

Connectivity

ON DEMAND BUSINESS

© 2008 IBM Corporation



Connectivity

- OTMA
- IMS Connect



OTMA

Highlights

- OTMA addresses high availability requirements through
 - ◆ Routing Enhancements
 - Destination Routing descriptors
 - Resume TPIPE security
 - ◆ Automatic flood detection and control of input messages
 - ◆ Time-out control
 - ◆ TPIPE storage clean-up
 - ◆ Member level security
 - ◆ Asynchronous message enhancements
 - ◆ Enhanced OTMA display information

4

OTMA enhancements address several high availability requirements. These include the areas listed on the visual.

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

5

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.

Routing Enhancements ...

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

D *destname* **keywords**

Where:

Destname is destination name and can be masked by ending in an "*"

Keywords are:


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

- ◆ Up to 50 lines can be used in the specification of a descriptor
 - Columns 1 through 10 must be the same for each line of a continuation
- Read and initialized at IMS startup

6

The new 'D' descriptor type for the DFSYDTx member of IMS.PROCLIB includes keywords as follows:

- TYPE= determines if output is destined for IMS Connect (IMSCON) or non-OTMA (NONOTMA). This is a required keyword.
- TMEMBER= 1 to 16 character client name. Required for TYPE=IMSCON. Ignored for TYPE=NONOTMA.
- TPIPE= 1 to 8 character TPIPE name. Optional for TYPE=IMSCON, defaults to Destination name. Ignored for NONOTMA.
- SMEM= indicates if this destination is a Super Member. Optional keyword for TYPE=ICON, defaults to SMEM=NO. Ignored for TYPE=NONOTMA. If "YES", the name defined in the TMEMBER keyword becomes the Super Member name and can only be 4 characters.
- ADAPTER= 1 to 8 character name of the IMS Connect Adapter to be used for the message, e.g., one example is an adapter for XML transformation. Optional for TYPE=IMSCON and ignored for TYPE=NONOTMA.
- CONVERTR= 1 to 8 character name of the Converter to be used by the Adapter. Required for TYPE=IMSCON if ADAPTER is specified. Ignored for TYPE=NONOTMA.

Information On Demand IMS Version 10 

Routing Enhancements ...

- Example

```

M HWSICON1                DRU=DFSYDRU0 INPUT=5000 T/O=5
D OTMACL99 TYPE=IMSCON TMEMBER=HWS1 TPIPE=HWS1TP01
D OTMACL*  TYPE=IMSCON TMEMBER=HWS2
D PRNTR3A  TYPE=NONOTMA
D SOAPGW1  TYPE=IMSCON TMEMBER=HWS2 TPIPE=HWS2SOAP
D SOAPGW1  ADAPTER=XMLADPTR CONVERTR=XMLCNVTR
D SOAPGW*  TYPE=IMSCON TMEMBER=HWS3 TPIPE=HWS3SOAP
D SOAPGW*  ADAPTER=XMLADPTR CONVERTR=XMLCNVT3

```

Masked descriptor →

continuation →

Masked descriptor ↪

Also note that for this specific example, all destination matches for any destination beginning with SOAPGW... will be routed to the single TPIPE HWS3SOAP

Note: any descriptors that result in syntax errors are ignored

7

Multiple OTMA descriptors can be defined in the same DFSYDTx member.

This example illustrates six descriptors:

- The first is a TMEMBER descriptor specifying the DRU exit for TMEMBER “HWSICON1”.
- The second is a Destination Routing descriptor for destination OTMACL99 that will be routed to IMS Connect TMEMBER “HWS1” with TPIPE “HWS1TP01”.
- The third is a Destination Routing descriptor for destinations matching the mask “OTMACL*”. The messages will be routed to a TMEMBER of “HWS2” with a TPIPE of the destination matching the mask, e.g., OTMACL04 if the destination using this descriptor was OTMACL04. Note that this along with the second descriptor illustrate that more specific destinations must be coded ahead of generic ones.
- The fourth is a Destination Routing descriptor for destination “PRNTR3A” that will be routed to legacy IMS.
- The fifth is a Destination Routing descriptor for “SOAPGW1” that will be routed to IMS Connect TMEMBER “HWS2” with TPIPE “HWS2SOAP” and will result in XML translation.
- The sixth is a masked Destination Routing descriptor for destinations that begin with “SOAPGW*”. This last descriptor will not be used for SOAPGW1 because the previous descriptor already specifically addressed that destination name. Note that all the destination matches for this descriptor will be routed to the one specified TPIPE “HWS3SOAP”.

Routing Enhancements ...


- Usage
 - ◆ Application programmers
 - Ensure IMS application ALTPCB destination name matches the name of a Destination Routing descriptor
 - ◆ System programmers
 - Define the new 'D' descriptors in DFSYDTx
 - Code more specific descriptors before generic definitions
 - Searched and used in the order coded
 - ◆ Error message enhancement
 - ◆ DFS2385E SYNTAX ERROR FOR DESCRIPTOR = *descriptor errortext*
 - New error text for the 'D' descriptors
 - Also issued if a previous release of IMS attempts to read DFSYDTx with the new descriptors

8

To correctly enable the use of the new callout descriptors, application programmers must code the destination of an asynchronous output message switch (ALTPCB) to match the destination name of a corresponding descriptor. IMS systems programmers or system administrators, on the other hand, will also need to create the sure descriptors in the DFSYDTx member of IMS.PROCLIB. Because descriptors can be masked (end in an "**") and are searched in the order they are coded, more specific descriptors should appear before generic ones.

The DFS2385E error message has been enhanced to catch syntax errors associated with the new descriptor keywords. New error text include:

- DESTINATION NAME NOT GIVEN OR BEGINS AFTER COLUMN 3
- DESTINATION NAME LONGER THAN 8 CHAR
- INVALID TYPE SPECIFIED
- DUPLICATE TYPE KEYWORD
- TMEMBER REQUIRED FOR TYPE=IMSCON
- TMEMBER MUST BE 1 TO 16 CHARACTERS LONG
- DUPLICATE TMEMBER KEYWORD
- TPIPE MUST BE 1 TO 8 CHARACTERS LONG
- DUPLICATE TPIPE KEYWORD
- SMEM MUST BE YES OR NO
- DUPLICATE SMEM KEYWORD
- ADAPTER MUST BE 1 TO 8 CHARACTERS LONG
- DUPLICATE ADAPTER KEYWORD
- CONVERTR MUST BE 1 TO 8 CHARACTERS LONG
- DUPLICATE CONVERTR KEYWORD
- CONVERTR REQUIRED WITH ADAPTER



Resume TPIPE Security

- Addresses security exposure
 - ◆ Asynchronous output messages retrieved by a Resume TPIPE request
- New RIMS SAF/RACF security resource class
 - ◆ Security definition association between
 - TPIPE name
 - Userid/group that can access the TPIPE
- OTMA security user exit routine DFSYRTUX
 - ◆ Invoked after the call to SAF/RACF regardless of result
 - ◆ Always invoked if it exists regardless of whether or not RIMS is defined

9

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.

Resume TPIPE Security ...

- Implementation
 - ◆ SAF/RACF security server
 - Supports a new resource class Rxxxxxxx in SAF/RACF
 - Where xxxxxx is the RCLASS value defined in the SECURITY macro or in the DFSDCxxx member
 - Default is RIMS
 - Authorizes TPIPE access from the userid/group
 - ◆ IMS
 - Provides a sample user security exit DFSYRTUX
 - Default routine always provides zero return code for compatibility
 - Must be modified if further protection is required
 - ◆ OTMA Client, e.g., IMS V10 IMS Connect
 - Passes information from the Resume TPIPE request to OTMA
 - TPIPE name in the OTMA CTL prefix and userid in the OTMA security prefix

10

To take advantage of Resume TPIPE Security, the following actions are required:

- Define a new resource class, TPIPE name, and user IDs in RACF (refer to SA22-7683 *Security Server RACF Security Administrator's Guide*) or applicable security server for the environment. The resource class comprises the resource class type of "R" and the resource class name whose value is taken from the RCLASS parameter of the SECURITY macro or in the DFSDCxxx member of Proclib. If RCLASS is omitted, the resource class name defaults to "IMS." The resulting class, then, is "Rxxxxxxx" where 'xxxxxxx' is the value of RCLASS or "RIMS" as the default.
- If needed, code, assemble, and bind the user exit in a library that is concatenated with IMS SDFSRESL under DD name STEPLIB or JOBLIB.
- Ensure that the OTMA client, e.g., IMS Connect, is at the correct IMS V10 level to pass in the TPIPE name in the OTMA CTL prefix and the userid/group in the security prefix header as part of the message for the Resume TPIPE command.

Resume TPIPE Security ...

- Remote client impact
 - ◆ Previous releases
 - Always returned the message if the Resume TPIPE request specified the correct clientid (TPIPE)
 - Userid authentication may have been done (RACF=Y) but no further check for authorization of userid access to the TPIPE
 - ◆ With Resume TPIPE security
 - Userid/group must be authorized to issue Resume TPIPE from a particular clientid (TPIPE)

11

The implementation of Resume TPIPE Security could possibly impact existing clients. In previous releases, a userid that was provided in the Resume TPIPE request was authenticated if IMS Connect was configured with security enabled. As long as the Resume TPIPE request, however, passed in the correct TPIPE name (clientid) then the associated asynchronous output messages could be retrieved.

In IMS V10, if the Rxxxxxxx | RIMS resource class is defined then security violations can occur where they previously did not. When retrieving the asynchronous output from IMS, the client signals IMS with a Resume TPIPE command and an IRM timeout value. This timeout value expires if there are no messages received by the client. Additionally, with the new support, a security check can result in success or failure. If the security validation is a success, normal processing takes place. On the other hand, if a security violation occurs, a new NAK message is sent to the client.

Message Flood Detection and Control

- Capability that automatically monitors the growth of active input messages
 - ◆ Sets a default max threshold of 5000 active input messages
 - ◆ If the IMS system has more than 5000 unscheduled, unprocessed, or orphaned input messages from an OTMA member
 - Any new input message from the same member is rejected

- Prevents possible S40D IMS Abends due to large number of OTMA control blocks associated with the queued requests

- Also delivered in IMS V8 and IMS V9:
 - ◆ V8: PK04461, V9: PK04463

12

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.

Message Flood Detection and Control ...




- Implementation
 - ◆ Flood detection and control capability is set on by default to 5000
 - ◆ Override order:
 - At initialization, DFSYDTx descriptor value is checked for override
 - During processing, /START TMEMBER command activates new override
 - As each OTMA member joins the group
 - The client-bid optionally provides a new INPUT value
 - Only accepted if less than the value in effect by descriptor or command
 - **Note:** an INPUT value of 0 in the descriptor or command deactivates the control
 - Subsequent client-bid values are ignored

13

By default, the message flood detection and control capability is always on and the maximum threshold value set to 5000. To override this default, several choices are available:

- The OTMA descriptor DFSYDTx in the IMS.PROCLIB library can provide a value which IMS detects at initialization. It is not, however, until the TMEMBER associated with the descriptor is actually started that IMS implements the override value.
- At any time, an operator can issue the /START TMEMBER command that not only starts the member but also provides an override flood detection value that supersedes anything provided in the descriptor.
- As a member joins the group, the client-bid protocol message can also provide an override. This value is honored only if it is less than the value that is already in effect based on the overrides provided by the descriptor or /START TMEMBER command.

Note that the deactivation of the input message flood control capability can be requested by either the DFSYDTx descriptor or the /START TMEMBER command by specifying an INPUT value of 0. A client-bid message cannot override this specification.

Message Flood Detection and Control ...

- Details
 - ◆ OTMA descriptor in **DFSYDTx** of IMS.PROCLIB:

M member-name DRU=...,INPUT= 0 to 9999
 - ◆ Commands:

/START TMEMBER member-name ... INPUT 0 to 9999
/STOP TMEMBER member-name | ALL
 - ◆ OTMA Member (e.g., IMS Connect) interface
 - **New settings in the state data section of the client-bid protocol message**
 - 2-byte field at offset **x'3E'** specifies the override value
 - Value must be less than the value set by IMS Command or OTMA descriptor
 - Flag **TMAMMAXI x'80'** in offset **x'2D'** specifies the new function request

14

The specifics of the implementation are as follows:

- A new INPUT parameter in the OTMA descriptor DFSYDTx in IMS.PROCLIB. The parameter allows values from 0 to 9999. If the value is set to 0 then the capability for the message flood detection is disabled. Values between 1 and 200 are set to 200 and anything over 9999 is set to 9999.

