleted include PRILOG, PRISLD, PRIOLD, etc. and the ALLOC records associated with these log records.

If you wish, you may specify abbreviated timestamps for most uses. DBRC will interpret the time correctly. Full precision timestamps are required in CHANGE and DELETE commands when a full precision timestamp is part of the RECON record key.

The timestamp precision value may be coded on the TIMEFMT parameter of the %SET statement in skeletal JCL. It is a value from 1 to 6. The default in previous releases was 1. In IMS V10 the default depends on the MINVERS value. MINVERS('10.1') sets the default to 6. MINVERS values less than '10.1' sets the default to 1. This could affect the output of GENJCL.USER. If you have specifically coded the TIMEFMT parameter, timestamp precision in the output will not change. If you have not coded TIMEFMT, the output may change when the MINVERS value is set to '10.1'.

Information On Demand
IMS Version 10
IBM

Database Change Accumulation Utility

- DB0 and DB1 control statements
  - Purge time is expanded
  - Number of DD names allowed reduced from 4 to 3
    - Position on control statement moved

Version 8 or 9 example:

```
----+----1----+----2----+----3----+----4----+----5----+----6----+----7---
//SYSIN    DD *
DB0DI32DB012006.175 12:00:00.1 -08:00DI320101 DI320102 DI320103 DI320104
DB0AB77DB012006.175 12:00:00.1 -08:00AB770101 AB770102
/*
```

Version 10 example:

```
----+----1----+----2----+----3----+----4----+----5----+----6----+----7---
//SYSIN    DD *
DB0DI32DB012006.175 12:00:00.123456 -08:00DI320101 DI320102 DI320103
DB0DI32DB012006.175 12:00:00.123456 -08:00DI320104
DB0AB77DB012006.175 12:00:00.1 -08:00     AB770101 AB770102
/*
```

- GENJCL.CA has been updated to create V10 control statements

Figure 15-78. Database Change Accumulation Utility CMA01.0

Notes:

The Database Change Accumulation utility DB0 and DB1 control statements have been modified to support timestamps with greater precision. The new expanded timestamp format may be used in V10 but it is not required. DB0 control statements are used to specify the database data sets that are accumulated to the new change accumulation data set. DB1 control statements are used to specify the database data sets that are written to the new output log data set. Since the timestamps now may occupy more columns in the control statements, the position of the database data set DDNAMEs have moved and only three may be specified on one control statement. In previous versions the timestamp was specified in columns 12-37 and DDNAMEs were in columns 38-45, 47-54, 56-63, and 65-72. In V10 the timestamp uses columns 12-42 and DDNAMEs are in columns 43-50, 52-59, and 61-68. If a database has more than three data sets that are in the same Change Accum group, multiple control statements are required in V10. The V10 example shows two control statements to replace the first control statement in the V9 example. Two control statements are required because four DDNAMEs are specified. The V10 example shows the greater precision timestamp. The last control statement in the V10 example does not specify the greater precision timestamp.

GENJCL.CA has been updated to create the new format of the control statements. These changes to the control statements will have no effects on users who create Change Accumulation JCL and control statements with GENJCL.CA. This is the vast majority of IMS installations.

Database Recovery Utility

- S control statement
  - Timestamp recovery time is expanded
  - Specification of "no image copy input" has been moved to column 63
  - Specification of time zone offset changed - space added before offset

Version 8 or 9 example:

```
----+----1----+----2----+----3----+----4----+----5----+----6----+----7---

//SYSIN     DD *
S  DI32DB01 DI320101          2006.175 12:00:00.1-08:00 C
/*
```

Version 10 examples:

```
----+----1----+----2----+----3----+----4----+----5----+----6----+----7---
//SYSIN     DD *
S  DI32DB01 DI320101          2006.175 12:00:00.123456 -08:00 C
/*
```

```
//SYSIN     DD *
S  DI32DB01 DI320101          2006.175 12:00:00.1 -08:00      C
/*
```

