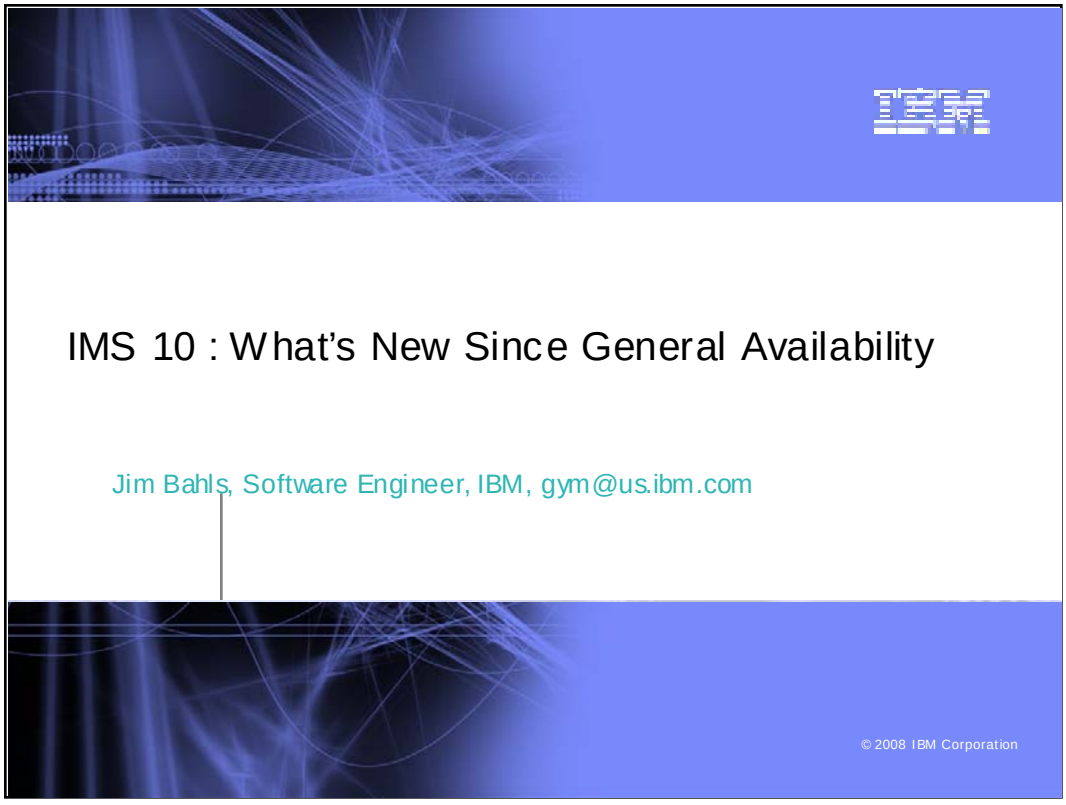


IMS 10 : What's New Since General Availability

Jim Bahls, Software Engineer, IBM, gym@us.ibm.com

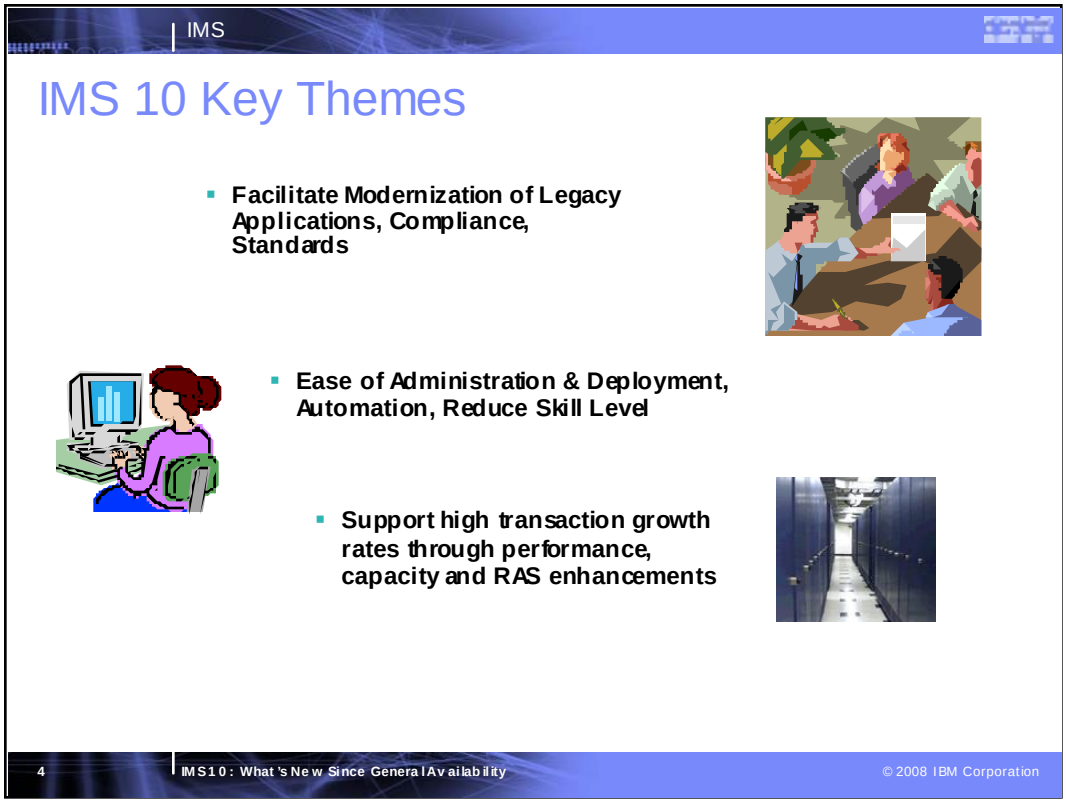
Disclaimer

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. IN ADDITION, THIS INFORMATION IS BASED ON IBM'S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION. NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM IBM (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF ANY AGREEMENT OR LICENSE GOVERNING THE USE OF IBM PRODUCTS AND/OR SOFTWARE.

Change Log for Presentation

Date	Initials	Change Description
2008-09-16	Jwb	Initial Presentation

This change log can be used to track changes to this presentation. The enhancement summary charts at the end will be updated with APAR #s and PTF #s as they are created.

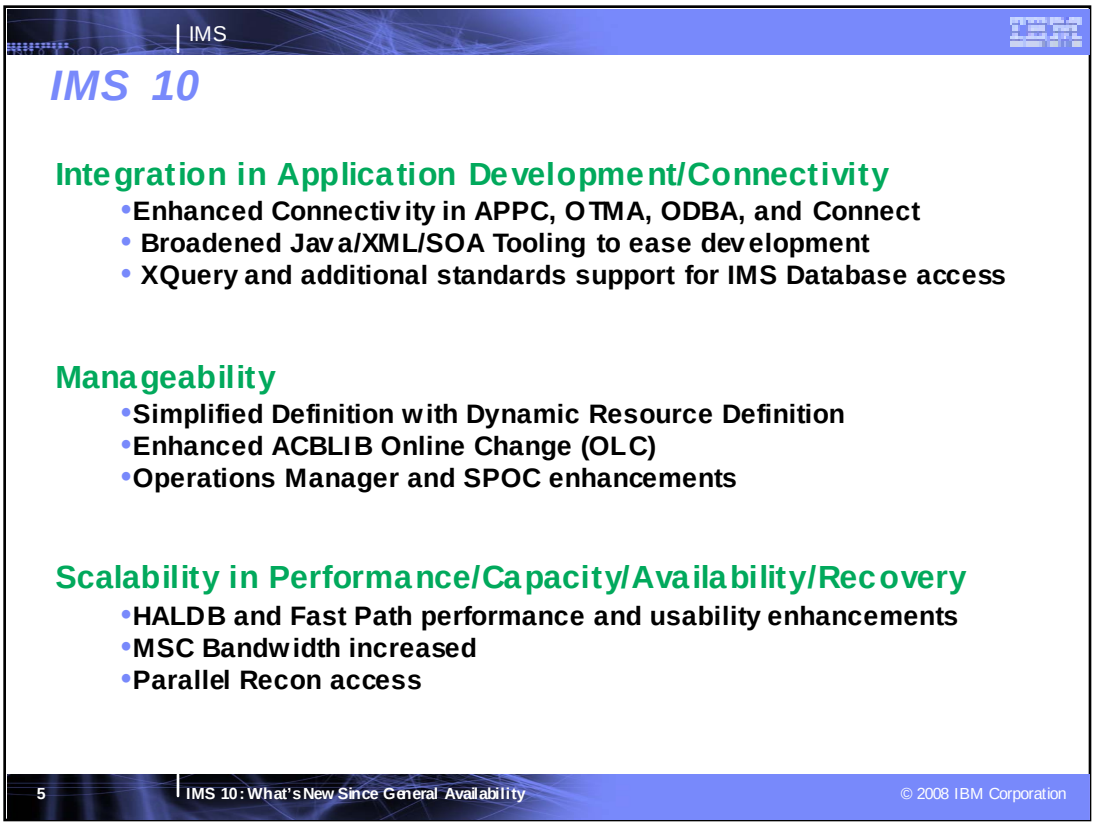


IMS Strategy

Improve Scalability, Capacity, Performance and Availability to allow IMS systems to grow faster than customer business needs.

Reduce the skill requirement needed to install, configure and operate IMS to make it easier to train new IMS system programmers. Make it easy to develop and deploy new applications that reuse existing IMS assets, reducing and/or eliminating the need for application programmers to understand IMS programming interfaces.

IMS has recently kicked off a new IMS SOA initiative to help clients modernize their applications. We are going to cities to talk to multiple clients and also to individual client locations. We are doing presentations and workshops to show them the power of IMS and SOA and how it can help their business. This is an effort to drive new application workloads on IMS.



Key Message: IMS 10 enhancements address integration, manageability, and scalability

IMS Enhancements are provided with new versions and service stream enhancements.

IMS 10 enhancements help you with Information On Demand business enablement, growth, availability, and systems management that today's and future environments and cost measures require. IMS 10 helps in addressing your On Demand business needs through integration/openness, manageability, and scalability, providing :

- Integration with other products and platforms across the internet, supporting open standards that benefit you, taking best advantage of the latest industry tooling for application development and connectivity.
- Manageability in staging users to autonomic computing, easing installation and use, eliminating/reducing outages, eased operations and systems management and minimizing the education curve for users of IMS.
- Scalability with virtualization in assuring flexibility for growth and expansion in a heterogeneous environment, utilizing the latest hardware/software facilities to optimize performance, capacity, availability and recovery.

Included for Integration in application development/Connectivity are:

- Integrated, Open Access with enhanced IMS XML database support, XQuery access to IMS data, and broadened Java and XML tooling to ease development.