- The /STOP TMEMBER member-name | ALL command which suppresses new input transactions or commands from a specific OTMA member or all OTMA members. This command does not affect the rest of communications between the stopped member and IMS. That means the following operations can still be performed for a stopped member: the client and server XCF connection remains unchanged; all of the running transactions currently scheduled in the IMS can still be processed, and the responses can be delivered; IMS conversational transactions can continue processing the existing conversation until it ends; all of the OTMA protocol commands including ACK/NACK can still be processed by OTMA. Note that after this command has been issued, a STO-INPUT status will be displayed on a subsequent /DISPLAY TMEMBER command for the appropriate OTMA members.

- The /START TMEMBER INPUT command with the same range and meaning of the values 0 to 9999.

- New fields in the client-bid protocol interface between the OTMA member and OTMA. For the client-bid protocol message, a new 2-byte field in the offset x'3E' of the state data can specify the maximum number of active input message for the member. The value can be specified between 0 and 9999. If the value is 0, OTMA will take the default of 5000 for the maximum active input message limit (note that specifying 0 does not turn the capability off). If the value is between 1 and 200, it will be treated as 200. If it is over 9999, it will be set to 9999. A further restriction is that the specified request must be less than any value previously set by a descriptor or command. If the client-bid tries to send in a higher value, it will be reset to the value established by the descriptor or command. Additionally, a new flag TMAMMAXI x'80' in the offset x'2D' of the state data needs to be set to inform the OTMA server that the maximum active input message checking function is requested. During processing, if OTMA detects that the active input message limit has been reached, a new NAK code of x'30' will be sent the OTMA member.

OTMA monitors the growth of the active input messages from members. A warning message DFS1988W is sent to the console to indicate that the input message buildup is approaching the maximum limit. The message will be sent starting at 80% of the message limit and every 5% thereafter. When the maximum limit is reached, an error message DFS1989E is sent to the console and any subsequent OTMA input messages are rejected with a new OTMA sense code x'30'. Once the situation has been alleviated, DFS0767I is issued.

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

- New Time-out control capability for CM1 (Send-then-Commit) interactions
 - ◆ For Synclevel=confirm or synclevel=syncpt processing
 - IMS waits for an ACK/NAK after sending the response
 - **“Wait-Syncpoint” or “Wait-RRS” status**
 - Locks are held, dependent region is occupied
 - ◆ New backout logic implements time-out capability
 - If ACK/NAK is not received within a time limit
 - Default time-out value is 120 seconds

15

The next capability that OTMA provides is a time-out control option that is applicable for CM1 message processing. For an OTMA send-then-commit (CM1) response message with synclevel=confirm or synclevel=syncpt, IMS expects an ACK/NAK from the OTMA client. Due to the possibility of a client programming error or a network failure or delay, the expected ACK/NAK may not be received by IMS. A missing or delayed ACK/NAK results in a “wait-syncpoint” condition for the IMS dependent region that processed the OTMA transaction. To resolve this situation, OTMA has been enhanced to detect this “wait-syncpoint” condition and take an appropriate time-out action. The default time-out value is 120 seconds.

CM1 (Send-then-Commit) Time-out Control ...

- Implementation

- ◆ Time-out is set on by default to 120 seconds
 - Range of values: 0 to 255
- ◆ Override order:
 - At initialization, DFSYDTx descriptor value is checked for override
 - During processing, /START TMEMBER command activates new value
 - As each OTMA member joins the group an override can be set
 - And optionally, each input message can set its own value
- **Note:** an INPUT value of 0 in the descriptor or command deactivates the control
 - Even if the feature is off, warning message DFS0808W will be issued to the system console when an expected ACK/NAK is not received within 120 seconds

16

The default 120 second timeout value for CM1 (send-then-commit) messages can be overridden in several ways:

- During IMS initialization with a new parameter in the OTMA descriptor member DFSYDTx. It is not until the member is actually started that the override value is honored.
- At any time with a new TIMEOUT specification in the /START TMEMBER command.
- By OTMA member request either when the member joins the group using the client-bid protocol message or, at a lower level of granularity, whenever a CM1 (send-then-commit message) flows into OTMA.

If none of the above methods is used to set the time-out value, the OTMA default of 120 seconds is used to determine when to perform the time-out action.

If needed, the /START TMEMBER TIMEOUT command and the OTMA descriptor can deactivate the OTMA time-out function by specifying a timeout value of 0. Once the function is deactivated, OTMA will not perform the time-out action. However, OTMA will still detect a long-waiting dependent region for a missing ACK or NAK and issue the following warning message: DFS0808W IMS REGION region-id IN BACK-END IMS aaaa HAS BEEN IN [WAIT-SYNCPOINT] or [WAIT-RRS] FOR otma-membername/tpipename FOR xx seconds.

CM1 (Send-then-Commit) Time-out Control ...

- Details
 - ◆ New parameter in the OTMA descriptor **DFSYDTx**:

M member-name ... T/O= 0 to 255
 - ◆ Command:




/START TMEMBER member-name TIMEOUT 0 to 255
 - ◆ OTMA Member (e.g., IMS Connect) interface
 - New settings in the state data section of the client-bid protocol
 - 1-byte time-out field at offset **x'41'** specifies time-out value
 - Value must be less than the value set by IMS Command or OTMA descriptor
 - Flag **TMAMTMOT x'20'** at offset **x'2D'** specifies that time-out is requested

17

The details of the implementation are as follows:

- A new T/O parameter in the OTMA descriptor member **DFSYDTx** in **IMS.PROCLIB**. The new parameter defines the time-out value in minutes for OTMA send-then-commit response messages. The value specified can be between 0 and 255 seconds. If the value is 0, OTMA will deactivate the time-out function. If it is over 255, it will be set to 120 which is the default.
- A new **TIMEOUT** specification in the **/START TMEMBER** command.
- New specification in the OTMA member request either when the member joins the group using the client-bid protocol message or, at a lower level of granularity, whenever a CM1 (send-then-commit message) flows into OTMA.

If a client-bid protocol message is used to specify the time-out value, the criteria of choosing the time-out value is provided through new flag specifications. Note that the client-bid cannot override the time-out specified by an OTMA descriptor or command. If the client-bid time-out value is equal to or greater than the current time-out value set by descriptor or command, OTMA will ignore the time-out value in the client-bid message. If the client-bid time-out value is less than the current time-out value set by the command or descriptor, the value from the client-bid will be used for the time-out action for this member.

CM1 (Send-then-Commit) Time-out Control ...

- Details ...
 - ◆ OTMA Member...
 - On an individual message level
 - 1-byte reserved field at offset **x'1E'** of the message control information prefix
 - Specifies time-out value for the input transaction
 - New flag **TMAMTTMO, x'08'** in byte 5 of the state data section
 - Specifies that OTMA can take the message level time-out value specified in the control data

18

On an individual message basis, additional flags have been provided for an even lower level of time-out specification. This capability supports a message time-out value which can be different from the time-out value set for the entire member. Note, however, that this value follows similar restrictions in that it cannot override the member time-out value set by an OTMA descriptor or IMS command.

When an OTMA time-out occurs, OTMA will first back-out the transaction in order to get out of the wait-syncpoint or wait-RRS condition for any missing ACK/NACK region. Subsequently, an OTMA CM1 deallocation message will be sent to the member with the existing ABORT flag and the new "time-out" flag. The IMS system console operator will also receive a DFS0809E message "IMS REGION region-id IN BACK-END IMS aaaaaaaa HAS TIMED OUT FOR otma-membname/tpipename FOR xx MINUTES". When OTMA takes the time-out action, byte 3 TMAMCCCI, of the OTMA commit-confirmation flag in the message control data prefix is set to TMAMCTMO, X'08', to indicate that the transaction was aborted due to the OTMA time-out condition.

TPIPE Storage Clean-up

- Enhancement to release unused storage
 - ◆ Supports the removal of unused TPIPEs
 - Idle for two checkpoints

- This capability is available in IMS V8 and IMS V9
 - ◆ IMS V8: PQ99983, IMS V9: PK00386

19

The next enhancement provides a more efficient way to control unused storage associated with idle TPIPEs. TPIPEs (Transaction Pipes) are OTMA control blocks that represent logical connections between the client and IMS. They are analogous to an IMS logical terminal (LTERM) and allow IMS to associate all input and output with a particular OTMA client. Once created, they occupy storage whether or not they are used again. This clean-up enhancement determines whether or not an inactive TPIPE can be deleted and its storage released. A TPIPE is considered inactive if it has been idle for 2 consecutive checkpoints.

TPIPE Storage Clean-up ...

- Implementation
 - ◆ New clean-up logic applies to consecutive IMS system checkpoints
 - The first checkpoint
 - Scans all the existing TPIPEs to see if input or output activities have occurred
 - If yes, then the TPIPE is not idle
 - If no activities, then the TPIPE is marked idle
 - In the subsequent checkpoint,
 - If an idle TPIPE is found and there has still been no activity the TPIPE is a candidate for removal
 - **Note:** Certain TPIPEs are never considered as candidates for clean-up
 - Synchronized TPIPEs from MQ
 - TPIPEs with status conditions such as TRA, STO, and TMP

20

The TPIPE storage clean-up function is activated when OTMA is activated in an IMS system. Logic added to IMS system checkpoint processing determines whether a TPIPE is active or idle based upon whether or not there are any input or output messages associated with the control block in addition to whether or not any TPIPE status conditions exist. TPIPEs that are idle for two consecutive checkpoints are deleted.

Active TPIPEs include those that are processing commit-then-send (CM0) messages in a shared queues environment, have incomplete send-then-commit (CM1) messages, or have queued commit-then-send (CM0) output messages.

Certain TPIPEs are never considered for removal. These include synchronized MQ TPIPEs and TPIPEs with outstanding status indicators.

Member Level Security

- New /SECURE OTMA command capability
 - ◆ Allows each OTMA member to define its own security setting
 - FULL, CHECK, NONE, or PROFILE
 - ◆ Dynamic change of security level
 - Note - Messages are processed with the security level that was in effect when the message was received

- Prior Releases
 - ◆ OTMA security was a system-wide setting for all OTMA members

21

Prior to IMS V10, OTMA does not allow different security levels defined for various members. The security setting requested is considered a system-wide setting for all of OTMA members. In V10, the OTMA command, /SECURE OTMA, has been enhanced to allow specification of member security so that each OTMA client can have its own security level.

Note - Messages are always processed with the security level that was in effect when the message was received. Even if a new security level is introduced by command, the security level associated with the message is based on the level in effect at the time of message receipt.

Member Level Security...

- Implementation

- ◆ `/SECURE OTMA security-option TMEMBER member-name`
 - Where *security-option* is FULL | CHECK | NONE | PROFILE

- ◆ `/DISPLAY TMEMBER` command
 - Enhanced to show security status for a specific member

<code>/DISPLAY TMEMBER MQ1</code>			
Response ET:			
<code>GROUP/MEMBER</code>	<code>XCF-STATUS</code>	<code>USER-STATUS</code>	SECURITY
<code>IMSGROUP</code>			
<code>-MQ1</code>	<code>ACTIVE</code>	<code>ACCEPT TRAFFIC</code>	CHECK

<code>/DISPLAY TMEMBER ICONN01</code>			
Response ET:			
<code>GROUP/MEMBER</code>	<code>XCF-STATUS</code>	<code>USER-STATUS</code>	SECURITY
<code>IMSGROUP</code>			
<code>-ICONN01</code>	<code>ACTIVE</code>	<code>ACCEPT TRAFFIC</code>	FULL

22

The addition of the TMEMBER keyword to the /SECURE OTMA command provides the ability to define any of the security options for a specific OTMA member.

The /DISPLAY command has also been enhanced to provide a mechanism to display the security option in effect for a specific member.

Asynchronous Message Enhancements

- Asynchronous messages
 - ◆ CM0 - Commit-then-Send messages
 - ◆ Undelivered IOPCB and all ALTPCB messages

- Enhancements - only apply to IMS Connect
 - ◆ Super Member capability
 - ◆ Purge
 - ◆ Reroute

23

Several enhancements are available to address asynchronous CM0 messages in OTMA environments that use IMS shared queues as well as those that implement load balancing or IP spraying techniques such as Sysplex Distributor or the WebSphere Edge Server. These enhancements extend the capabilities of IMS Connect.

