E15

OTMA Shared Queues for IMS Connect, MQSeries, WebSphere

Jack Yuan

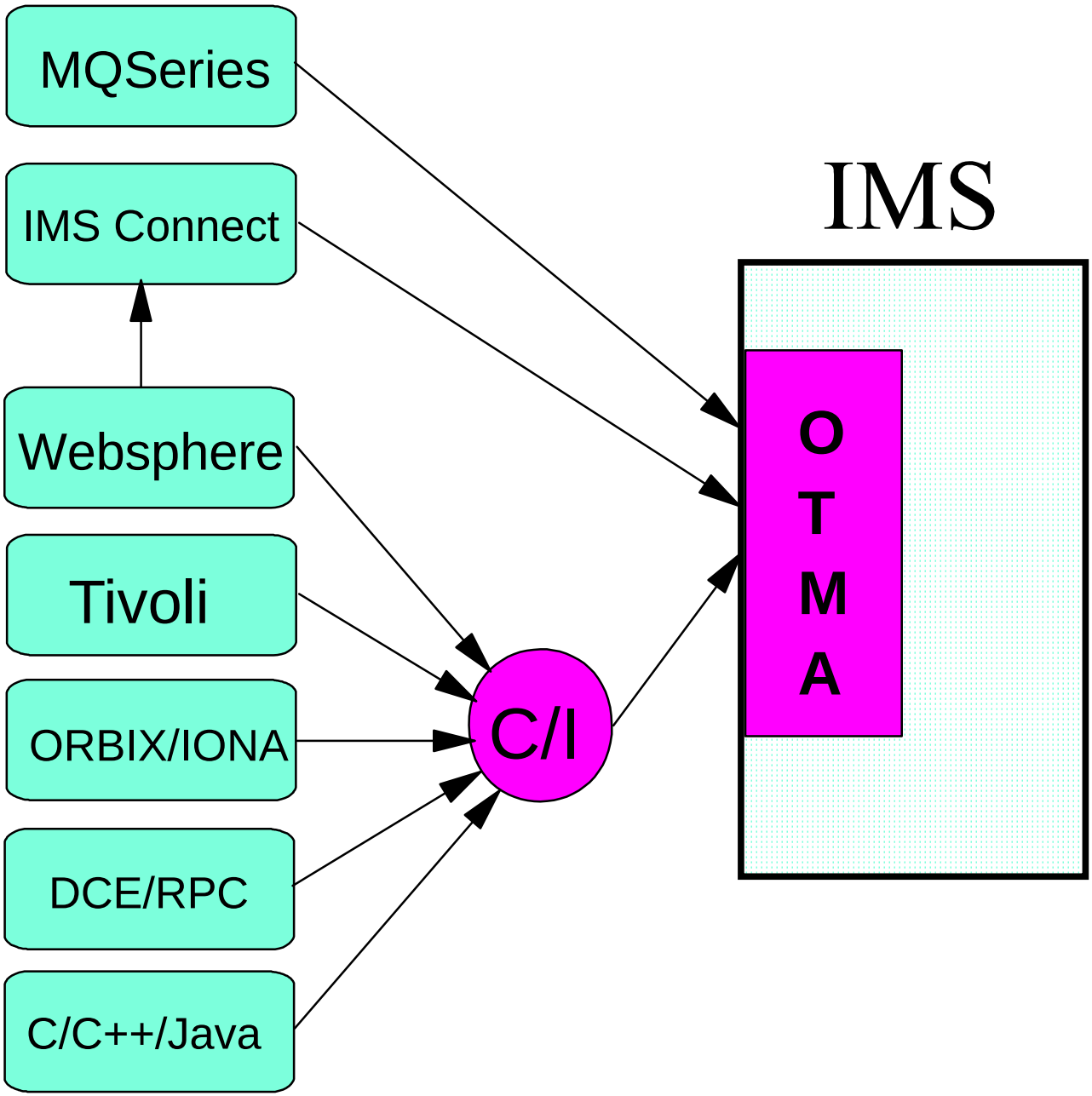
St. Louis, MO **Sept. 30 - Oct. 3, 2002**

RETURN TO INDEX

Agenda

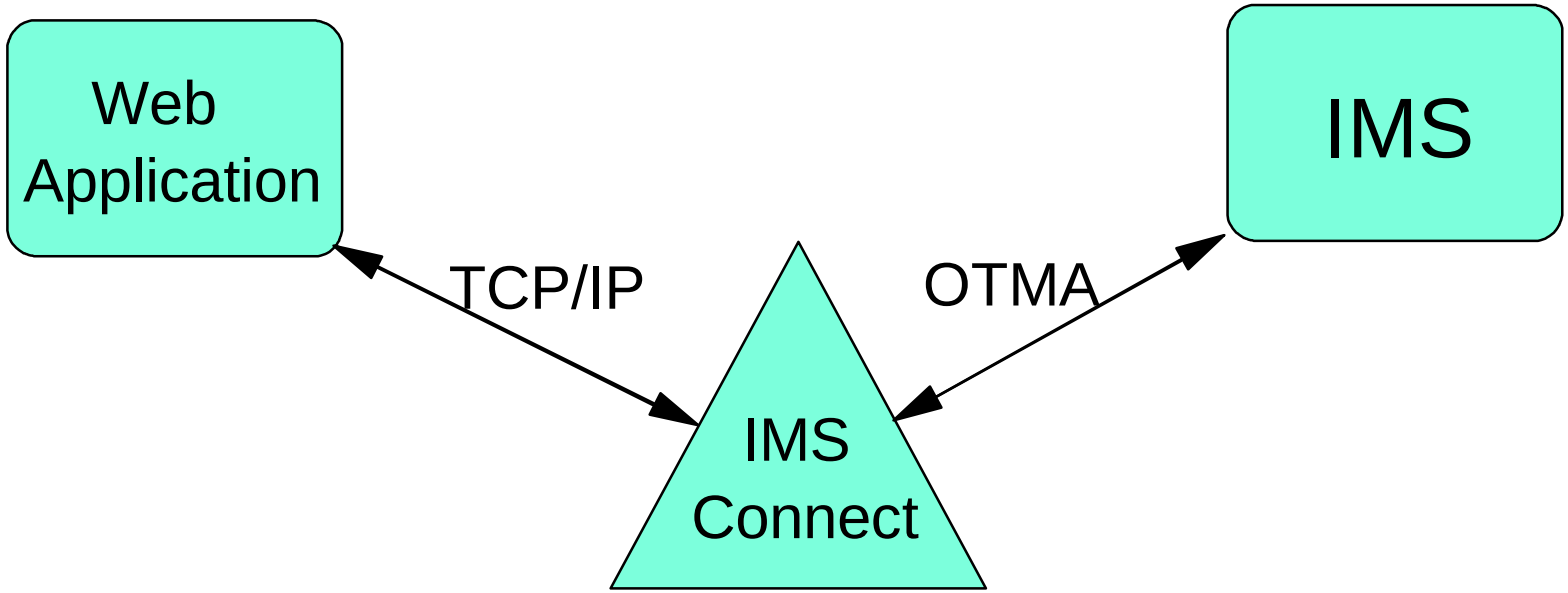
- **IMS OTMA Overview**
  - How IMS Connect, MQSeries and Websphere connect to IMS OTMA
- **IMS Sysplex implementation for OTMA**
  - How OTMA supports Sysplex in IMS V6, V7 and V8
- Summary

Big Picture with IMS OTMA

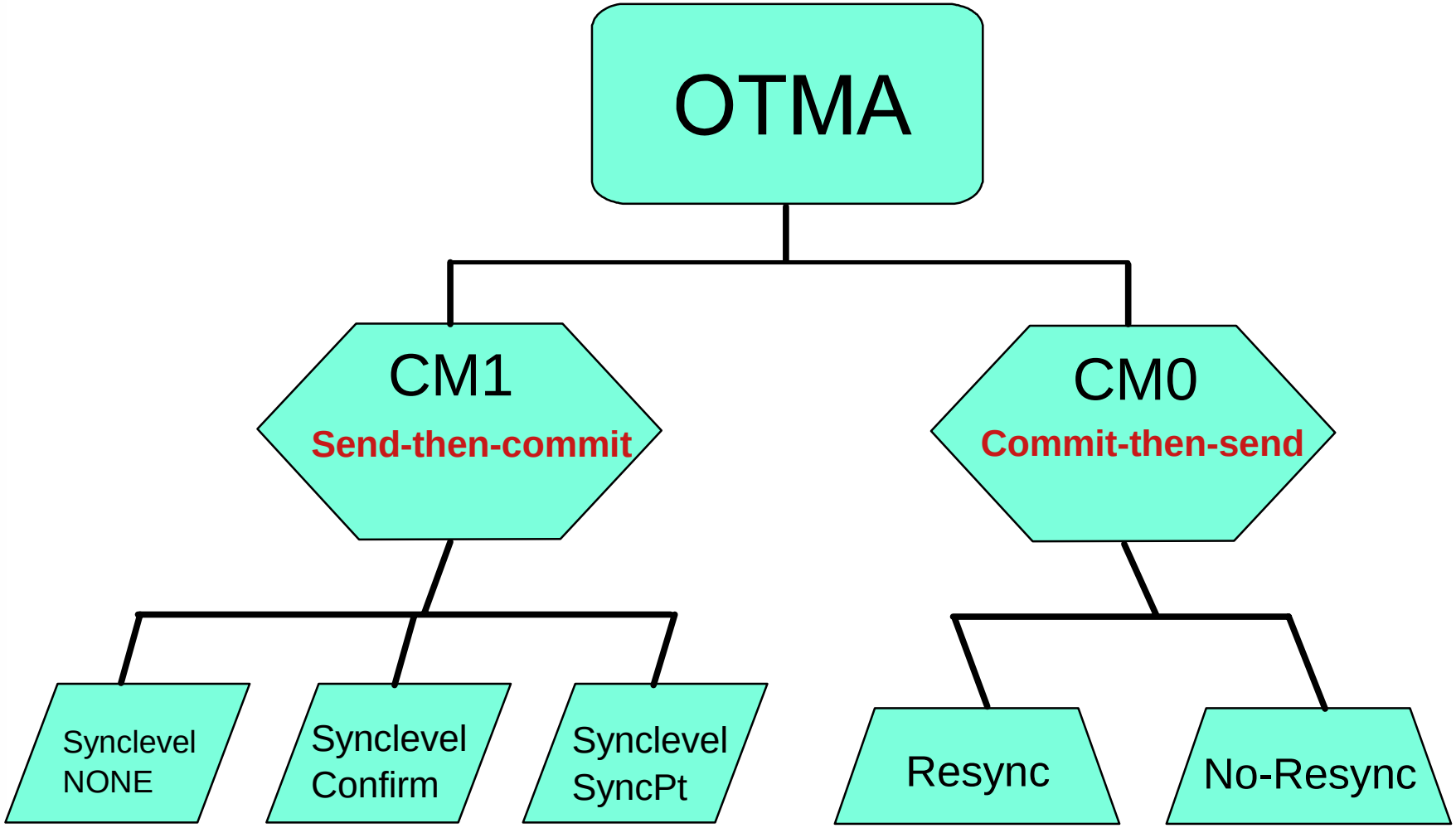


OTMA

- The Open Transaction Manager Access (OTMA) provides a high performance connectionless protocol for IMS to communicate efficiently with MVS applications without using the traditional SNA protocol.