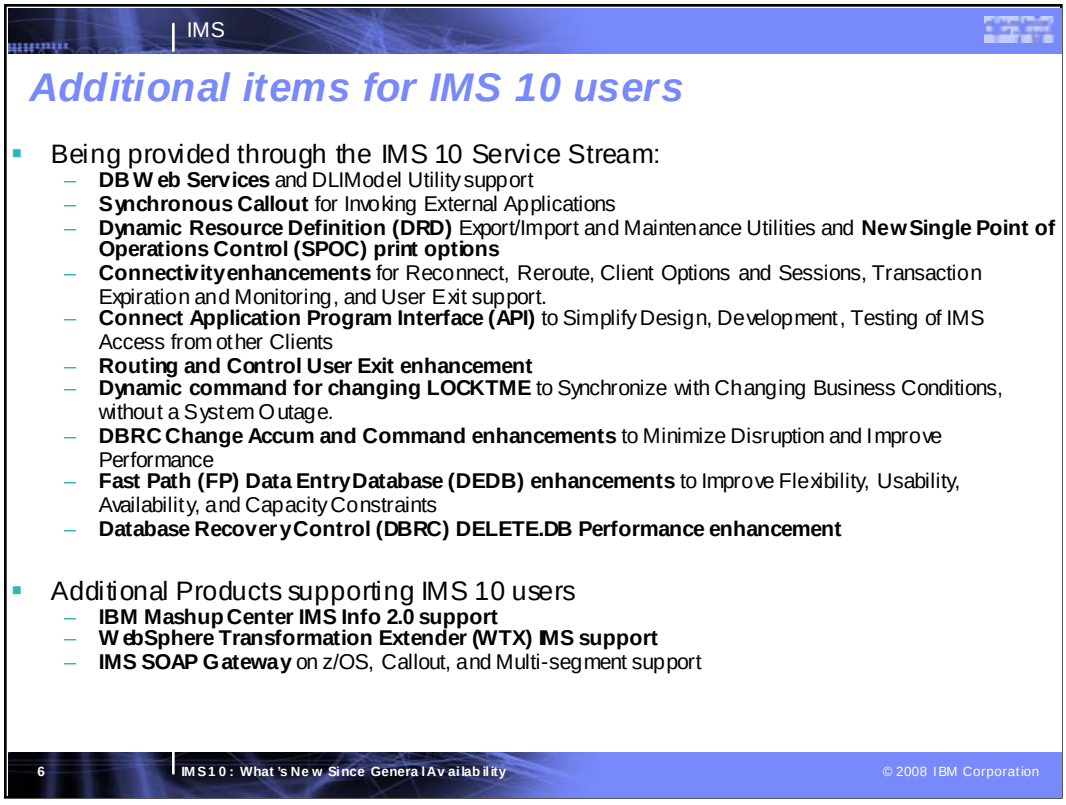
Included for Easing IMS Manageability are:

- Simplified Installation and Definition using new Dynamic Resource Definition facilities and additional online change capabilities help to simplify the method to define IMS systems and resources. These allow dynamic additions, changes and deletions, reduce the number of required user decision points and use intelligence to determine system default values while allowing users to define and override definition choices.
- Additional enhancements are also provided to ease systems/operations management, installation, serviceability and usability.

Scalability enhancements for Performance/Capacity/Availability/Recovery include:

- Fast Path enhancements continually improve performance and usability
- The Multiple Systems Coupling (MSC) facility, connecting IMS systems across the network, is enhanced to improve bandwidth performance across the network.
- Parallel Recon Access relieves contention for the Database Recovery Control (RECON) data sets in the IMS Sysplex. As the Sysplex grows, this can relieve bottlenecks to growth.

.



Key Message: These are many additional enhancements being provided for IMS 10 users.

IMS 10 enhancements are being provided through the IMS 10 service process or as enhancements to separate related products.

Customers have asked for still more IMS connectivity enhancements that extend access to IMS TM while reducing existing complexity and resource requirements. These enhancements can improve reliability and serviceability and enhance resilience, performance and availability.

The IMS Connect API shields IMS Connect client application developers from complexities by providing a simple API which allows them to interact with IMS Connect. This is being accomplished by providing simple ways of describing the connections that they want made and the interactions that they want performed along with the data that they want sent to IMS for those interactions using re-usable profiles and by providing them with simple methods which will perform those interactions.

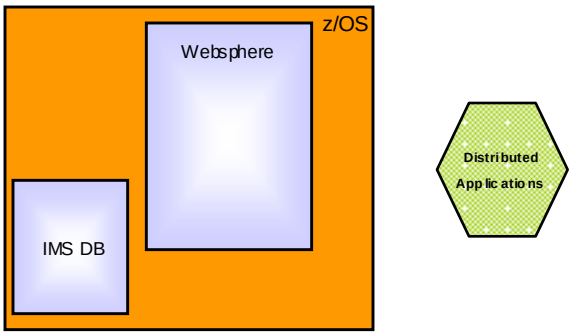
Separate related products, like the WebSphere Transformation Extender (WTX), provide support to invoke IMS transactions while leveraging standards-based transaction support on distributed platforms of complex data formats and unique industry requirements. This support provides faster standards compliance and improved data quality with automated data validation using industry and regulatory standards.

I'll expand on a few of these items.


IBM
Integration and Connectivity

Access IMS data without TM?

- DBCTL Customer wants to access IMS data from the distributed world
  - Tremendous value in Data stored in IMS
  - Must write a Web Service Application to expose desired data
  - TM Resource Adapter require IMS TM

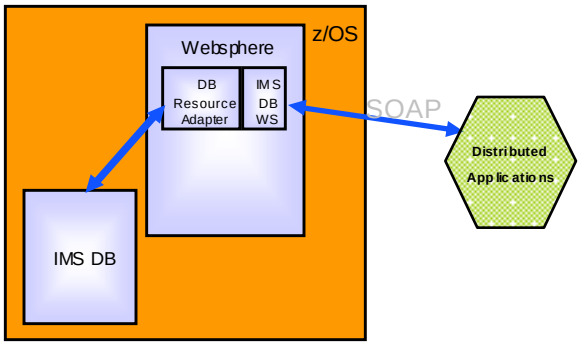


IMS DB only customers (DBCTL) cannot expose their IMS data to the distributed world without writing their own Web Service Application . IMS provides JAVA libraries for the low-level DL/I access, but a Web Service Application must be written in order to make the IMS data available to distributed applications.

IMS TM Resource Adapter can not be used as solutions because it require IMS TM.

IMS DB Web Services

- Access to IMS Databases through IMS DBCTL
  - Web Service Application and WSDL is generated
    - specifying data using DLI syntax
  - IMS TM not required
    - Requires Websphere z
    - Additional required software included with IMS



Additional required components that are included with IMS

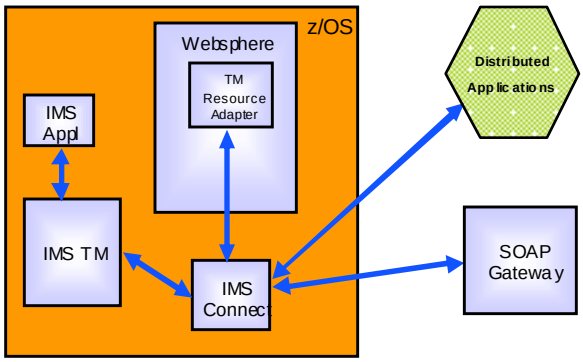
- IMS Java Libraries (for low-level DL/I IMS access)
- IMS XML Database (for IMS/XML data transformations)
- DLI Model Utility (for tooling)

The IMS Java Libraries (which include the IMS XML Database support) make up the IMS DB Resource Adapter that is deployed into Websphere. This gives Websphere applications access to IMS data.

A Config file defines the data you wish to expose is used as input to the DLI Model Utility. The DLI Model utility generates two artifacts: an EAR file and a WSDL file. The EAR file is deployed into Websphere, while the WSDL can be passed on to SOA Programmers.

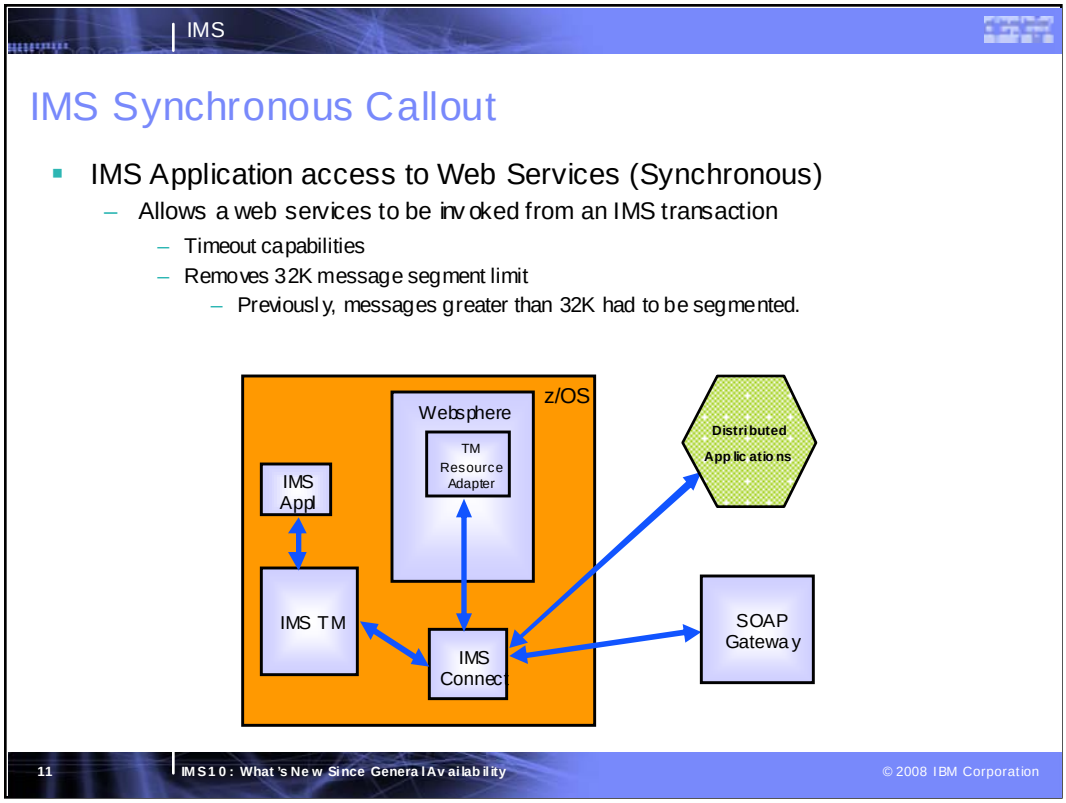
IMS Asynchronous Callout

- IMS Application access to Web Services (Asynch)
  - IMS 10 introduced Asynchronous callout
    - Opens up web services to IMS transactions
    - One transaction initiates service, another transaction receives output
  - How about the use of web services within a transaction?



IMS 10 introduced asynchronous callout to IMS. It gives IMS applications the ability to initiate a web service by inserting a message to an ALT-PCB. The response to the web service request is processed by another transaction.

Customers also want to be able to invoke web services within a transaction.



The synchronous callout function allows an application to invoke a web service from within a transaction instance. The output is received synchronously (within the same transaction instance).

