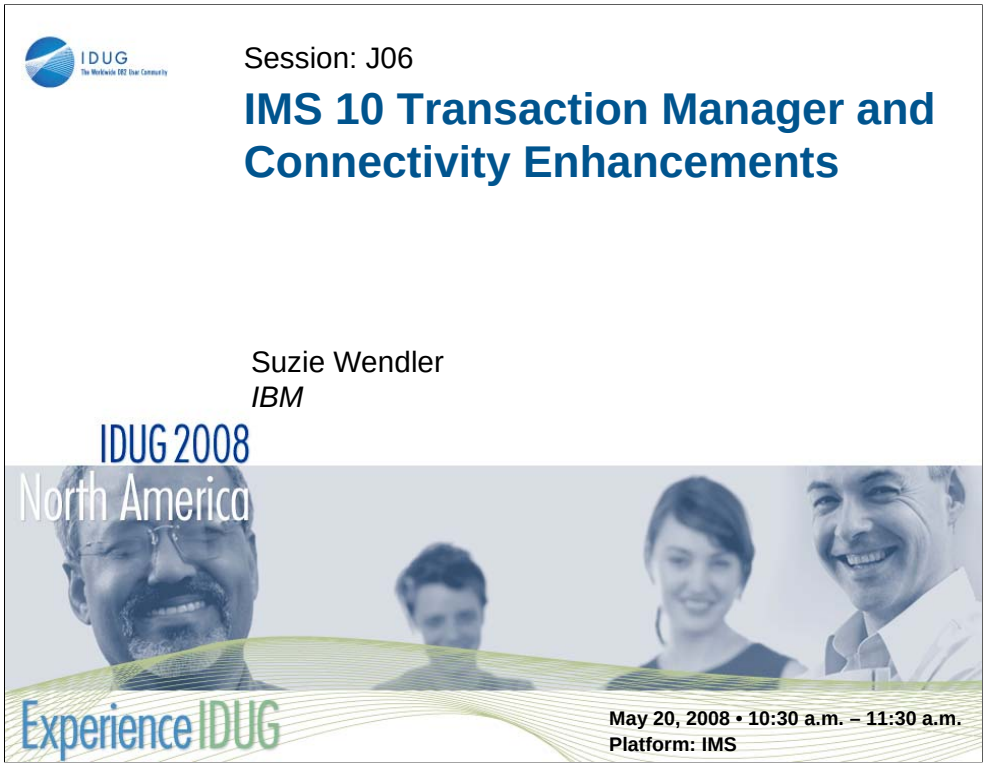
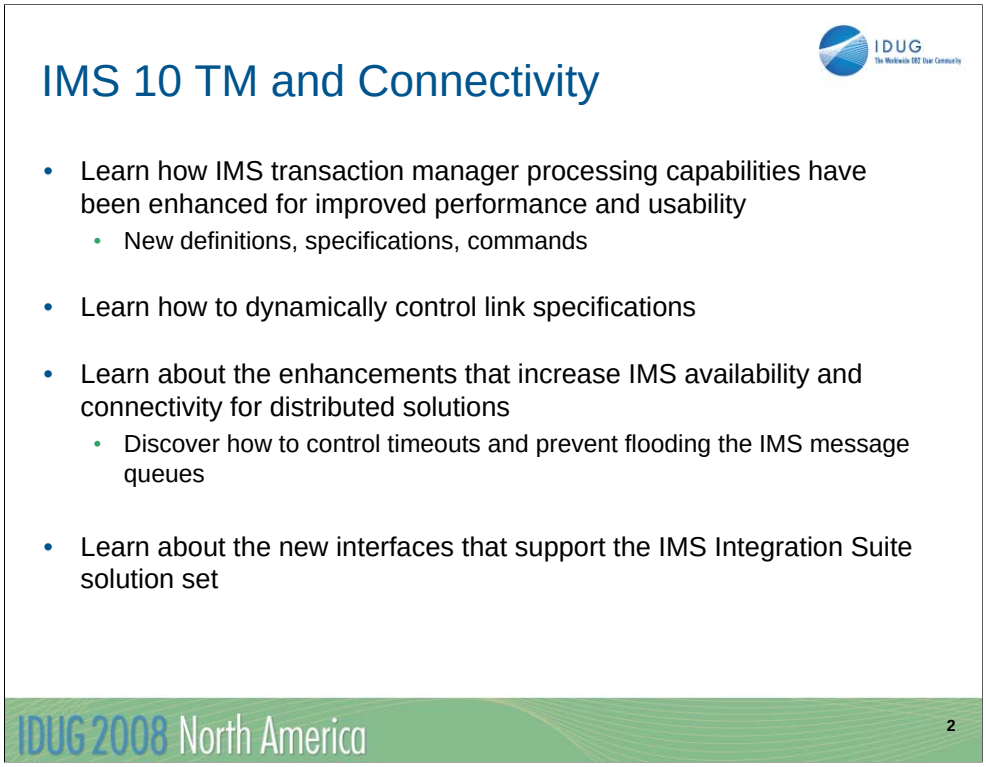
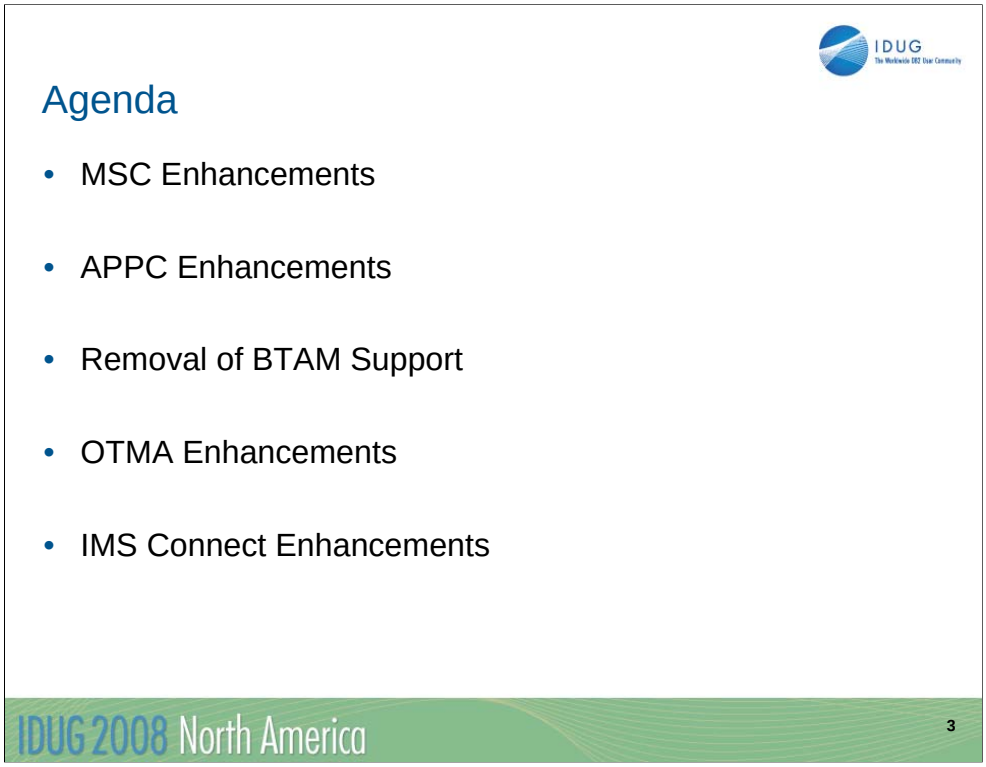


As IMS environments continue to grow and be integrated into distributed solutions, performance, availability and connectivity continue to key areas of focus. This presentation describes how IMS Version 10 has enhanced the transaction manager processing capabilities to address these areas to allow IMS environments to continue to protect and enhance their investment in IMS solutions. This presentation includes details on new capabilities in areas such as MSC, APPC, OTMA and IMS Connect.

Copyright IBM Corp. 2006

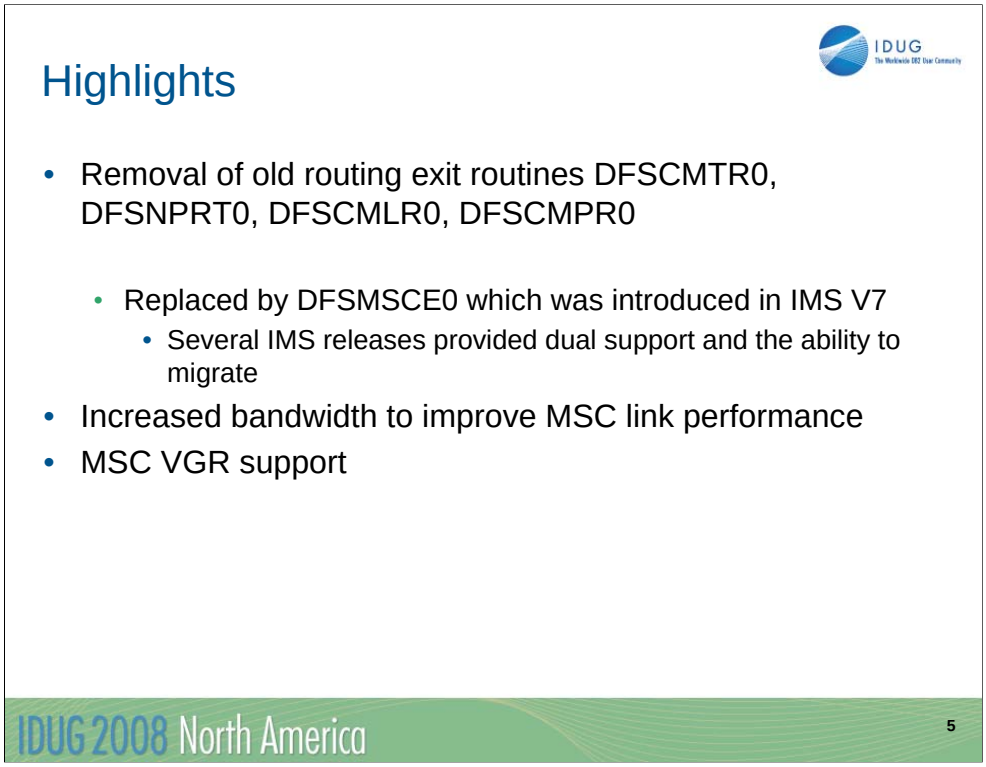


IMS Version 10 introduces many enhancements in the transaction manager as well as connectivity areas. This presentation provides an overview as well as some detail of the major capabilities.



The topics shown on this visual outlines the major areas that are discussed in the presentation.

MSC users will be happy to discover that IMS 10 provides several major enhancements that address bandwidth and performance considerations.