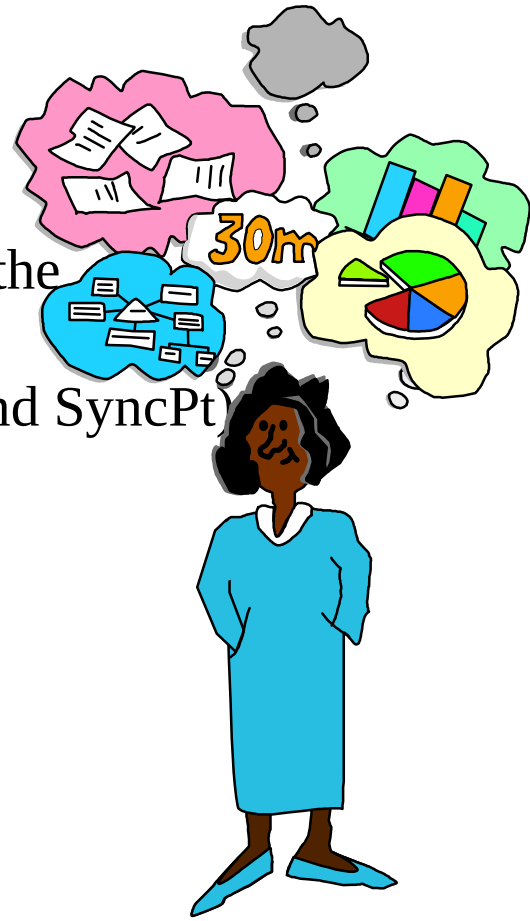


Modes for OTMA



CM0 or CM1

- Commit Mode 0, CM0, or Commit-then-send
  - IMS processes the transaction and commits the data before sending the output to the OTMA client.
  - Provide "Resynch" feature to prevent data from being reprocessed. (use the special SYN tpipe)
- Commit Mode 1, CM1, or Send-then-commit
  - IMS processes the transaction and sends the output to the OTMA client before committing the data.
  - Support 3 levels of synchronization (None, Confirm and SyncPt)

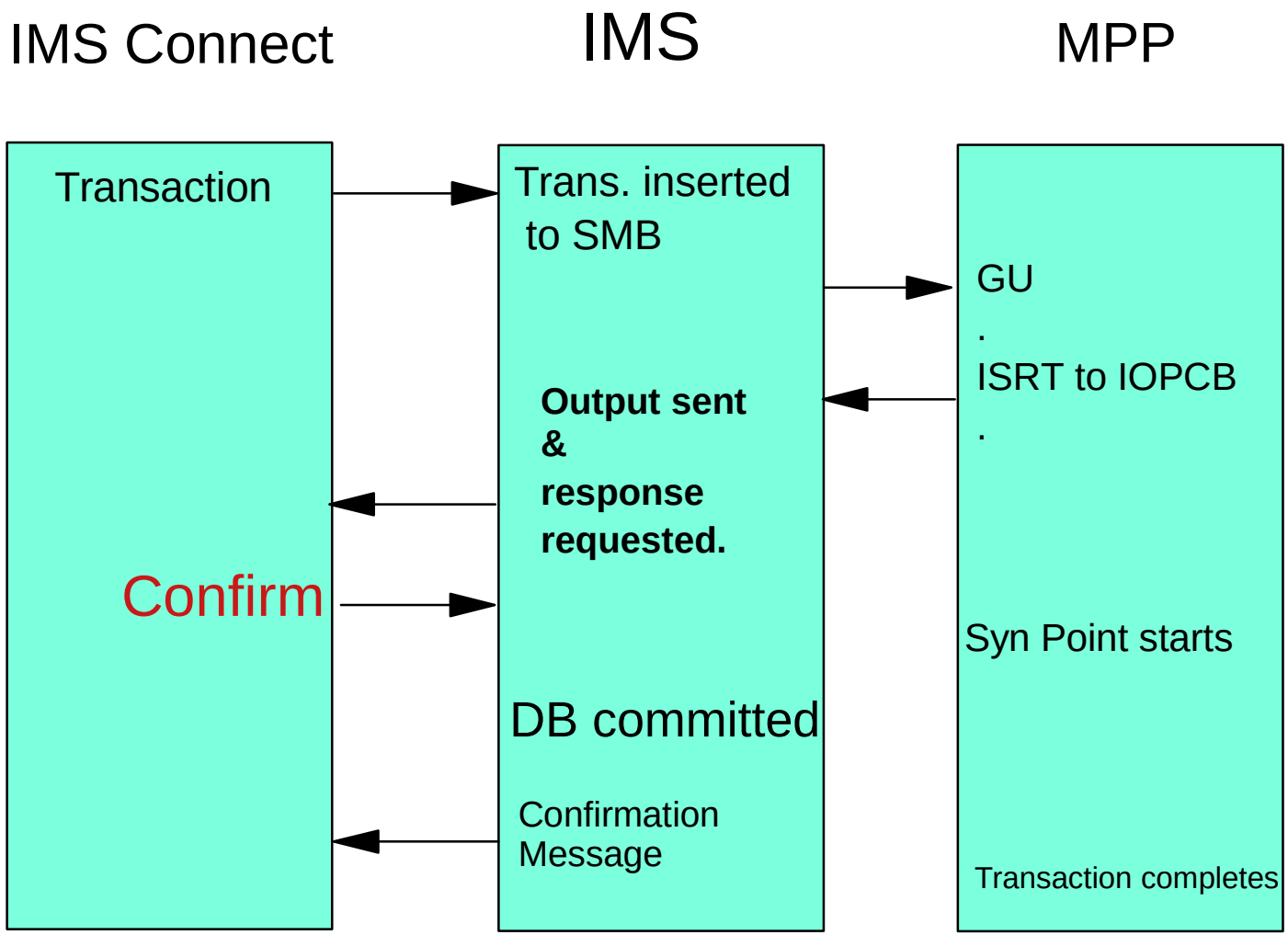


Levels of OTMA Synchronization

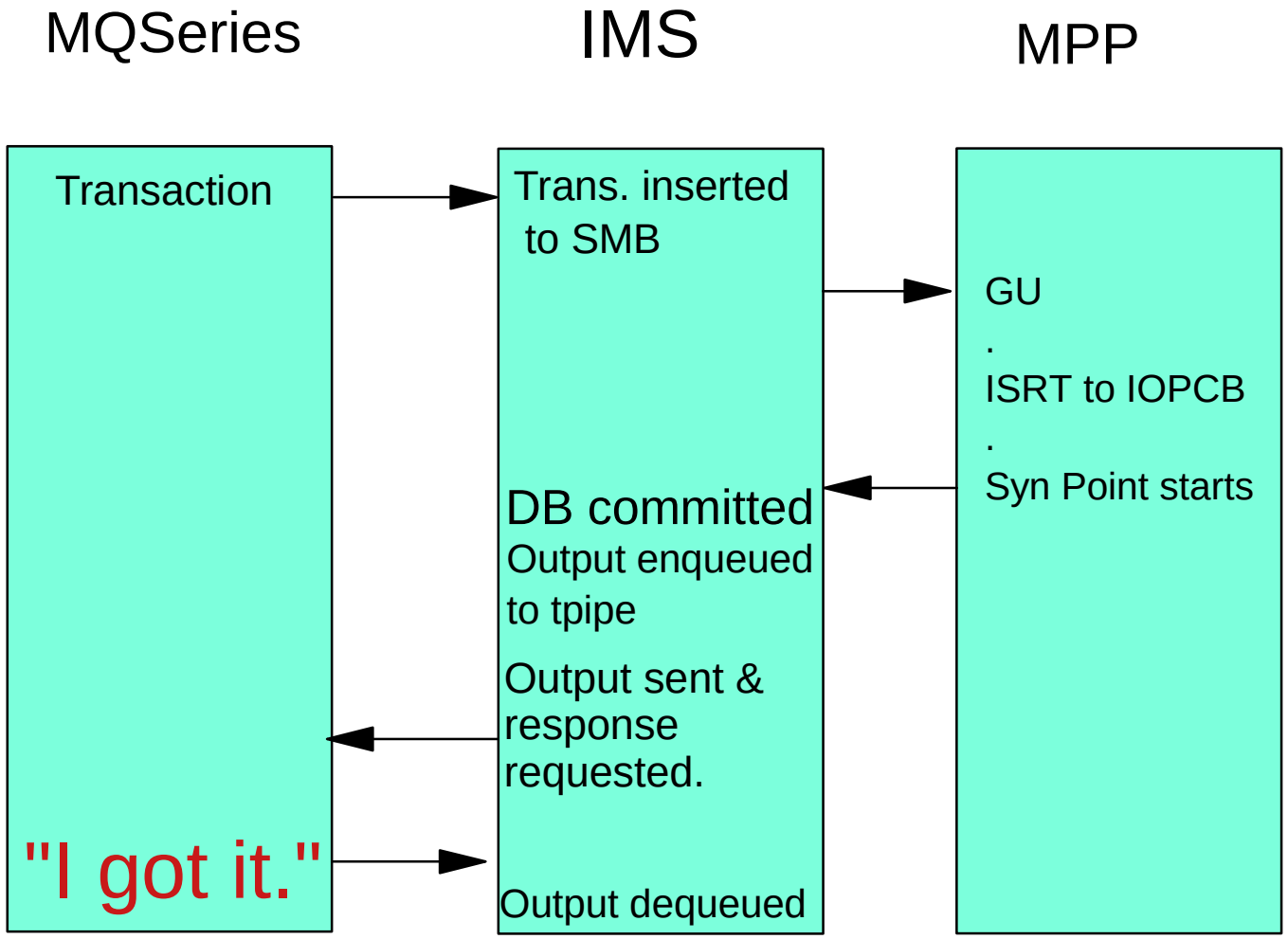
- NONE
  - Synchronization or confirmation not required
- CONFIRM
  - Confirmation is a part of the conversation
- SYNCPT
  - Coordinated Two-Phase commit flows using RRS

Send-then-commit message

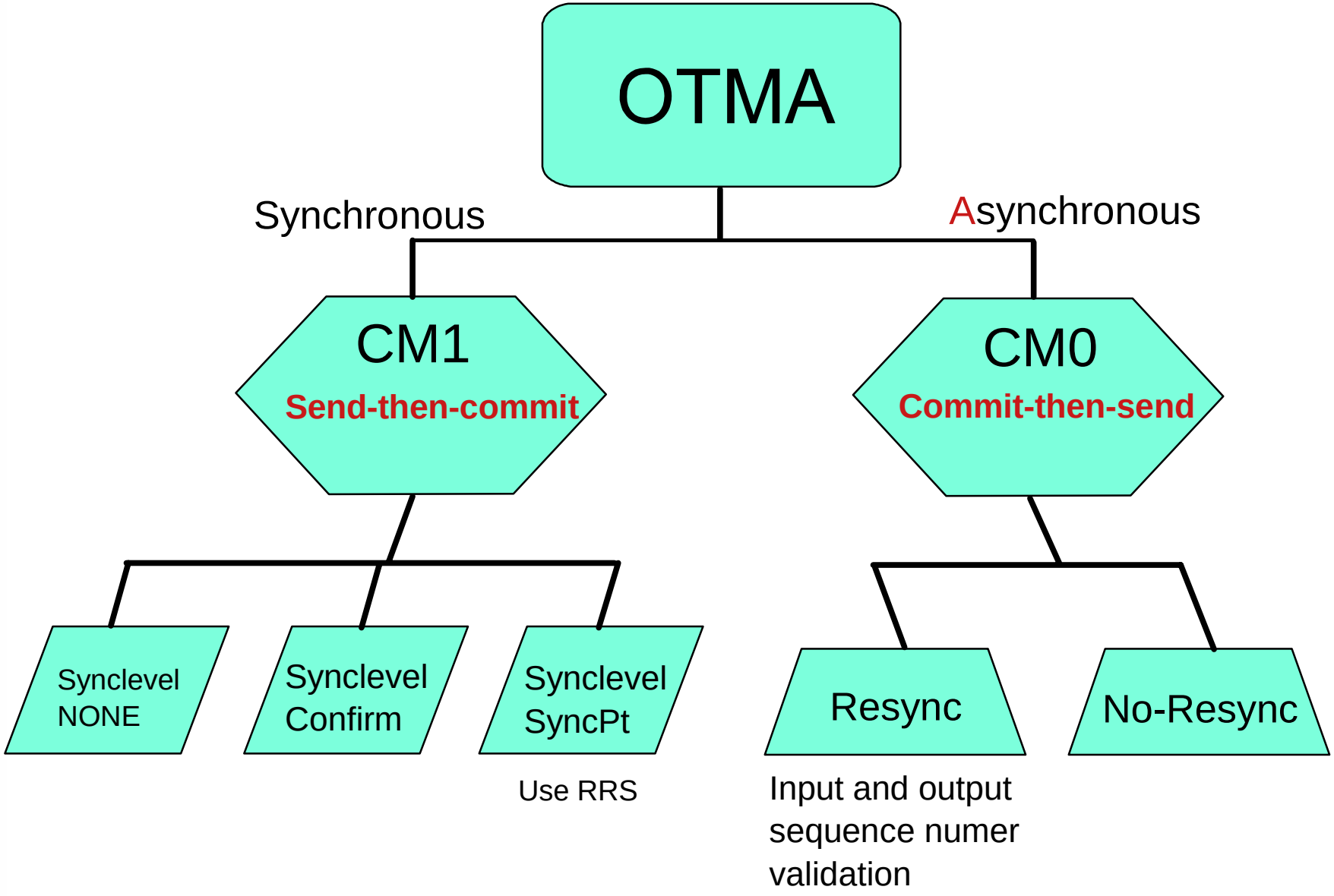
(synch level = CONFIRM)