- GENJCL.RECOV has been updated to create V10 control statements

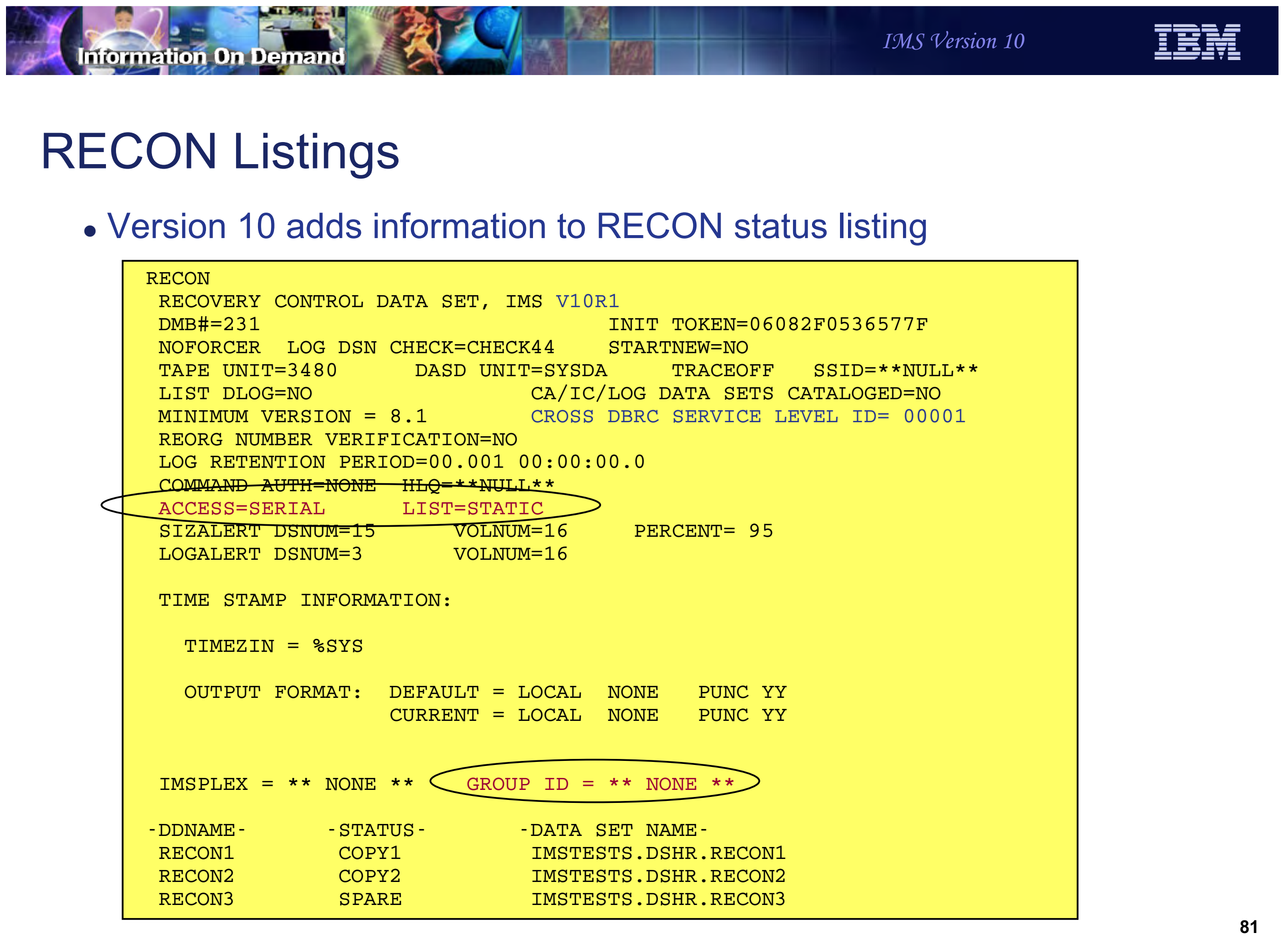
Figure 15-79. Database Recovery Utility CMA01.0

Notes:

The S control statement is used to specify the database and DDNAME for the database data set that is to be recovered. If the recovery is a timestamp recovery, the timestamp is also specified. The new expanded timestamp format may be used in V10 but it is not required. In previous releases column 57 was used for an indicator. The indicator could specify that a user image copy had been restored or that an RSR receive was done. Since the timestamp may use column 57, the indicator is now coded in column 63 when needed. The first example for V10 shows the use of the expanded timestamp. The second shows the use of the shorter timestamp. Both examples have the ‘C’ which indicates that there is no image copy input.