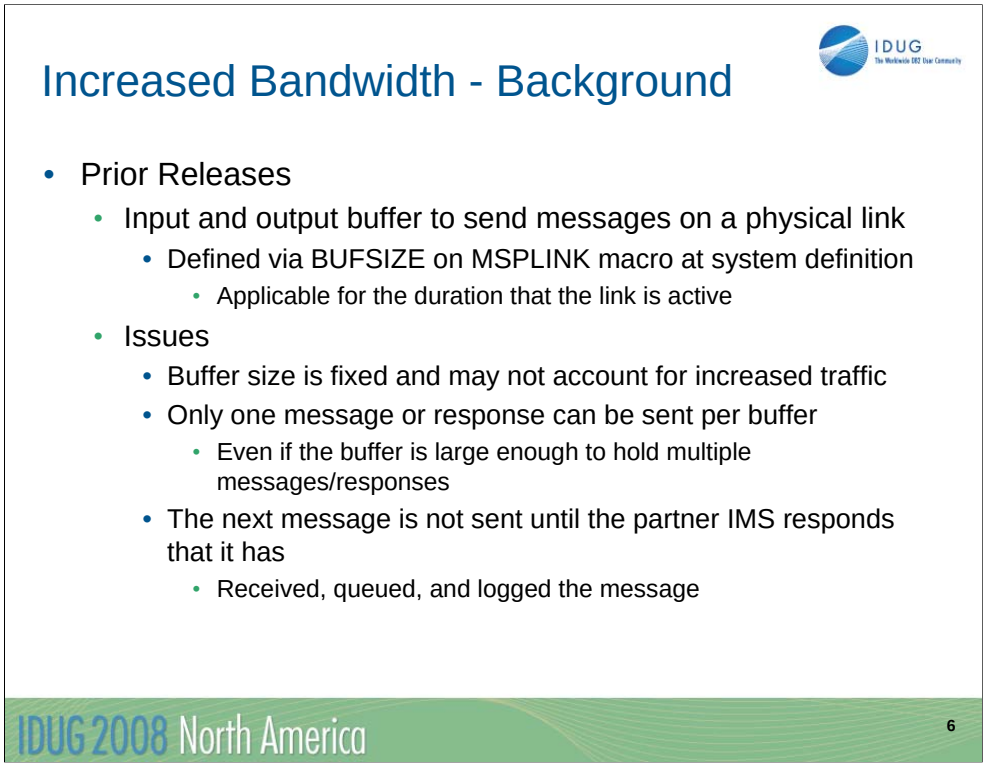
The MSC enhancements fall into three major categories as listed on this visual.

The first piece of information and a key migration consideration is that IMS 10 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 which include: the MSC Terminal Routing Exit (DFSCMTR0), MSC Input Message Routing Exit (DFSNPRT0), MSC Link Receive Routing Exit (DFSCMLR0), and the MSC Program Routing Exit (DFSCMPR0). As a reminder, DFSMSCE0 was introduced in IMS V7 to provide an opportunity to migrate over several releases. Note: DFSMSCE0 can be used in both MSC and non-MSC environments, although not all routing options apply to non-MSC systems.

The second bullet on this visual references an improvement that addresses the performance and bandwidth requirements of high-volume MSC systems. Many of your systems today, those of you that use MSC, may have had to define a large number of parallel MSC links between pairs of IMS systems to support your throughput needs. This scheme can complicate the operations of the IMS systems and can further add to the complexity of the system configurations as well as load balancing schemes. IMS 10 introduces several enhancements to increase MSC bandwidth.

Additionally as shown on the third bullet, IMS 10 enhances the VTAM Generic Resources (VGR) capability to include the MSC environment.

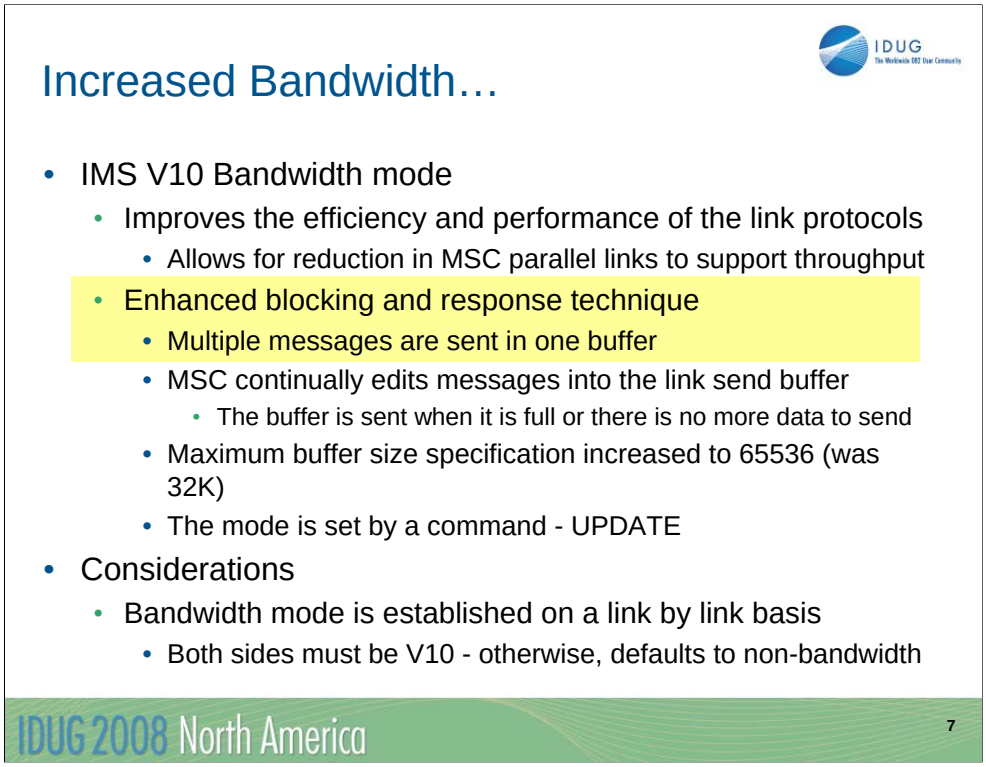
Copyright IBM Corp. 2006



To understand the changes for MSC bandwidth, a review of the issues in releases such as IMS version 8 and IMS version 9 are provided on this visual.

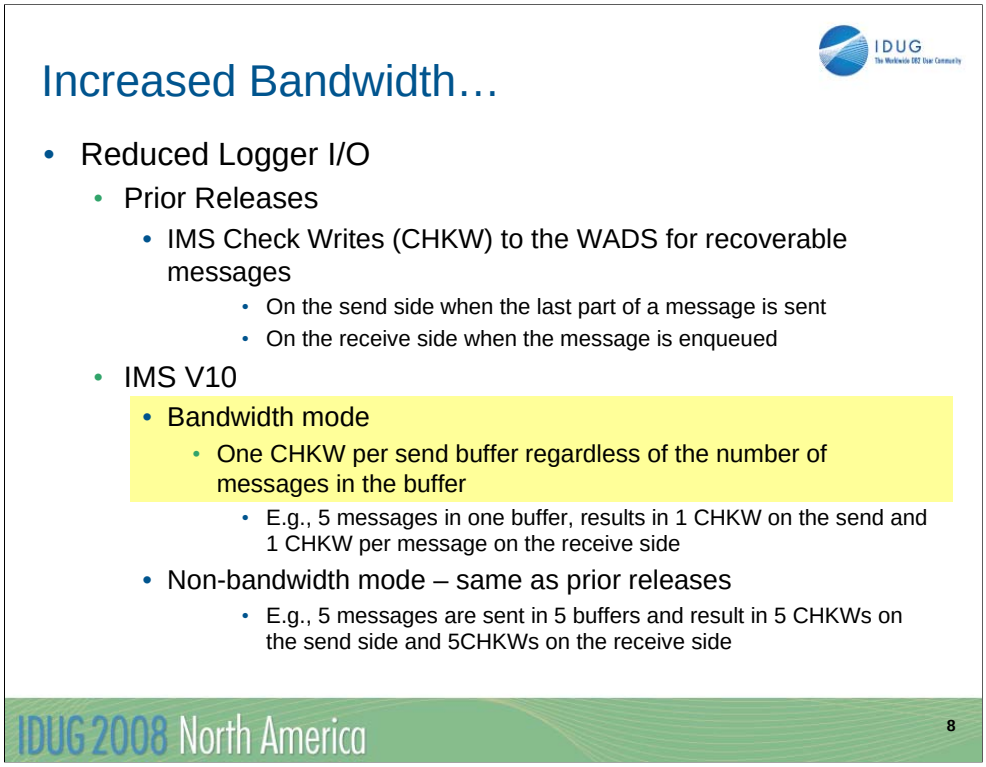
The definition of MSC physical links provides a specification for a BUFSIZE on the MSPLINK system definition macro. The buffer sizes specified have therefore been fixed at system definition. When the link is started, an input and output buffer is acquired and held for the duration of the link restart (i.e. link active), at the specified size to send and receive data, that is messages, across the link. Since each side, or 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.

Copyright IBM Corp. 2006



The MSC bandwidth mode in IMS 10 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 maximum size to 65536 from 32K, more and more messages and responses may be sent simultaneously.

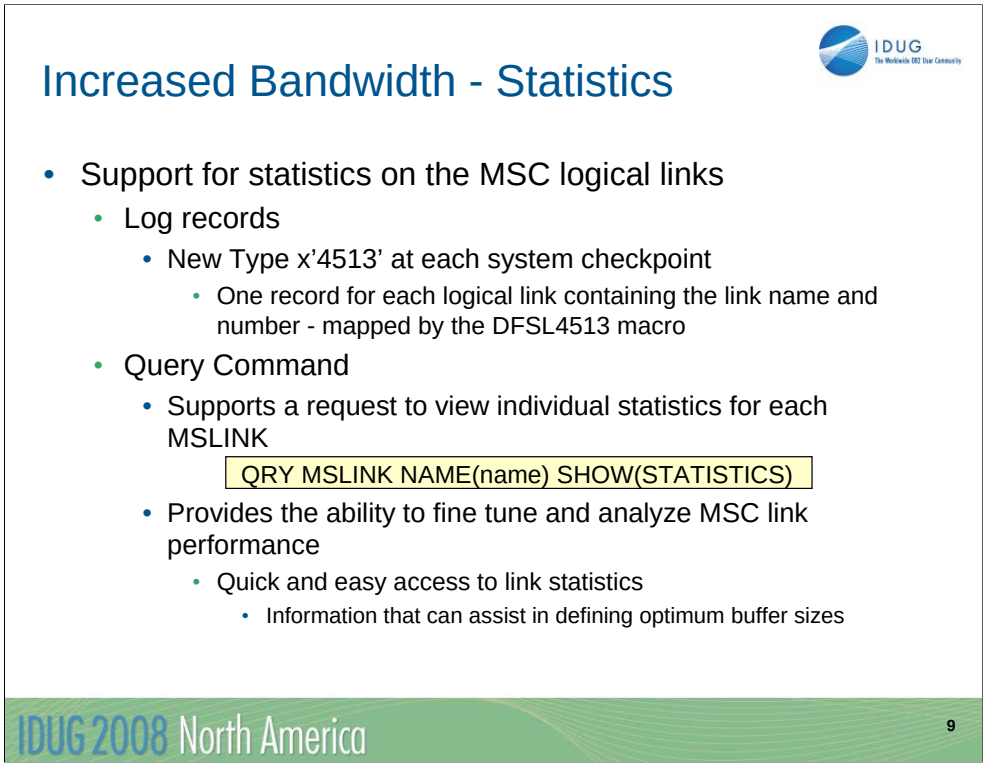
Note that BANDWIDTH mode is not a system definition option. It can only be set ON/OFF by the IMS UPDATE command on a link by link basis for all the CTC, MTM, or VTAM links. The specification is only valid for IMS 10 to IMS 10 connections. The default is non-Bandwidth mode which allows MSC to function as in previous releases.