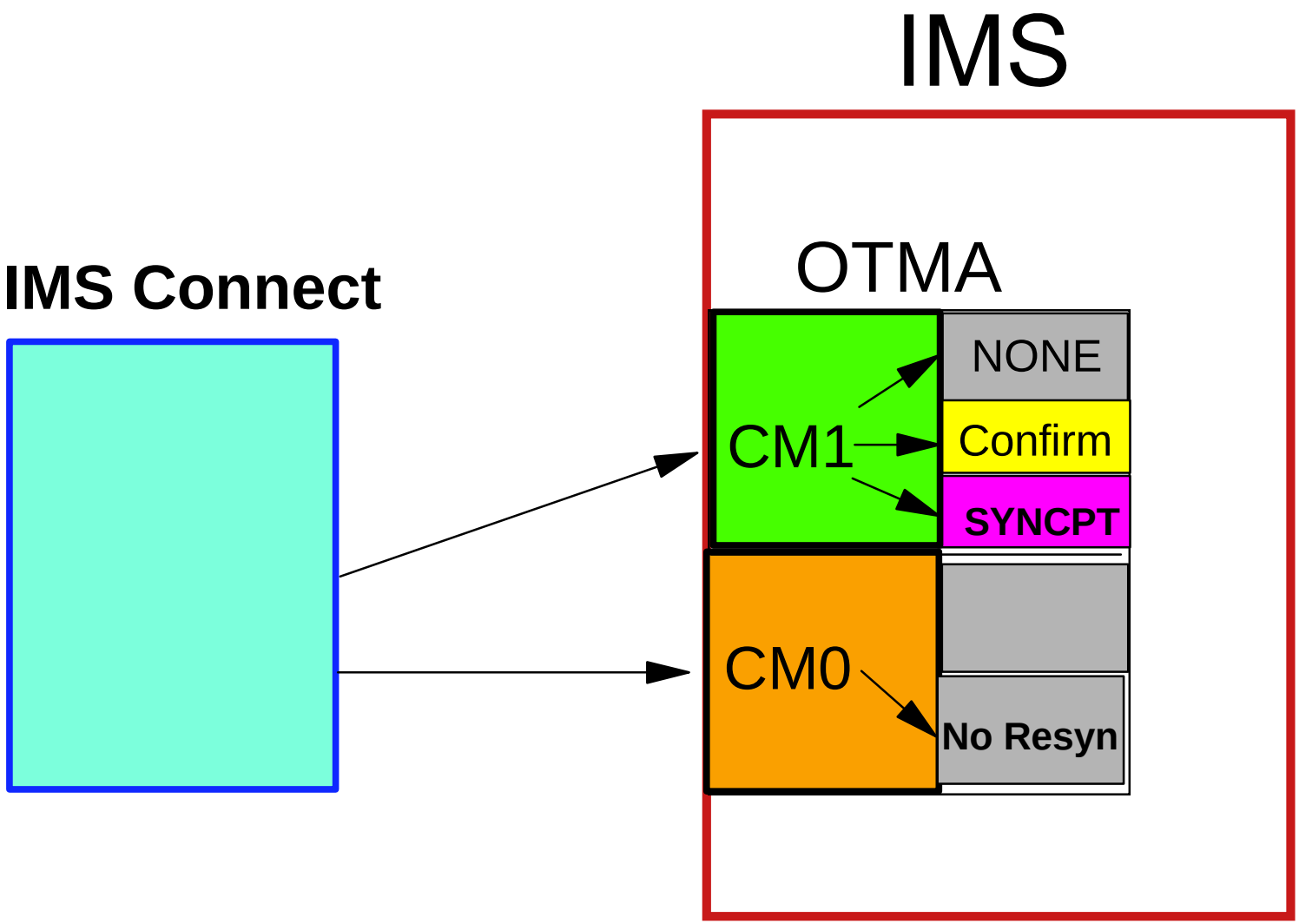
Commit-then-send message



Modes for OTMA



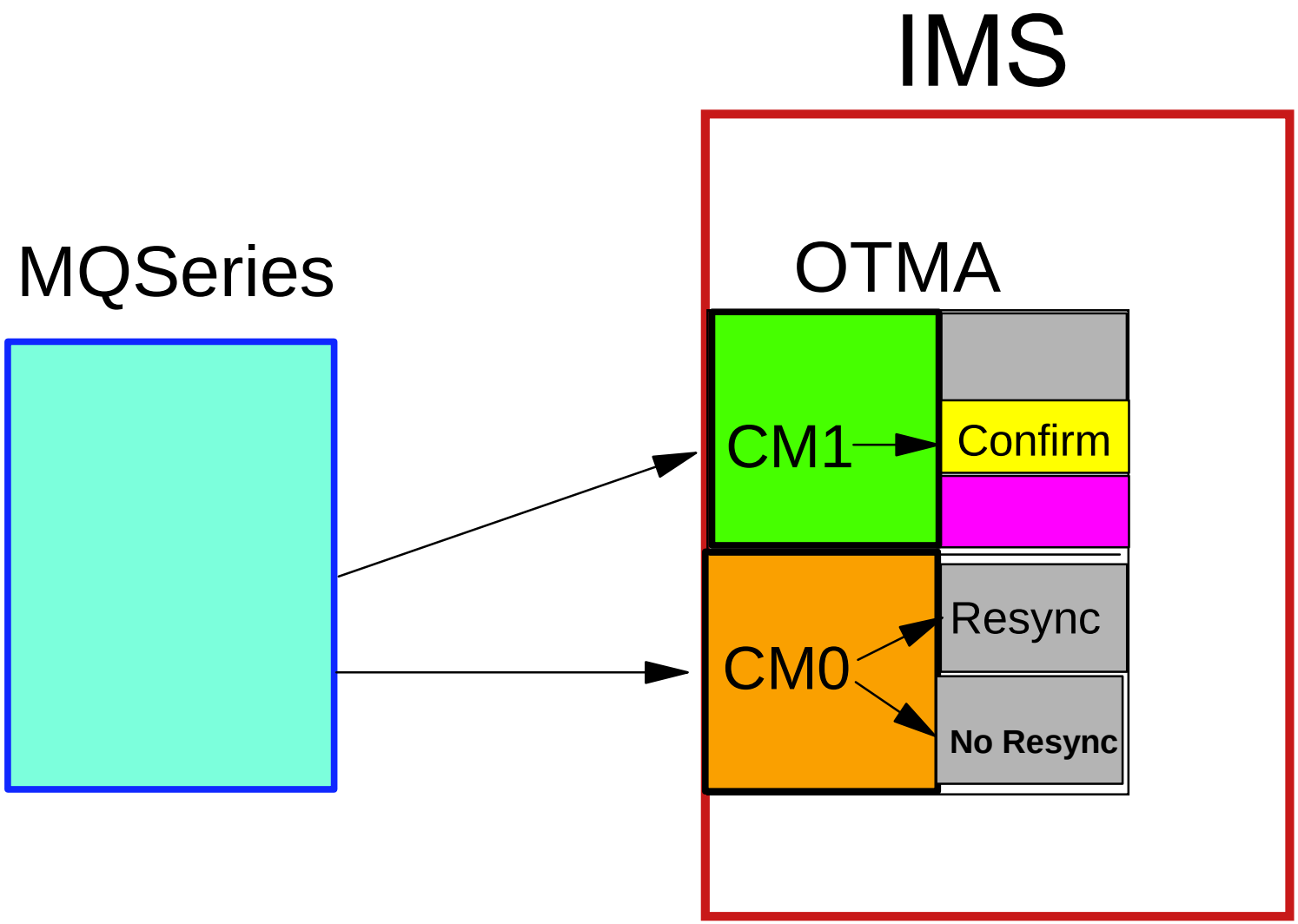
How IMS Connect uses OTMA



CM1 : Send then Commit

CM0 : Commit then send

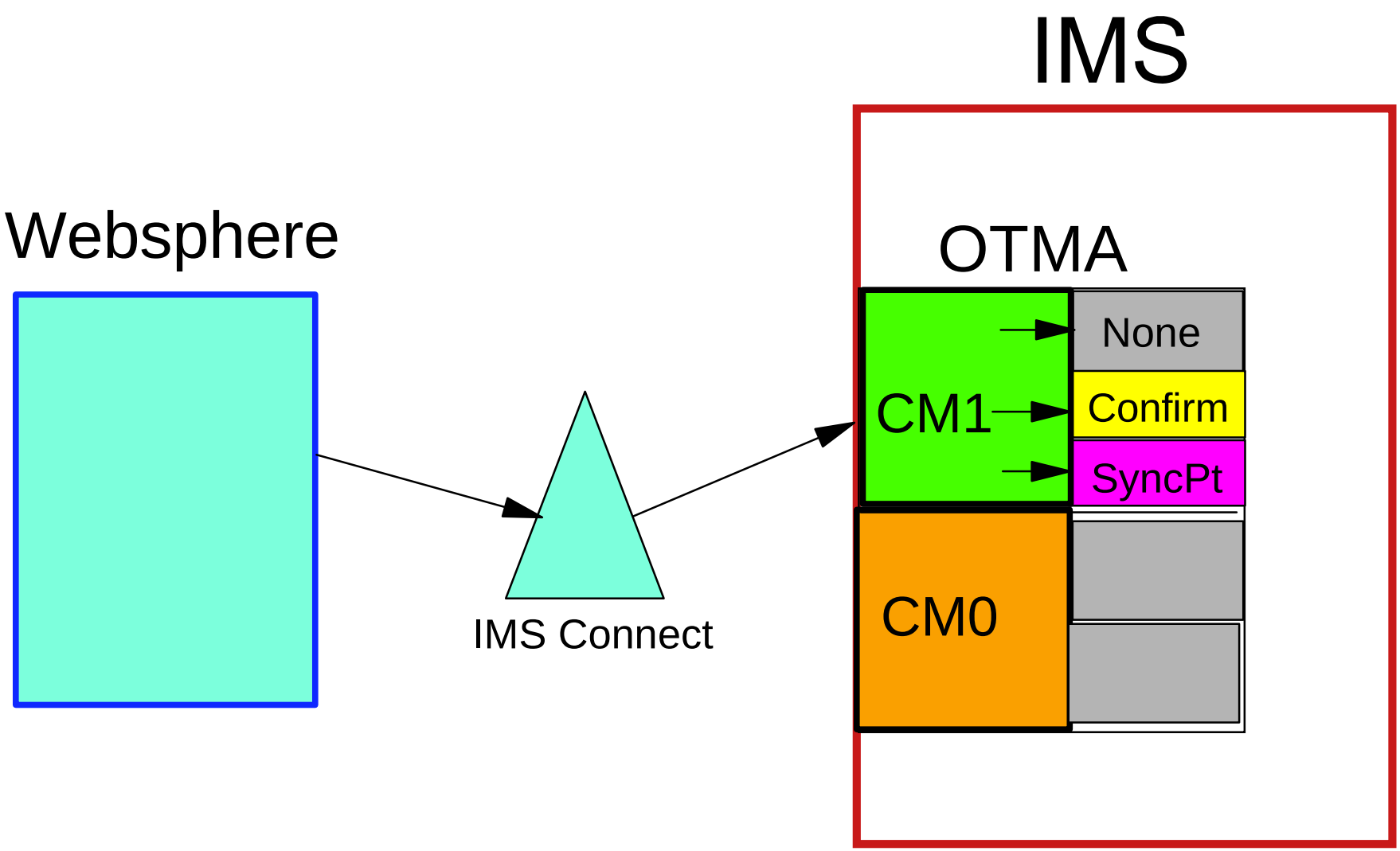
How MQSeries uses OTMA



CM1 : Send then Commit

CM0 : Commit then send

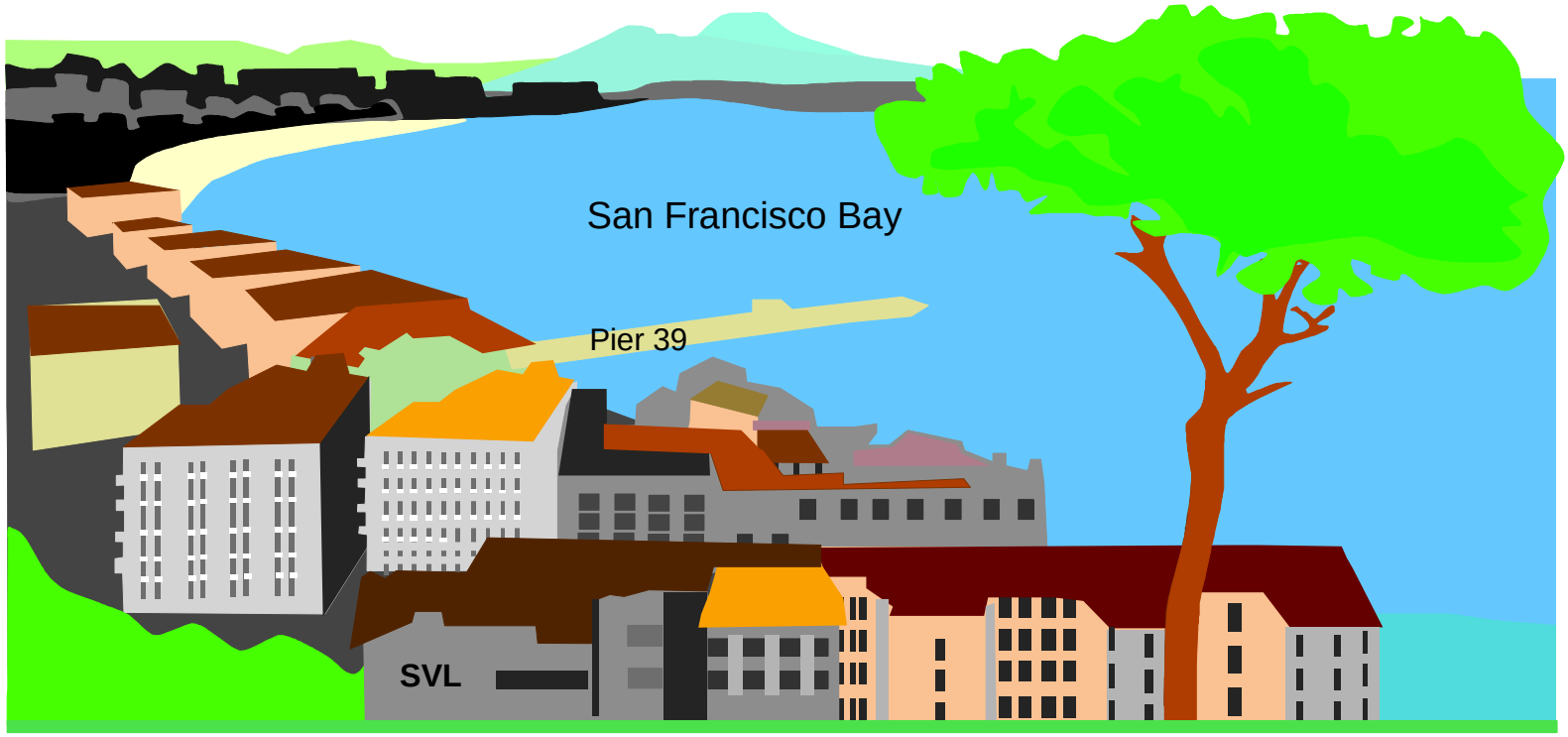