Asynchronous Message Enhancements ...

- Super Member capability
 - ◆ Facilitates delivery of IMS asynchronous messages
 - By a set of OTMA clients running in the sysplex
 - Multiple instances of IMS Connect
 - Supports load balancing / IP Spraying solutions such as Sysplex Distributor
 - By IMS systems in a shared queues environment
 - Multiple front-end and back-end IMS systems
 - Removes affinity to a particular IMS
- Also delivered in previous releases:
 - IMS V9 and IMS Connect: PK09946, PK30086, PK10911
 - IMS V8: PK09944, PK30103, IMS Connect V2.2: PK10910

24

The Super Member capability in OTMA facilitates the delivery of asynchronous (CM0 Commit-then-send) messages by any instance within a set of IMS Connects and IMS subsystems. With this function, affinity to a particular IMS or IMS Connect is removed and the use of shared queues as well as solutions such as Sysplex Distributor become more viable.

Asynchronous Message Enhancements ...

- Super Member capability ...
 - ◆ A group name given to a set of OTMA members
 - E.g., IMS Connect instances
 - Allows any IMS Connect instance to retrieve the message
 - ◆ Generic structure name in IMS on which the messages are queued
 - OTMA members that share a super member name
 - Recognized by IMS by both their specific name and super member name
 - For Shared Queues, no affinity to an IMS system
 - Any IMS can deliver the message

25

A super member is a special OTMA member name which can be shared by a set of IMS Connects to handle the CM0 hold queue messages.

When an IMS Connect attaches to IMS, OTMA creates a regular member structure name unique to that instance to track the connection status and to record the connection options for later transaction processing. If a super member name, which could be considered a group name for a set of IMS Connects using Sysplex Distributor or similar mechanism, is given by an IMS Connect during the connection time, OTMA will additionally create a member structure called the super member structure. If the super member structure already exists, OTMA will use it instead of creating a new one. A regular member structure is dedicated exclusively to the IMS Connect for which it was created. However, a super member structure is shared among a set of IMS Connects so that a Resume TPIPE can be issued from any IMS Connect. The role of the super member is to store and deliver the CM0 hold queue messages.

Asynchronous Message Enhancements ...

- Super Member - Implementation
 - ◆ IMS Connect HWSCFGxx configuration file
 - HWS statement

HWS ... SMEMBER= *smember-name*


 - *Smember-name* can be 1-4 characters
 - Cannot be the same name as an existing OTMA member
- Command support
 - ◆ /DISPLAY OTMA
 - Display output includes a new SMEM column
 - ◆ /TRA TMEMBER... TPIPE..., /STA or /STO OTMA
 - Can be issued to a super member name
 - When issued with a regular member name
 - Output is expanded to include any related super members

26

IMS Connect has been enhanced to support the Super Member capability. The HWSCFGxx configuration file member provides a new SMEMBER= parameter in the HWS statement. The value provided must be different than the value provided in the MEMBER= parameter of the DATASTORE statement and must follow the OTMA naming conventions.

When the client-bid protocol message is sent from an IMS Connect member that has specified a Super Member value, the state data of the OTMA prefix carries the defined SMEMBER value in a new 4-byte field at the offset of x'36' in the state data section. In addition to the super member name in the client-bid message, a new flag TMAMFGSM (x'08') at the offset x'2D' of the state data informs OTMA that the super member processing is requested.

Additionally, the /DIS, /TRA, /STA and /STO commands have been enhanced to support the Super Member capability. The /DISPLAY OTMA command output includes a new SMEM column to display the Super Member name if one exists. The /TRA TMEMBER ... TPIPE, /STA OTMA and /STO OTMA commands can be issued against either a Super Member name or regular IMS Connect.

Information On Demand IMS Version 10 


Asynchronous Message Enhancements ...

- Super member capability - the issue

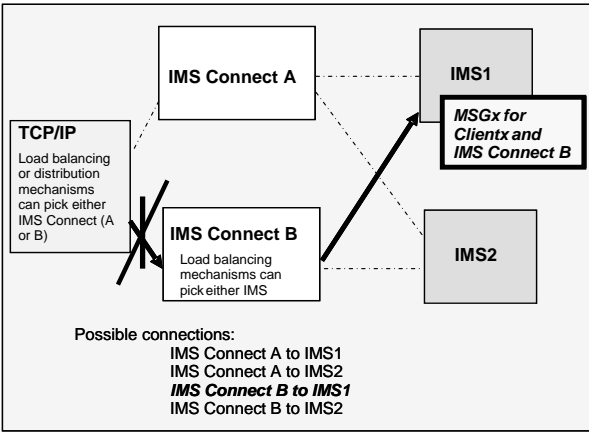
Resume TPIPE
for clientx through
IMS Connect (generic)
and IMS (generic)

Will only retrieve MSGx if
the connection is correctly
established with IMS
Connect B and IMS1

If connection is established
through any other path, this
program will either
disconnect or timeout at
some point.



Sysplex

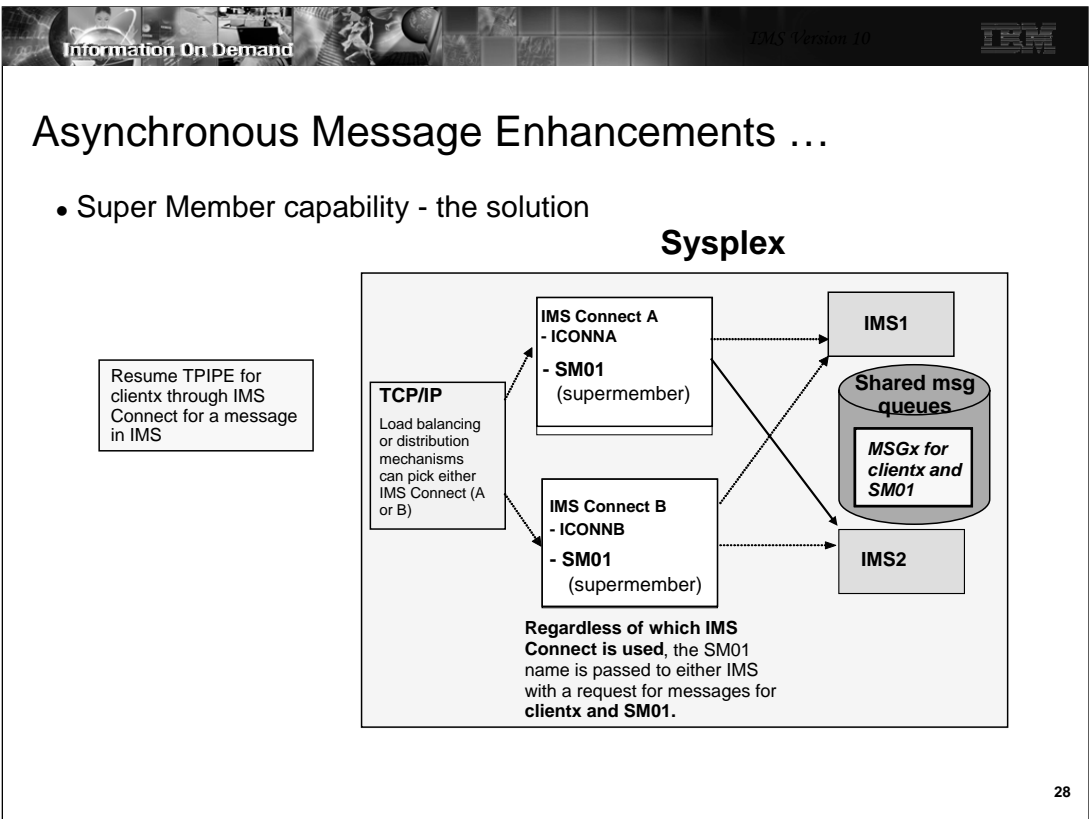


Possible connections:
 IMS Connect A to IMS1
 IMS Connect A to IMS2
IMS Connect B to IMS1
 IMS Connect B to IMS2

27

This and the next visual attempt to illustrate the value of the Super Member support.

Asynchronous CMO output messages in IMS are queued to a message queue construct that is identified by TMEMBER and TPIPE and, therefore, associated with a specific IMS Connect instance. As a result, there is a potential issue when using any of the load balancing or sysplex distribution mechanisms or even when a message is processed in a back-end IMS in a shared queues group. To retrieve the message, the remote program needs to establish a connection through the appropriate IMS Connect to the actual IMS system that queued the message. This can be a challenge because there is no easy mechanism for a remote program to discover the required connection path, i.e., a specific IMS Connect to a specific IMS. Additionally, the remote programs may not want to know specific connection paths to IMS because that would negate the value of using load balancing and distribution mechanisms.



The OTMA Super Member function resolves the issue by allowing a RESUME TPIPE request to retrieve CM0 output across all combinations of IMS Connect and IMS systems. If multiple IMS systems are involved, then those IMS systems must also have IMS shared queues implemented. If there is only one IMS system but multiple IMS Connects, then shared queues support is not required. As shown in the illustration on this visual, the Resume TPIPE request for clientx can be routed through any load balancing or distribution mechanism to either IMS Connect A or IMS Connect B. Both systems are identified to IMS1 and IMS2 by their unique XCF member names as well as the global Super Member name of SM01. The request to retrieve the output message for TPIPE clientx can be sent to either IMS1 or IMS2 because both have access to the shared queues and, more specifically, to all the messages under the shared queues construct for SM01 and TPIPE clientx.

Asynchronous Message Enhancements ...

- REROUTE and PURGE Support
 - ◆ For OTMA hold queue capable clients ONLY
 - OTMA members, e.g. IMS Connect
 - That keep asynchronous messages on the IMS message queue until requested by the remote application
 - ◆ Extended support that allows remote applications to
 - Reject (NAK) or request action on asynchronous messages
 - **Purge** from the IMS message queue
 - **Reroute** to an alternate destination
 - When both REROUTE and PURGE are specified for a message
 - Neither action is performed
 - OTMA issues DFS2407W message
 - Stores the message onto the hold queue of the inputting TPIPE
- Also delivered in previous releases

29

OTMA asynchronous message enhancements has been extended to provide greater flexibility for messages that cannot be delivered or are rejected by the remote client. The two actions, which are mutually exclusive of each other, are to reroute messages that cannot be delivered to an alternate destination or to purge them.

The REROUTE enhancement provides a mechanism for an IMS Connect client to request that undeliverable Commit Mode Zero (CM0) output associated with a send/receive from the client application be rerouted to an alternate IMS Connect destination. When a Client Reroute Request is made, IMS Connect notifies OTMA to remove the message from the current queue and requeue it to the provided Reroute name queue.

The Purge Not Deliverable extension allows the remote client application or the appropriate User Message Exit to specify whether or not the output should be purged if it is not deliverable. Note that if both capabilities are requested, neither action is performed and DFS2407W message is issued.

This capability was delivered in previous releases as follows: IMS V9 and IMS Connect: PK16934, PK22480, PK24907, PK09543, PK12013; and for

IMS V8: PK21868, PK09542, IMS Connect V2.2: PK12012

Asynchronous Message Enhancements ...

- REROUTE and PURGE - Implementation ...
 - ◆ REROUTE
 - OTMA
 - OMUSR_FLAG1 new setting of OMUSR_REROUT (X'01')
 - OMUSR_ARCLEV new setting of OMUSR_AL02 (X'02')
 - Field OMUSR_REROUT_NM holds the reroute name
 - IMS Connect
 - IRM_ARCH new setting of IRM_ARCH1 (X'01')
 - IRM_F3 new setting of IRM_F3_REROUT (X'08')
 - New field IRM_REROUT_NM holds the reroute name
 - ◆ PURGE
 - OTMA
 - OMHDRPND EQU X'10' purge if not deliverable
 - IMS Connect
 - IRM_F3 has a setting of IRM_F3_PURGE (X'04')

Both the OTMA headers and the IMS Connect headers have been enhanced to request either the PURGE or REROUTE capability.

Asynchronous Message Enhancements ...