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.



IMS 10 also provides capabilities to gather and show statistics on the MSC logical links. The enhancements allow quick and easy access to link performance. The first capability includes a new log type x'4513' that is written at each system checkpoint to provide information on each logical link. The second enhancement provides additional parameters on the QUERY command to dynamically request statistics. The information in the resulting display can be used to determine the efficiency of the link and assist in deciding on an optimum buffer size as well as fine tuning the performance environment. Individual statistics for each MSLINK are collected in the DFSMSCWA workarea.

There are three categories of statistics:

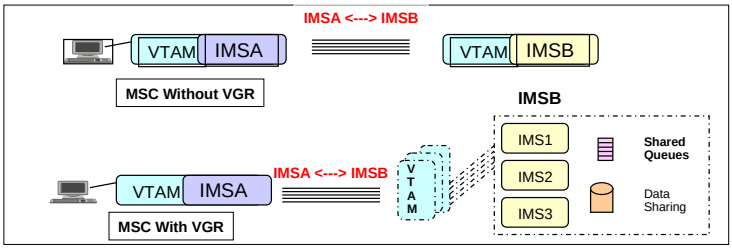
•General information- such as, statistics start time, ITASK dispatch counts, ITASK processing times, and the rate and number of logger check writes

•Send statistics - such as, messages sent, byte count sent, send message sizes, queue manager get counts and times, and send I/O times

•Receive statistics - such as messages received, byte count received, receive message sizes, QMGR insert counts and times, and receive I/O times

MSC VGR Support

- Single MSC system image for the IMS instances in an IMSplex
  - New enhancement to the existing IMS VGR functionality
    - Access through MSC links using VTAM Generic Resources
  - Initial implementation for MSC VGR
    - Local mode with VTAM-managed affinities
      - Local mode - No MSC data in CF, no support for the Resource Manager



IDUG 2008 North America

10

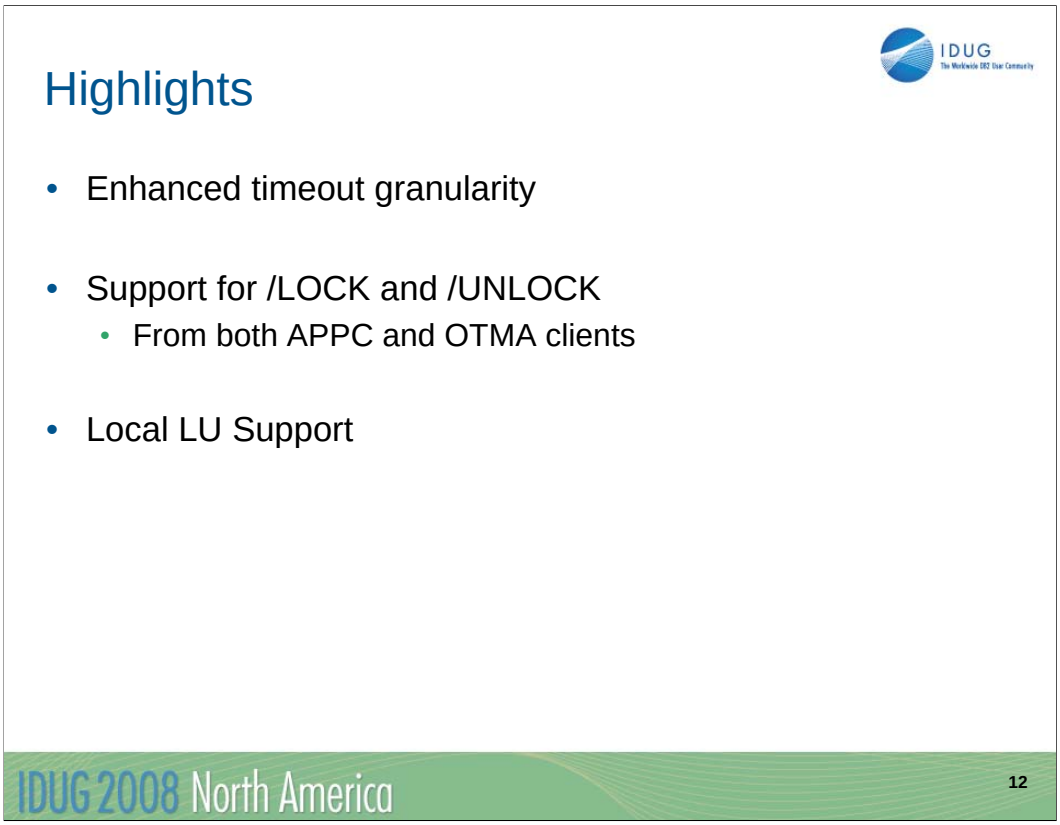
Another major MSC enhancement in IMS 10 is MSC support for VTAM Generic Resources (VGR). This capability enhances the VGR capability in IMS which already includes other terminal types. With the new support, IMS 10 provides a single system view for all the MSC IMS systems in an IMSplex environment.

This function adds the MSC environment as a local mode VGR with VTAM-managed affinities. Additionally, the use of MSC VGR eases the requirements on the definitions that are needed. As shown on this visual, a remote IMS such as IMSA can use a single set of MSC link definitions to access any of the IMS systems in the VGR group IMSB. Likewise, IMS systems that are part of the local VGR group IMSB can clone definitions for MSC access to the single remote IMS.

Note that 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.

APPC Enhancements

Copyright IBM Corp. 2006

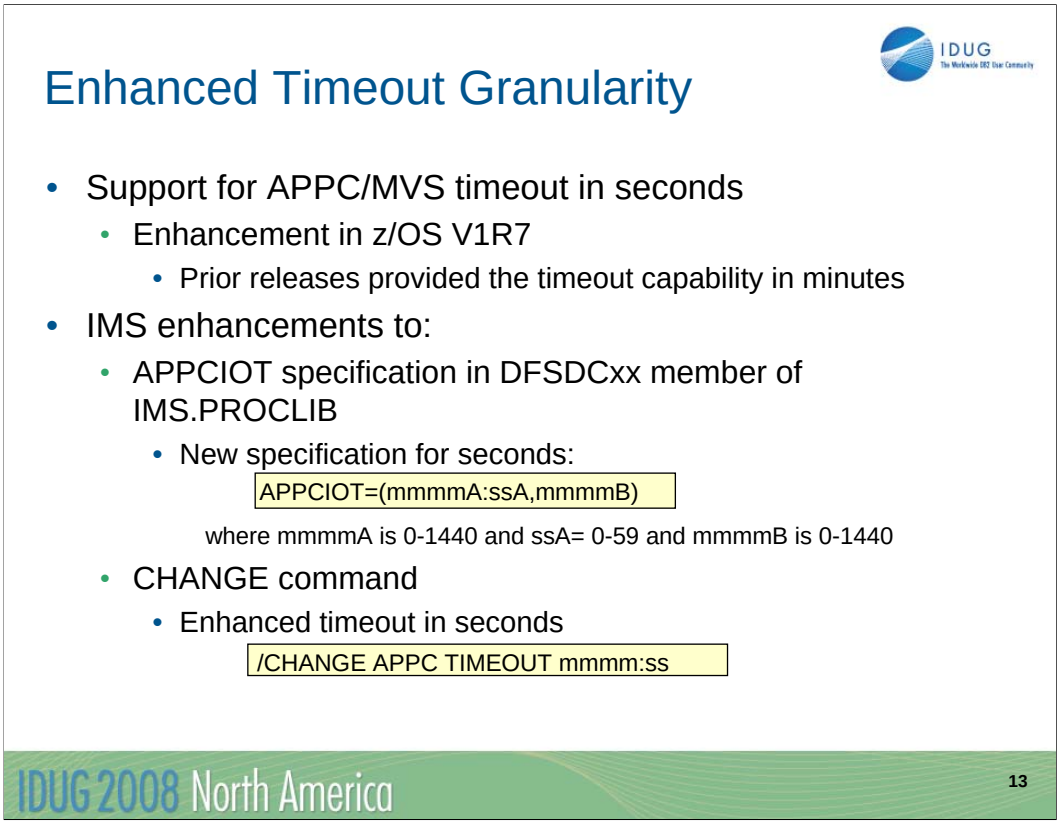


APPC enhancements in IMS 10 include the following three areas:

•First, a greater level of timeout granularity with support for APPC/MVS timeout in seconds to allow resources to be released more quickly

•Second, the ability for APPC and OTMA clients to issue the /LOCK and /UNLOCK commands

•And third, Local LU Support which 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.



The APPC/MVS capability for timeout has been supported by IMS since version 6. In z/OS 1.7, 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 10 and z/OS 1.7, 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 transaction 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.

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
  - The /LOCK and /UNLOCK command functionality remains unchanged

APPC:

```
ALLOCATE LUNAME=IMSLU, TPN=/LOCK. ...
SEND_DATA   DB database1
RCV_AND_WAIT                    <------DFS058 LOCK COMMAND COMPLETED
DEALLOCATE
```

IMS Connect/OTMA:

```
CONNECT
WRITE   LLLL llzz IRM (/LOCK in IRM_TRNCOD)
          llzz /LOCK DB database1
        EOM
READ                            <------DFS058 LOCK COMMAND COMPLETED
DEALLOCATE
```


The next enhancement lifts the restriction to sending in the /LOCK and /UNLOCK commands from APPC and OTMA clients. The support allows the DATABASE, PROGRAM, and TRANSACTION keywords. Note that the commands themselves are not changed.

The 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.

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 second example on this visual shows a request coming in from a remote application through IMS Connect.