Synchronous callout allows your IMS application to invoke one of the following external applications and synchronously get the output back:

(1) an J2EE application (like an Enterprise Java Bean/EJB or an Message Driven Bean (MDB)) or Web service providers running in the WebSphere Application Server using the IMS TM Resource Adapter

(2) other Web service providers (e.g. like Microsoft .NET or SAP XI) using IMS SOAP Gateway

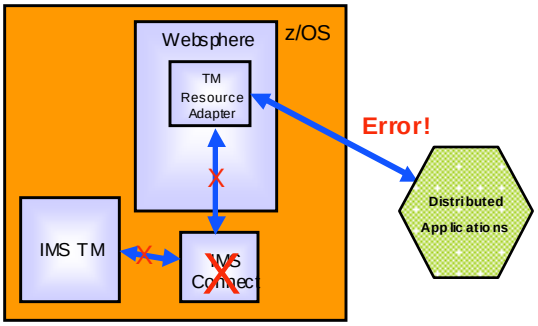
(3) any other applications (like RYO, SAP apps) using the IMS Connect interfaces

An IMS application makes a DLI call to send out a synchronous callout request. This is routed by IMS to one of the one of the outbound destinations through IMS Connect – i.e. the WebSphere Web Services, SOAP Gateway, or RYO applications. After the callout request has been processed, the output data would be returned back to the same IMS transaction instance.

Initial release will not support more complex environments (such as 2 phase commit). It will support Shared Queues, but each back-end IMS must have an IMS Connect. The outbound destination cannot go through the Shared Queue.

IMS TM Resource Adapter Externalizes Connection Error

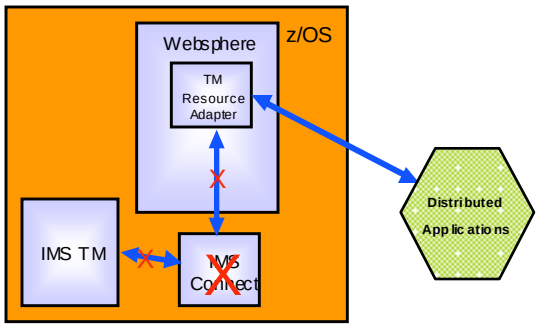
- Recycling IMS Connect causes WAS applications to fail because of connection failures
  - System maintenance can force IMS Connect to be recycled
  - WAS applications fail because of the lost connection



IMS Connect needs to be recycled periodically in order to apply system maintenance. This can result in an error message being sent to distributed applications that are connected to IMS via the TM Resource Adapter.

IMS TM Resource Adapter : Socket Reconnect

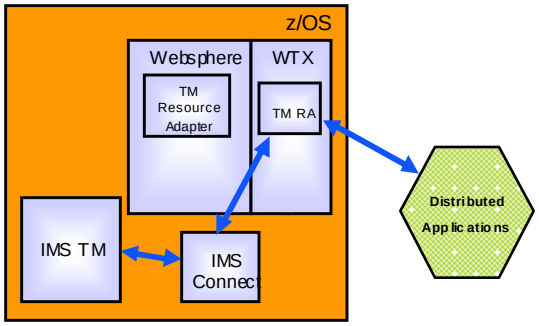
- Socket Reconnect provides a capability to reconnect when a communications failure is encountered
  - Communications with IMS Connect
  - Hides the connection failure from the application



The IMS TM Resource Adapter Socket Reconnect enhancement hides the connection failure from the distributed application by handling the failure and reconnecting to IMS Connect.

IMS TM Resource Adapter : WTX Support

- Websphere Transformation Extender
  - Supports complex data formats on distributed platforms
  - Helps customers to be compliant with industry standard data formats
- Can invoke IMS Transactions from distributed application and be compliant with industry standard data formats

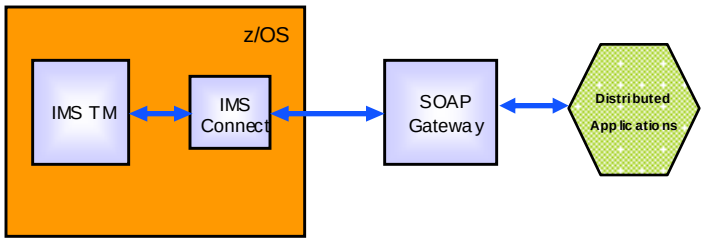


IMS integration with WTX enables an IMS distributed applications to be compliant with complex data formats like SEPA, SWIFT (financial services), HIPAA (healthcare), and EDI (cross-industry).

This support is included in

SOAP Gateway Multi-segment support

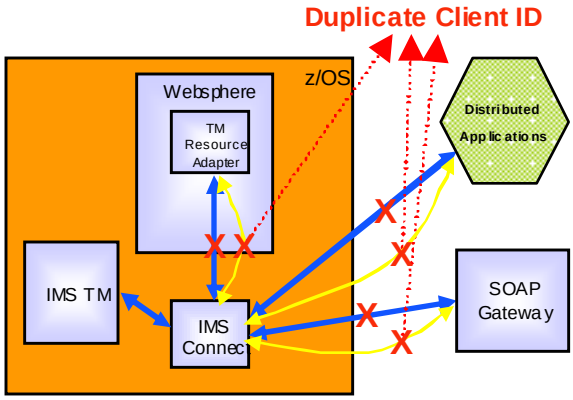
- Unable to expose multi-segment transaction to SOA applications
  - Single segment support only
- SOAP Gateway enhanced to support multi-segment transactions



Multi-segment support is provided by IMS SOAP gateway and IMS Connect with tooling support from the Rational Developer for System z v7.5. With this support, IMS multi-segment applications can participate in SOA using the IMS SOAP Gateway. Note that this support is provided only for the “provider” (inbound – both request/response) scenario and not for the “consumer” (callout) scenario

IMS Connect Receives "Duplicate ID" Error

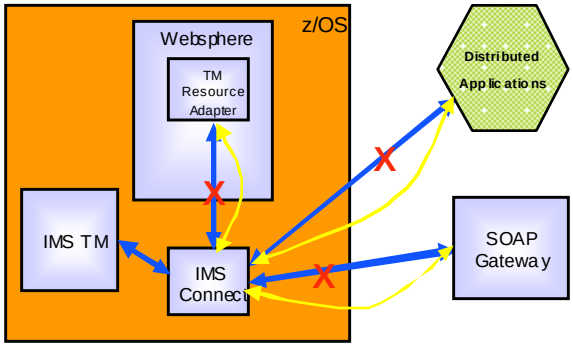
- Reconnect can return 'duplicate client id'
  - Each IMS Connect client must handle this 'normal' error
  - Higher network traffic, higher cost



If an IMS Connect client is disconnected (for example, because of network problems), it is unable to reconnect until IMS Connect cleans up the old session. This results in the client receiving a 'duplicate client id' error from IMS Connect until the old connection is cleaned up.

IMS Connect Cancel Client ID Option

- Client tell IMS Connect to 'reconnect' client ID
  - IMS Connect handles duplicate client ID
  - Lower network traffic, lower cost



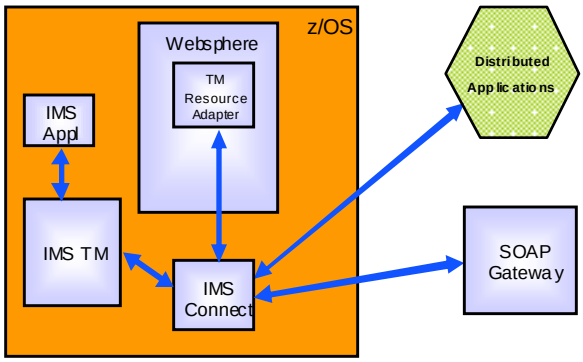
The cancel client id is an option which can be set in the first input message.

IRM_F3_CANCID flag (X'80') in the IRM header

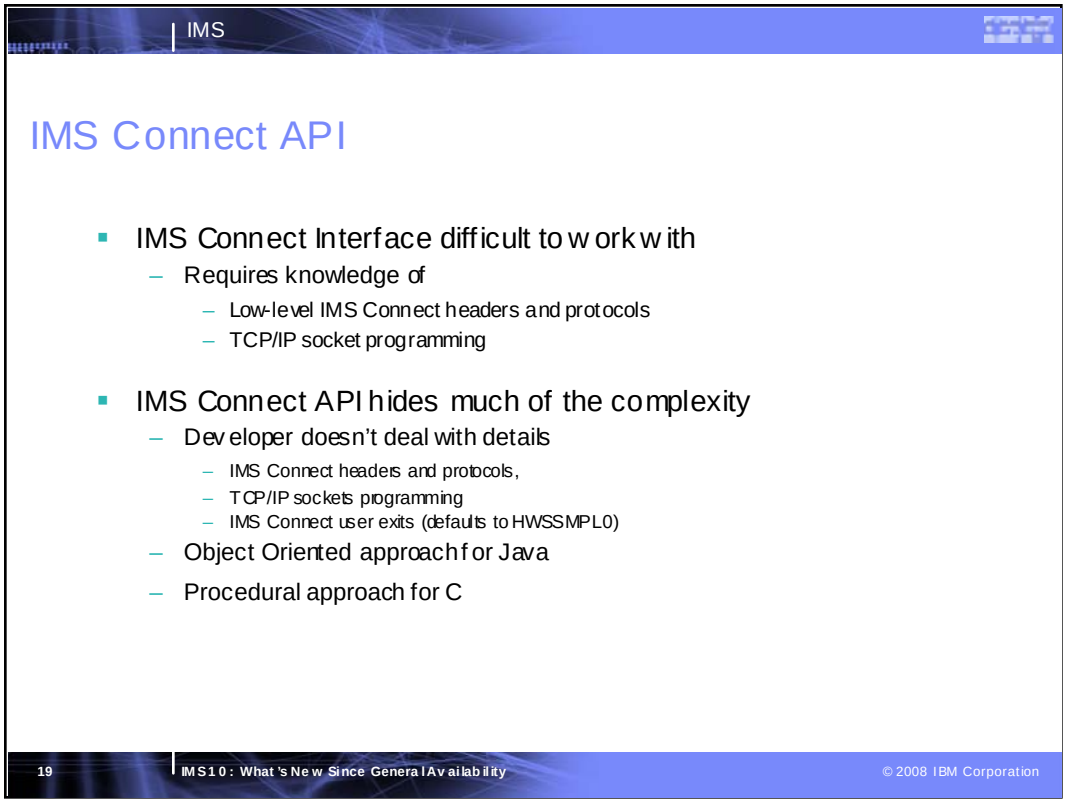
If the duplicate session exists, it will be cleaned up before the new one is established and HWSS0743W is issued

IMS Connect Client ID Hash Table

- Client ID search modified to use a hash table
  - Performance improvement
    - Especially for customers with large number of clients



The client id search was previously done sequentially. A hash table can be much more efficient, especially when there is a large number of client ids being searched (such as an environment with a large number of concurrent client sockets). This change can result in a faster response time and less CPU being used for each message that is processed.

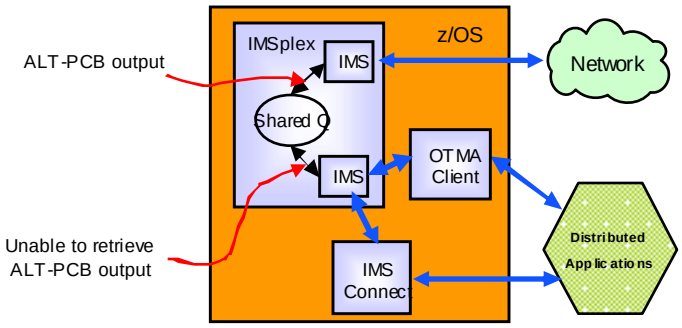


This item addresses the need for a simplified way to interact with IMS Connect. Currently, IMS Connect client application developers must understand the complexities of both the IMS Connect IRM header and IMS Connect protocols for all of the types of interactions supported by IMS Connect as well as the complexities of TCP/IP socket programming.