- IMS Connect support
 - ◆ Resume TPIPE
 - Protocol provides flags for PURGE and REROUTE requests
 - When a Resume TPIPE request is in progress
 - Subsequent Resume TPIPE request is queued instead of rejected
 - ◆ Send-Only messages
 - Can specify a reroute queue name for the output
 - Also delivered in IMS V9: PK17421, PK18555

31

IMS Connect has been enhanced to take advantage of the new capabilities. The Resume TPIPE protocol provides flags to indicate one or the other type of request.

Another enhancement in this area allows RESUME TPIPE requests to be queued when requested for the same TPIPE name.

Additionally, the Send-Only protocol allows specification of a reroute queue name for the output in the case of an initial rejection of the output reply. This capability allows IOPCB output from the Send-Only input transaction to be rerouted to a dedicated TPIPE instead of the inputting TPIPE. The user of the IMS Connect Send-Only transaction will need to turn on the reroute flag and specify a reroute TPIPE name in the input stream to activate the capability. Note that this is not supported by the local option capability, HWSIMSO0, HWSIMS01 and HWSJAVA0.

OTMA /DISPLAY Command Enhancements

- /DIS OTMA and /DIS TMEMBER output has been expanded
 - ◆ Additional information
 - Message flood threshold value, Current number of active input messages, Time-out value, Super Member name, DRU exit name
 - ◆ New USER-STATUS indicators
 - SMQ BACKEND
 - STO-INPUT
 - FLOOD
- /DIS TMEMBER TPIPE
 - ◆ Enhanced to display the number of input messages
 - New column "INPCT"
 - Number wraps after 65535

32

The /DISPLAY OTMA and /DISPLAY TMEMBER command outputs have been enhanced to provide more information about the OTMA environment and specific TMEMBERS. To contain all the information, the single line display output has been increased to two lines.

Several new USER-STATUS indicators provide the following information:

- SMQ BACKEND - This status indicator on a TMEMBER line shows up on a back-end IMS in a shared queues group. It shows that OTMA has duplicated the specific TMEMBER environment and control blocks needed on the back-end to process a message that was received from that TMEMBER which is attached on the front-end. The same member name on the front-end IMS is shown in 'connected' state and no SMQ BACKEND indicator. Whereas the corresponding control blocks on the back-end IMS which processes the message is shown in 'disconnected' state with an SMQ BACKEND indicator.
- STO-INPUT - This status shows that the /STOP TMEMBER command has been issued for a specific member-name and no new input can be accepted.
- FLOOD - This status indicator shows that a specific TMEMBER is in a message flood condition and that the maximum input message count that was specified has been reached.

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

OTMA Processing during Restart

- New OTMA=M option in DFSPBxx
 - ◆ Option to control OTMA functionality during all restarts including ERE
 - IMS does not enable OTMA during the system initialization
 - /START OTMA commands are not recovered during restarts

- /START OTMA NOCHECK
 - ◆ Command to start OTMA as a non-recoverable request during restart
 - Capability is introduced for OTMA=N users

- Also delivered in IMS V8 and IMS V9
 - ◆ IMS V8: PK14679, IMSV9: PK14680

33

In addition to the existing OTMA values of Y and N, IMS V10 introduces the option of OTMA=M (manual). When IMS first initializes, the value of OTMA=M functions similarly to OTMA=N such that OTMA is not started. Once IMS is up and running, the /STA OTMA command can be issued but becomes non-recoverable. The setting of OTMA=M, therefore, takes effect when IMS terminates either normally or abnormally and has to be restarted. During restart processing, OTMA is not automatically restarted. This capability was introduced to prevent looping abend situations where IMS may have terminated as a result of OTMA error conditions.

An additional impact on OTMA restart processing is introduced for environments where IMS is initialized with OTMA=N. If a /START OTMA NOCHECK command is issued, the command is also not recovered during either a warm start or emergency restart.

Migration Considerations

- Message flood control
 - ◆ Default limit of 5000 is set at initialization
 - To deactivate the function, define the INPUT parameter in the DFSYDTx descriptor or issue the /STA TMEMBER INPUT command
- Time-out
 - ◆ Default is set to 5 minutes
 - To deactivate the function, define the T/O parameter in the DFSYDTx descriptor or issue the /STA TMEMBER TIMEOUT command
- /DIS OTMA and /DIS TMEMBER command enhancements
 - ◆ Single line output has been expanded to two-line output in order to include DRU exit name info and time-out info.
 - ◆ New information in the /DISPLAY TMEMBER TPIPE provides the input message count

34

The considerations listed on this visual address issues that should be considered when migrating from a previous release of IMS. The assumption is that migration to IMS V10 is based on existing functionality without adding any new capabilities during the migration process.

The message flood control enhancement in IMS V10 is automatically enabled with a default limit of 5000 messages. To provide compatibility with previous releases and deactivate the support, either specify an INPUT value of 0 in a descriptor or issue the /STA TMEMBER INPUT command.

Likewise, the timeout support for synchronous CM1 message is automatically enabled with a default value of 5 minutes. To provide compatibility with previous releases and deactivate the support, specify T/O value of 0 in a descriptor or issuing the /STA TMEMBER TIMEOUT command.

Note that the /DISPLAY command output associated with OTMA and TMEMBER requests has been expanded to two output lines and includes new information. As a migration consideration, this is a key issue for automated operations.

OTMA Enhancements - Benefits

- Destination Routing support
 - ◆ Facilitates outbound messages through an architected OTMA layer
 - Minimizes the need for coding OTMA exit routines through descriptors
 - Enables growth of e-business environments using the IMS Connector for Java and the IMS SOAP Gateway

- Resume TPIPE security
 - ◆ Ensures that only authorized users can retrieve output messages in the hold queue
 - Supports IMS Connect clients using Resume TPIPE commands

- Message Flood protection
 - ◆ Provides a mechanism to prevent OTMA clients from flooding the IMS message queues and causing S40D abends

35

The next three visuals summarize the enhancements in this section.

OTMA Enhancements - Benefits...

- Time-out
 - ◆ Allows automatic detection and resolution of “hung” conditions identified as wait-synpoint or wait-RRS

- TPIPE clean-up
 - ◆ Greater efficiency of storage usage for OTMA control blocks

- Member level security
 - ◆ Greater control of security environment for OTMA
 - Allows different options on a member level

OTMA Enhancements - Benefits...

- Asynchronous message enhancement
 - ◆ Super Member support
 - Ensures delivery of CM0 messages in a Shared Queues environment or with the use of IP load balancing techniques
 - ◆ Purge and Reroute
 - Provide greater control of output messages that cannot be delivered to original destination

- /DIS OTMA and /DIS TMEMBER output display enhancements
 - ◆ Provide more comprehensive information on the environment

- OTMA=M option and /STA OTMA with NOCHECK option
 - ◆ Protect IMS from restarting an OTMA system that has caused IMS to terminate abnormally



IMS Connect Enhancements

Highlights

- ACEE aging value support
- Client password change request
- RACF mixed case password
- CM1 timeout
- Message flood control
- Asynchronous message enhancements
 - ◆ Super member, Reroute and Purge Not Deliverable
 - ◆ Port affinity
 - ◆ Alternate clientid
- XML Adapter support
- IMS SOA Composite Business Application Support

IMS Connect provides several usability, availability and security enhancements.

Information On Demand IMS Version 10

ACEE Aging Value Support

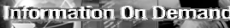


- New OAAV parameter in the DATASTORE statement of the HWSCFGxx file
 - ◆ OTMA ACEE aging value in seconds
 - 0 to 2147483647 (x' 7FFFFFFF' in OTMA)
 - Values between 0 and 300 are reset to 300
 - Example: DATASTORE ID=... **OAAV=360**
 - ◆ OTMA aging value in effect can be displayed
 - ◆ VIEWHWS/VIEWDS command output
 - ◆ MVS Modify Command QUERY MEMBER and QUERY DATASTORE output
 - Example: RACF APPL NAME=
OTMA ACEE AGING VALUE=360

40

The Access Control Environment Element (ACEE) is a control block that represents a verified userid to IMS. The ACEE is used to determine the user's authorization to the IMS command or IMS transaction requested in the input message. Once built in OTMA, the ACEE for each userid is cached and the aging value associated with each OTMA client, e.g., IMS Connect, is kept in a table. The aging value is then used to determine when the cached control block should expire and be refreshed. IMS re-creates the ACEE if a message associated with the userid is received but the age of the current ACEE is greater than the aging value. The aging value is used to balance performance (possible RACF I/O to refresh the ACEE) and integrity. For IMS Connect, the ACEE expiration value is specified during the client-bid process and is set to a default of no expiration.

IMS V10 provides a new parameter, OAAV, in the DATASTORE statement of the HWSCFGxx file for specification of an OTMA ACEE aging value. If not specified, the default continues to be 2147483647 which, in essence, means no expiration and is the maximum value supported by OTMA.

The VIEWHWS, VIEWDS, QUERY MEMBER and QUERY DATASTORE command output displays have all been enhanced to show the aging value that is in effect for the associated environment.

Client Password Change Request

- New mechanism for a remote client to request that a SAF/RACF password be changed

HWSPWCH old-password / new-password-1 / new-password-2

 - Where new-password-1 and new-password-2 are the same value

- HWSPWCH
 - ◆ Defined keyword supported by HWSSMPL0, HWSSMPL1, HWSJAVA0
 - ◆ To enable the function
 - HWSPWCH0 address must be established in the exit routine
 - Include the HWSPWCH0 object code
 - Define 'INCLUDE TEXT(HWSPWCH0)' statement in the Binder JCL

41

IMS Connect provides a new mechanism that allows a remote client to request that the SAF/RACF password associated with a userid be changed.

As provided, the new capability will be supported in the HWSSMPL0, HWSSMPL1 and HWSJAVA0 exit routines. The routines check for a leading keyword of 'HWSPWCH' to determine whether it is a request to change the password. This 'HWSPWCH' string can be viewed as a transaction code but new logic in the routine, HWSPWCH0, is called to process the special request. HWSSMPL0, HWSSMPL1 and HWSJAVA0 can be modified to define the HWSPWCH constant with a unique keyword value other than 'HWSPWCH'.

The exit routines pass HWSPWCH0 the request keyword length in IMSEA_PWCHKEYL field. This allows HWSPWCH0 to process the password change request independently from the request keyword. After regaining control, the exit routines check the return code in register 15. A zero return code means the request is successful. If it is a non-zero return code, the exit routine checks the IMSEA_ERCD field for a valid error code, and replies back to the client with an error message specified in IMSEA_MSGTEXT and IMSEA_MSGLEN fields.

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

Client Password Change Request ...

- Remote client support
 - ◆ IMS TM Resource Adapter (formerly IC4J)

```
LLLL | IRM | OTMA | LLZZ HWSPWCH old-password / new-password1 / new-password2 / EOM
```

- ◆ Other clients

```
LLLL | IRM | LLZZ HWSPWCH old-password / new-password1 / new-password2 / EOM
```

42

The password change support in HWSJAVA0 can be invoked by the IMS TM Resource adapter (formerly called IMS Connector for JAVA or IC4J) client. After the OTMA headers, the message sent begins with the defined keyword HWSPWCH followed by the old and new passwords.

Similarly, other clients that invoke exit routines based on HWSSMPL0 or HWSSMPL1 can supply the HWSPWCH request after the IRM header.

Information On Demand IMS Version 10

RACF Mixed Case Password Support

- Enhancement to enable RACF mixed case passwords
 - ◆ PSWDMC parameter in HWS statement in IMS Connect HWSCFGxx

HWS ...PSWDMC = Y | N
 - ◆ IMS Connect command

SETPWMC ON | OFF
 - ◆ IMS Connect UPDATE command

F imsconnproc, UPDATE MEMBER TYPE(IMSCON) SET (PSWDMC(ON | OFF))
- Requires that RACF support is enabled

RACF SETROPTS(MIXEDCASE)

43

The support for RACF mixed case password in IMS Connect is aligned with the IMS V10 support for mixed case passwords. The capability in IMS Connect allows the password to be preserved exactly as the remote client provided and pass the string to RACF without translation to upper case.

The function can be turned on as follows:

A new parameter, PSWDMC= in the HWS= statement, defines the option of mixed case passwords where PSWDMC=N is the default. This setting can be changed using the IMS Connect SETPWMC or UPDATE command.

IMS Connect also provides a command, SETPWMC, that can override the HWSCFGxx specification.