Copyright IBM Corp. 2006

Local LU

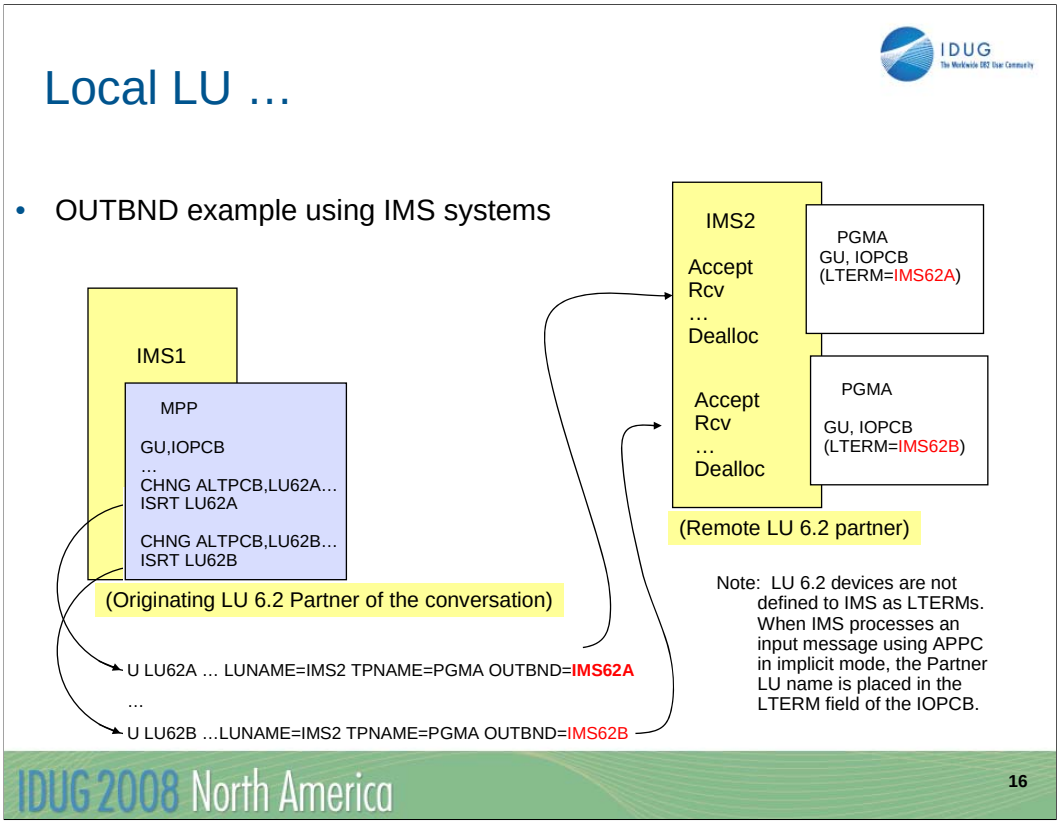
- Enhancement that allows greater control when specifying which LU name is to be used for asynchronous outbound conversations
  - Allows IMS application to request any LOCAL LU name through a new OUTBND descriptor keyword for ALTPCB requests
  - Allows incoming LU name (if not the BASE) to be used for outbound asynchronous responses to the IOPCB

APPCLLU= Y|N in DFSDCxxx

- 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

The third enhancement has to do with Local LU support. To understand this, let me explain some terminology. The APPC/IMS support has always 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. IMS 10 provides greater controls in determining which LU name, if two or more exist, is to be used for asynchronous outbound conversations. The controls can impact both ALTPCB and IOPCB output messages.

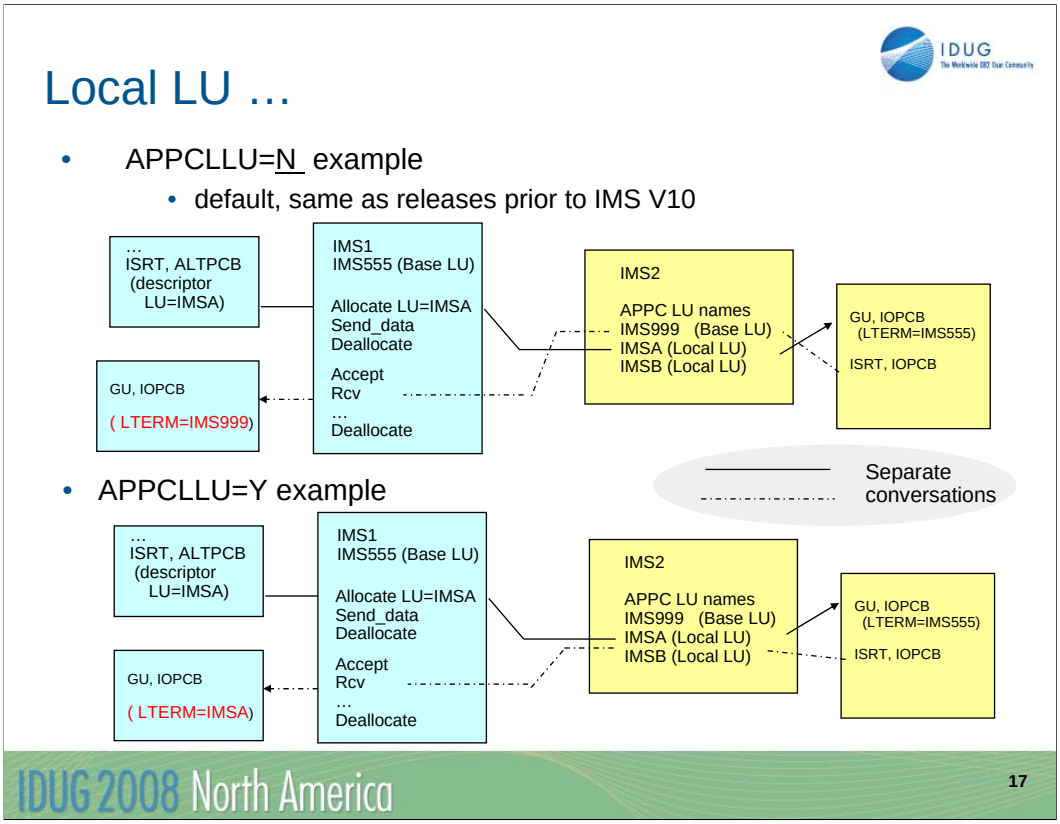
Copyright IBM Corp. 2006



Let us first discuss the controls for the ALTPCB output. 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. On the other hand, the LU 6.2 Edit Exit Routine (DFSLUEE0), if it exists, is always called for inbound and outbound conversations managed by IMS.

The example in this visual shows how specifying different Local LUs in the OUTBND parameter of two descriptors results in the same remote partner seeing the different names. For purposes of the example, the remote partner on the right part of the visual is also an IMS system. In IMS2, the remote partner, the target application PGMA, when it is invoked as a different instance for each message, sees a different LTERM name associated for the first message versus the second. This difference could be important if PGMA uses the LTERM name to make different decisions, e.g., decide which printer to send output or to branch to different logic in the application.

Although the remote partner on this visual is IMS, it can actually be any LU 6.2 partner.

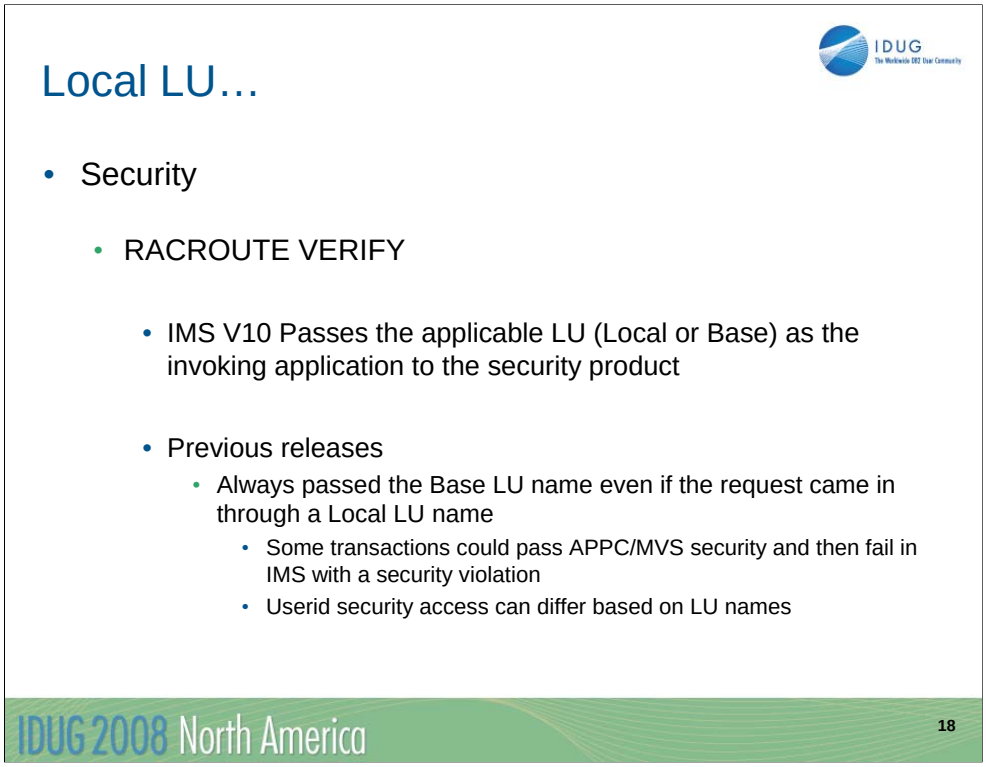


IMS 10 also provides a new startup parameter in the DFSDCxxx member of IMS.PROCLIB. APPCLLU affects asynchronous outbound messages that are inserted to the IOPCB. For this situation to occur, the message that the IMS application processed with a GU IOPCB had to originate in an APPC partner as an asynchronous inbound message.

The two examples on this visual illustrate the difference between the APPCLLU specifications of N and Y.

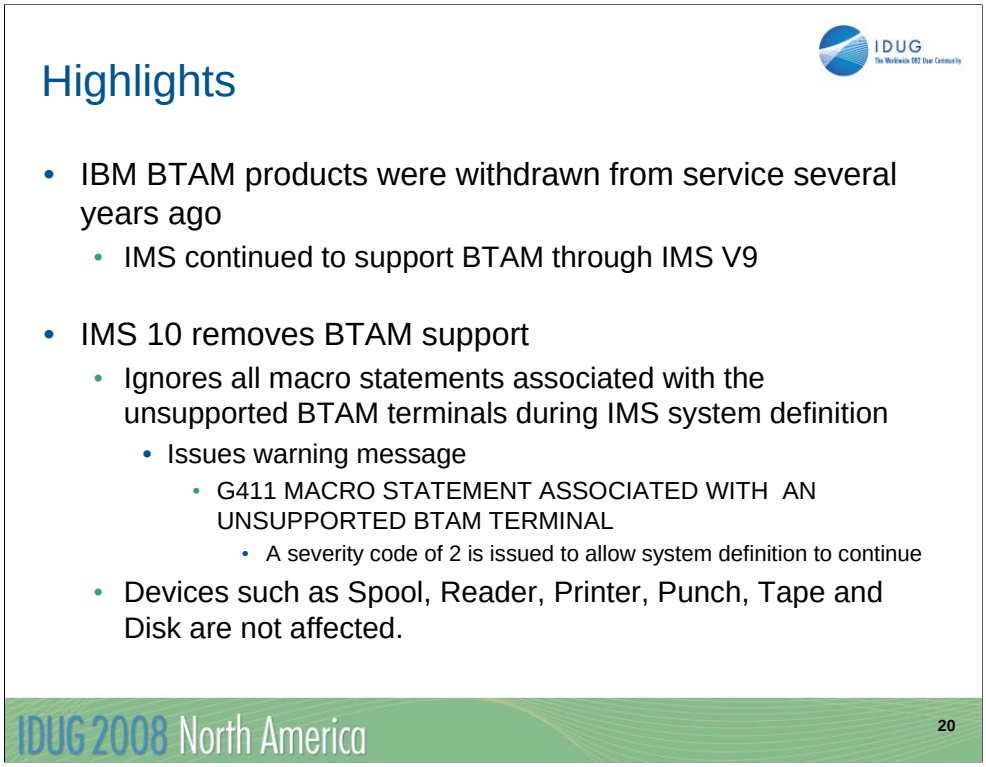
In the first example, APPCLLU=N, the IOPCB asynchronous output reply is sent using the Base LU name associated with IMS2 regardless of which LU name was used for the inbound asynchronous conversation.