How Websphere uses OTMA



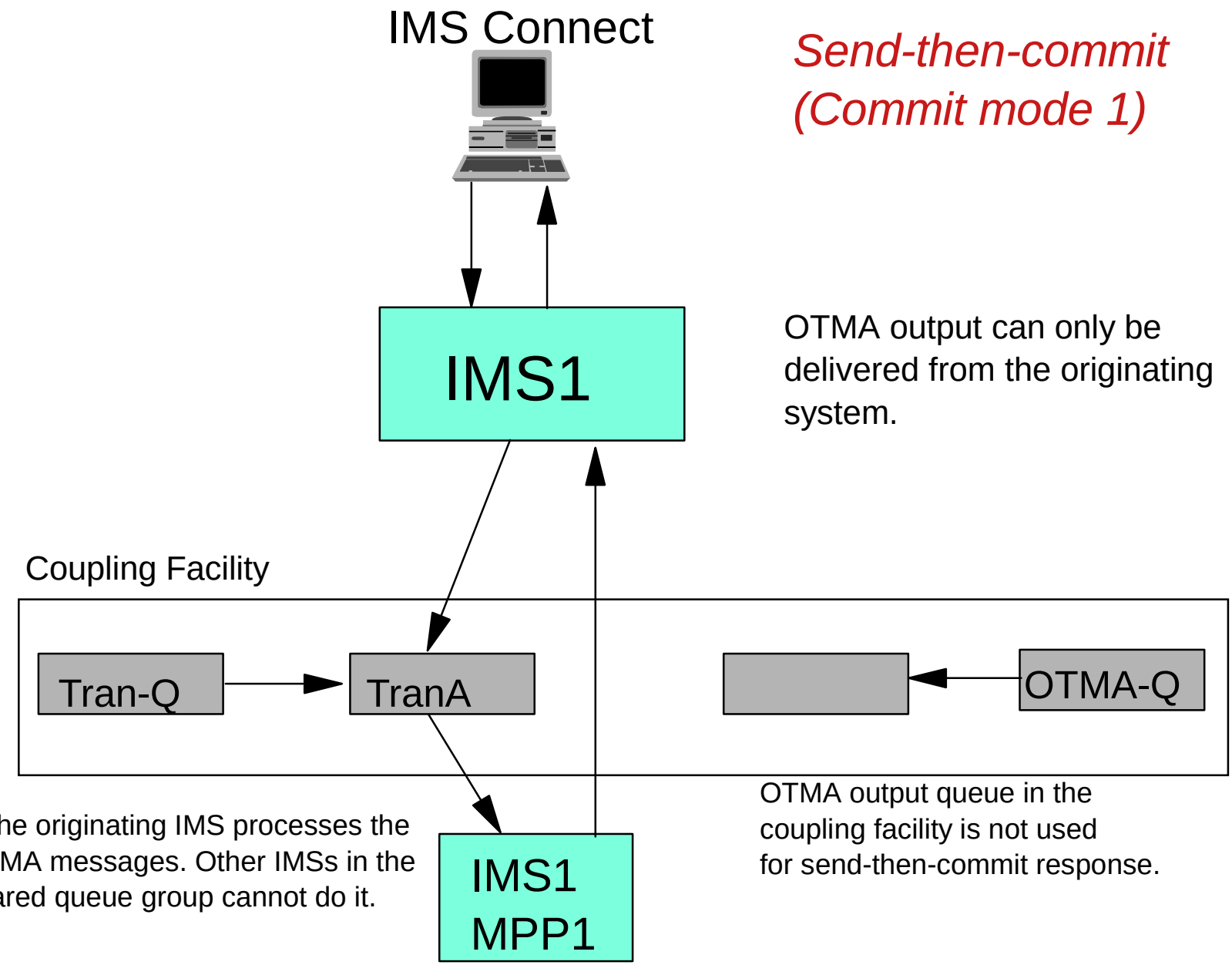
CM1 : Send then Commit

CM0 : Commit then send

OTMA Sysplex and Considerations



OTMA messages in IMS V6 Shared Queue Environment



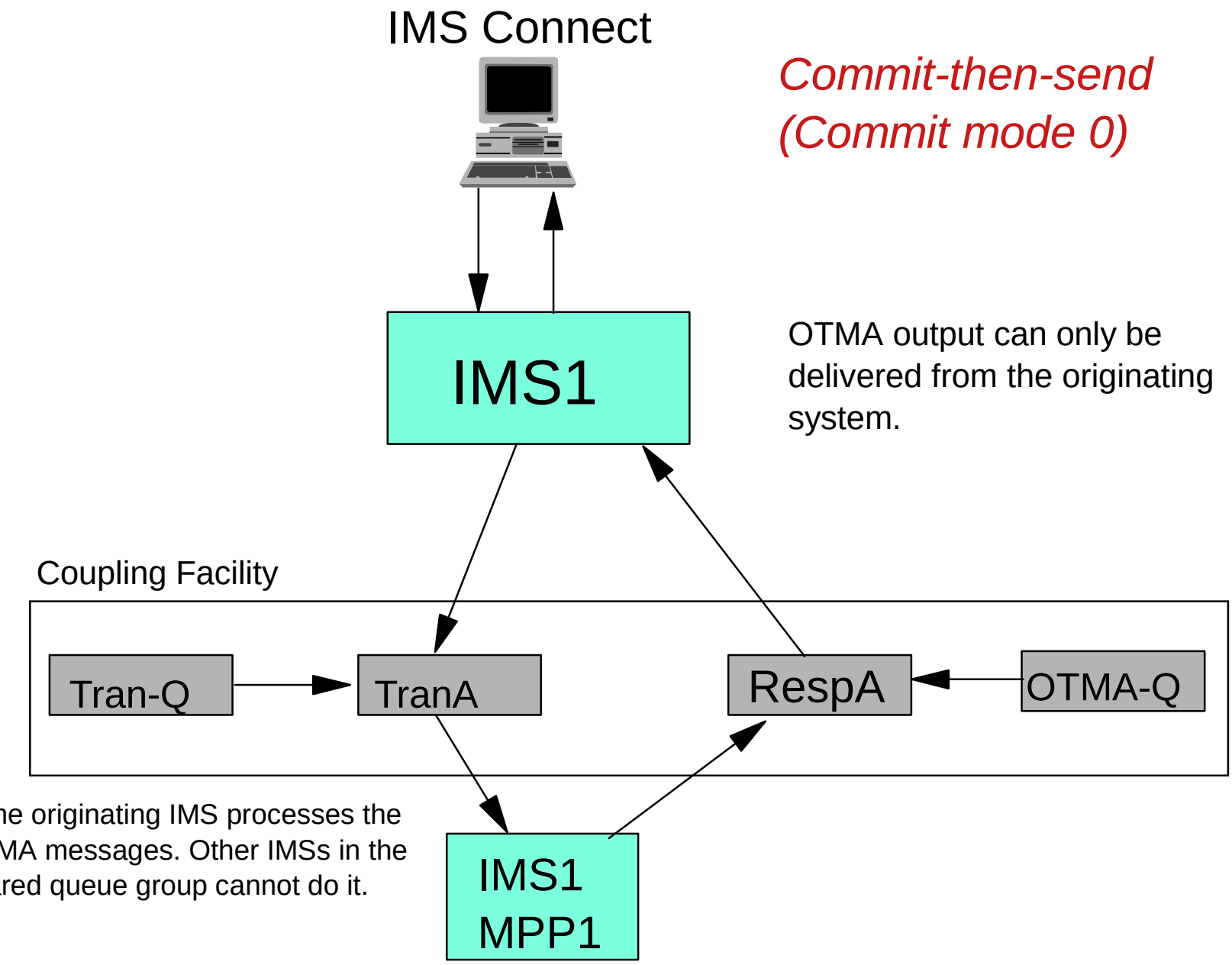
Send-then-commit
(Commit mode 1)

OTMA output can only be delivered from the originating system.