The PSWDMC keyword is also available to the IMS Connect UPDATE command as another way to request the support.

The IMS Connect support requires that RACF enable mixed case passwords through the RACF SETROPTS(MIXEDCASE) command. Note that the RACF enablement of this support does not constitute the IMS Connect usage of this support. Also note that the mixed case support for IMS Connect can only take effect when RACF is enabled.

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

- Time-out Control capability for CM1 interactions
 - ◆ Supports the associated function in the OTMA Enhancements section
 - Resolves “Wait-Syncpoint” and “Wait-RRS” situations

- New **ACKTO=** parameter in the DATASTORE statement of the HWSCFGxx file
 - ◆ ACKTO = 0 to 255 seconds. Default in OTMA is 120.
 - Example:

DATASTORE ID=..., ACKTO=120

 - Adjustment of values:
 - If specified as 0, reset to 120 or value specified in OTMA
 - If specified outside the 0-255 range, reset to the OTMA value
 - Not accepted if value is greater than value in OTMA set by descriptor or command

44

IMS Connect supports the new OTMA time-out control function for send-then-commit CM1 interactions. OTMA provides a default value of 120 seconds after which transactions that are held in “Wait-Syncpoint” or “Wait-RRS” status are released and backed out. If provided, the value in the ACKTO parameter of the IMS Connect configuration DATASTORE statement is passed to OTMA during client-bid processing.

Note the following considerations:

A specified value of 0 is reset to 5.

Any value specified in error between 60 and 255 is reset to 60.

Any value outside the 0-255 range Results in an Abend U3401.

Value cannot be greater than what is defined in OTMA - If the CM1TO value is equal to or greater than the time-out specified in IMS by an OTMA descriptor or /STA TMEMBER command, OTMA will ignore the IMS Connect request. On the other hand, if the CM1TO value is less than the current time-out value set by the OTMA descriptor or command then the value which is passed to IMS by the IMS Connect client-bid process will be used for the time-out action for this member.

CM1 (Send-then-Commit) Time-out Control ...

- Command support
 - ◆ VIEWHWS, VIEWDS, QUERY MEMBER, QUERY DATASTORE
 - Display output

```
...  
OTMA ACEE AGING VALUE=value  
OTMA CM1 TIMEOUT VALUE=value
```

- If time-out occurs
 - ◆ Remote Client receives a deallocate of the connection and an RSM status message

45

IMS Connect command output has been enhanced to display the CM1 timeout value. The applicable commands include: VIEWHWS and VIEWDS output command and the MVS MODIFY command for QUERY MEMBER and QUERY DATASTORE.

Message Flood Control

- Capability that monitors the growth of active input messages
 - ◆ Supports the associated function in the OTMA Enhancements section
 - Prevents flooding IMS with input messages if they cannot be processed in a timely manner

- New **MAXI=** parameter in the DATASTORE statement of HWSCFGxx
 - ◆ MAXI = 0 to 9999. Default in OTMA is 5000.
 - Example: `DATASTORE ID=..., MAXI=5000`
 - Adjustment of values:
 - If specified as 0, reset to 5000
 - Between 0 and 200, reset to 200
 - Between 9999 and 65535, reset to 9999
 - If specified outside the 0-65535 range, Abend U3401
 - Not accepted if value is greater than value in OTMA set by descriptor or command

46

IMS Connect also takes advantage of the OTMA Message Flood Control capability to automatically monitor the growth of active input messages. OTMA provides a default value of 5000 messages after which input messages from a specific IMS Connect instance will be rejected. If provided, an override value in the MAXI parameter of the IMS Connect configuration DATASTORE statement is passed to OTMA during client-bid processing.

Note the following considerations:

A specified value of 0 is reset to 5000.

A value between 0 and 200 is reset to 200.

Any value specified in error between 9999 and 65535 is reset to 9999.

Any value outside the 0-65535 range Results in an Abend U3401.

Value cannot be greater than what is defined in OTMA - If the MAXI value is equal to or greater than the INPUT value specified in IMS by an OTMA descriptor or /STA TMEMBER command, OTMA will ignore the IMS Connect request. On the other hand, if the MAXI value is between 1 and 200, a value of 200 will be sent to IMS. If the IMS Connect value is less than the value specified in OTMA, then the IMS Connect MAXI value will be used.

Message Flood Control ...

- Command support
 - ◆ VIEWHWS, VIEWDS, QUERY MEMBER, QUERY DATASTORE
 - Display output

```
...  
OTMA ACEE AGING VALUE=value  
OTMA CM1 TIMEOUT VALUE=value  
OTMA MAX INPUT MESSAGE=value
```
- If input messages are rejected due to message flood protection
 - ◆ Remote Client receives RSM status message

47

IMS Connect command output has been enhanced to display the maximum input message value. The applicable commands include: VIEWHWS and VIEWDS output command and to the MVS MODIFY command for QUERY MEMBER and QUERY DATASTORE.

Asynchronous Message (CM0) Enhancements

- Functions that are documented and described in the OTMA Enhancements section
 - ◆ Super member support
 - ◆ Reroute
 - ◆ Purge

There are several enhancements for CM0 processing that are described in detail in the OTMA Enhancements section. The detail in the OTMA section includes specifics for IMS Connect.

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

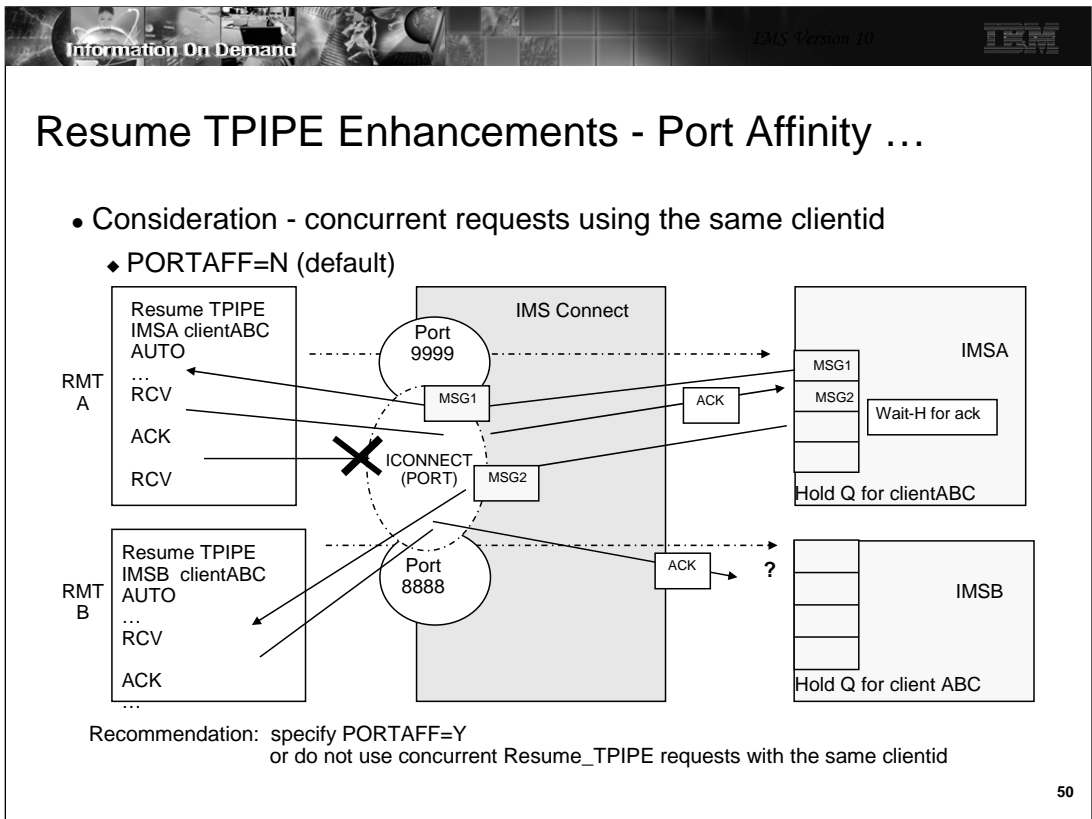
49

Using concurrent Resume TPIPE connection requests of the same clientid across several ports may cause problems such as keeping an IMS OTMA TPIPE in WAIT-H status. To address the issue and support this concurrency requirement, IMS Connect provides a new parameter to enforce all the correlated interaction such as the retrieval of a message and associated ACK or NAK to the same remote client instance. The PORTAFF parameter in the TCPIP statement controls whether commit-then-send (CM0) output messages sent by IMS to an IMS Connect system have affinity to the port on which IMS Connect received the original input message.

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

When PORTAFF=N is specified, IMS Connect attempts to return the CM0 output to the first client it finds on any available port with an outstanding request from this clientid.

NOTE: If running in a sysplex environment that has implemented redundancy, load balancing, and supermember support, etc., PORTAFF=N is a reasonable choice. Using the same instance of a single clientid across multiple ports is not recommended.

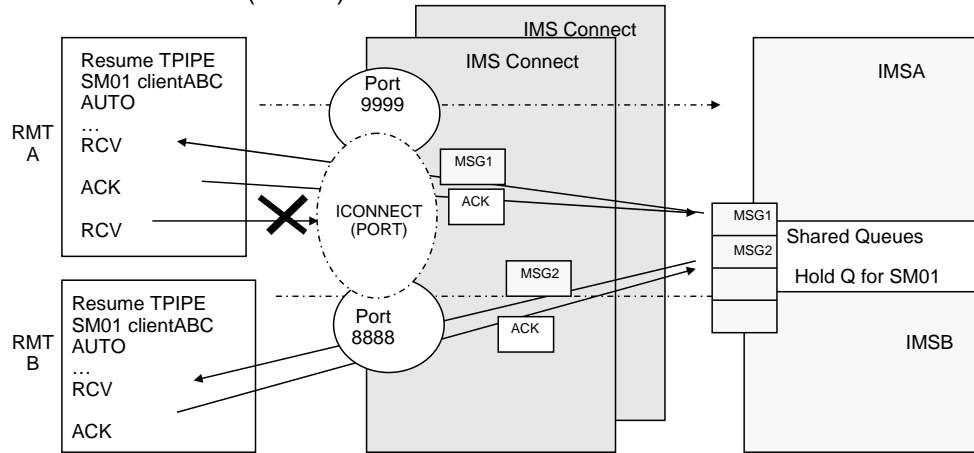


As mentioned in the previous visual, when PORTAFF=N, IMS Connect attempts to return the CM0 output to the first port found on which the client ID of this IMS connect is present. PORTAFF=N is the default. There is a consideration in this area because if a message is retrieved from the Hold Queue, then the PORTID in OMUSER_PORTID is ignored and IMS Connect assumes the portid is the generic ICONNECT port. This assumption tends not to be a problem unless an error such as a connection failure occurs. When that happens and a message has been received in IMS Connect for delivery, IMS Connect scans all the ports under the generic ICONNECT port and delivers the message to the first one that it finds.

The example on this visual shows a situation where two Resume_TPIPE AUTO requests specifying the same clientid, clientABC, are sent into a single IMS Connect. One request (from RMTA) is sent to IMSA and the other (from RMTB) is sent to IMSB. IMSA has two messages, MSG1 and MSG2, on the Hold Queue. IMSB has no messages at the moment and so clientABC on RMTB just waits. IMSA sends MSG1 to IMS Connect which delivers the message to the outstanding request for clientABC on RMTA which responds with an ACK. In this scenario the connection fails for one of several reasons - ACK timeout, network problem, etc. Since the Resume_TPIPE had originally specified AUTO, IMSA sends MSG2 after the ack for MSG1 is received. When IMS Connect receives the message, it detects that the connection to RMTA is no longer there and scans the ports under ICONNECT to find the first one available. IMS Connect sends MSG2 to the 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.

Resume TPIPE Enhancements - Port Affinity ...

- In a Sysplex Distributor, Super Member, Shared Queues environment
 - ◆ PORTAFF=N (default) is feasible



In an environment that supports full redundancy including Shared Queues, Super Member support, Sysplex Distributor, etc., specifying PORTAFF=N is feasible. This configuration with concurrent Resume_TPIPE clients using the same clientid presumes that any of the remote clients can retrieve any of the messages.