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.



Security in this area has also been enhanced. 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 (Access Control Environment Element - in essence a control block representing the user), 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.

This next bit of information in IMS 10 should not be news to anybody.



Although IBM withdrew marketing and service of BTAM products several years ago, IMS continued to support the BTAM macros through IMS version 9. IMS 10 removes this support.

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 10 Messages and Codes manual.

The IMS documentation 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

OTMA Enhancements

Copyright IBM Corp. 2006

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
  - TMEMBER / USER level security enhancements
  - Asynchronous message enhancements
  - Enhanced OTMA display information
  - OTMA restart options
  - CM0 Ignore Purge

OTMA introduces many new capabilities in IMS 10. This visual provides a laundry list of enhancements which I will go through briefly.

Copyright IBM Corp. 2006

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

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.

This capability also provides the foundation for the Integration Suite's asynchronous callout function that allows ALTPCB messages to be sent to remote TCP/IP applications and web services.

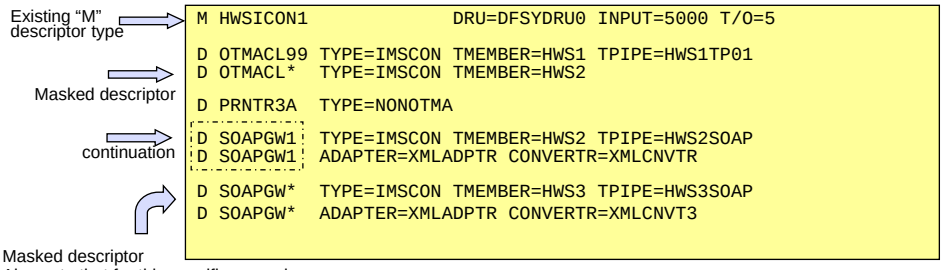
Copyright IBM Corp. 2006

Routing Enhancements …

- New ‘D’ descriptor type in DFSYDTx member of IMS.PROCLIB

```
D destname        TYPE={IMSCON|NONOTMA} TMEMBER=name TPIPE=name
                  SMEM={NO|YES} ADAPTER=adapname CONVERTR=convname
```

  - Example



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

Routing destinations are specified through a new ‘D’ descriptor type in the DFSYDTx member of IMS.PROCLIB. Multiple OTMA descriptors can be defined in the same DFSYDTx member.

This example illustrates six descriptors:

•The first is a TMEMBER descriptor with an M is the first position. This descriptor specifies the DRU exit for TMEMBER “HWSICON1”.

•The second is a Destination Routing descriptor, with a D in the first column, destination OTMACL99 that will be routed to the specified IMS Connect instance.

•The third is a Destination Routing descriptor for destinations matching the mask “OTMACL*”. The messages that resolve to this descriptor are anything beginning with OTMACL except for OTMACL99. Note that this along with the second descriptor illustrate that more specific destinations must be coded ahead of generic ones.

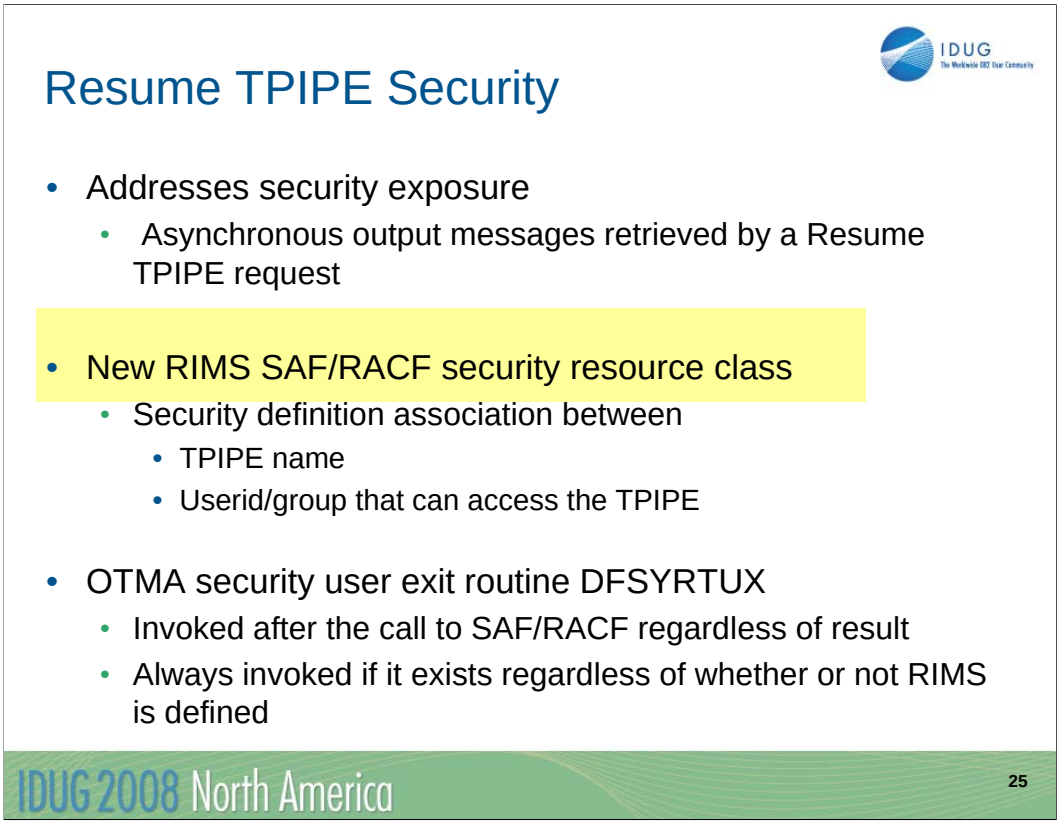
•The fourth is a Destination Routing descriptor for destination “PRNTR3A” that will be routed to legacy IMS.

•The fifth is a Destination Routing descriptor for “SOAPGW1” that will be routed to IMS Connect with a specification for XML translation.

•The sixth is a masked Destination Routing descriptor for destinations that begin with “SOAPGW*”.

From a usage perspective, IMS application programmers will need to ensure that the ALTPCB destination name matches the name of a Destination Routing descriptor. System programmers will also need to ensure that they define the new ‘D’ descriptors in DFSYDTx to correctly route the application messages. Destinations are searched and used in the order coded.

Copyright IBM Corp. 2006



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.

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 > 5000 unprocessed input messages from an OTMA member, new input from the same member is rejected
  - Overrides provided
    - Descriptor, /STA TMEMBER command
      - INPUT= 0 to 9999
        - 0 turns off the support,
        - 0-200 resets to 200
        - >9999 resets to 9999
    - OTMA client-bid
  - Prevents possible S40D IMS Abends due to large number of OTMA control blocks associated with the queued requests
    - Also delivered in IMS V8 (PK04461) and IMS V9 (PK04463)

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.

Copyright IBM Corp. 2006

CM1 (Send-then-Commit) ACK Time-out

- New Time-out control for CM1 (Send-then-Commit)
  - 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
    - Backout occurs if ACK/NAK is not received within a time limit
      - Default time-out value is 120 seconds
      - Values range from 0 to 255 seconds
  - Overrides provided through
    - T/O= descriptor keyword
    - Timeout= keyword in the /STA TMEMBER command
    - Client-bid protocol message
    - Individual message override
  - Note: A value of zero turns off the capability and is accepted only via the descriptor or command

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, depending if the interaction is sync_level=confirm or sync_level=syncpoint, results in a "wait-syncpoint" or "wait-RRS" 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.

Copyright IBM Corp. 2006

TPIPE Storage Clean-up

- Enhancement to release storage associated with unused TPIPEs
  - Idle for two checkpoints
    - At each checkpoint
      - All existing TPIPEs are scanned to see if input or output activities have occurred
    - In the subsequent checkpoint,
      - Idle TPIPEs with still no activity are candidates for removal
      - Note: Certain TPIPEs are never considered for clean-up:
        - Synchronized TPIPEs from MQ
        - TPIPEs with status conditions such as TRA and STO
        - Not in Resume TPIPE Auto or Single wait mode for a msg
- This capability is available in IMS V8 and IMS V9
  - IMS V8: PQ99983, IMS V9: PK00386

IMS10 also 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. IMS is aware that there are certain TPIPEs which should never be considered for clean-up.

Copyright IBM Corp. 2006

TMEMBER / USER Level Security

- New /SECURE OTMA command capabilities
  - Allows each OTMA member to define its own security setting
    - FULL, CHECK, NONE, or PROFILE
    - Dynamic change of security level

      /SECURE OTMA security-option TMEMBER member-name

      (where security-option is FULL | CHECK | NONE | PROFILE)
    - Prior Releases
      - OTMA security was a system-wide setting for all OTMA members
  - New option for a specific user profile

    /SECURE OTMA REFRESH USER userid

    (where acee may be in multiple TMEMBERs)
    - Prior Releases provided a command to refresh
      - All ACEEs for all TMEMBERS in OTMA
      - All ACEEs in a specific TMEMBER

Security is always an area that piques a lot of interest especially with respect to messages that can be sent in from remote servers and from the internet. Prior to IMS 10, OTMA did not allow different security levels to be defined for various members or even unique instances of the same member type. The security setting requested was always considered to be a system-wide setting for all of OTMA members. In IMS 10, 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.

Additionally, a new option allows a specific user profile to be refreshed across all instances of OTMA clients or TMEMBERs. In previous releases, the refresh capability allowed a recycling of all the ACEEs for all TMEMBERs or all ACEEs in a specific member but nothing to the level of granularity of a single ACEE.

Note, however, that 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.