The IMS Connect API is used to execute interactions with IMS Connect. In order to shield client applications (and their developers) from the complexities of interacting with TCP/IP sockets and IMS Connect, the IMS Connect API generates the IMS Connect input message header internally and manages the interaction with IMS Connect according to the IMS Connect message protocols.

OTMA Unable to Retrieve all ALT-PCB Output

- Application sends output to alternate destination
  - Different from where the transaction originated
- OTMA can retrieve if same IMS
- Unable to retrieve if transaction ran on another IMS
  - Transaction must run on same IMS as OTMA

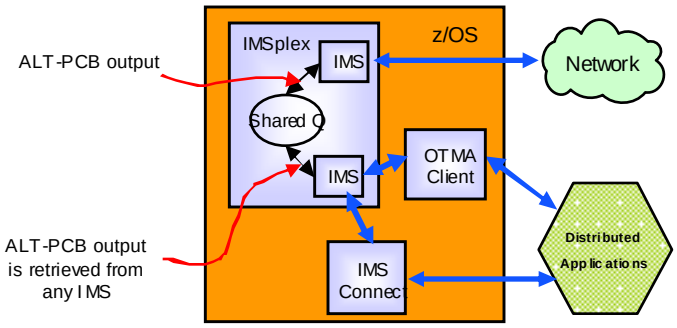


ALT-PCB output messages generated at the shared queues back-end IMS cannot be retrieved by the front-end IMS with IMS Connect using the OTMA Resume Tpipe AUTO request. Since messages generated at the front-end IMS can be retrieved, all the transactions that generate the ALT-PCB output must run on the same IMS as the OTMA connection. The makes the shared queues IMSplex less flexible for distributed applications.

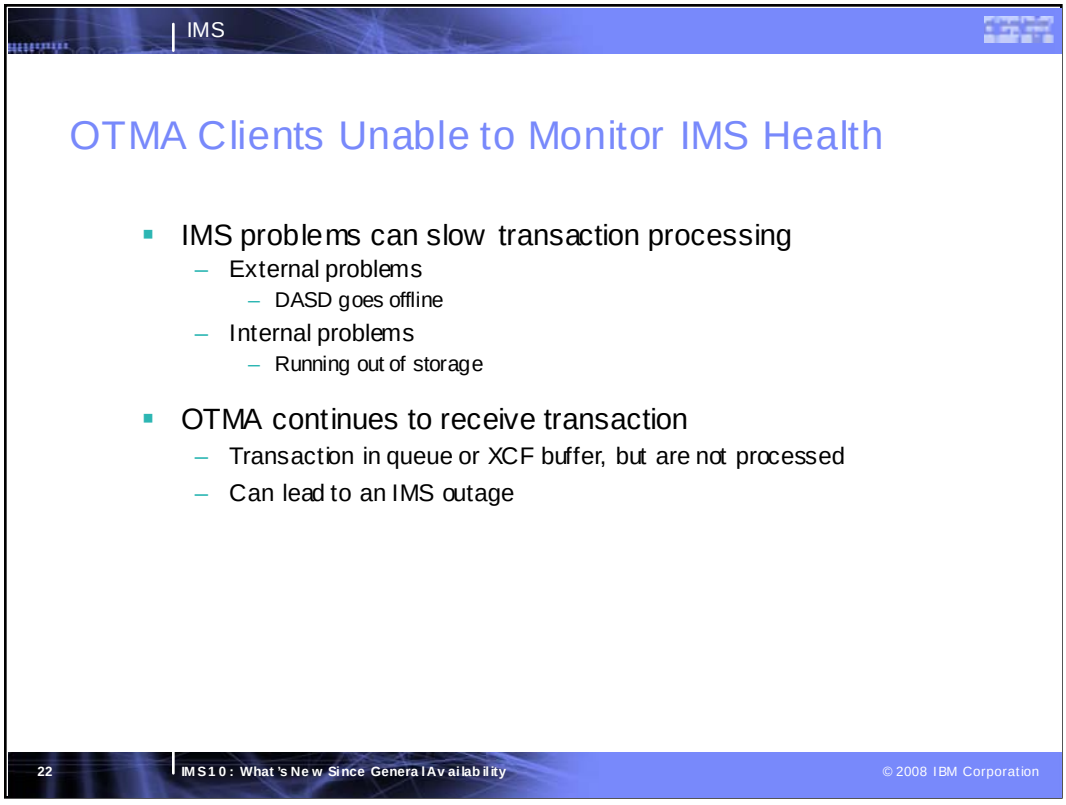
A back end IMS is any IMS in the shared queues group other than the OTMA local IMS.

OTMA Retrieve ALT-PCB Output

- OTMA can retrieve ALT-PCB output that originated from any IMS in the shared queues group

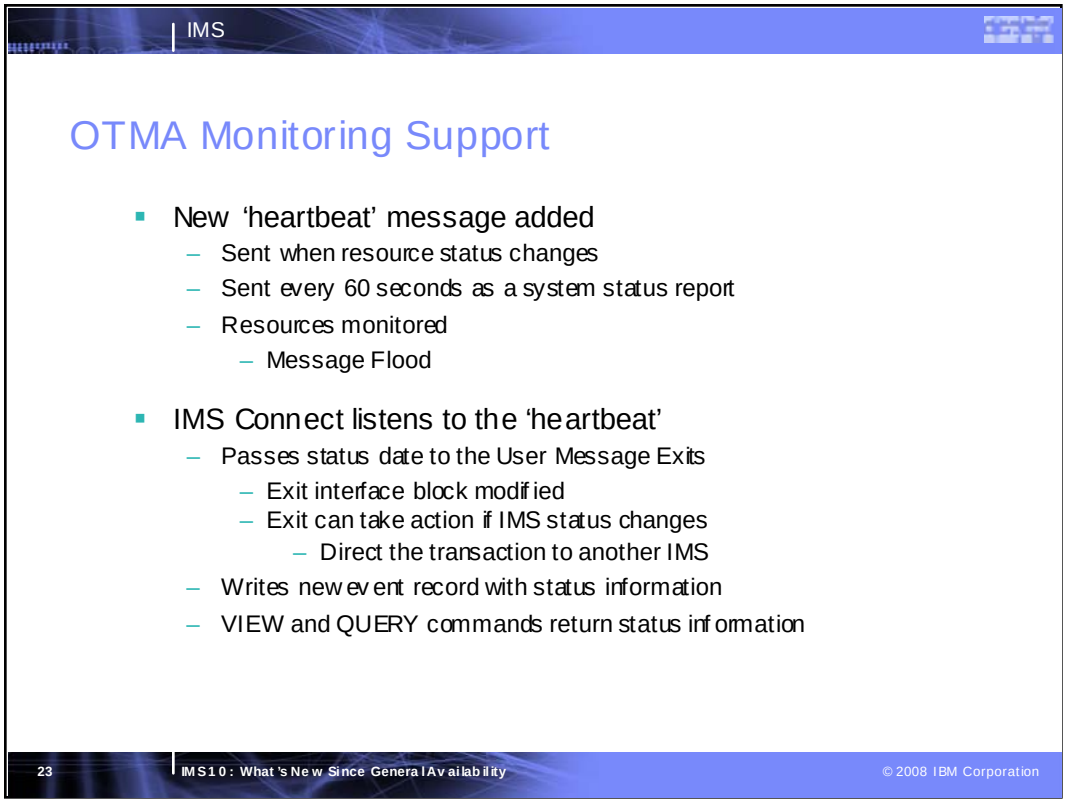


With this enhancement, ALT-PCB output can be retrieved from any IMS in the shared queues IMSplex. The OTMA connections does not need to be on the same IMS that ran the transaction.



An IMS system can encounter problems that slow or halt its processing of transactions without IMS actually taking an outage. This is often a result of a problem somewhere else on the z/OS system that cascades to IMS. For instance, a problem with the DASD that contains the IMS logs can cause problems for the entire IMS system. The source of the problems can also be internal to IMS. For example, a shortage of storage can slow down IMS until that shortage is relieved.

Even though IMS is not functioning at 100%, OTMA will continue to receive transactions. Since these transactions are not being processed at a normal rate, they will queue up and can cause an IMS outage. Even if IMS stays up, the work that is represented by these transactions is not being done.



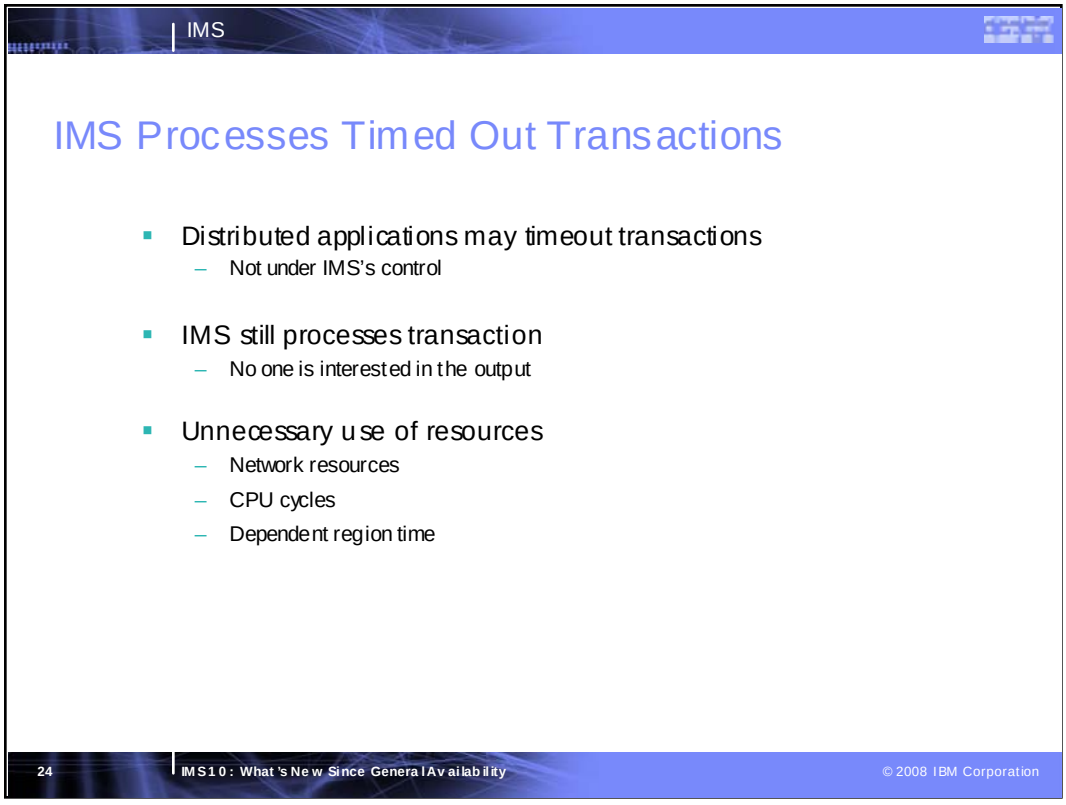
A system status message is added to OTMA. This message is sent to all OTMA clients :

•Any time a resource status changes. For any resource that OTMA is monitoring, it will set one of two bits in the resource status section of the message header. If the resource status degrades to a certain threshold, the 'warning' bit is set for that resource. If the resource is not available at all, the 'unavailable' bit is set for that resource.