Resume TPIPE Enhancements - Alternate Clientid

- Capability to request and retrieve asynchronous output messages that are queued to another client
 - ◆ Supports a generic or server application that retrieves messages originally destined for another application
 - ◆ Resume TPIPE request specifies an alternate clientid
 - OTMA delivers output messages queued to the alternate name to the requesting application
 - ◆ Note: this new support differs from but leverages the Reroute capability which only addresses undeliverable and/or NAK'ed messages

52

IMS Connect introduces a new protocol that allows client applications to specify an alternate clientid in the RESUME TPIPE request. IMS Connect forwards the alternate clientid to OTMA, and OTMA returns the asynchronous messages that are queued to the TPIPE of the alternate clientid name to the client application that issued the Resume TPIPE request.

This Alternate Clientid function differs from the Reroute capability that was discussed earlier. Reroute requests address the situations when messages cannot be delivered or are NAK'ed. In these situations, OTMA queues the message onto the TPIPE name associated with the reroute request when the undeliverable condition occurs. The Alternate Clientid function, on the other hand, supports Resume TPIPE requests that retrieve messages which are already queued to a TPIPE name but the name is different. This new capability could be used to provide a programmable solution in the OTMA/IMS Connect environment that is comparable to the way that IMS users can assign an LTERM and all messages queued to it to a different node.

Resume TPIPE Enhancements - Alternate Clientid ...

- Supported by IMS Connect user message exits
 - ◆ HWSSMPL0, HWSSMPL1, HWSSOAP1, HWSJAVA0
 - IRM Header
 - IRM_RT_ALTCID 8 bytes specifying the alternate clientid
 - Occupies same offset as reroute name IRM_REROUT_NM field
 - OTMA header
 - OMUSR_RT_ALTCID 8 bytes specifying the alternate clientid
 - Occupies the same offset of OMUSR_REROUT_NM field
- Resume TPIPE security (discussed in the OTMA section)
 - ◆ IF RIMS|Rxxxxxxx resource class is enabled
 - Required userid access to the alternate clientid (TPIPE)

53

This support applies to: remote client applications that invoke user message exits HWSSMPL0 and HWSSMPL1; IMS SOAP Gateway client applications using user message exit HWSSOAP1; IMS Connector for Java (IC4J) client applications using user message exit HWSJAVA0; and any other user-written IMS Connect message exits. No support is provided for the local option, HWSIMSO0 and HWSIMSO1.

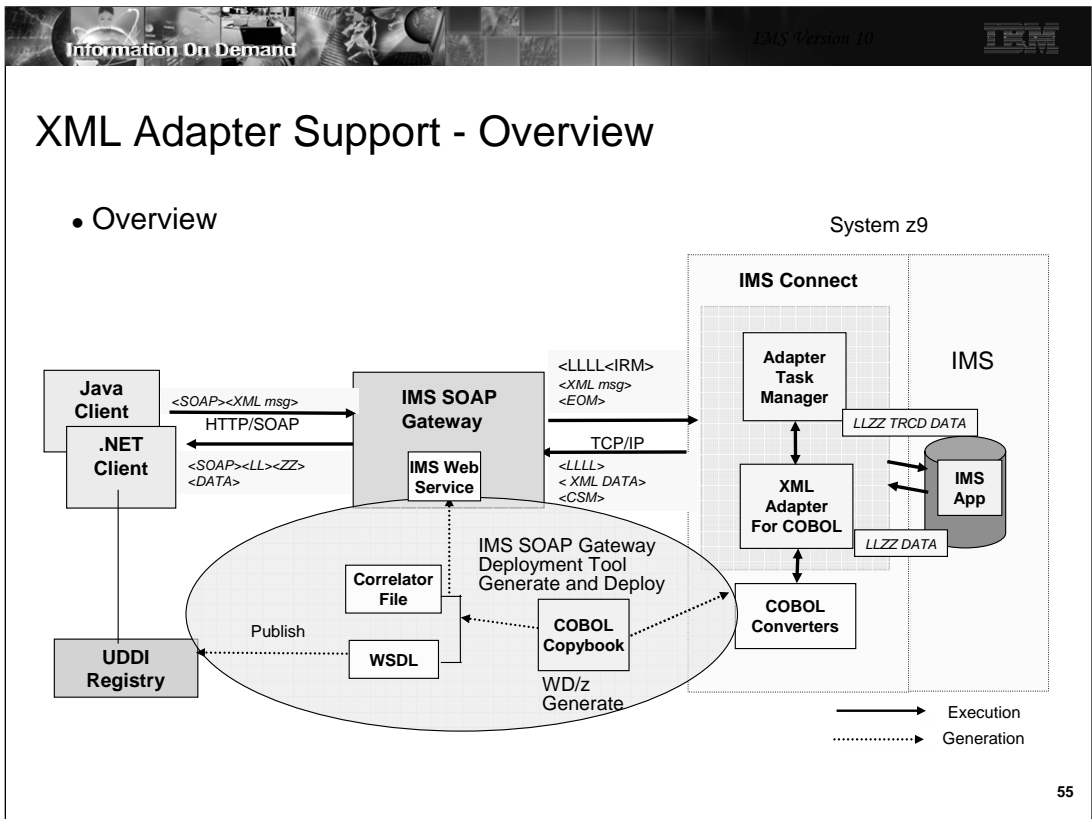
To take advantage of the new capability, optional fields have been added to both the IRM and OTMA headers. Either field, IRM_RT_ALTCID in the IRM or OMUSR_RT_ALTCID can be used to set a valid alternate clientid value. The feature is not enabled if the fields are both set to blanks.

XML Adapter Support

- Capability that supports translation between XML messages and IMS messages
 - ◆ IMS Connect client, e.g., IMS SOAP Gateway (available with IMS V9)
 - Sends an XML message with a request for translation
 - ◆ IMS Connect
 - Inbound: invokes the XML Adapter to translate message for IMS
 - Removes XML tags
 - If necessary, convert from UNICODE to EBCDIC
 - Outbound: invokes the XML Adapter to prepare an XML message
 - If necessary, convert from EBCDIC to appropriate UNICODE encoding schema
 - Create XML tags
- IMS V9 Support - PK24912, PK29938

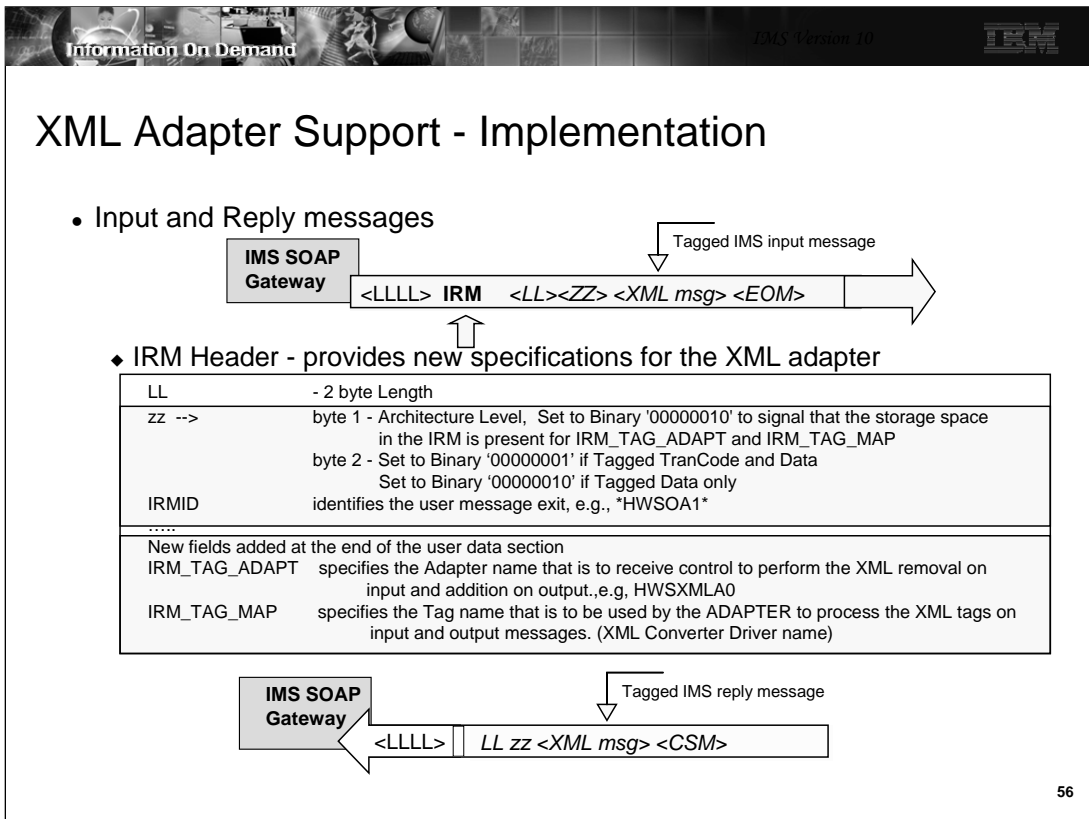
54

The XML Adapter support opens up IMS Connect to: receive and recognize messages containing XML tags; provide the conversion into a message format for IMS including unicode to EBCDIC translation if needed; receive IMS Application reply messages that do not contain XML tags; and perform the conversion to XML as well as any EBCDIC to unicode translation prior to replying to the remote client.



This overview visual shows the current supported configuration for the XML Adapter Support in IMS Connect. The IMS Soap Gateway is an IBM-provided function that must be at a minimum level of V9.2 to take advantage of the XML Adapter in IMS Connect.

WebSphere Developer for zSeries (WDz) supports the generation of XML Converters for COBOL applications using COBOL copybooks. The XML Adapter support in IMS Connect, in conjunction with the generated WDz XML Converters, facilitates the conversion of XML transactional requests into byte stream application data structures, and vice versa. Clients can send transactional requests in XML to IMS Connect, where the XML Adapter calls the user specified WDz XML Converter to convert it to byte streams that the IMS application understands, and then IMS Connect sends it to IMS. Responses when received by IMS Connect will be converted to XML before being sent back to the clients. Each IMS application expects its messages to be in a certain data structure; therefore, one XML Converter is needed for each IMS application.



Input messages must adhere to the application protocol, as defined and documented for IMS Connect, by providing an IRM header that provides the name of the message exit routine that is to be called. For XML Adapter support, the IRM provides flags and values that can be used to invoke the appropriate functions.


The IRM header always begins with an LL (2 byte length field) followed by 2 bytes that are usually zeroes for non-XML message processing. When the XML Adapter is to be invoked, the first zz byte can be used to signal the inclusion of 16 bytes in the user data section of the IRM header for specification of the TAG and ADAPTER names. The second byte details whether the tagged data includes the tranocode or not.

At the end of the of the user data section, two fields provide the information for the XML environment:

The IRM_TAG_MAP field provides 8 bytes to specify the TAG name. This is the name of the COBOL Driver, the XML or whatever is required for the Adapter to call or use to perform the XML transformation.

The IRM_TAG_ADAPT field provides 8 bytes to specify the ADAPTER name. This is the name of the routine that is to gain control from IMS Connect to remove/add the XML tags, defined by the TAG name. The same Adapter that processes the input message (removes the XML Tags) will be required to process the output message (add the XML Tags).

Reply messages are sent back by IMS connect in tagged format. When the message is sent by IMS to IMS Connect, the user message exit invokes the Adapter to process the byte array data and convert it to XML tagged data before sending the message to the remote client.

Information On Demand IMS Version 10 

XML Adapter Support - Implementation ...

- Configuration file HWSCFGxx
 - ◆ ADAPTER statement


```
...
DATASTORE ID=...
ADAPTER XML=Y|N
```
- New PROCLIB member, e.g., HWSEXIT0
 - ◆ EXITDEF statement
 - Contains XML Adapter (HWSXMLA0) definitions


```
EXITDEF(TYPE=XMLADAP,EXITS=(HWSXMLA0),
        ABLIM=8,COMP=HWS)
```
- BPE Configuration file BPECFGxx
 - ◆ EXITMBR statement


```
EXITMBR=(HWSEXIT0,HWS)
```

57

To enable the support in IMS Connect, several definitions have to be created:

A new ADAPTER statement in the HWSCFGx file provides the option of identifying the presence or absence of the XML adapter support. The parameter XML= provides two values. XML=Y/N. Y(es) requests the XML adapter function. The default of N(o) disables the function.