There is another small change in the coding of the timestamp. In previous releases there was not a space between the time and the sign for the time zone offset. In the V8 or V9 example on this page the sign is in column 50. In V10 there is a blank between the time and the sign for the offset. In the two examples on this page the blank is in column 55 for the first V10 example and column 50 for the second example.

GENJCL.RECOV has been updated to create the new format of the control statement for timestamp recovery. The change in the control statement will have no effects on users who create Database Recovery JCL and control statements with GENJCL.RECOV. This is the vast majority of IMS installations.



Information On Demand
IMS Version 10
IBM

RECON Listings

- Version 10 adds information to RECON status listing

```
RECON
 RECOVERY CONTROL DATA SET, IMS V10R1
 DMB#=231                           INIT TOKEN=06082F0536577F
 NOFORCER  LOG DSN CHECK=CHECK44    STARTNEW=NO
 TAPE UNIT=3480      DASD UNIT=SYSDA     TRACEOFF   SSID=**NULL**
 LIST DLOG=NO                 CA/IC/LOG DATA SETS CATALOGED=NO
 MINIMUM VERSION = 8.1        CROSS DBRC SERVICE LEVEL ID= 00001
 REORG NUMBER VERIFICATION=NO
 LOG RETENTION PERIOD=00.001 00:00:00.0
 COMMAND AUTH=NONE  HLQ=**NULL**
 ACCESS=SERIAL      LIST=STATIC
 SIZALERT DSNUM=15      VOLNUM=16     PERCENT= 95
 LOGALERT DSNUM=3       VOLNUM=16

 TIME STAMP INFORMATION:

   TIMEZIN = %SYS

   OUTPUT FORMAT:  DEFAULT = LOCAL  NONE   PUNC YY
                   CURRENT = LOCAL  NONE   PUNC YY


 IMSPLEX = ** NONE **    GROUP ID = ** NONE **

-DDNAME-      -STATUS-       -DATA SET NAME-
 RECON1        COPY1          IMSTESTS.DSHR.RECON1
 RECON2        COPY2          IMSTESTS.DSHR.RECON2
 RECON3        SPARE          IMSTESTS.DSHR.RECON3
```

81

Figure 15-80. RECON Listings CMA01.0

Notes:

The RECON status or header listing has some added and changed information.

Of course, the IMS version is listed as "V10R1". This means that the RECONs have been upgraded to IMS V10.

There is a new line which lists the type of RECON access, either SERIAL or PARALLEL. On the same line the default for the DBRC LIST command, either STATIC or CONCURR, is shown.

On the line where the IMSPLEX value is shown, the DBRC Group ID value is also shown. In this example, these parameters have no values so "** NONE **" is listed.

The sample listing shown here includes the "CROSS DBRC SERVICE LEVEL ID". This also appears on IMS V9 RECON listings when the maintenance for APARs PQ98655 and PK01097 is applied and on IMS V8 RECON listings when the maintenance for APARs PQ98654 and PK01096 is applied. The service level ID is used to invoke functions which require a consistent level of maintenance on all IMS systems using the RECONs.