•Every 60 seconds as a 'heartbeat' of the current status of IMS.

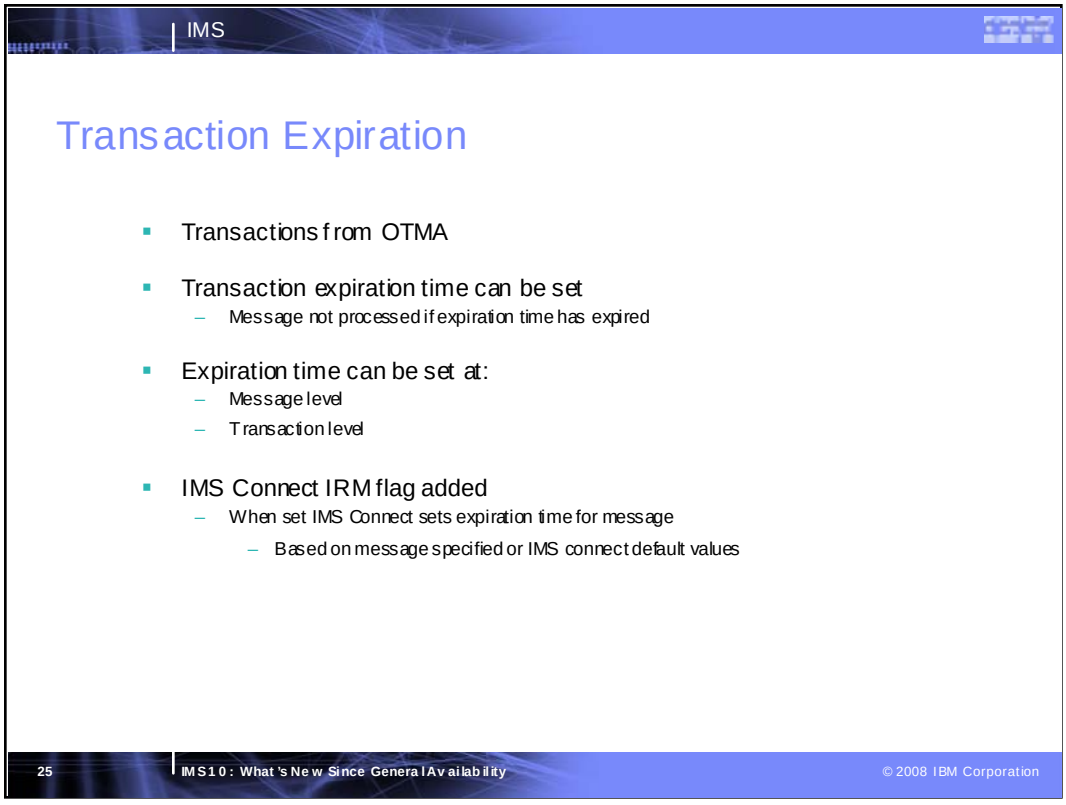
IMS 10 adds 'Message Flooding' status to the resources being monitored. If the message queues become 80% full, the 'warning' flag is set. If the message queues fill up, the 'unavailable' flag is set.

IMS connect listens to this new system status message from OTMA. It passes the system status data to its User Message Exits and returns the data with the VIEW and QUERY commands.



A client application can submit a transaction request to IMS. IMS receives the transaction, processes it, and sends a reply. If IMS does not have the resources to process the transaction in the allotted timeframe, the client application might time out the transaction call. By the time the transaction is processed and a reply is sent, the client application no longer wants the response message.

Processing unwanted transactions in IMS increases processing costs and CPU cycles.



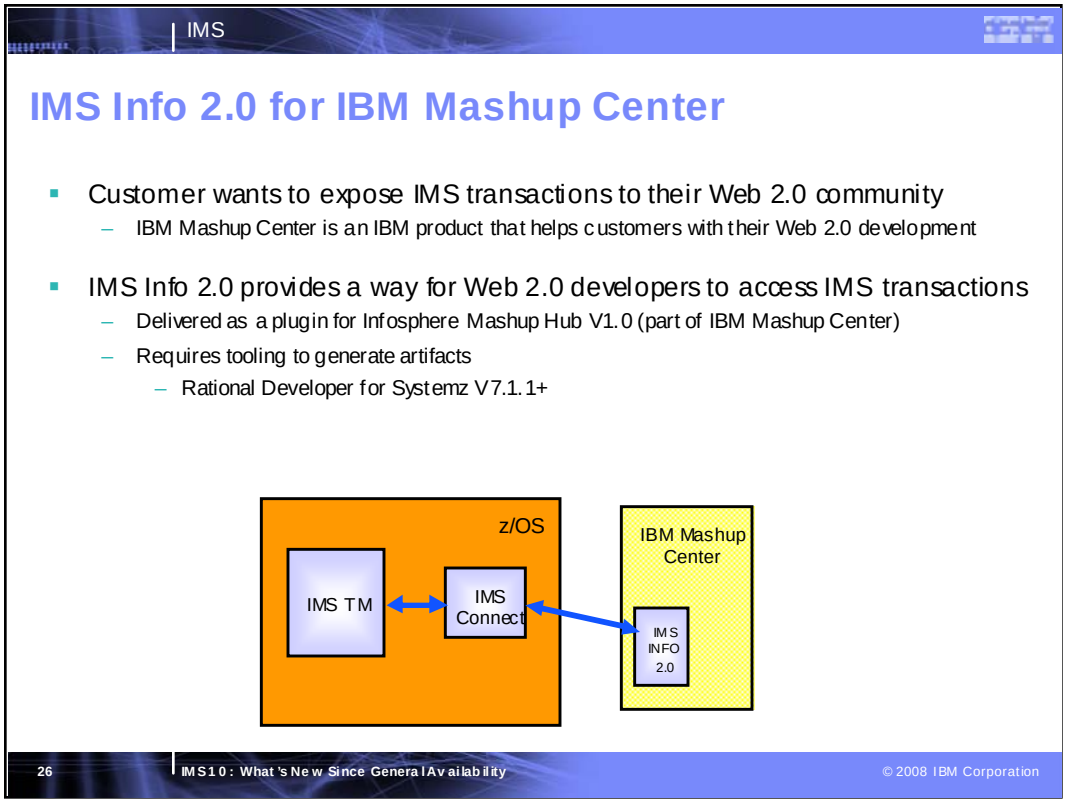
OTMA Transaction Expiration gives the customer the ability to set an expiration time for an individual message or for a transaction. If the expiration time has expired, the message is not processed.

Transaction level expiration time can set in the following ways:

•TRANSACT macro from the IMS Stage 1

•The output destination creation exit DFSINSX0 can be set the expiration time when dynamically creating a transaction.

IMS Connect exploits this function by introducing a new flag in the IRM (IMS Request Message). When this flag is set IMS Connect computes the transaction expiration time based on the IRM_TIMER value that is specified in the message or the default timeout value for IMS connect (set with the TIMEOUT parameter in the IMS connect configuration member).

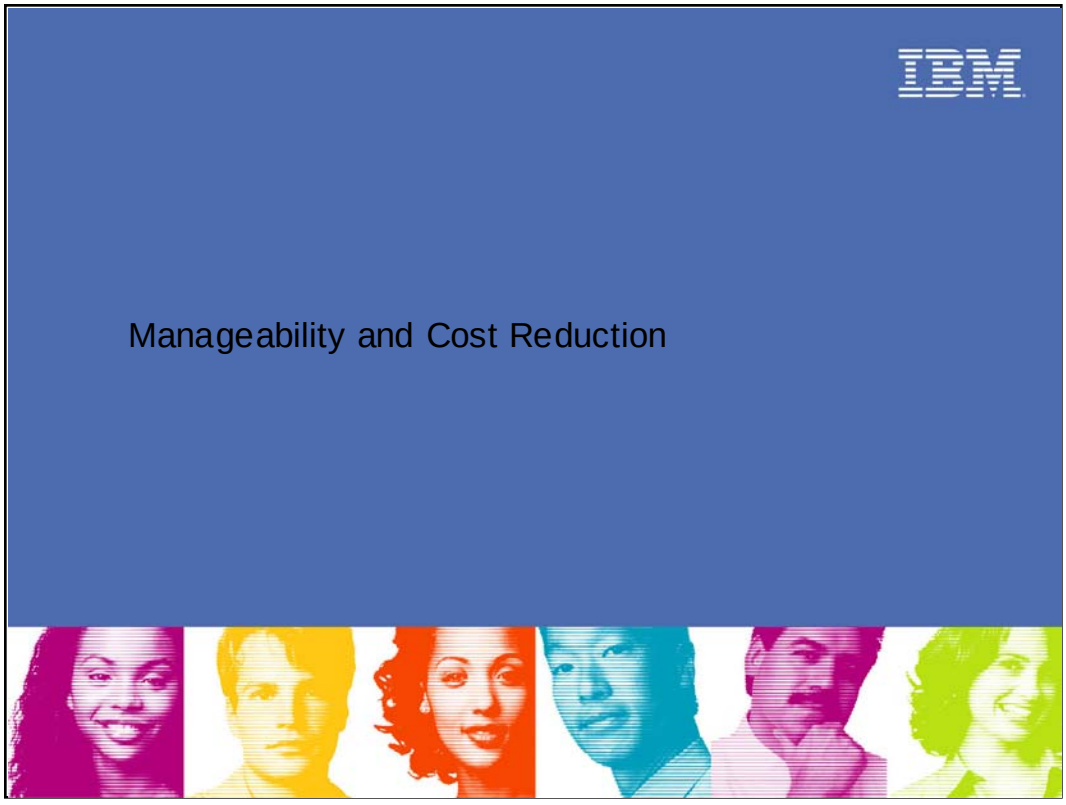


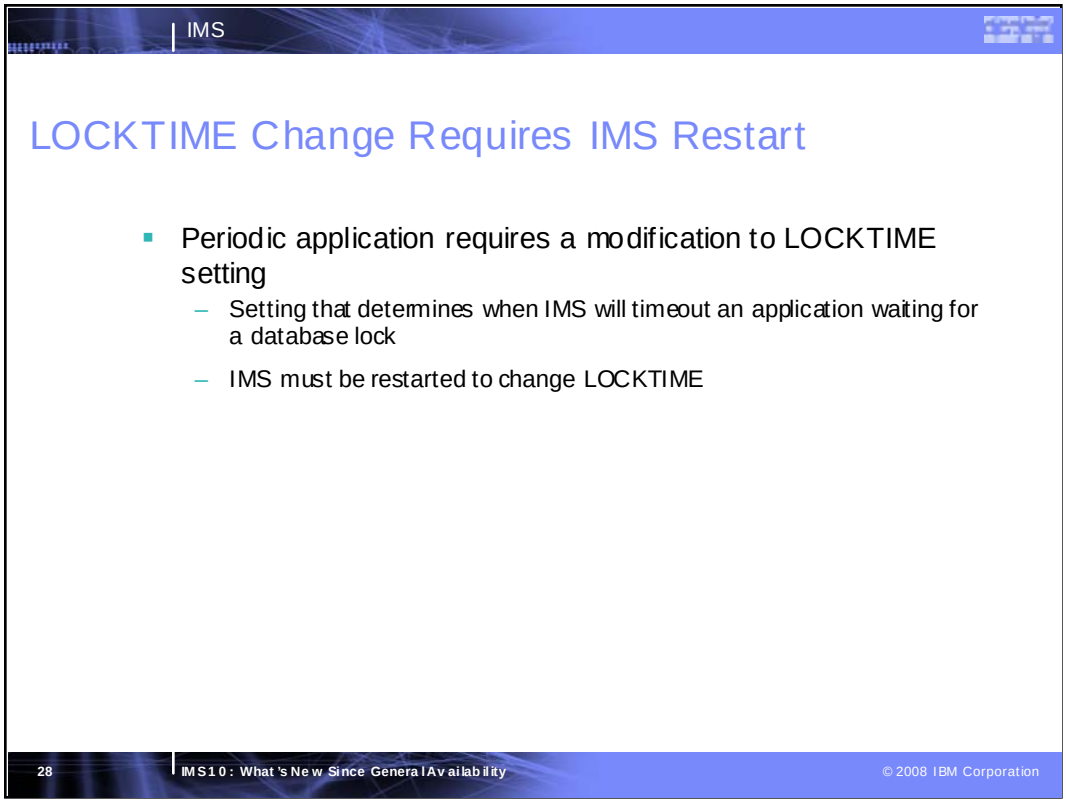
IMS Info 2.0 allows the IMS TM Customer to bring their IMS transactions into the Web 2.0 community as RESTful services.