The only adapter that is currently supported is the IBM-provided Cobol Adapter HWSXMLA0. The definitions for the adapter are provided in an EXITDEF statement in a special PROCLIB member. The name of the member can be HWSEXIT0, as shown in this example, or it can be any name that is meaningful to the installation. The name that is chosen must also be defined to IMS in the BPE configuration file statement EXITMBR. The values of the EXITDEF statement are as follows:

TYPE=XMLADAP must be coded as is and defines the exit as an XML adapter plug-in to IMS Connect.

EXITS=HWSXMLA0 also must be coded as shown for the IBM-supplied Cobol Adapter.

ABLIM = 0 (unlimited), 1, up to 2,147,483,647 defines the abend limit which is the number of times the XML ADAPTER can abend before it is disabled

COMP=HWS must also be coded as is and requests the HWS component of IMS Connect.

IMS Connect detects the name of the PROCLIB member which contains the adapter information in the EXITDEF statement in the BPE configuration file.

Information On Demand IMS Version 10

XML Adapter Support - Implementation ...

- User Message Exit HWSSOAP1
 - ◆ OCO
 - ◆ IRMID *HWSOA1*
 - Existing functions as in other exits
 - New functions
 - RXML - Process client XML input
 - EXML - Process adapter output error
 - New fields in the EXPRM parameter list
 - EXPRXML_TAGNM
 - EXPRXML_ADPTNM
 - EXPRXML_RETCODE
 - EXPRXML_RSNCODE

IMS Connect

```

graph TD
    subgraph IMS_Connect [IMS Connect]
        HWSSOAP1[HWSSOAP1 Exit routine]
        ATM[Adapter Task Manager]
        XML_Adapter[XML Adapter For COBOL]
        HWSSOAP1 <--> ATM
        ATM <--> XML_Adapter
    end
  
```

58

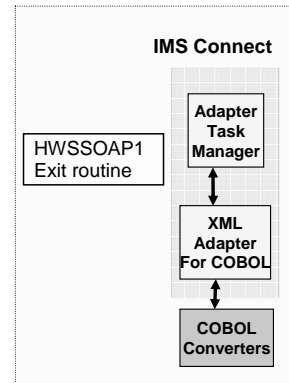
IMS Connect provides a new user message exit routine that specifically understands the XML interface. In addition to the existing message functions of READ, XMIT, INIT and TERM, the new exit routine, HWSSOAP1, includes new interfaces: RXML and EXML. On input from the IMS Connect Client Application, IMS Connect calls HWSSOAP1 with the function type set to 'RXML' which translates the IRM header and returns the ADAPTER name and the TAG name back to IMS Connect. Upon returning to IMS Connect, error analysis will be performed. If an error exists, function 'EXML' is invoked to process the error message. If no error exists, IMS Connect calls and passes control to the ADAPTER passing a set of parameters which determine: if this ADAPTER should process the message; if translation is required or not; and what XML processing is required. When the reply is ready to be sent back, IMS Connect receives the message and invokes the 'XMIT' function. Again the ADAPTER is called to prepare the reply in a tagged format.

New fields in the EXPRM parameter list include the following:

- EXPRXML_TAGNM Defines an output field and contains the TAG NAME to be used by the ADAPTER, the TAG name would represent one of the following: XML name for user purposes, COBOL Driver name for IMS Connect support of COBOL ADAPTER, user Map Name for user purposes
- EXPRXML_ADPTNM Defines an output field and contains the Adapter name to be used by IMS Connect to load the correct ADAPTER.
- EXPRXML_RETCODE - Return code returned by the User Message Exit back to IMS Connect.
- EXPRXML_RSNCODE - Reason code returned by the User Message Exit.

XML Adapter Support - Implementation ...

- XML converter routines
 - ◆ Cobol source code
 - Provide the information needed to perform conversion from tagged data to a byte stream
 - Unique to each message definition
 - Can be generated by WDz toolkit
 - ◆ Compiled and bound into file that is concatenated into IMS Connect STEPLIB



59

To perform correct conversion for each message, the Adapter requires information on how this is to be done. This information is called an XML Converter. XML Converters are Cobol programs generated from WebSphere Developer for zSeries (WDz) using the target IMS COBOL application's copybook. For each copybook, WDz generates three COBOL programs - an Inbound Converter (XML schema), an Outbound Converter (XML schema), and a Driver (Cobol code). These three programs are combined into one file and are referred to as an XML Converter. An XML Converter has to be created for each IMS COBOL transaction application that wants to support XML transaction messages. For details on generating XML Converters using WDz, see the WDz documentation.

On an inbound request to IMS, the XML Adapter calls the COBOL XML Converter Driver and passes it the inbound function code. The COBOL XML Converter Driver calls the Inbound Converter to parse the incoming XML message, convert the parsed message into the COBOL data structure byte streams, and then return it to the XML Adapter. On an outbound reply from the IMS application program, the XML Adapter calls the COBOL XML Converter Driver with an outbound function code, which then calls the Outbound Converter. The Outbound Converter converts the IMS output message into XML and passes it back to the XML Adapter which then sends it to IMS Connect to be sent back to the client.

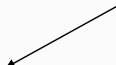
XML Adapter Support - Implementation ...

- IMS Connect JCL

```
                IMS Connect

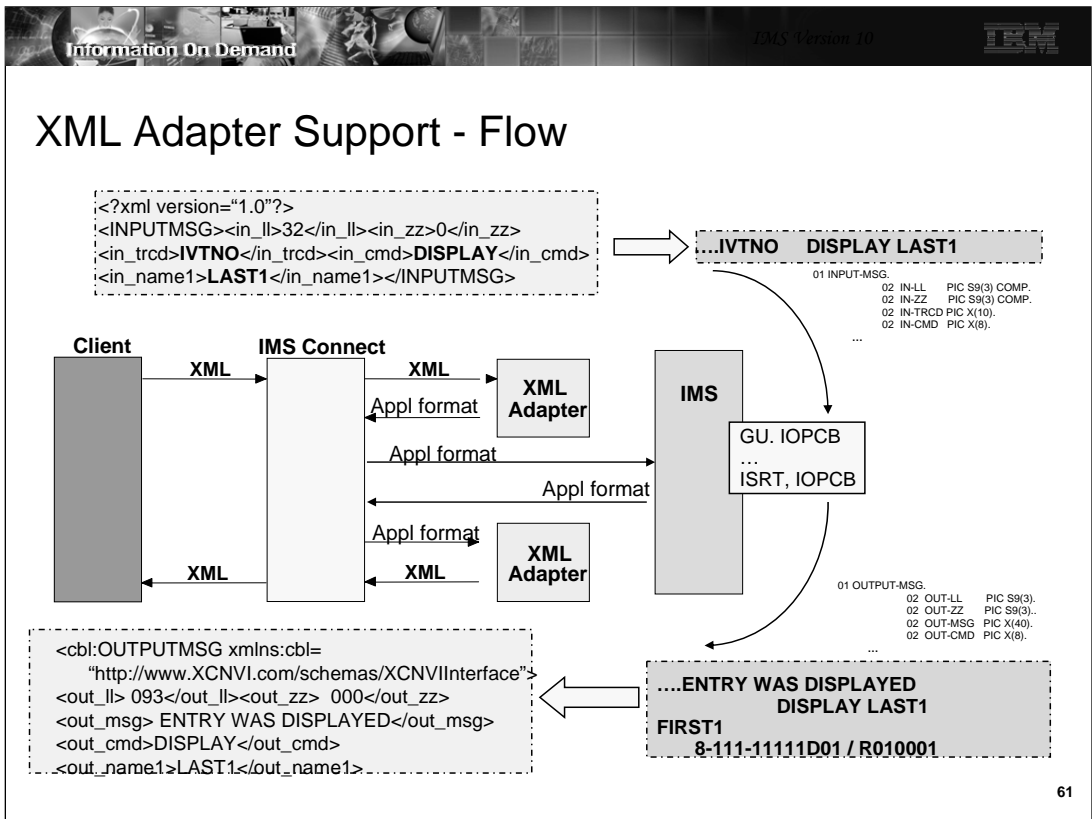
//HWS  PROC
//STEP1 EXEC PGM=HWSHWS00, REGION=5M,
// PARM='BPECFG=BPECFG00, HWSCFG=HWSCFG00'
//STEPLIB DD DISP=SHR,DSN=IMS10.SDFSRESL
//*        SSL SUPPORT DATASETS
//        DD DISP=SHR,DSN=CEE.SCEERUN
//        DD DISP=SHR,DSN=SYS1.CSSLIB
//        DD DISP=SHR,DSN=GSK.GSKLOAD
//*        COBOL XML CONVERTER DRIVERS
//        DD DISP=SHR, DSN=IMS.XML.DRIVERS
//PROCLIB DD DISP=SHR,DSN=IMS10.PROCLIB
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//HWSRCRD DD DISP=SHR,DSN=IMS10.HWSRCRD
```

APF authorized



60

The Cobol XML Converter routines must be added into an APF-authorized library that it concatenated into the STEPLIB list of the IMS Connect startup JCL.



This visual gives a high level view of the flow of messages. When an IMS Connect Client sends a transaction message in XML to IMS Connect, IMS Connect receives the input message and checks a flag in the IRM that indicates whether the input message is in XML or not. If it is XML, IMS Connect calls the XML Adapter. The XML Adapter converts the XML input message into the COBOL Application input data structure, sends it back to IMS Connect. IMS Connect processes the returned stream and forwards it on as an LLzz_trancode_data message to IMS so that the IMS Application can process the input message. The IMS COBOL Application returns an output message to IMS Connect. IMS Connect again calls the Adapter. The XML Adapter converts the IMS COBOL Application output message into an XML output message, sends it back to IMS Connect. IMS Connect processes the message from the XML Adapter and sends the XML output message to its Client.

The IMS COBOL Application is not aware of any message conversions. It receives messages in the same application specific format as it normally expects. The diagram below shows the message formats received by each component.

XML Adapter Support - Considerations ...

- Error messages are sent back to the remote client in XML tags
 - ◆ XML Converter Driver and XML Adapter errors
 - `<XMLAdapterError> </XMLAdapterError>`
 - If message can be converted to codepage
 - `<XMLAdapterError>HWS..... </XMLAdapterError>`
 - Several new HWS... error messages have been added
 - Otherwise, XML Adapter Status Message (XASM)
 - `Ilzz*XADPST* rc rs`
 - where rc= 4 byte return code, rs= 4 byte reason code
 - ◆ IMS errors
 - `<IMSDFSMMessage>DFS.....</IMSDFSMMessage>`

62

There are two types of runtime errors generated in the XML Adapter - the errors from WDz COBOL XML Converter Drivers, and the errors from the XML Adapter. Both types of error messages are returned from the XML Adapter in XML-tagged messages if the output codepage is available. IMS Connect does not distinguish between the two types of errors. If the output codepage is not available, the XML Adapter returns a XML Adapter Status Message (XASM). The XASM includes a return and reason code to indicate the type of error.

Besides these two types of errors, the XML Adapter also passes other error messages from/to IMS Connect, for example, the DFS error messages generated by IMS transactions. For any DFS messages, either errors or non-errors, the XML Adapter wraps the messages with XML tags and passes them back to the caller. The DFS message in XML format is defined as `<IMSDFSMMessage>DFS.....</IMSDFSMMessage>`.

XML Adapter Support - Considerations ...

- IMS Connect Restart
 - ◆ Required when new XML Converters are to be changed
 - Adding or deleting an XML Converter does not require a restart

- BPE UPDATE command to refresh the Adapter exit
 - F ICONN01, REF USRX NAME(XMLADAP)
 - Value:
 - Refresh request to bring in a new version of the ADAPTER plug-in user exit routine

- LE environment is required
 - ◆ XML Adapter brings up an LE environment during initialization

- Restriction
 - ◆ Single-segment messages only (multi-segment support is coming)

63

An IMS Connect restart is required to recognize any changes that have been made to the dataset containing the XML Converters. On the other hand, a refresh of the Adapter exit routine can be done while IMS Connect is active by using the BPE refresh capability.

Note that the COBOL XML Converter Driver requires an LE environment to run. The XML Adapter brings up an LE environment during its initialization processing.