DBRC API Security Authorization

- DBRC command authorization security is extended to include DBRC API requests
  - Requests are treated like commands for security checking

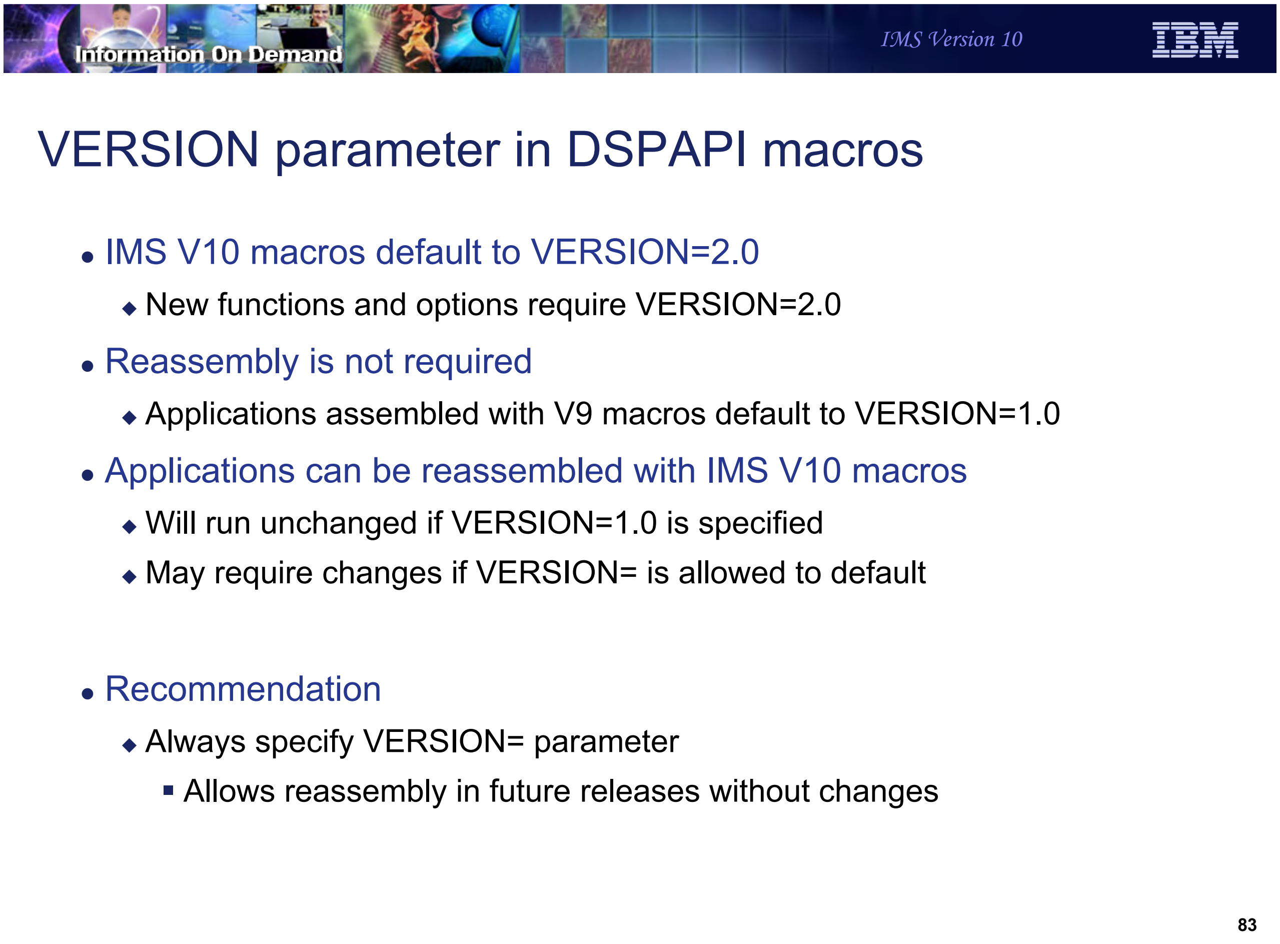
- Migration
  - Security was not checked in V9
    - Security checking in V10 might cause application which ran successfully in V9 to fail in V10

Figure 15-81. DBRC API Security Authorization CMA01.0

Notes:

DBRC command security was introduced in IMS V8. It may be used to invoke authorization checking for DBRC commands. RACF, or any SAF security product, may be used. Alternatively, an exit routine may be invoked or both the security product and the exit routine may be invoked. Commands are authorized by defining a resource representing the command. In RACF this is done with an RDEFINE statement.