* The originating IMS processes the OTMA messages. Other IMSs in the shared queue group cannot do it.

OTMA output queue in the coupling facility is not used for send-then-commit response.

OTMA messages in IMS V6 Shared Queue Environment



Commit-then-send
(Commit mode 0)

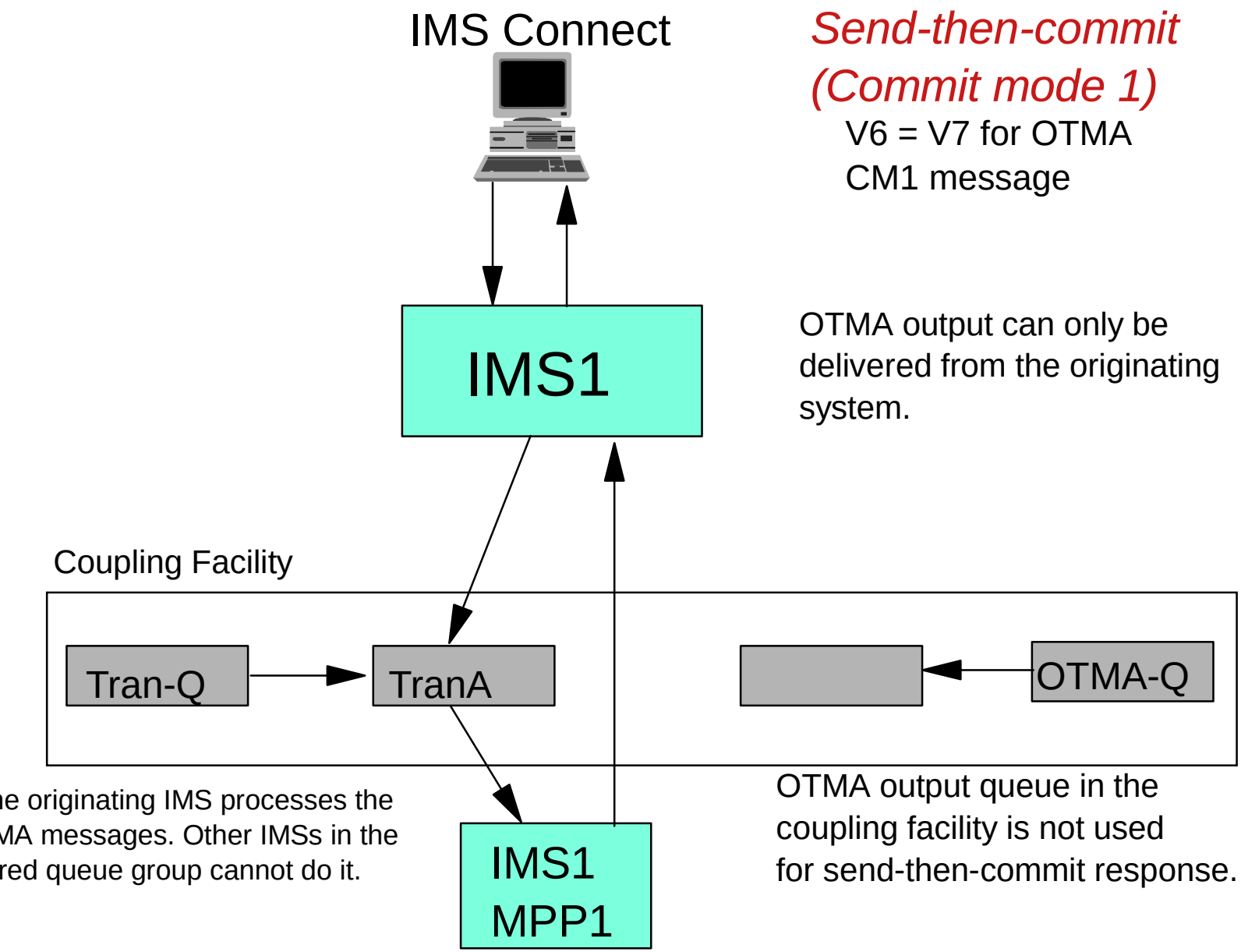
OTMA output can only be delivered from the originating system.

* The originating IMS processes the OTMA messages. Other IMSs in the shared queue group cannot do it.

OTMA messages in IMS V6 Shared Queue Environment

- **Non-MSC transactions entered from OTMA client**
  - Transaction must run on the IMS which receives the OTMA messages.
  - Program switches also run on the same IMS.
  - Output can only be delivered from the same IMS, the originating IMS.
- **MSC transactions entered from OTMA client**
  - MSC link may be defined on any IMS in the shared queue group.
  - Output can only be delivered from the originating IMS.

OTMA messages in IMS V7 Shared Queue Environment



Send-then-commit (Commit mode 1)

V6 = V7 for OTMA

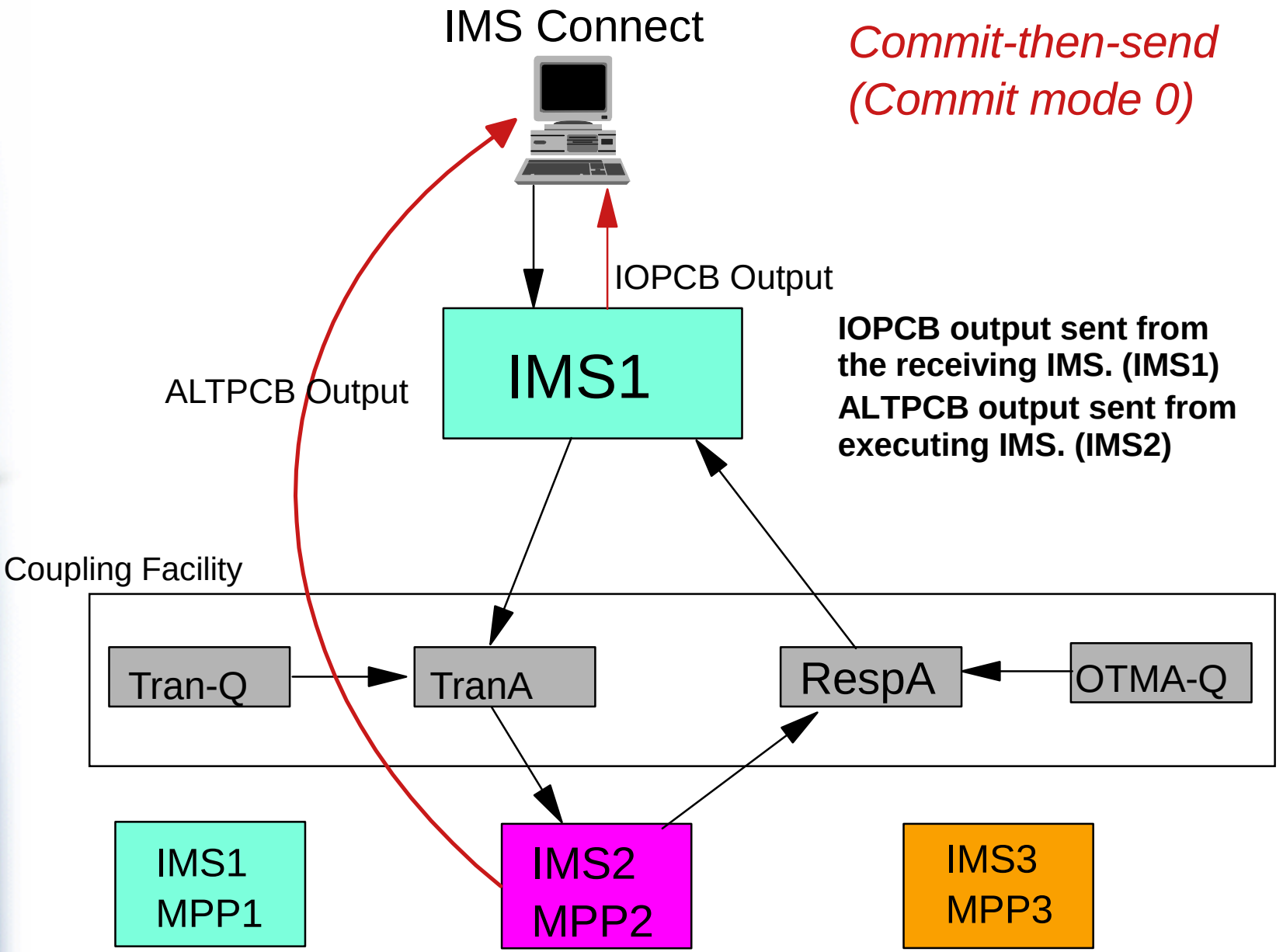
CM1 message

OTMA output can only be delivered from the originating system.