Copyright IBM Corp. 2006

OTMA /DISPLAY Enhancements

- /DIS OTMA and /DIS TMEMBER output
  - 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

IMS 10 also provides more information about the OTMA environment with enhancements to the information provided in the /DISPLAY OTMA and /DISPLAY TMEMBER command output. 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:

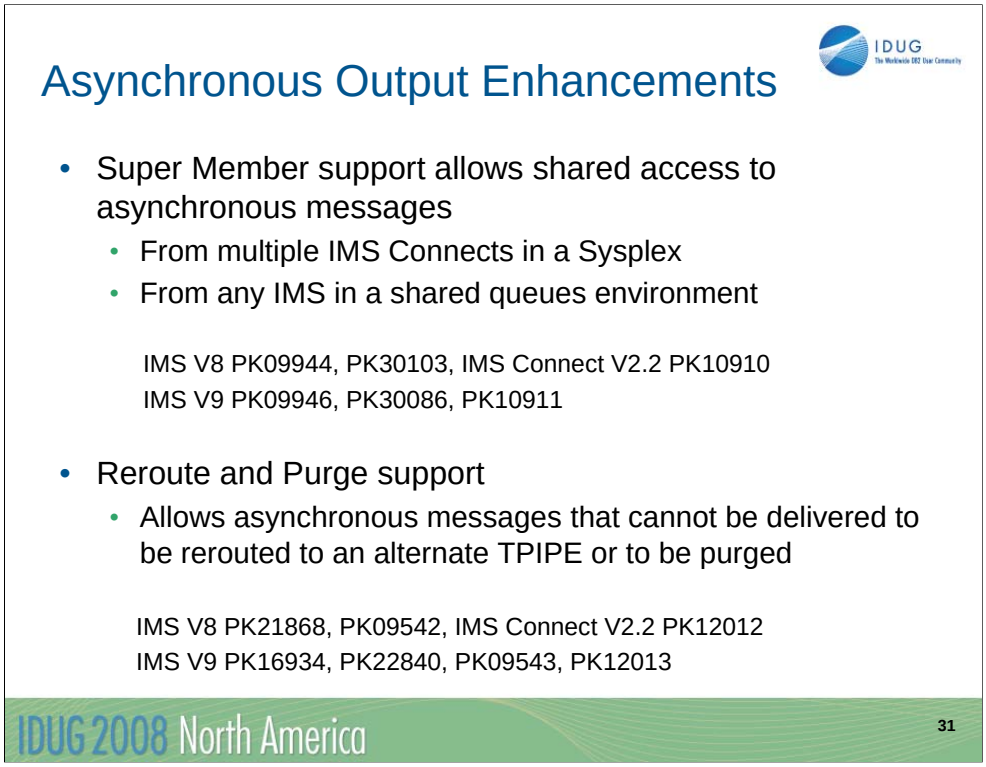
•SMQ BACKEND indicates that the TMEMBER information reflects a back-end IMS in a shared queues group.

•STO-INPUT is a status that shows that the /STOP TMEMBER command has been issued for a specific member-name and no new input can be accepted.

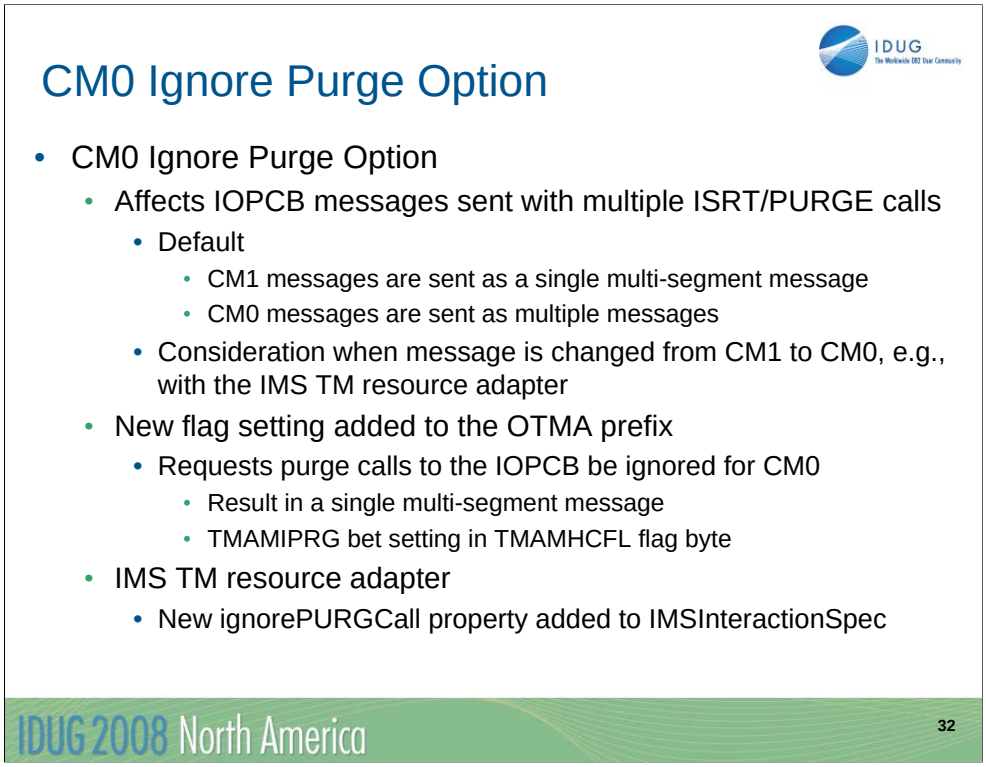
•FLOOD indicates that a specific TMEMBER is in a message flood condition and that the maximum input message count that was specified has been reached.

Additionally, the /DIS TMEMBER TPIPE command has been enhanced to show the number of input messages currently on the queue.

Copyright IBM Corp. 2006



This next visual is simply a reminder that over the past several years, OTMA enhanced its support for IMS and IMS Connect implementation scenarios that included IMS shared queues and load balancing algorithms or IP spraying techniques through the use of capabilities such as the sysplex distributor. Additionally. OTMA in conjunction with IMS Connect allows asynchronous messages that are sent by IMS but cannot be delivered to be either rerouted to an alternate destination or purged from the IMS message queue.



This brings us to another enhancement in IMS 10 with respect to asynchronous output messages. OTMA has always handled multiple output messages that are inserted to the IOPCB differently for Send-then-Commit (CM1) versus Commit-then-Send (CM0) specifications. This affects programs that issue repeated iterations of ISRT followed by PURGE calls to the IOPCB. Ordinarily, a PURGE call triggers the delineation between output messages. With OTMA, multiple ISRT/PURGE combinations for CM1 interactions result in one multi-segment output message. The same scenario, when invoked in a CM0 mode, results in multiple output messages, one corresponding to each PURGE call.

Remote programs that are coded to use one or the other mode, CM1 versus CM0, should be aware of this anomaly. IMS 10 provides a new flag that can be set in the OTMA prefix of a message to request that the IOPCB output be processed as one message even when the mode is CM0 and multiple ISRT/PURGE iterations have been issued. This enhancement allows remote programs to consistently expect only one output message and then parse the information to determine if the message contains more than one segment.

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

In addition to the existing OTMA values of Y and N, IMS 10 introduces the option of OTMA=M (manual) to request that OTMA not be automatically started in an IMS restart situation. 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.

Copyright IBM Corp. 2006

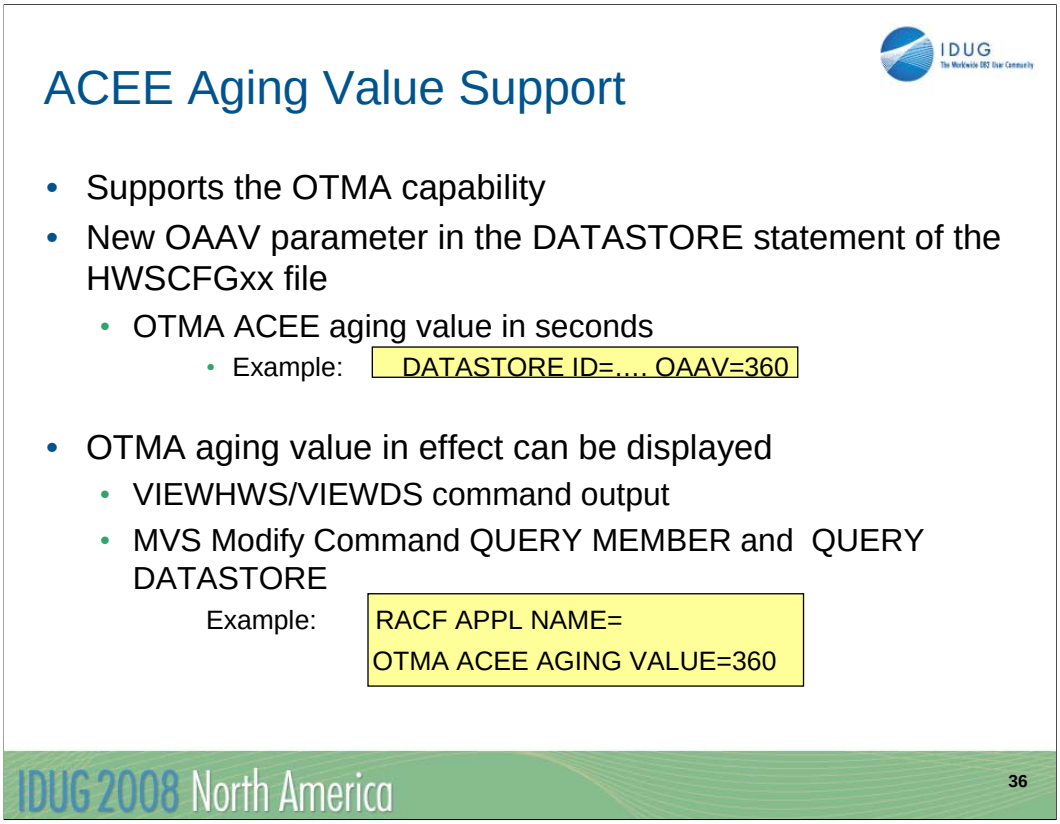
Now let us turn towards IMS Connect.

Highlights

- IMS Connect enhancements include
  - ACEE aging value support
  - Client password change request
  - RACF mixed case password
  - Message flood control
  - CM1 timeout ack support - ACKTO
  - Resume TPIPE Enhancements
    - Alternate clientid
    - PORTAFF
  - XML Adapter support
  - IMS SOA Composite Business Application Support

IMS 10 also introduces many enhances for IMS Connect as listed on this visual.