This authorization is extended to the DBRC API in IMS V10. API requests invoke command authorization checking. Command authorization checking uses resources which are defined to secure specific commands or API requests.


Information On Demand
IMS Version 10

VERSION parameter in DSPAPI macros

- IMS V10 macros default to VERSION=2.0
  - New functions and options require VERSION=2.0
- Reassembly is not required
  - Applications assembled with V9 macros default to VERSION=1.0
- Applications can be reassembled with IMS V10 macros
  - Will run unchanged if VERSION=1.0 is specified
  - May require changes if VERSION= is allowed to default
- Recommendation
  - Always specify VERSION= parameter
    - Allows reassembly in future releases without changes

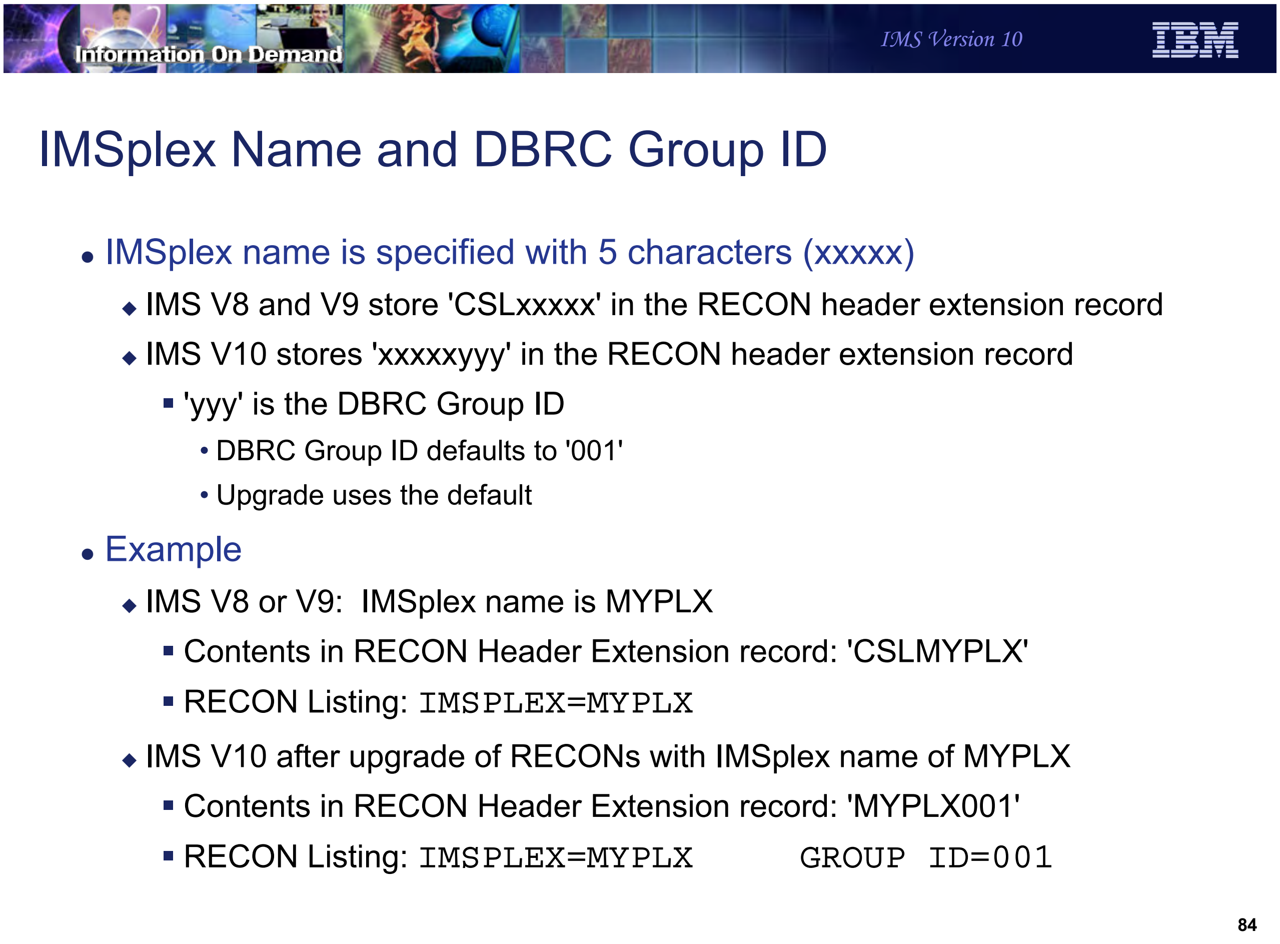
Figure 15-82. VERSION parameter in DSPAPI macros CMA01.0