IBM Mashup Center is a web based application running on WebSphere V6.1.13+ and Apache Derby. It contains the IMS Info 2.0 plugin that will allow a Web Application Developer to create an IMS RESTful service.

Rational Developer for System z V7.1.1+ is required to generate an IMS Info 2.0 correlator file based on COBOL copybooks or PL/I source files which is imported into the IMS Info 2.0 plugin on Mashup Hub and the XML converters for converting the IMS Message from XML to bytes and vice versa.

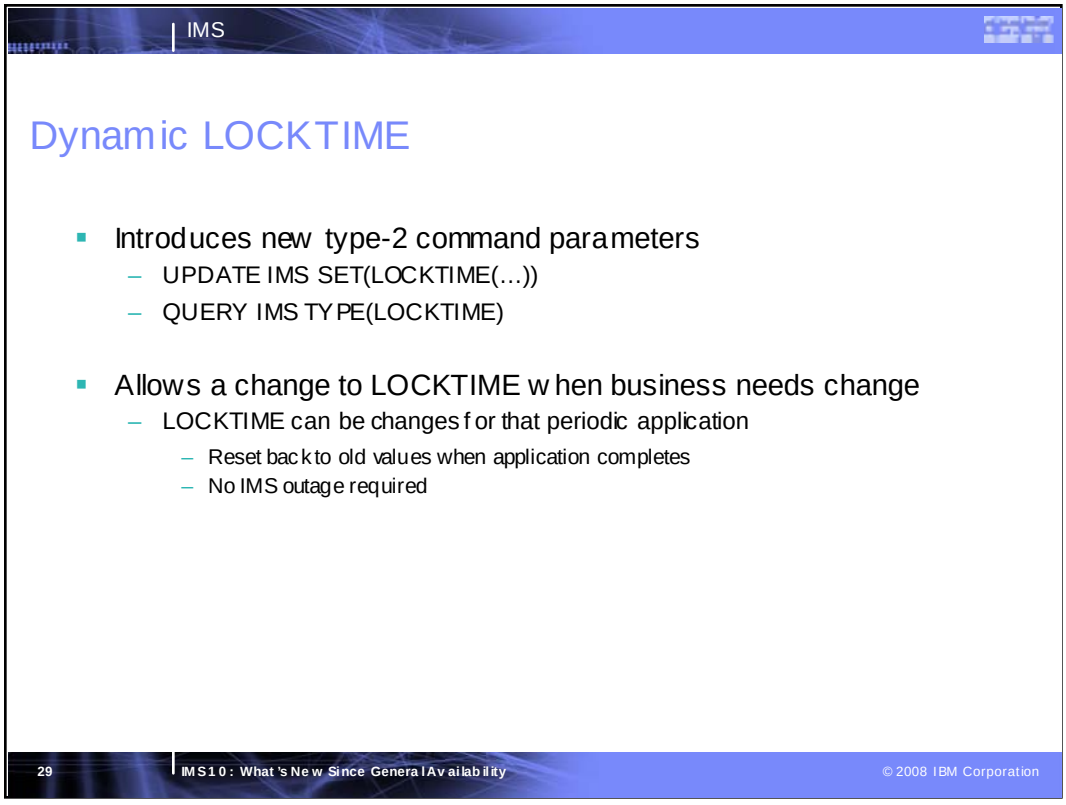
Once the service has been created, IBM Mashup Center can be used to create composite feeds and services based off of the IMS RESTful service which can then be consumed, remixed, and mashup in the Web 2.0 community.


IBM
Manageability and Cost Reduction



The LOCKTIME parameter tells IMS and IRLM how long to hold a database lock before it is released. A database lock can hold up other applications that are trying to access the same data.

A periodic application (for instance, one that runs once a quarter) with a large number of database updates may require a different setting for LOCKTIME. An outage must occur because IMS must be recycled in order to change the LOCKTIME value.

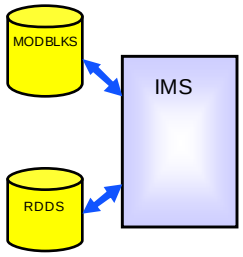


A parameter is added to the UPDATE type-2 command that allows the LOCKTIME parameter to be dynamically modified. Another parameter is added to the QUERY type-2 command that returns that current setting of LOCKTIME.

These commands allow the user to change the LOCKTIME value without taking an IMS outage.

DRD : Unable to Port Resource Definitions

- Automatic Import/Export is part of base IMS 10
  - All MODBLKS resource definitions exported or imported
  - System RDDS cannot be used by another IMS
    - IMS ID part of data
  - RDDS always overwritten, cannot append
- Customer Requirements
  - Move resource definitions to another IMS
  - Work with subset of resources

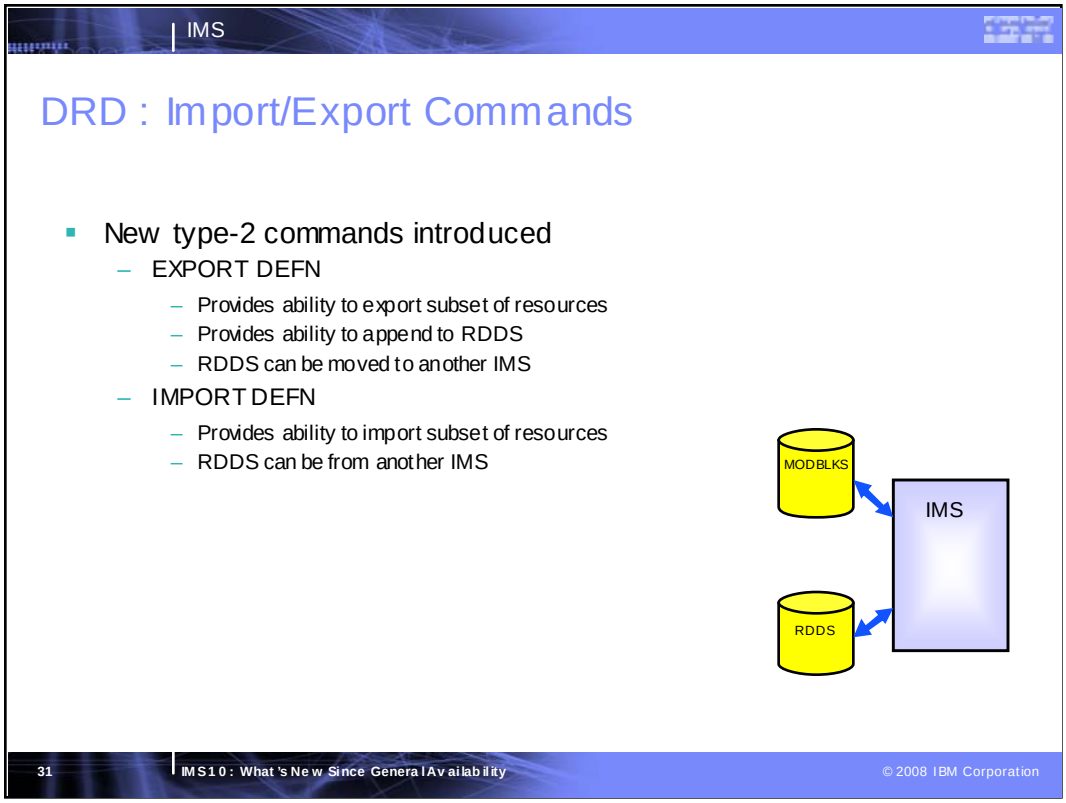


Dynamic Resource Definition is an IMS 10 enhancement that allows users to dynamically add, update, and delete MODBLKS resources (databases, programs, routing codes and transactions) while IMS is active. All changes made dynamically are logged and recoverable across a warm or emergency restart. In order to recover the changes across an IMS cold start, the current resource definitions must be saved to an external data source while IMS is up, and then recovered from the external data source during or after the cold start.

The base V10 code provides an automatic export function which can be enabled so that all of an IMS's resource definitions are automatically exported to a system RDDS at checkpoint time (if there have been definitional changes made since the last checkpoint). The base V10 code also provides an automatic import function which can be enabled so that an IMS's resource definitions are automatically imported from a system RDDS during cold start.

A Resource Definition Data Set (RDDS) is the dataset into which IMS saves its resource definitions. Automatic import and automatic export require a set a system RDDSs. A system RDDS belongs to the IMS and cannot be shared between IMSs. It contains all of the resource definitions for the IMS.

Customers would like to be able to export definitions with a command. They would also like to be able to port definitions from one IMS to another, for example when cloning a system.



The EXPORT DEFN and IMPORT DEFN type-2 commands are introduced by this enhancement.

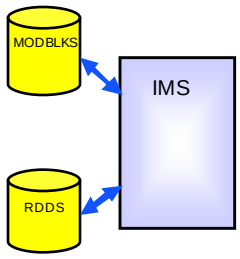
The EXPORT DEFN command can be used to export resource definitions to a resource definition data set (RDDS) and the IMPORPT DEFN command can be used to import resource definitions from an RDDS. The RDDS can be one of the system RDDSs used by automatic import and automatic export, or the RDDS can be a non-system RDDS. A non-system RDDS can be shared between IMSs and can contain a subset of an IMS's resource definitions.

When exporting to a system RDDS, all of the IMS's resource definitions must be exported. When exporting to a non-system RDDS, all or some of the IMS's resource definitions may be exported. The non-system RDDS can then be used by another IMS to import the resource definitions. When exporting to a system RDDS, all definitions in the system RDDS are overwritten with the definitions being exported. When exporting to a non-system RDDS, the user has the option of overwriting the existing definitions with the new definitions, or appending the new definitions to the end of the data set.

When importing from a system RDDS, all of the IMS's resource definitions must be imported. When importing from a non-system RDDS, all or some of the resource definitions may be imported. The resource definitions in the non-system RDDS can be from the same IMS or another IMS.