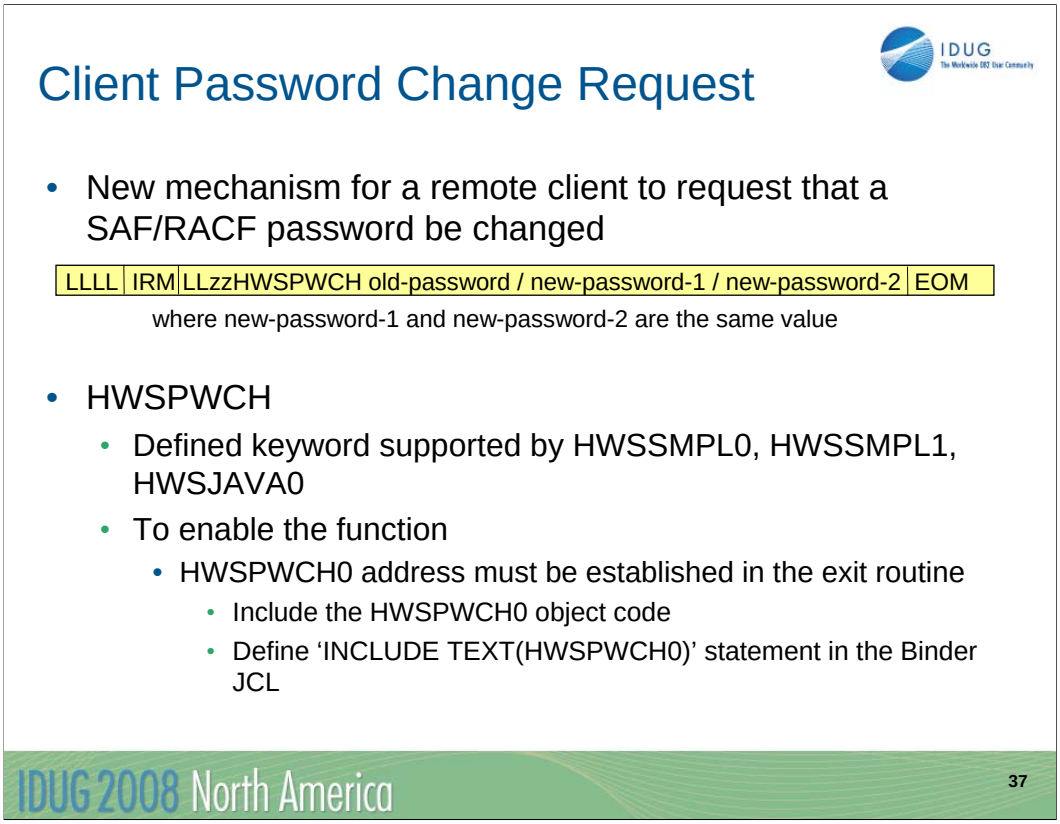
Copyright IBM Corp. 2006



IMS Connect takes advantage of an OTMA capability to override the ACEE aging value. 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.

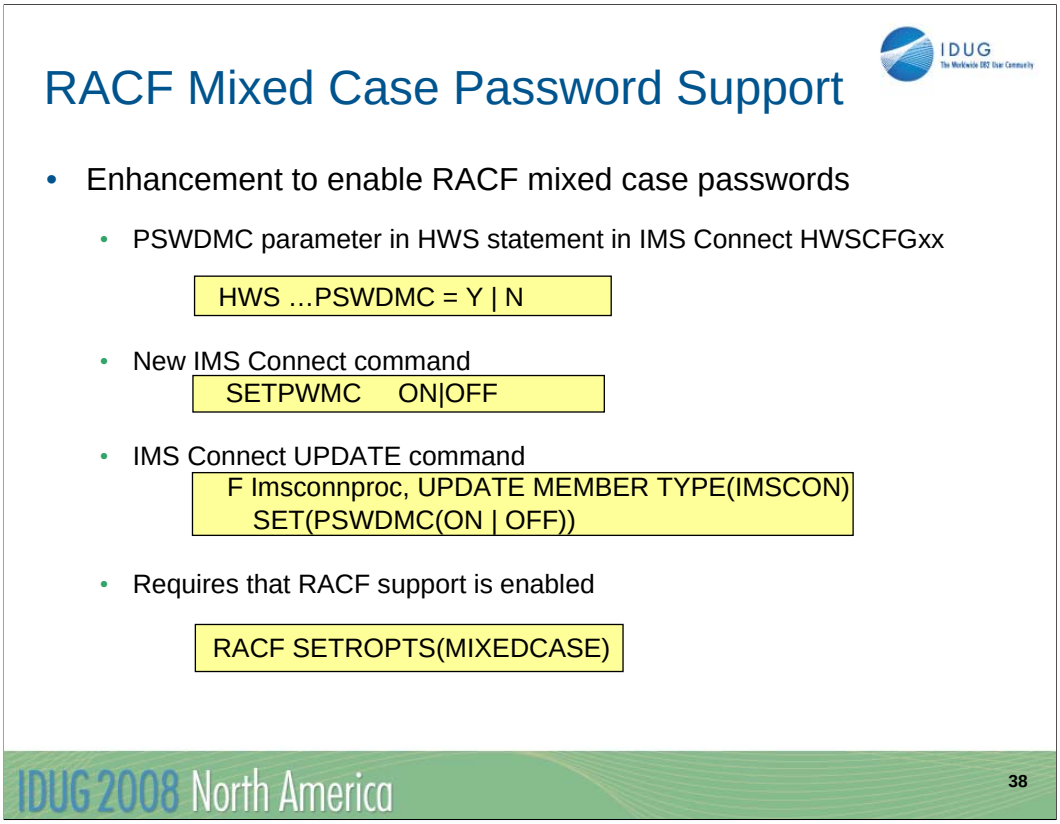
IMS Connect 10 provides a new parameter, OAAV, in the DATASTORE statement of the HWSCFGxx file for specification of an OTMA ACEE aging value. If not specified, the default continues to be 2147483647 which, in essence, means no expiration and is the maximum value supported by OTMA. The VIEWHWS,VIEWDS, 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.

Copyright IBM Corp. 2006

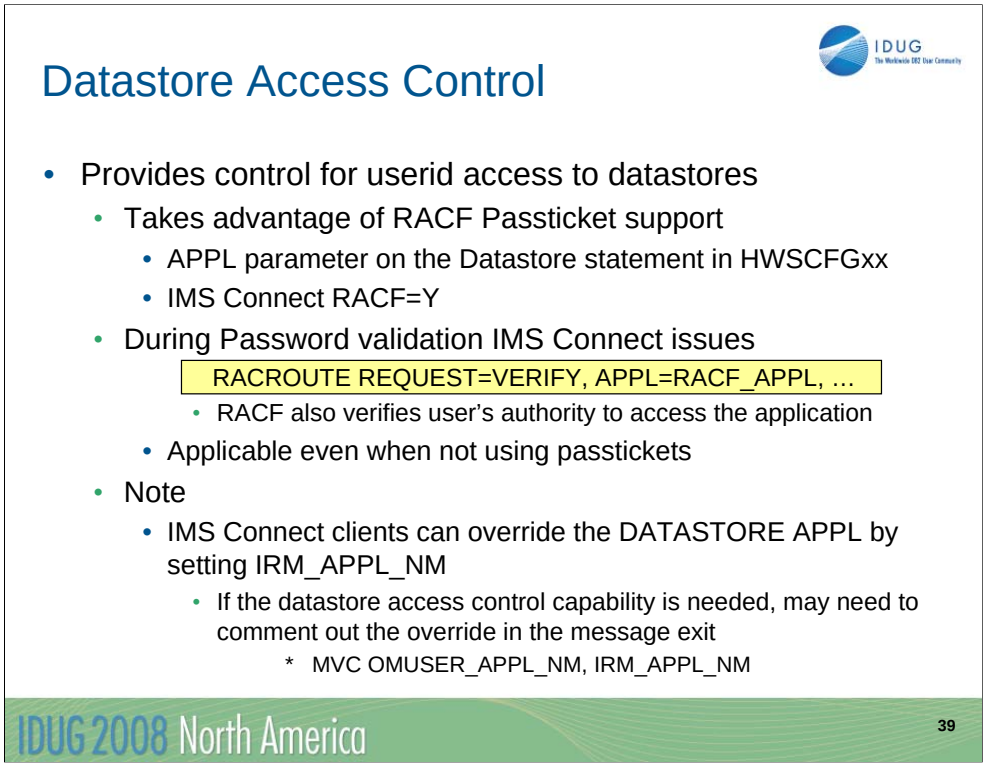


IMS Connect also provides a new mechanism to allow a remote client to request that the SAF/RACF password associated with a userid be changed. As provided, the new capability will be supported in the HWSSMPL0, HWSSMPL1 and HWSJAVA0 exit routines. The routines check for a leading keyword of 'HWSPWCH' to 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.

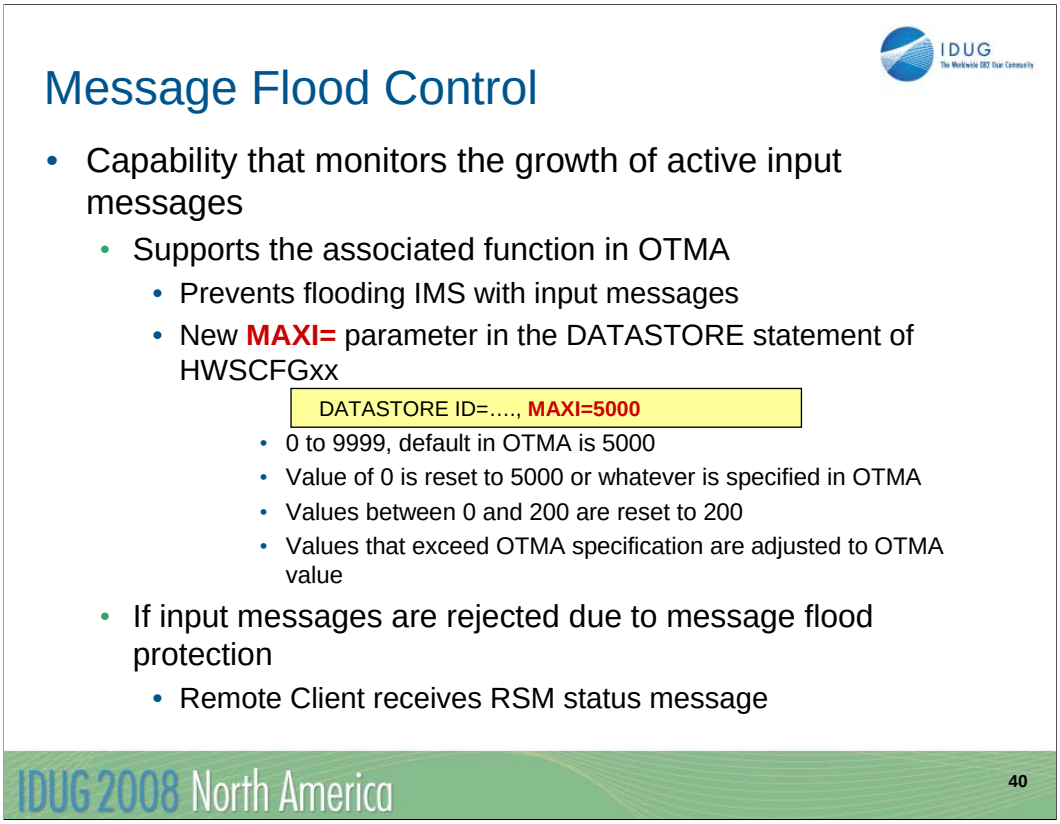
In order to establish the HWSPWCH0 address, the HWSPWCH0 object code must be included in the exit routine and an INCLUDE statement added to the exit routine JCL for the binder (link-edit) step. During execution, if a request for password change is received and the HWSPWCH0 address does not exist, the exit routine will send a message back to the client stating that the password change function is not supported.