Notes:

Each DBRC API macro includes the VERSION= parameter. New functions, such as AUTH, and new options, such as READONLY=YES, require VERSION=2.0. In IMS V9 the only valid value for VERSION was 1.0. It was also the default. In IMS V10 VERSION= defaults to 2.0.

Applications written in IMS V9 will continue to run in V10 without change. Reassembly is not required. In fact, reassembly could cause the program to change due to the change in the default for VERSION=. In some cases, the default of VERSION=2.0 may cause different results from the previous default of VERSION=1.0. This is not always the case. Some of the changes are only the use of previously reserved bytes in the control blocks that are produced. In any case, if you wrote a program for IMS V9 and reassemble it using an IMS V10 macro library, it is safest to specify VERSION=1.0 on the DFSAPI macros before the reassembly.

Since the VERSION= parameter defaults to the latest level of the macros and later levels may produce different results, it is safest to specify the VERSION= parameter value explicitly. This will ensure that future assemblies of DBRC API programs will produce the same results.


Information On Demand
IMS Version 10
IBM

IMSplex Name and DBRC Group ID

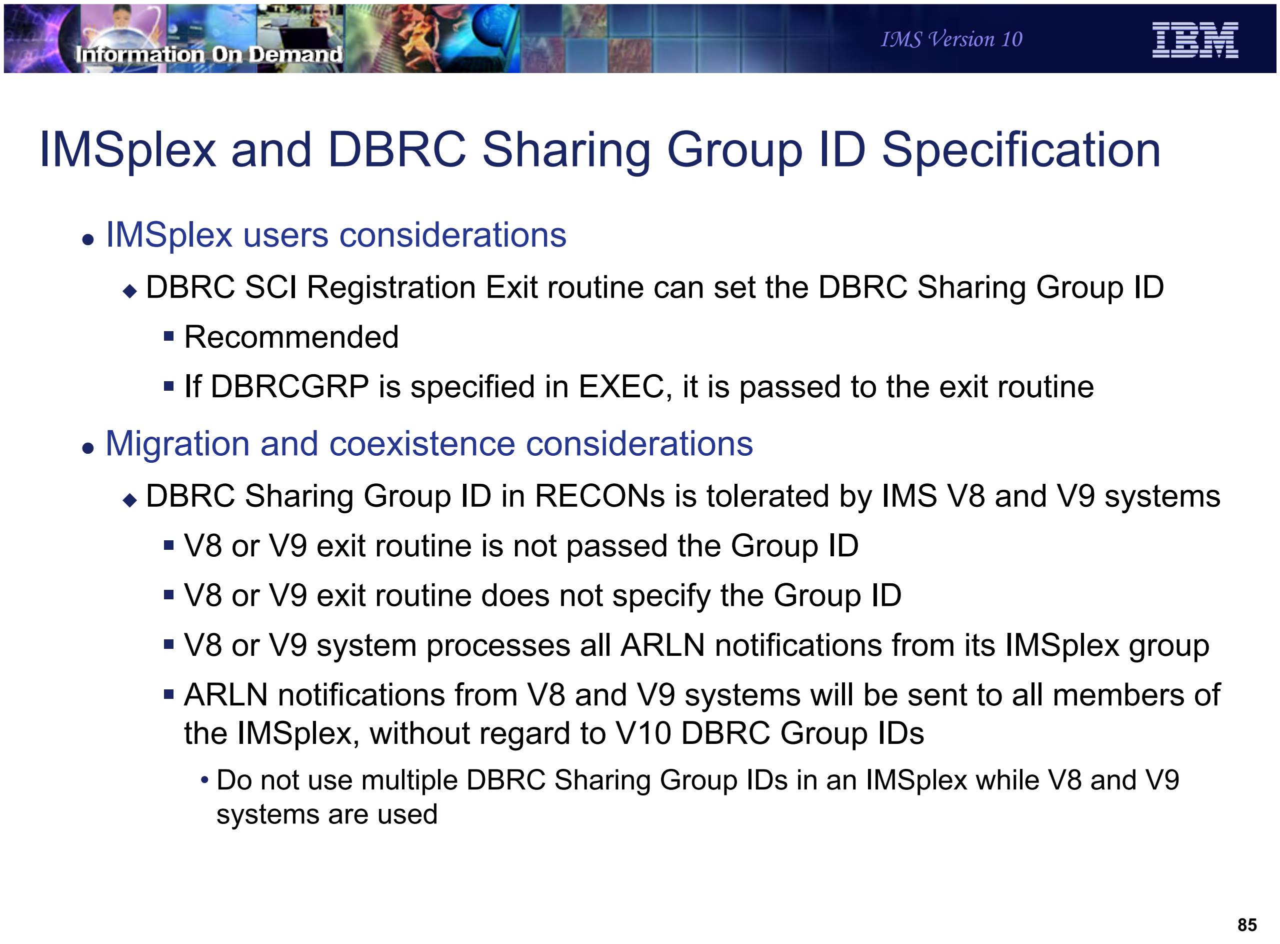
- IMSplex name is specified with 5 characters (xxxxx)
  - IMS V8 and V9 store 'CSLxxxxx' in the RECON header extension record
  - IMS V10 stores 'xxxxxyyy' in the RECON header extension record
    - 'yyy' is the DBRC Group ID
      - DBRC Group ID defaults to '001'