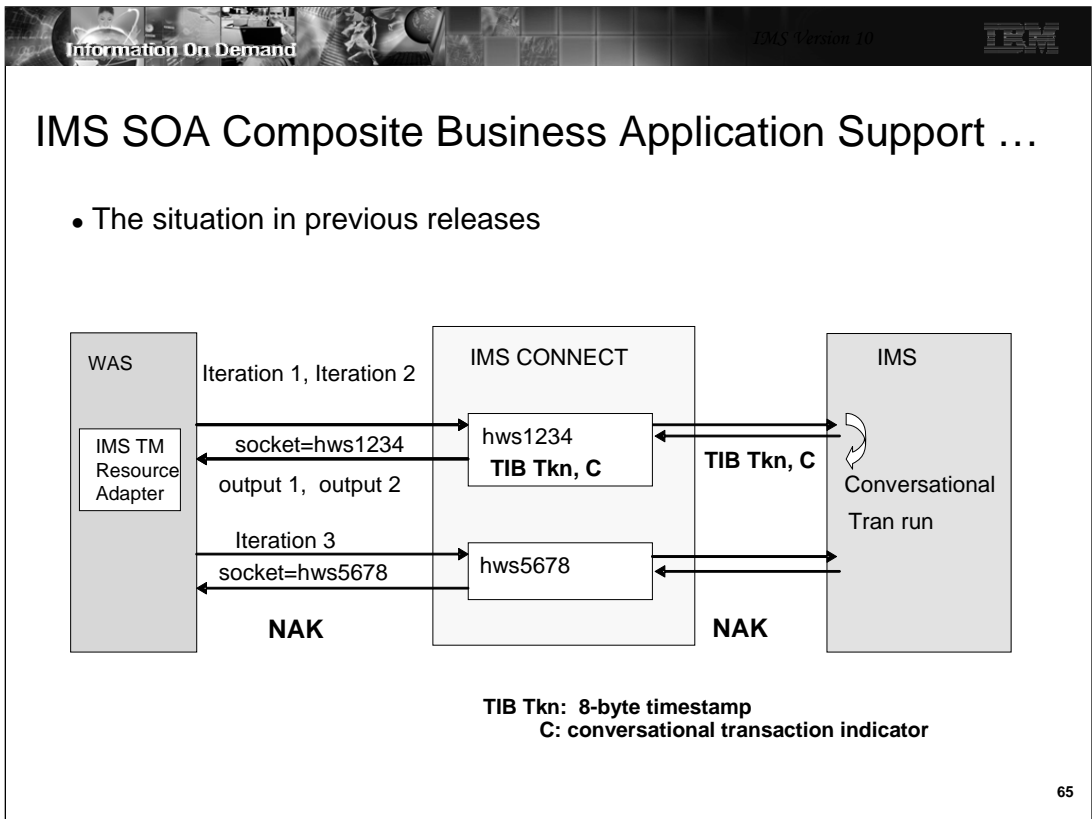
IMS SOA Composite Business Application Support

- New capability for IMS TM Resource Adapter clients that invoke IMS conversational transactions
 - ◆ Allows iterations of the conversation to span shareable persistent sockets
 - Application Server on which IMS TM Resource Adapter runs may use different sockets for each iteration
 - IMS TM Resource Adapter client does not have control of the socket that is used
- Supports the WebSphere Process Server
 - ◆ Process Choreography function

64

IMS TM Resource Adapter (formerly IC4J) applications running in a WebSphere Application Server (WAS) can take advantage of either dedicated or shareable persistent sockets. A dedicated persistent socket remains dedicated to a particular user-specified clientid for commit-then-send CM0 interactions until released by a client application's request. Shareable persistent sockets, on the other hand, can be shared (serially reused) by multiple applications in the WAS server running either send-then-commit CM1 or commit-then-send CM0 interactions. For the latter type of socket, IC4J generates a clientid and does not allow user-specified clientid's.

Although the IMS TM Resource Adapter has supported interactions with IMS conversational transactions, IMS V10 enhances this capability by allowing the iterations between a conversation to span multiple shareable sockets.





This visual discusses the flow and a potential issue in the existing implementation.

The first request from the IMS TM Resource Adapter application invokes the first iteration of an IMS conversational transaction flow. The IMS TM Resource Adapter either uses an existing shareable persistent socket connection or generates a new one. For this example, the generated clientid is hws1234. At this point, neither the remote client nor IMS Connect are aware of whether the input is a conversational transaction. When the first reply message for this conversational iteration is created, IMS includes a conversational indicator (C) and a Transaction Instance Block token (TIB Tkn) in the message prefix that is sent back to IMS Connect. IMS Connect then saves these two pieces of conversational information in an internal control block associated with the clientid, hws1234. IMS Connect also sends the C indicator and the TIB Tkn with the output of the first iteration to the IMS TM Resource Adapter which returns them to the Java application.

When the second iteration of the same conversation begins, the message is routed to one of the shareable sockets in the pool. If the second iteration comes in through the same socket hws1234, IMS Connect uses the saved information in its internal control block to associate this conversation iteration with the previous one.

If the third iteration is routed to a different socket in the WAS pool, for example, HWS5678, IMS Connect cannot associate the request with the intended internal control block (hws1234) and the conversation information saved in it. IMS Connect, therefore, creates a different internal control block for hws5678 which has no knowledge of the existing conversation. When IMS receives the the input, it rejects the request with a NAK.

IMS SOA Composite Business Application Support ...

- New conversation option for IMS TM Resource Adapter requests
 - ◆ IMS conversational transaction process choreography request
 - Set in the first input message

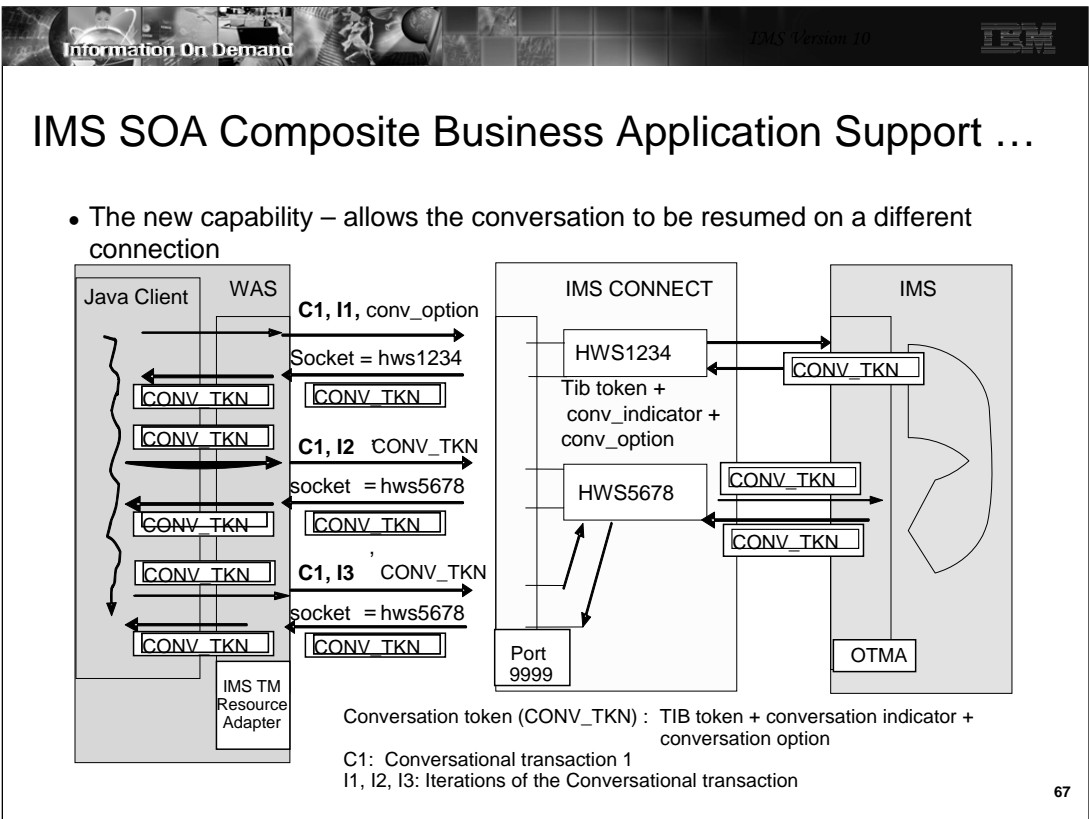
HWSOMUSR: User Data Header				
Field	Length	Hex Offset	FieldValue	Description and Settings
OMUSR_FLAG1	1	3C	OMUSR_CONV_OPTION X'80'	Use IMS Conv txn with Process choreography

- Requests indicators to be passed back to the IMS TM Resource Adapter client
 - TIB token, IMS conversational transaction indicator, new conversation option
 - Client has the responsibility of passing all the information back in to continue the IMS conversation
 - IMS Connect does not keep track of conversation information

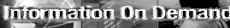


66

IMS Connect defines a new conversational option which is available to IMS TM Resource Adapter applications. Setting the OMUSR_CONV_OPTION to true (X'80') invokes the function. If the function is not enabled, IMS Connect uses the existing conversational logic. Note that setting the OMSUSR_CONV_OPTION to false (X'00') does not imply a non-conversational transaction.

IMS Connect creates the conversation token (the TIB token, Conversation indicator and conversation option flag), cleans up the internal control blocks, and passes the conversation token to the IMS TM Resource Adapter. The Adapter passes the information to the application which has to maintain the information to pass back on the next iteration of the conversation. This architecture allows the conversation to be continued by IMS Connect and passed to IMS on any of the sockets that are available to the remote client.



This visual illustrates the new capability. When receiving a transaction reply from OTMA, IMS Connect checks to see whether the reply message contains the conversational transaction indicator, the Transaction Instance Block (TIB) token, and the new conversational option flag. If so, IMS Connect passes the conversation token back to the IMS TM Resource Adapter and cleans up all the internal knowledge about this conversational iteration. The Adapter then returns the conversation token back to the Java client. The Java client maintains responsibility for sending the conversation token back in the subsequent iterations of a conversation or in the SYNC_END_CONVERSATION interaction to end a conversation. The conversation token is composed of the 8-byte timestamp of the TIB token, the conversational transaction indicator, and the new conversational option indicator. The conversation token is passed back and forth between the Java client, the IMS TM Resource Adapter, IMS Connect, and OTMA. IMS Connect uses the conversation token to track the conversation within a single iteration.

IMS SOA Composite Business Application Support ...

- **Error conditions**

HWSP1510E MESSAGE CONTAINS INVALID CONVERSATION INFORMATION
C=*clientid*, M=*mc*

where C is the client name and M identifies the module issuing the message

 - ◆ Issued when IMS Connect detects that the conversational transaction information in the input message conflicts with the conversation being managed
 - Possible causes:
 - Conversational option flag is reset from one iteration to the next
- **Command Output**
 - ◆ VIEWHWS and QUERY MEMBER TYPE(IMSCON)
 - When the conversational option is specified
 - CONV status displayed only if command is entered during an iteration
 - Conversational status is not kept across iterations and the CONV status is not displayed

68

If the new conversational option flag and other conversational information are set in an input message, IMS Connect assumes that this is a subsequent iteration of a conversational transaction and uses the conversational information to resume the conversation. If IMS Connect determines that the conversational transaction information from the input message does not conflict with any conversation it manages, IMS Connect continues processing the input. Otherwise, IMS Connect issues the HWSP1510E error message. This scenario could happen when the conversational option flag is flip-flopped from FALSE in one conversational iteration to TRUE in a subsequent one.

The IMS Connect commands VIEWHWS and QUERY MEMBER TYPE(IMSCON) display the CONV status if a client is running an IMS conversational transaction. When the new conversational option is specified, IMS Connect no longer keeps track of the conversational status of the client across iterations of the conversational transaction. The output of the commands, therefore, will not be able to display the CONV status. Only when the display command is entered during the instance that a conversational iteration is in progress will the CONV status be displayed.

IMS Connect Enhancements Benefits

- Better control and management of IMS Connect environment
 - ◆ HWSCFGxx parameters for ACEE aging value, CM1 Timeout value, and Message Flood Control

- Improved security capability
 - ◆ Ability to change passwords and support mixed case

- Integrated XML support
 - ◆ XML Adapter opens the IMS environment to XML applications

- IMS SOA Composite Business Application support
 - ◆ Expands conversational processing to environments with connection pooling, sysplex distributor, and cloned application servers