DRD : Unable to Build or Recover RDDS Offline

- RDDS Extraction Utility (DFSURDD0) – base V10
  - Offline Batch Utility to convert RDDS data into:
    - SYSGEN macro statements
      - APPLCTN, DATABASE, RTCODE, TRANSACT
    - Type-2 CREATE commands
      - CREATE DB, CREATE DBDESC, CREATE PGM…
- What happens if the RDDS is lost?
  - How can I create a new RDDS offline?
    - MODBLKS dataset
    - SYSGEN macros
    - IMS Log



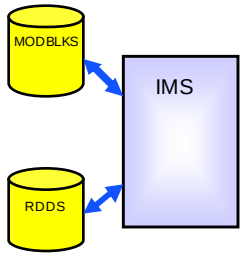
The RDDS Extraction utility is provided as part of the IMS V10 base code. It is an offline batch utility that can be used to convert the definitions in an RDDS into either SYSGEN macro statements, or type-2 CREATE commands. This utility could be used if you had to fall back from DRD to non-DRD. You could use the utility to generate SYSGEN macros for the resources defined in the RDDS. The macros could then be included in a MODBLKS gen to generate a MODBLKS data set with the current definitions.

What happens if you are running in a DRD environment and you lose the RDDS that contains your most current resource definitions? Currently, IMS must be active in order to create an RDDS (automatic export or EXPORT command).

Customers have requested the ability to generate an RDDS without requiring an IMS to be up. The want to be able to create an RDDS from SYSGEN macros, or from a MODBLKS dataset or from the IMS log.

DRD : Maintenance Utilities

- DRD Maintenance Utilities provide a way to move resource information
  - Input
    - MODBLKS dataset
    - SYSGEN macros
    - IMS log
    - RDDS
  - Output
    - RDDS
    - SYSGEN macros
    - CREATE commands
- ISPF panel driven

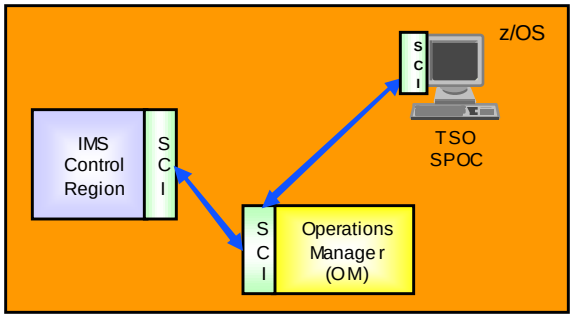


With the enhancements being added with the DRD Maintenance Utilities and the functions provided by the RDDS Extraction Utility, the customer now has the ability to move resource information between an RDDS, a MODBLKS dataset, and a set of SYSGEN macros. The utilities allow a customer to recreate an RDDS without an active IMS.

These ISPF panel driven utilities allow a customer to create an RDDS by using a MODBLKS dataset, SYSGEN macros, or an IMS log. The RDDS can be used as input to create the equivalent SYSGEN macros or CREATE commands that will create the resources in an active IMS.

TSO SPOC Print Options

- TSO SPOC users have single options for command output
  - Some commands have large amount of output
- TSO SPOC print options allow user to set output style
  - Wrap, by column, or by resouce



This enhancement allows the user to control the format of printed output from TSO SPOC, REXX SPOC, batch SPOC and related programs.

The printed output can be formatted in three styles:

WRAP

Lines of data are fit to the width of the print file and wrapped to the next line as needed

Works this way today

BYCOL

Lines of data are grouped together by column

Example: all records that include columns 3-8 are printed followed by all records that include columns 9-12

BYRSC

Lines of data are grouped together by resource

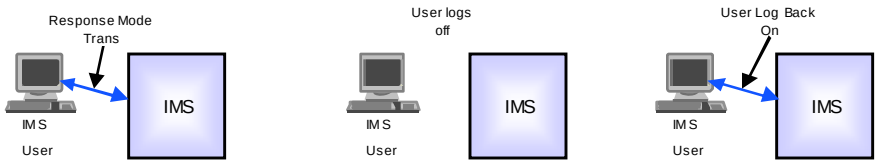
Example: all records that include resources 1-7 are printed followed by all records that include resources 8-15

The new formatting options can be selected from the TSO SPOC preferences panel or as input parameters to the batch programs.


IBM
Growth and Scalability

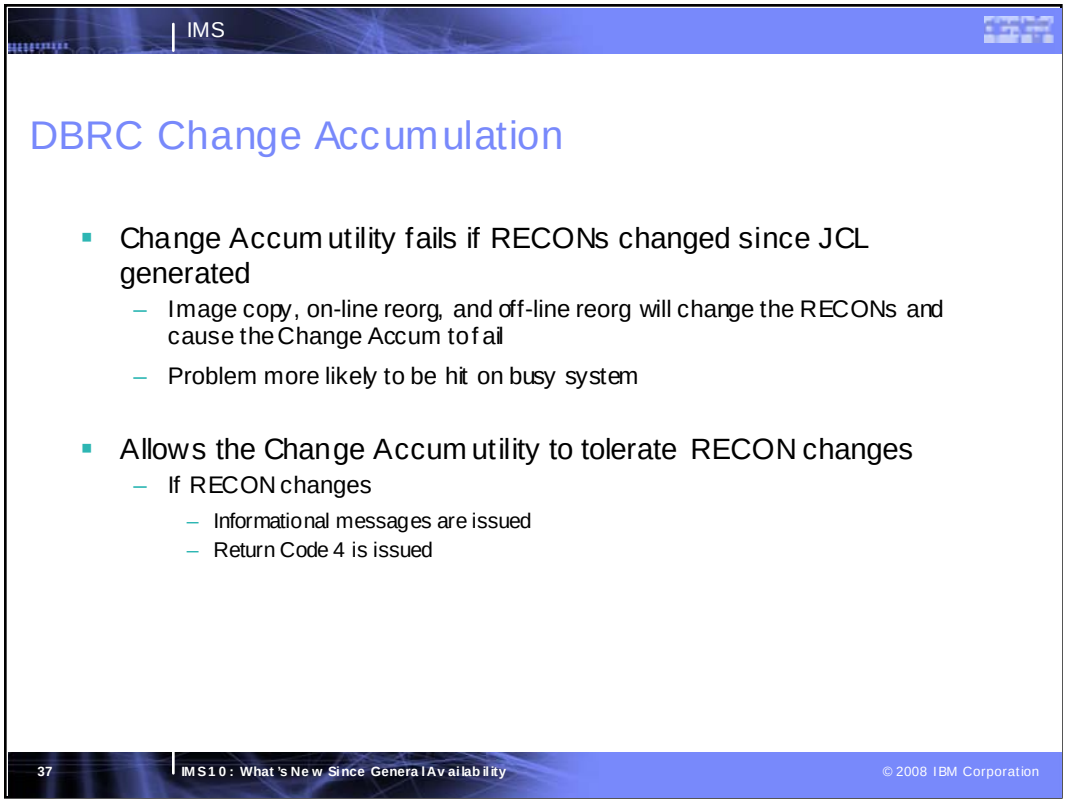
FF Response Mode Recovery Option

- Full Function Response Mode not retained after signoff
  - User not in response mode when they log back on
    - Can enter new transactions without receiving output from original transaction
- Option added to recover when user or terminal signs back on
  - User in response mode when they log back on
    - Cannot enter new transaction until output received
  - Does not recover across a restart



A response mode transaction prevents a user from entering another transaction until they have received the output from the first transaction. If the user logs off before they receive the response and then logs back on, the response mode state is lost. If they enter a non-response mode transaction before they receive the output that is queued from the previous transaction, the response from the previous transaction can be received before the response to the transaction that was just entered. This can be confusing.

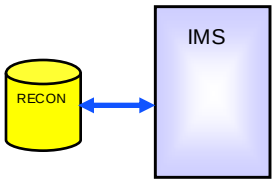
When this option is used, the user logs back on and they are still in response mode. Another transaction cannot be entered until the output from the first transaction is received.



DBRC will use the purgetime in the DB0 statement from the Change Accumulation JCL during validation of the JCL. If DBRC finds a later image copy, online reorganization, or offline reorganization, new informational messages are issued to inform the user and return code 4 is issued. New error messages are received if DBRC selects an earlier purgetime or if changes after an offline reorganization are found and the Change Accumulation execution will fail. DSP0711I (invalid purgetime) is no longer issued.

DBRC DELETE.DB Performance Improvements

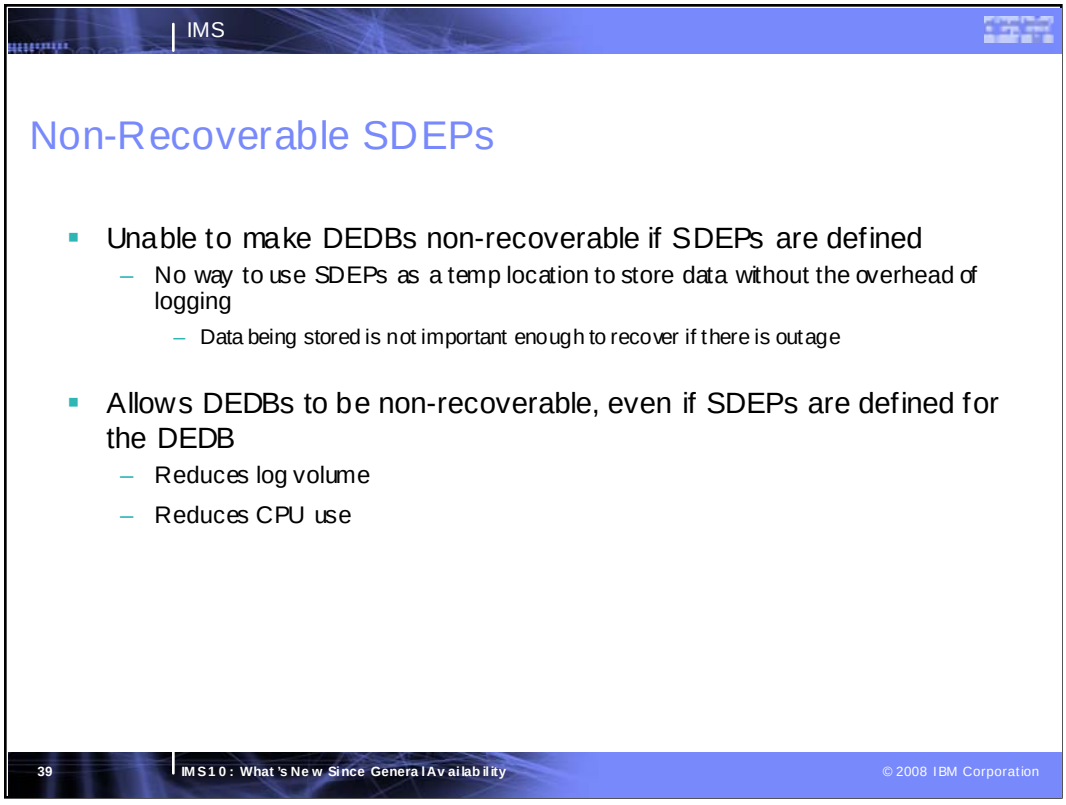
- Deleting a large FP DB from the RECONs can cause a system slow down
  - DEDB with large number of areas
  - Ties up the RECONs with large amount of I/O
    - Can prevent other work from processing
- Method of processing DELETE.DB modified
  - Fewer RECON accesses, less I/O