      - Upgrade uses the default
- Example
  - IMS V8 or V9: IMSplex name is MYPLX
    - Contents in RECON Header Extension record: 'CSLMYPLX'
    - RECON Listing: `IMSPLEX=MYPLX`
  - IMS V10 after upgrade of RECONs with IMSplex name of MYPLX
    - Contents in RECON Header Extension record: 'MYPLX001'
    - RECON Listing: `IMSPLEX=MYPLX      GROUP ID=001`

Figure 15-83. IMSplex Name and DBRC Group ID CMA01.0

Notes:

The IMSplex name is optional. It is required for Automatic RECON Loss Notification and Parallel RECON Access. The IMSplex name is specified with up to 5 characters. If it was set under IMS V8 or V9, it will be maintained when the RECONs are upgraded to IMS V10. If it was not set under IMS V8 and V9 it may be set in IMS V10 by a CHANGE.RECON command. It is stored in the RECONs. IMS V8 and V9 store the IMSplex name as 'CSLxxxxx' where 'xxxxx'. When the RECONs are upgraded to V10, the value stored is 'xxxxxyyy' where 'yyy' is the DBRC Group ID. The upgrade sets the DBRC Group ID to '001'. It may be changed with a CHANGE.RECON IMSPLEX(plexname,groupid) command.

IMS V8, V9, and V10 list only the 5 characters of the IMSplex name in listings of the RECON header. These listing include a line with IMSPLEX=xxxxx when an IMSplex name has been stored in the RECONs. If there is no value stored, the line includes IMSPLEX=**NONE**. IMS V10 listings also include the DBRC Group ID on this line. If there is no IMSplex name the Group ID is listed as GROUP=**NONE**. If there is an IMSplex name, the Group ID is listed as GROUP=yyy where yyy is the Group ID.