The support for RACF mixed case password in IMS Connect is aligned with the IMS 10 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 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.



Several releases ago, IMS Connect provided support for RACF Passtickets. To allow RACF to correctly decode the Passticket, a DATASTORE APPL variable was also provided to specify the IMS APPL name defined to RACF in the PTKTDATA statement. In IMS 10, IMS Connect takes further action with the APPL specification to validate the user's authority to access the IMS system associated with the DATASTORE specification. This capability to validate user authority to access an IMS system is applicable by specifying an APPL value even when Passtickets are not being used.



A few visuals ago, the OTMA enhancement for Message Flood Control as specified globally for an IMS system was discussed. IMS Connect also takes advantage of this capability to automatically monitor the growth of active input messages from a specific instance of IMS Connect. 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. IMS Connect cannot turn off the OTMA support nor provide an override value greater than what OTMA specified, but it can adjust a value from 200 to whatever OTMA provides. The override only applies to this IMS Connect instance.

CM1 (Send-then-Commit) ACK Time-out

- Time-out Control capability for CM1 interactions
  - Supports the associated function in OTMA
    - Resolves *"Wait-Syncpoint"* and *"Wait-RRS"* situations
  - New **ACKTO=** parameter in the DATASTORE statement of HWSCFGxx

    DATASTORE ID=...., **ACKTO**=120
    - 0 to 255 seconds, default in OTMA is 120 seconds
    - Value of 0 is reset to 120 or whatever is specified in OTMA
    - Values that exceed OTMA specification are adjusted to OTMA value
  - If time-out occurs
    - Remote Client receives a deallocate of the connection and an RSM status message

Similarly, IMS Connect supports the new OTMA time-out control function for send-then-commit CM1 interactions. If provided, the value in the ACKTO parameter of the IMS Connect configuration DATASTORE statement is passed to OTMA during client-bid processing. The override value for IMS Connect is applicable only to this instance of an IMS Connect execution and will be accepted as long as the IMS Connect ACKTO is not 0 and is less than the OTMA value.

Copyright IBM Corp. 2006

Resume TPIPE Enhancements - Alternate Clientid

- Capability to request and retrieve asynchronous output messages that are queued to another client
  - Supports a server application that retrieves messages originally destined for another application (clientid)
    - Resume TPIPE request specifies an alternate clientid
      - OTMA delivers output messages queued to that alternate name
  - Supported by IMS Connect user message exits
    - HWSSMPL0, HWSSMPL1, HWSSOAP1, HWSJAVA0
  - Note: this support differs from but leverages the existing Reroute capability which only addresses undeliverable and/or NAK'ed messages
    - IMS V9 support added with PK24907

Another IMS Connect enhancement that leverages the existing Reroute capability but adds additional functionality is the Alternate Clientid capability. Client applications can now specify an alternate clientid in the RESUME TPIPE request to retrieve asynchronous messages that are queued to a TPIPE by that name.

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 where the name is different that the clientid associated with the remote requestor. 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.

Copyright IBM Corp. 2006

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

Another Resume TPIPE enhancement is support for Port Affinity. Using concurrent Resume TPIPE connection requests of the same clientid across several ports may cause problems. IMS Connect addresses this situation with 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.

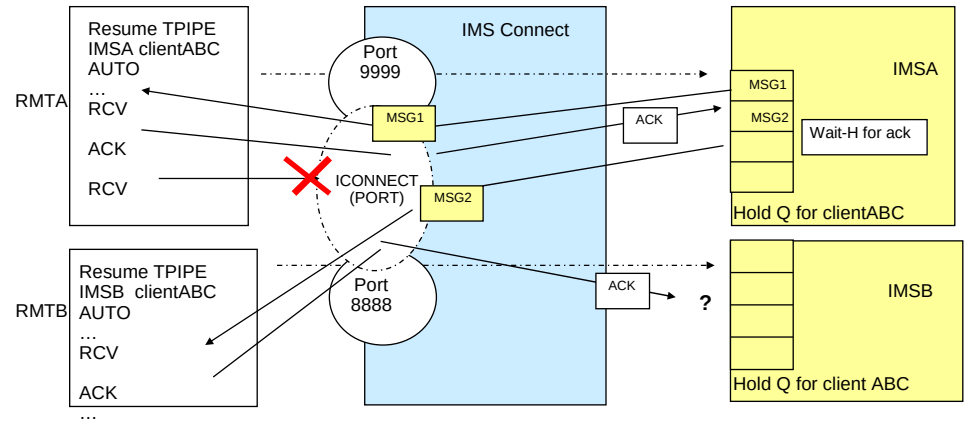
When PORTAFF=Y is specified, IMS Connect returns all CM0 output for this IMS Connect client through the same port on which it received the original input message.

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.

Copyright IBM Corp. 2006

Resume TPIPE Enhancements - Port Affinity …

- Consideration - concurrent requests using the same clientid
  - PORTAFF=N (default)



Recommendation: specify PORTAFF=Y
or do not use concurrent Resume_TPIPE requests with the same clientid

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. When multiple ports are defined to an IMS Connect instance, a generic port ICONNECT is created under which all actual ports are managed. 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 waiting 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.

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.

Copyright IBM Corp. 2006

XML Adapter Support

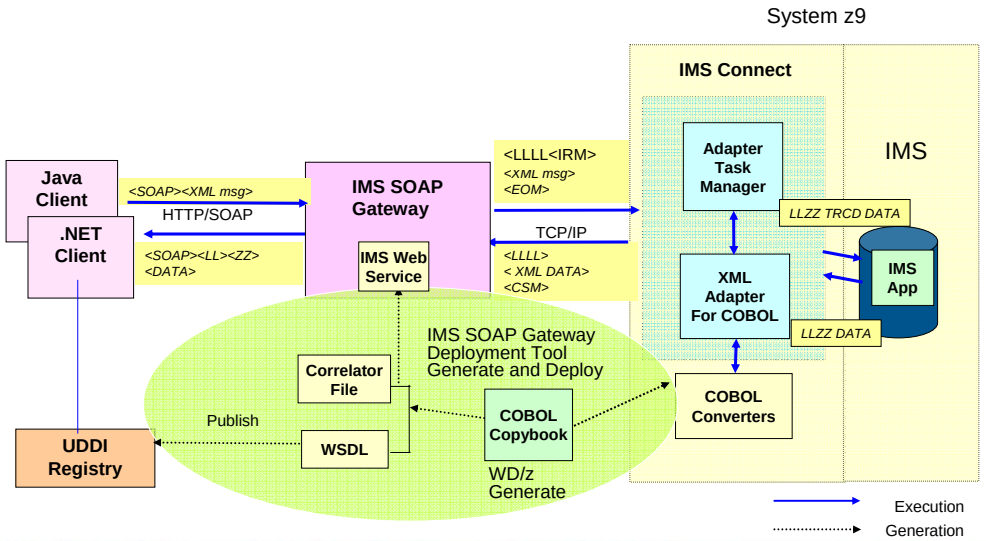
- Capability that supports translation between XML messages and IMS messages
  - IMS Connect client, e.g., IMS SOAP Gateway
    - Sends an XML message with a request for translation
  - IMS Connect
    - Inbound: invokes the XML Adapter to translate message for IMS
      - Removes XML tags
      - If 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

In support of the IMS Soap Gateway, IMS Connect now supports the translation between XML messages and IMS Messages. This capability strips off the XML tags on the inbound to IMS side and adds them back to the message on the output. The support was also retrofitted back to IMS version 9.

Copyright IBM Corp. 2006

XML Adapter Support …

- Overview



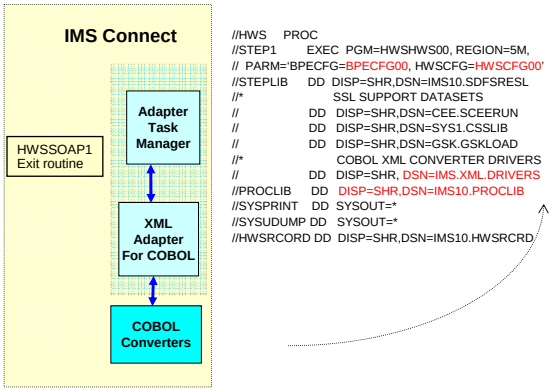
This next visual provides a picture of what occurs. The IMS Soap Gateway provides the SOAP endpoint to remote clients such as Java and .NET clients and passes XML messages to IMS Connect as well as receives XML messages in response.

For IMS Connect to correctly interpret the XML into a layout for the Cobol or PL/I application, the Rational Developer for zSeries (RDz) supports the generation of XML Converters from copybooks. Because each IMS application expects its messages to be in a certain data structure; one XML Converter routine is needed for each IMS application.

Copyright IBM Corp. 2006

XML Adapter Support …

- User Message Exit HWSSOAP1
  - OCO
  - IRMID *HWSOA1*
- 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



Once the converter files are generated by RDz, the files must be added to an authorized library which is concatenated to IMS Connect's set of STEPLIB libraries. This assures that IMS Connect invokes the correct processes.

Just to let you know, per the announcement letter, two limited usage licenses of RDz are being provided with IMS 10 for this support of Web services. The associated terms and conditions are outlined in the RDz licensing.

Copyright IBM Corp. 2006

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

IMS SOA Composite Business Application support enhances the IMS interaction with the process choreography function of WebSphere Process Server. Prior to this enhancement, process choreography with IMS was limited to non-conversational transactions. With IMS 10, process choreography can now include conversational transaction support. IMS Connect supports this new capability that allows iterations of a conversational transaction to span different sockets.

Copyright IBM Corp. 2006

Summary

- IMS V10 Transaction Manager and Connectivity Enhancements
  - Continue to open IMS up to new architectures
  - Enhance existing functionality
  - Address increasing demands for availability, scalability and performance

This brings us to the summary. I hope that what you have gotten from today's session is a perspective of all the new functions in the TM and Connectivity area of IMS 10. As you saw, IMS 10 continues to open IMS up to new architectures while enhancing existing functionality.

Copyright IBM Corp. 2006

J06

IDUG

IMS 10 Transaction Manager and Connectivity Enhancements

Suzie Wendler

IBM

wendler@us.ibm.com

Copyright IBM Corp. 2006