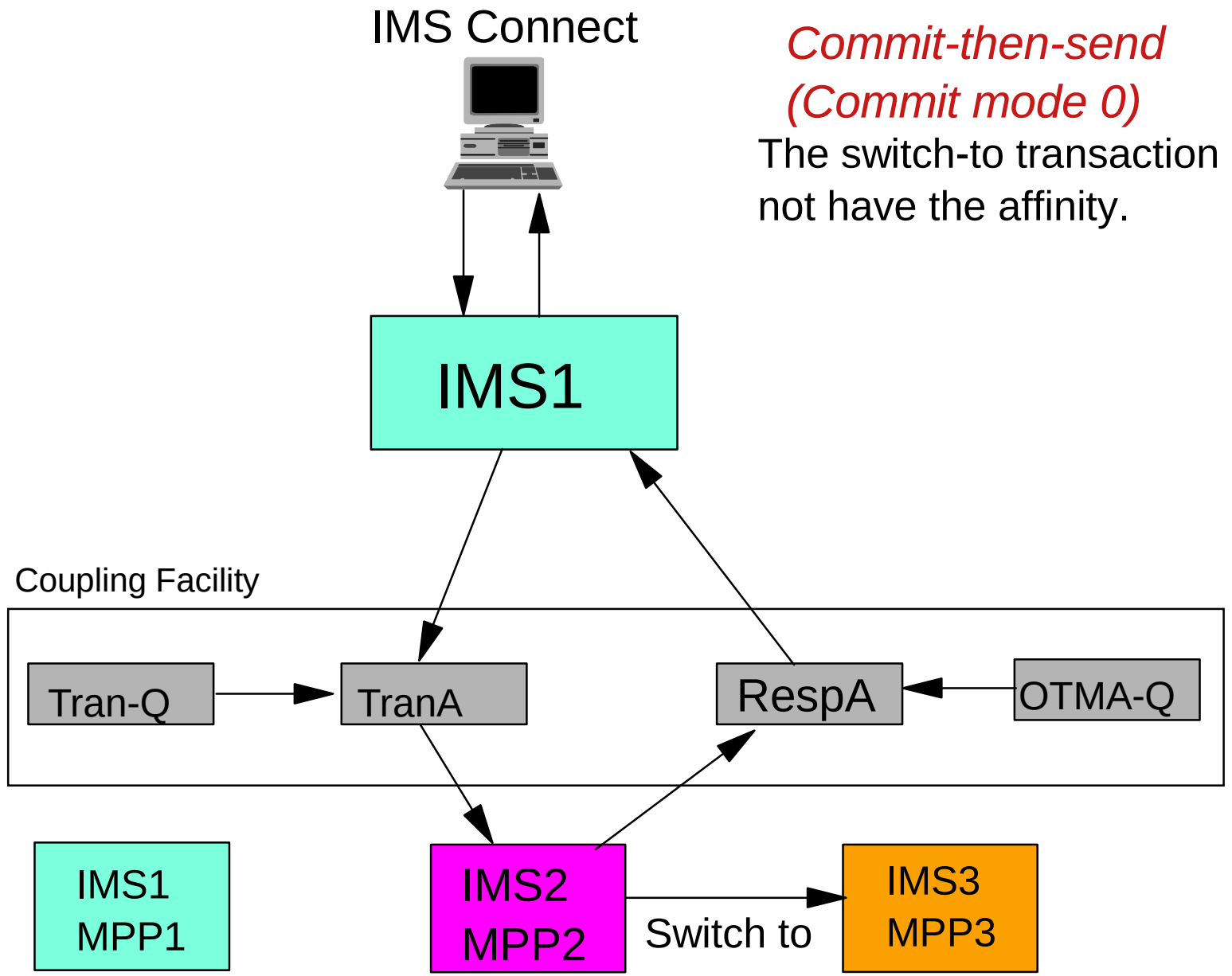
* The originating IMS processes the OTMA messages. Other IMSs in the shared queue group cannot do it.

OTMA output queue in the coupling facility is not used for send-then-commit response.

OTMA messages in IMS V7 Shared Queue Environment



OTMA messages in IMS V7 Shared Queue Environment



Commit-then-send
(Commit mode 0)

The switch-to transaction does not have the affinity.

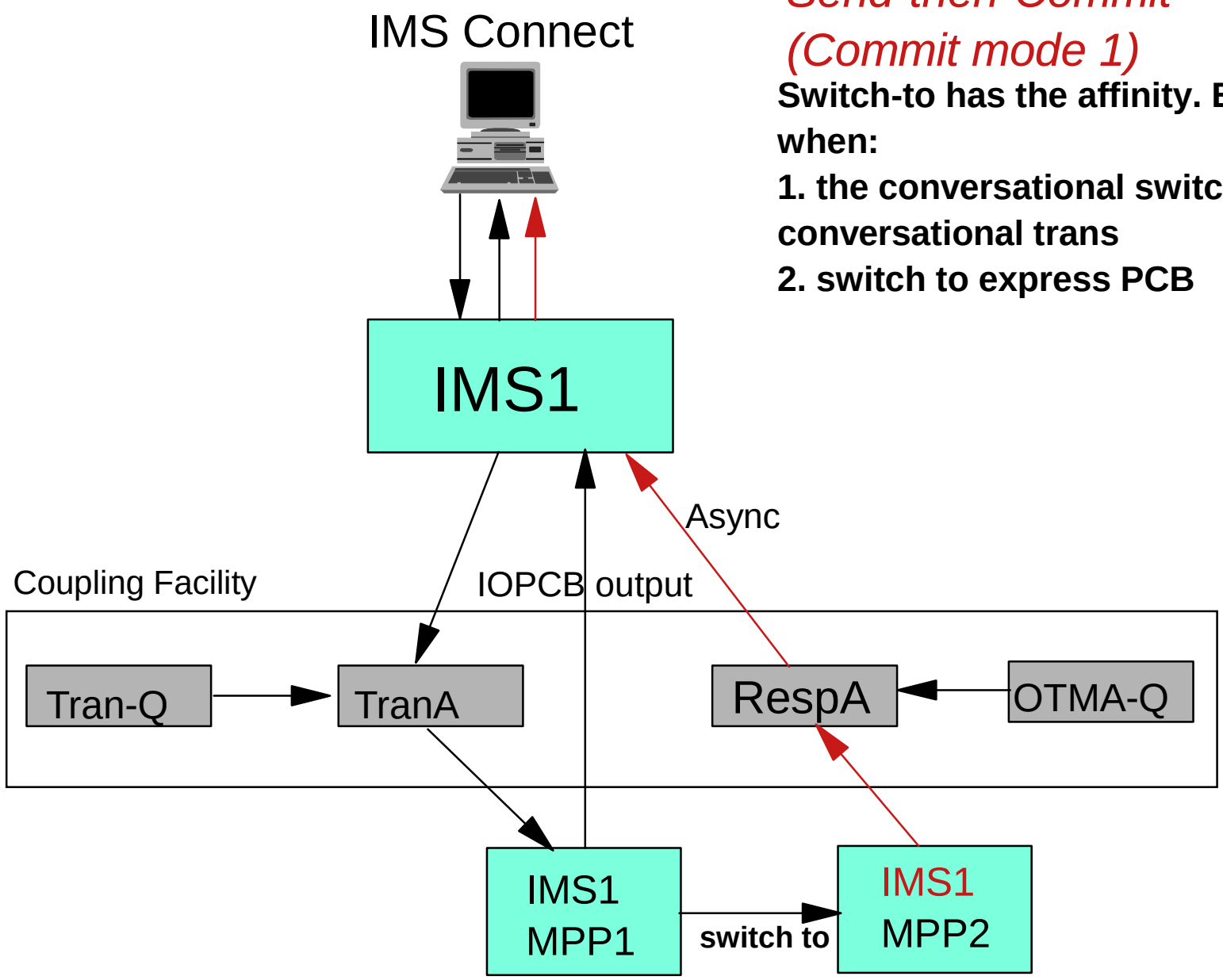
OTMA messages in IMS V7 Shared Queue Environment

Send-then-Commit (Commit mode 1)

Switch-to has the affinity. Even when:

1. the conversational switch to non-conversational trans

2. switch to express PCB



OTMA messages in IMS V7 Shared Queue Environment

- **Commit-then-send transactions entered from OTMA client**
  - Transaction can run on any IMS in the shared queue group.
  - Program switches can also run on any IMS.
  - IOPCB Output can only be delivered from the originating IMS, the receiving IMS.
  - ALTPCB Output is sent from the executing IMS.
- **Send-then-commit transaction processing remains the same. Only the receiving IMS can process it.**
- **Switch-to transaction which initiated by the CM1 (send-then-commit) message will always run on the receiving IMS.**

/DIS TRAN ALL QCNT

- /DISPLAY TRAN ALL QCNT displays all of the transactions on the shared queue with a global queue count. (V7 APAR PQ40720)
- It also displays an AFFINITY column which gives the IMS SYSID for transaction which have affinity for a particular IMS system.

Entry:
/DISPLAY TRAN ALL QCNT

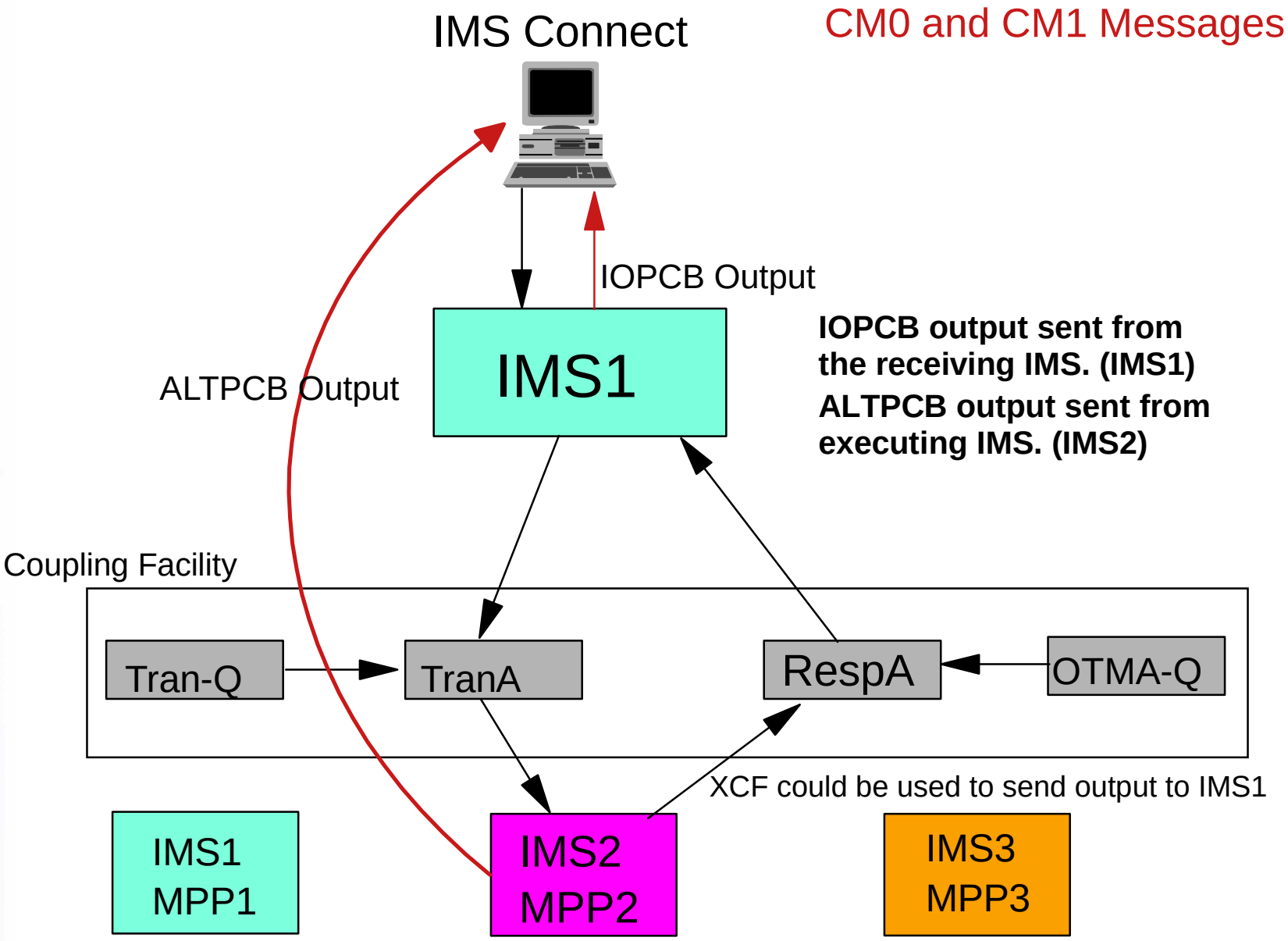
Response:

TRAN	GBLQCT	AFFINITY
SKS1	1234	IMS1
SKS1	56	IMS2
SKS1	78	

/DIS QCNT OTMA MSGAGE

- The output from the /DIS QCNT OTMA MSGAGE command was changed to include the TMEMBER and TPIPE associated with each output message.
- The new format is shown below. A '+' is used to indicate the lines that are continued on the next line.
- The solution is provided via V8 APAR PQ62784 or V7 APAR PQ64257.