Information On Demand

IMS Version 10

IMSplex and DBRC Sharing Group ID Specification

- IMSplex users considerations
  - DBRC SCI Registration Exit routine can set the DBRC Sharing Group ID
    - Recommended
    - If DBRCGRP is specified in EXEC, it is passed to the exit routine
- Migration and coexistence considerations
  - DBRC Sharing Group ID in RECONs is tolerated by IMS V8 and V9 systems
    - V8 or V9 exit routine is not passed the Group ID
    - V8 or V9 exit routine does not specify the Group ID
    - V8 or V9 system processes all ARLN notifications from its IMSplex group
    - ARLN notifications from V8 and V9 systems will be sent to all members of the IMSplex, without regard to V10 DBRC Group IDs
      - Do not use multiple DBRC Sharing Group IDs in an IMSplex while V8 and V9 systems are used

Figure 15-84. IMSplex and DBRC Sharing Group ID Specification CMA01.0

Notes:

The DBRC SCI Registration exit routine (DSPSCIX0) may be used to specify the IMSplex name, as in previous releases, and the new DBRC Group ID. The use of the exit routine is recommended for users of IMSplex. It removes the requirement to specify IMSPLEX= for the execution of all IMS jobs which use DBRC. This includes batch jobs and utilities. With IMS V10 the exit also may specify the DBRC Sharing Group ID. This removes the requirement to specify DBRCGRP= for IMS executions.

IMS V8 and V9 systems can tolerate the specification of the DBRC Group ID in the RECONs. DBRCGRP= is not a valid parameter on the EXEC statement for V8 and V9. When the exit routine is invoked in a V8 or V9 environment, the DBRC Group ID is not passed to it. The exit routine cannot specify the DBRC Group ID. Even though a V8 or V9 instance cannot specify the DBRC Group ID, it can join an IMSplex where V10 instances are using DBRC Group IDs. The V8 or V9 instance will be passed all ARLN notifications from the IMSplex group. If a V8 or V9 system reconfigures its RECONs, its ARLN notification will be processed by all members of the IMSplex. This will include all V10 systems. If there are multiple DBRC Groups, all members of all groups will process the notification. For these reasons, you should not use multiple DBRC Group IDs in an IMSplex while you are still using IMS V8 or V9 systems.

Bibliography

Manuals:

SC18-9699	IMS Version 10: Application Programming API Reference
SC18-9698	IMS Version 10: Application Programming Guide
SC18-9697	IMS Version 10: Application Programming Planning Guide
SC18-9700	IMS Version 10: Command Reference, Volume 1
SC18-9701	IMS Version 10: Command Reference, Volume 2
SC18-9702	IMS Version 10: Command Reference, Volume 3
SC18-9703	IMS Version 10: Communications and Connections Guide
SC18-9704	IMS Version 10: Database Administration Guide
SC18-9705	IMS Version 10: Database Utilities Reference
GC18-9706	IMS Version 10: Diagnosis Guide
GC18-9707	IMS Version 10: Diagnosis Reference
SC18-9708	IMS Version 10: Exit Routine Reference
SC18-9709	IMS Version 10: IMSplex Administration Guide
GC18-9710	IMS Version 10: Installation Guide
GC18-9782	IMS Version 10: Licensed Programming Specifications
SC18-9711	IMS Version 10: Master Index and Glossary
GC18-9712	IMS Version 10: Messages and Codes Reference, Volume 1: DFS Messages
GC18-9713	IMS Version 10: Messages and Codes Reference, Volume 2: Non-DFS Messages
GC18-9714	IMS Version 10: Messages and Codes Reference, Volume 3: IMS Abend Codes
GC18-9715	IMS Version 10: Messages and Codes Reference, Volume 4: IMS Component Codes
SC18-9716	IMS Version 10: Operations and Automation Guide
GC18-9717	IMS Version 10: Release Planning Guide
SC18-9718	IMS Version 10: System Administration Guide
GC18-9998	IMS Version 10: System Definition Guide
GC18-9966	IMS Version 10: System Definition Reference
SC18-9967	IMS Version 10: System Programming API Reference

SC18-9968	IMS Version 10: System Utilities Reference

Web URLs:

http://www.ibm.com/ims

IBM’s IMS web site