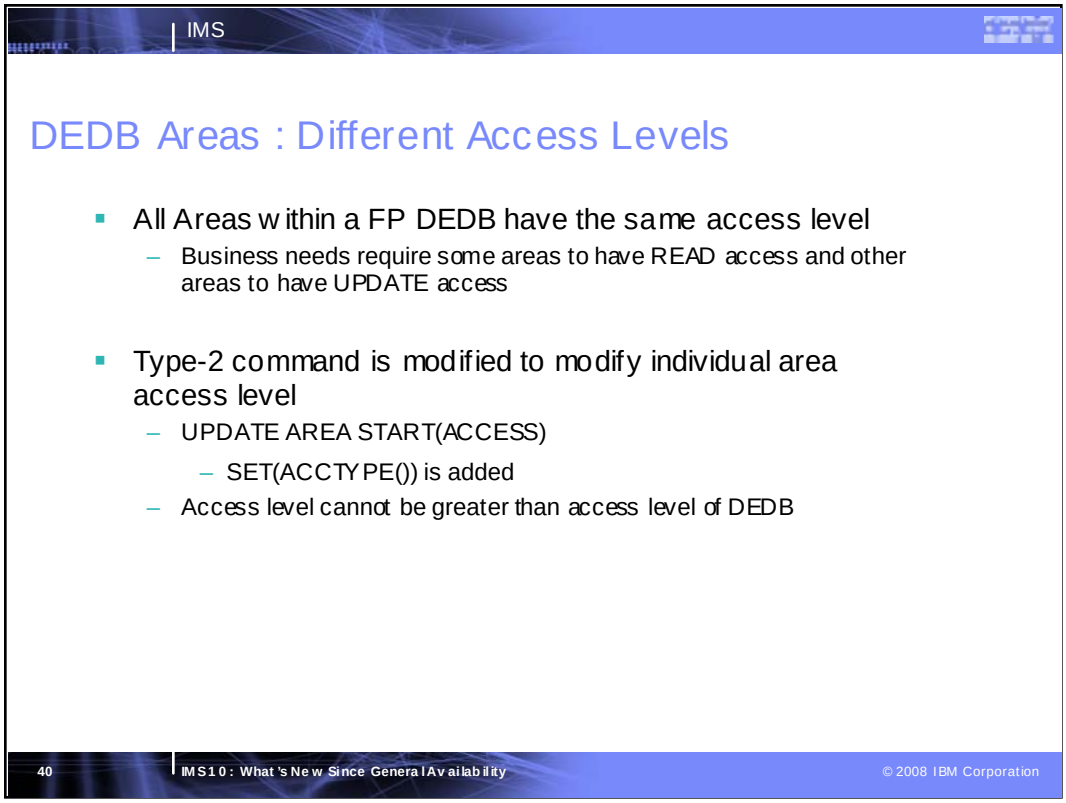
This enhancement reduces the time it takes to process the DELETE.DB function. DELETE.DB is often done for a DEDB with a large number of areas with the intent to add new areas to the 'middle' of the DB as part of a reorganization. When the new areas have been added, the database is added back to the RECON with a INIT.DB/DBDS/ADS/etc.

The main reason the processing of DELETE.DB takes so long is that some records are changed many times and a before image is saved before each change. This results in a large number of I/Os and can tie up the RECON for a long time, preventing other accesses to the RECON (for instance, when a database is allocated).

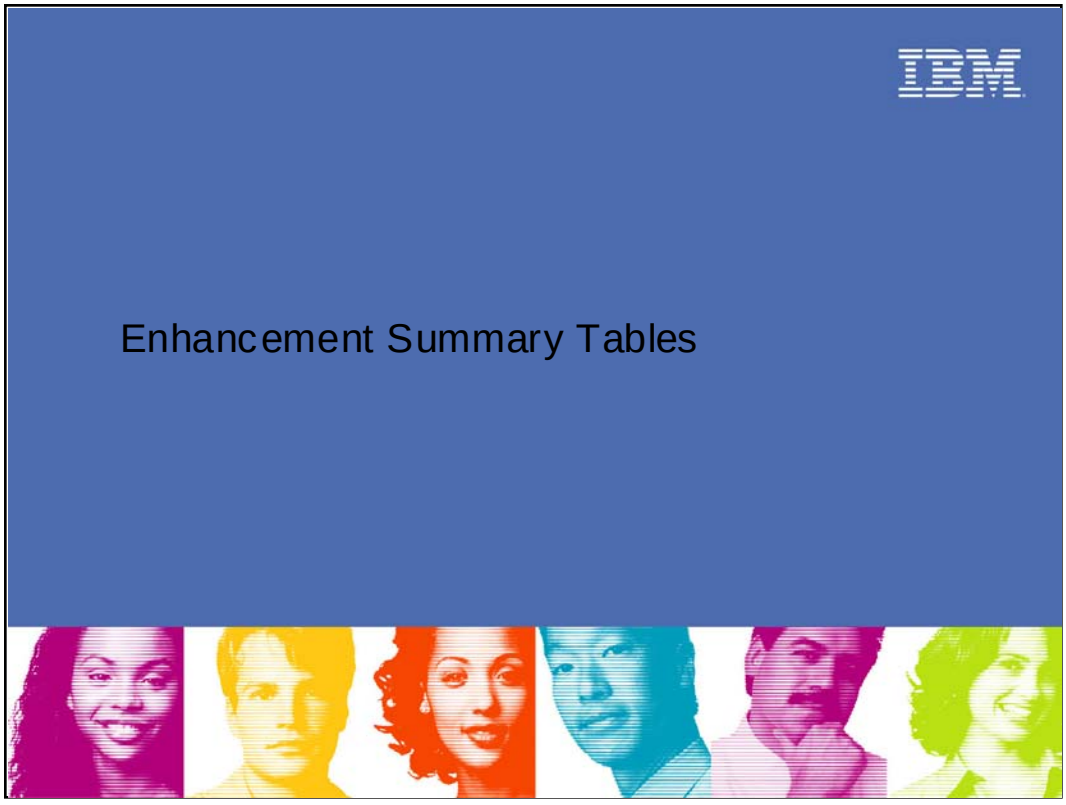
This enhancement reduces the number of I/O to the RECON. This allows other processes to access the RECON much more quickly.

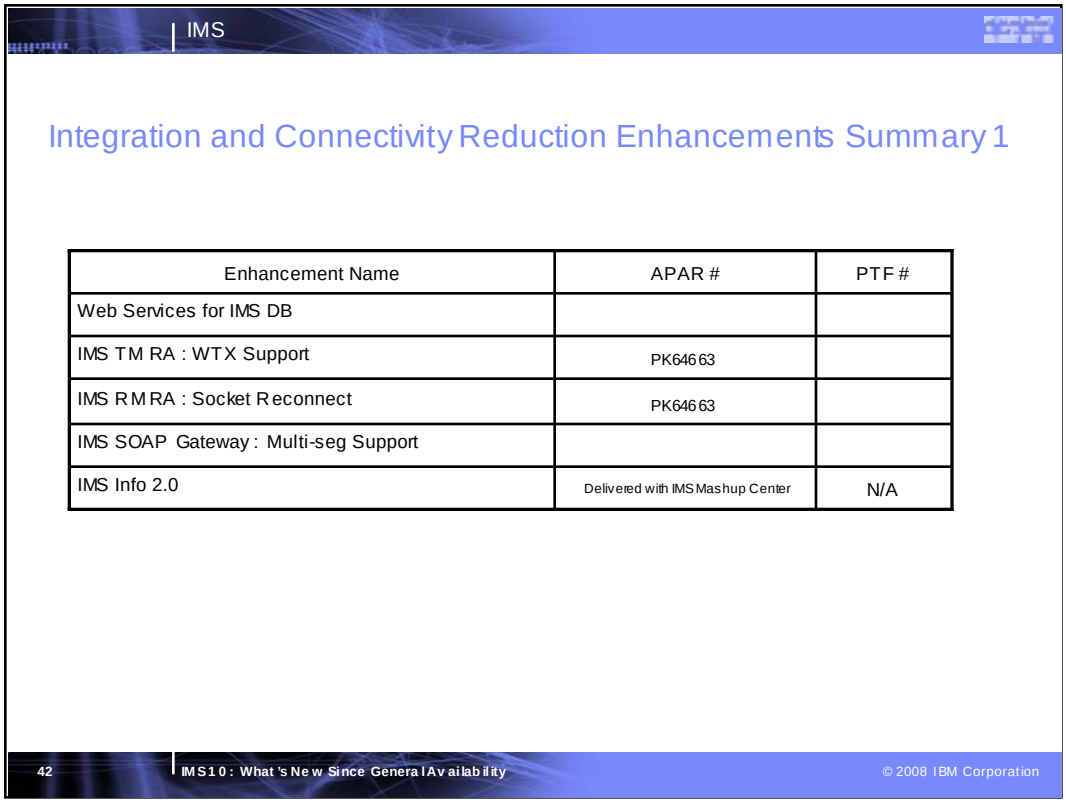


Prior to this change, Fast Path allowed Nonrecoverable DEDBs as long as no SDEPs were defined. This change lifts that restriction. It allows the SDEP portion of an area to be used to hold expendable data not needing recovery if IMS should fail. An example of how this can be used is as a large scratch pad for journaling and debugging info. This becomes an attractive means for storing sequential data without all the overhead of logging database change records.



Currently, all Areas of a DEDB have the same access of the DEDB. This enhancement allows users to have the capability to change Area access of one or more DEDB Areas of the DEDB to have different access via a type-2 command interface


IBM
Enhancement Summary Tables



Here is a list of Integration and Connectivity enhancements that are for IMS components that are not part of IMS Connect and IMS TM. It also includes enhancements that are delivered with products that are not part of IMS.

A PTF number or 'N/A' in the PTF # column indicates that the enhancement is available.

Integration and Connectivity Reduction Enhancements Summary 2

Enhancement Name	APAR #	PTF #
Synch Callout		
IMS Connect API		
IMS Connect Client ID Hash Table Support	PK57574	
IMS Connect Duplicate Client ID Option		
OTMA Monitoring Support	PK70458 (IMS TM) PK70960 (IMS Connect)	
OTMA Transaction Expiration Support		
Retrieve ALT-PCB Output	PK61174	

Here is a list of Integration and Connectivity enhancements that are for IMS Connect and IMS TM.

A PTF number in the PTF # column indicates that the enhancement is available.

Manageability and Cost Reduction Enhancements Summary

Enhancement Name	APAR #	PTF #
Dynamic LOCKTIME		
DRD IMPORT Command	PK66682	
DRD EXPORT Command	PK66704	
DRD Maintenance Utilities	PK63186	
SPOC Print Options	PK50292	UK35086

Here is a list of Manageability and Cost Reduction enhancements.

A PTF number in the PTF # column indicates that the enhancement is available.

Growth and Scalability Enhancements Summary

Enhancement Name	APAR #	PTF #
FF Response Mode Recovery	PK53989	UK32360
Change Accum	PK53223	
DBRC DELETE.DB Perf Enh		
Non-Recoverable SDEPs	PK56321	UK35862
DEDB Area Different Access		

Here is a list of Growth and Scalability enhancements.

A PTF number in the PTF # column indicates that the enhancement is available.