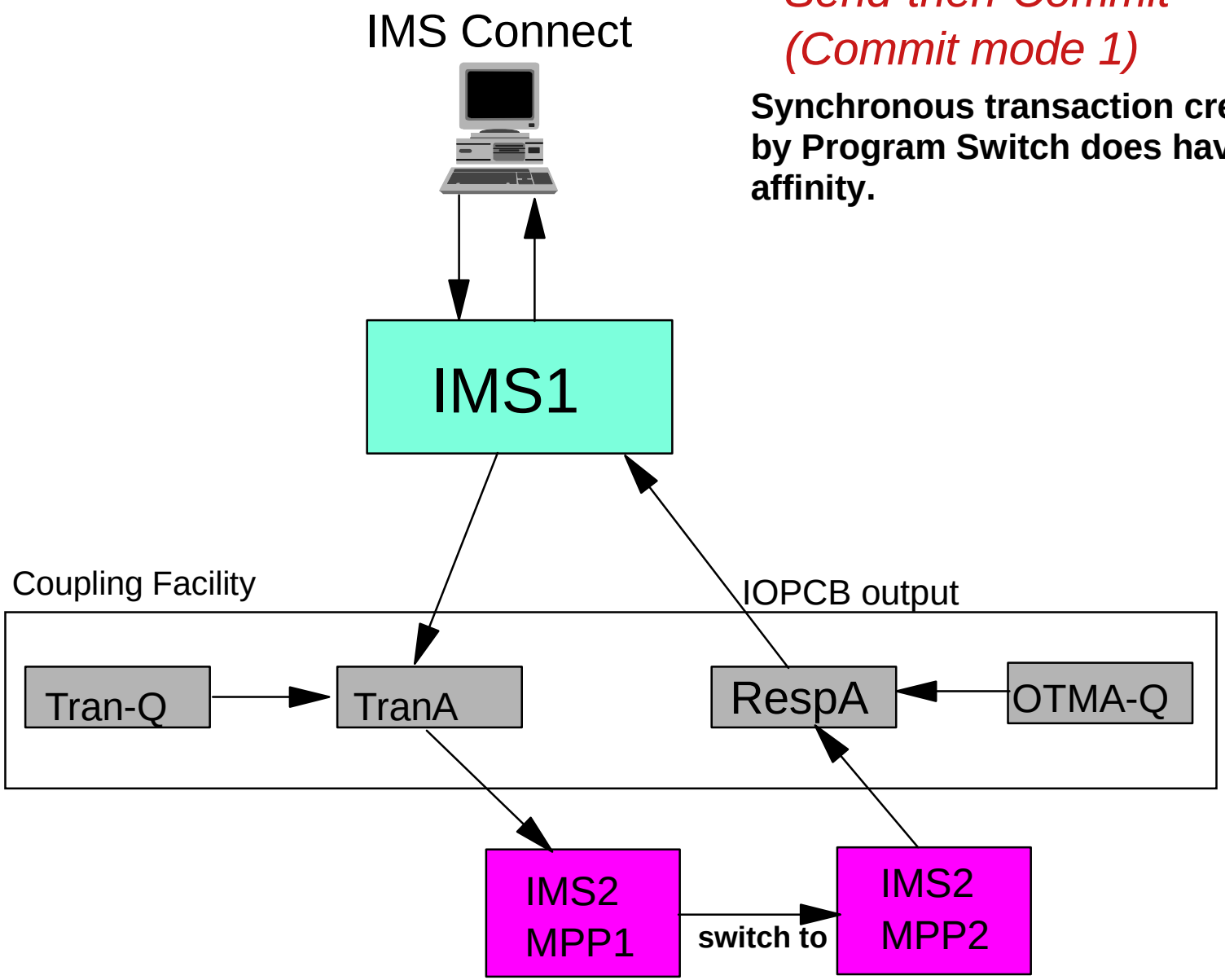
OTMA messages in IMS V8 Shared Queue Environment



OTMA messages in IMS V8 Shared Queue Environment

Send-then-Commit (Commit mode 1)

Synchronous transaction created by Program Switch does have the affinity.



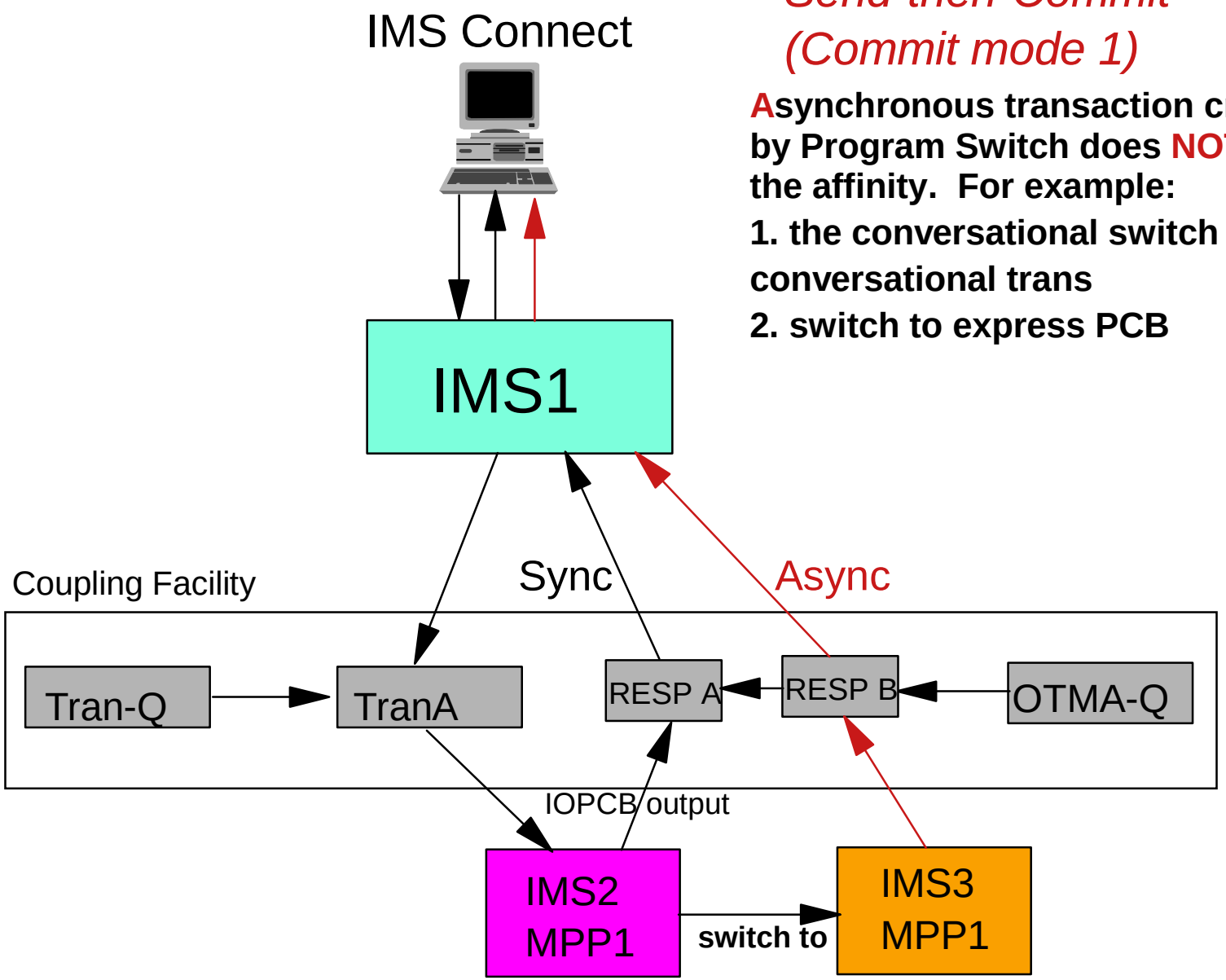
OTMA messages in IMS V8 Shared Queue Environment

Send-then-Commit (Commit mode 1)

Asynchronous transaction created by Program Switch does NOT have the affinity. For example:

1. the conversational switch to non-conversational trans

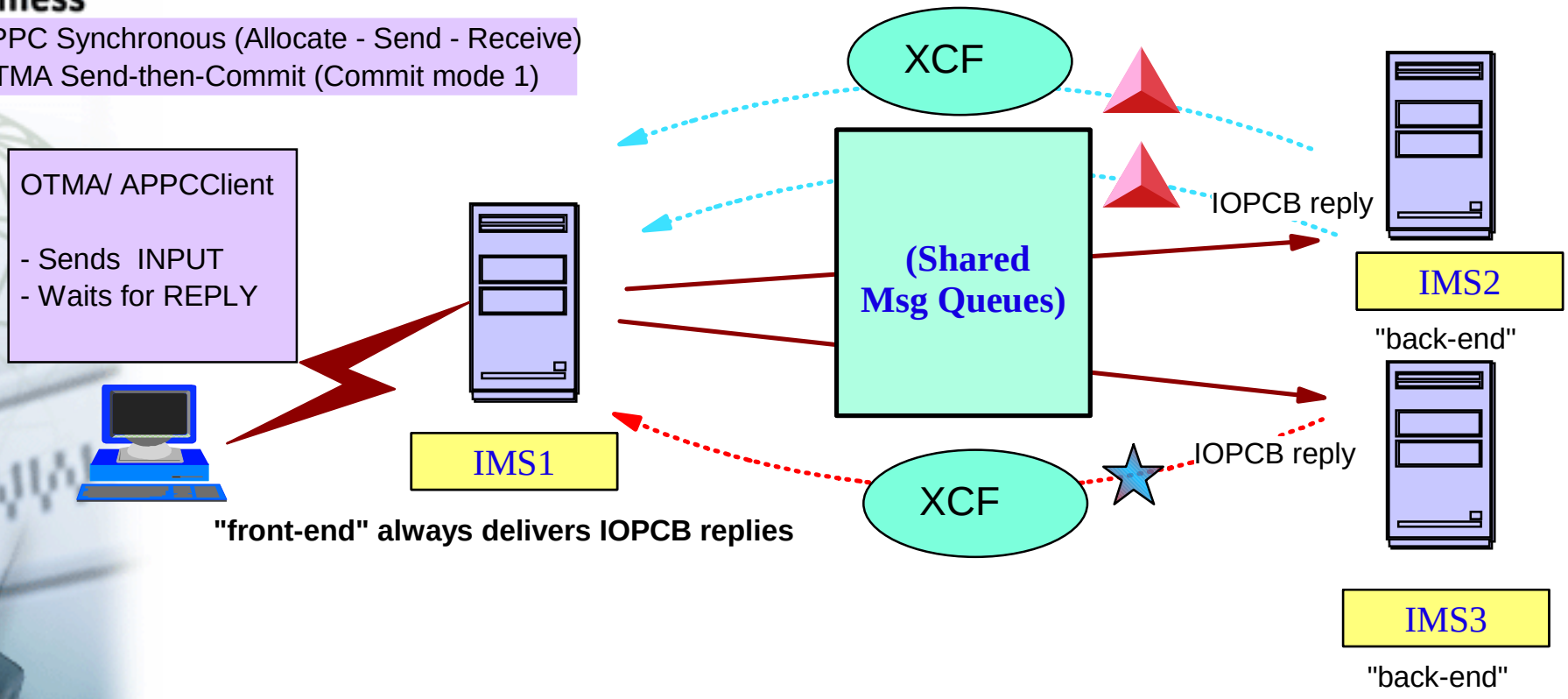
2. switch to express PCB



Using XCF for sending output

e-business

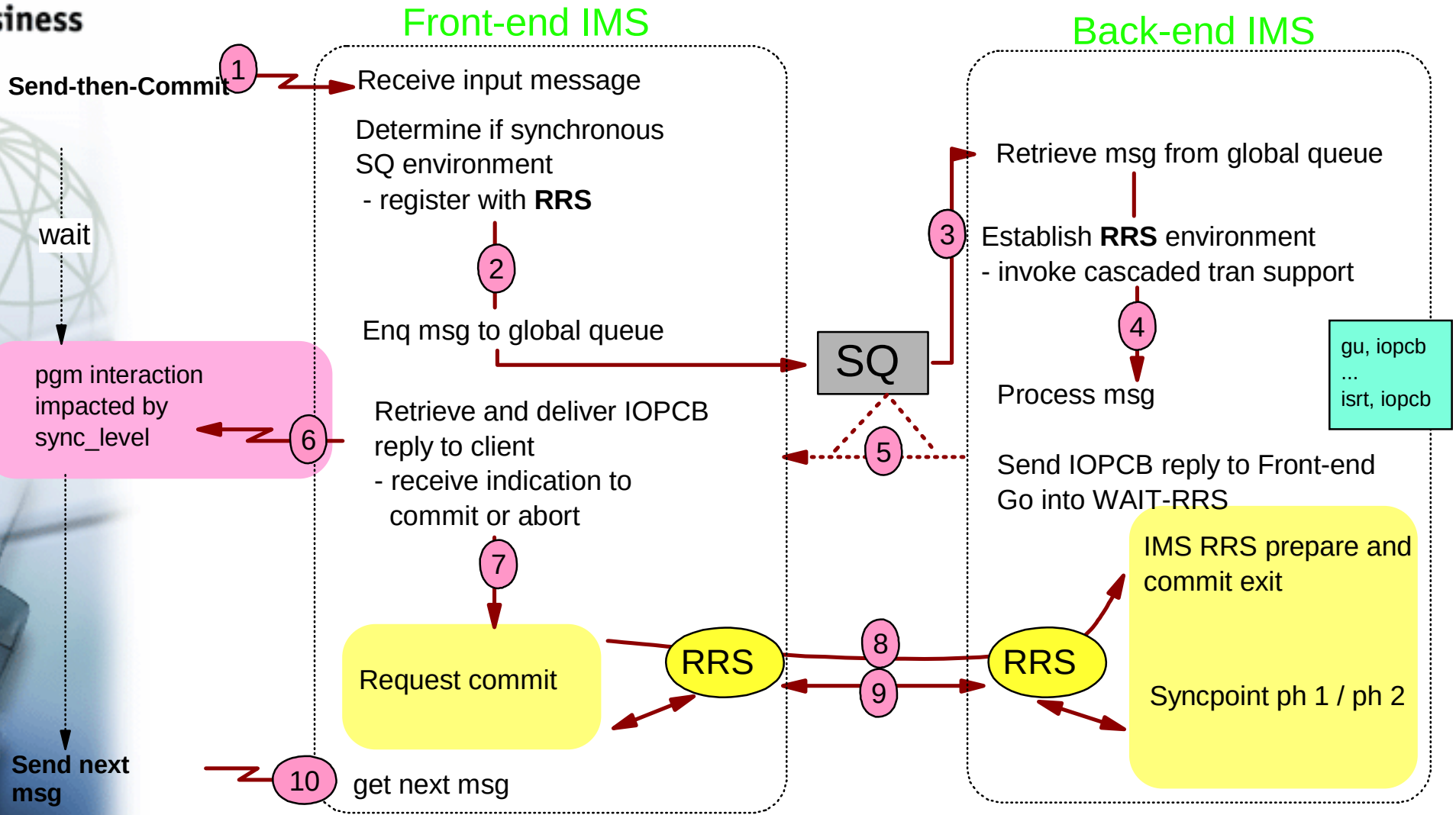
APPC Synchronous (Allocate - Send - Receive)
OTMA Send-then-Commit (Commit mode 1)



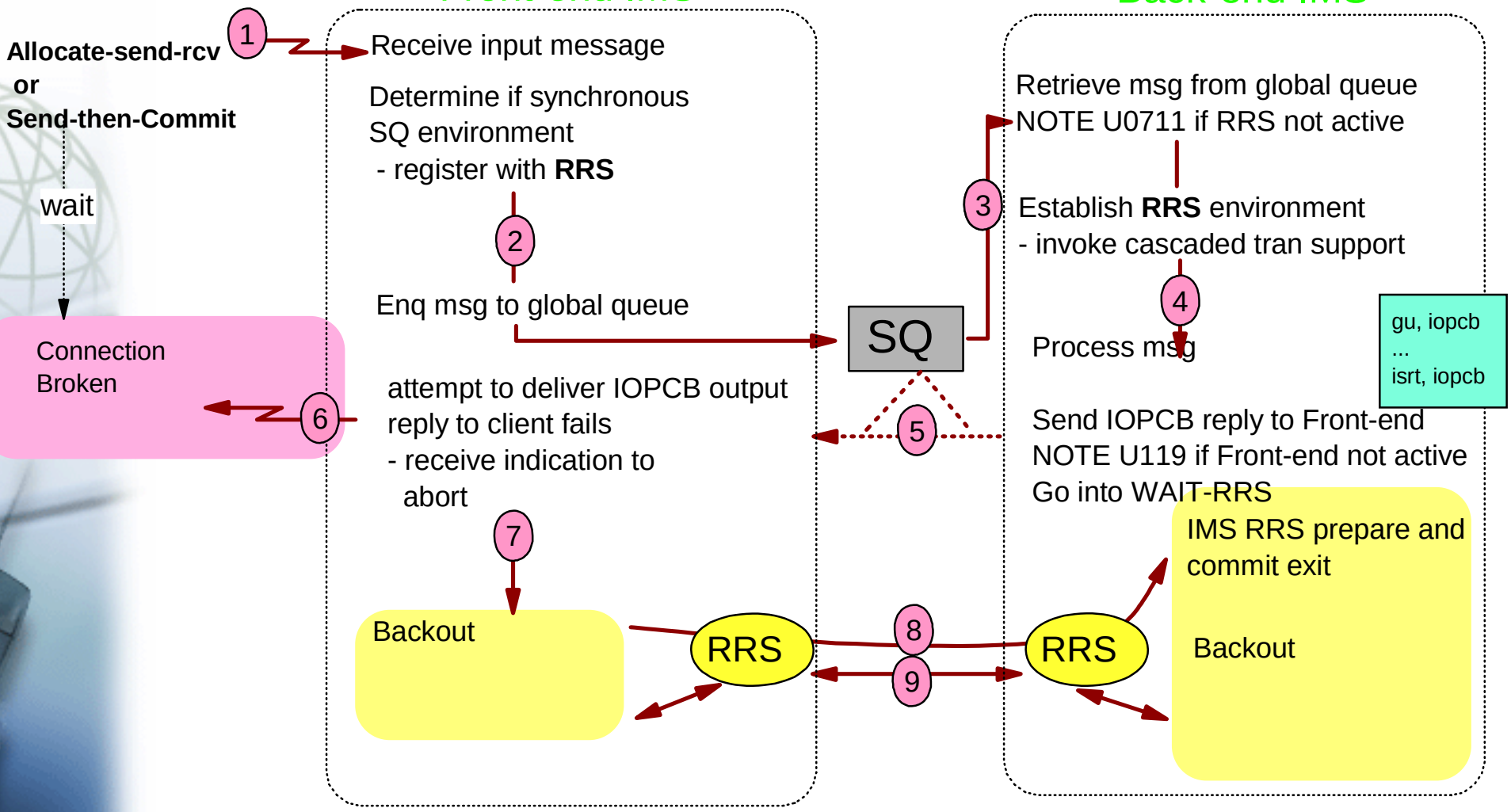
Non-conversational IOPCB reply messages (less than 61K) are sent to the front-end using XCF services

Conversational IOPCB reply messages or any messages greater than 61K are sent to the front-end using Shared Queues along with a special NOTIFY message that is sent using XCF

IMS OTMA/APPC Shared Queue with RRS cascaded transaction (commit flow)



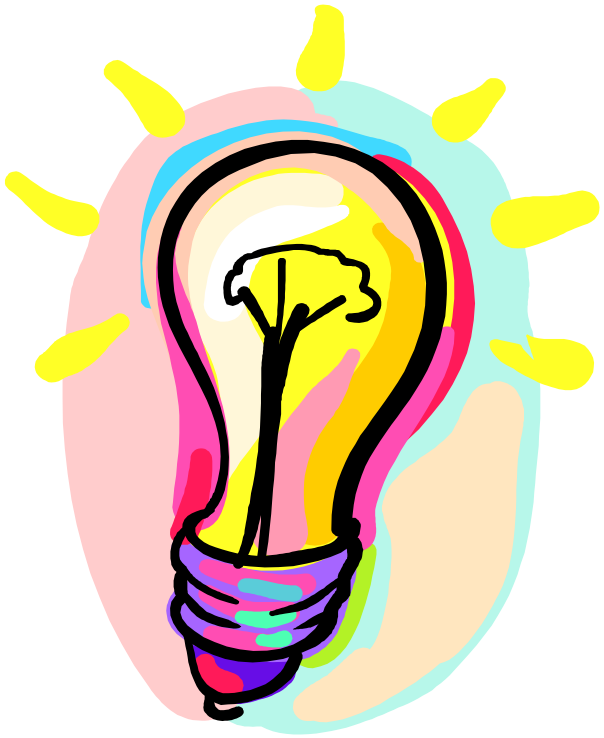
IMS OTMA/APPC Shared Queue with RRS cascaded transaction (backout flow)



IMS V8 OTMA CM1 Shared Queue Function Enablement

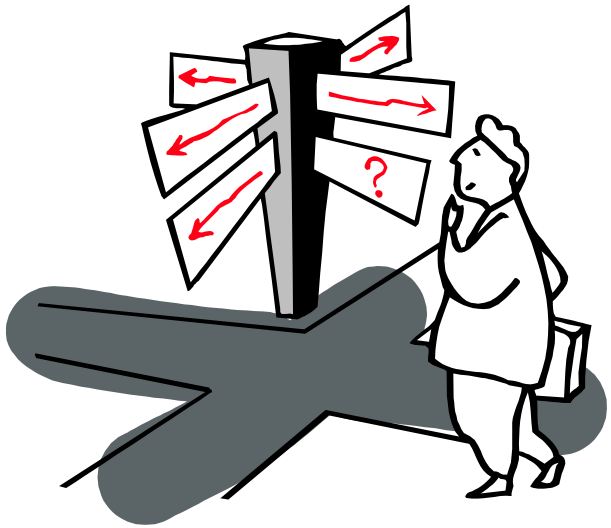
Enabled if:

- DBRC change.recon minvers(81)
- Shared Queues is enabled and
- All IMS systems in the SQ group are IMS V8 and
- All environments are z/OS V1.2 with RRS enabled



OTMA Shared Queue Considerations

- IMS V6 and V7 OTMA does not support all the shared queue features.
- IMS V8 completes the shared queue support for OTMA. However, it requires z/OS 1.2 RRS function.
- Performance will be impacted by using RRS.
- Send-then-commit transactions created by program-to-program swtiches are still running on the same IMS, the executing IMS. (No workload balancing for this type of program switches.)



Summary

- IMS OTMA allows you to use IMS TM as a server to communicate with many different MVS applications without the traditional SNA.
- IMS V8 OTMA/APPC Shared Queue support allows Send-then-commit (CM1) workload to be distributed and executed on any of the IMS systems in the Shared Queues group.

RETURN TO INDEX