IBM Information On Demand 2007

What you need to know about IMS and RRS

Jack Yuan, IBM, Senior Engineer

jackyuan@us.ibm.com

1136

Act.Right.Now.

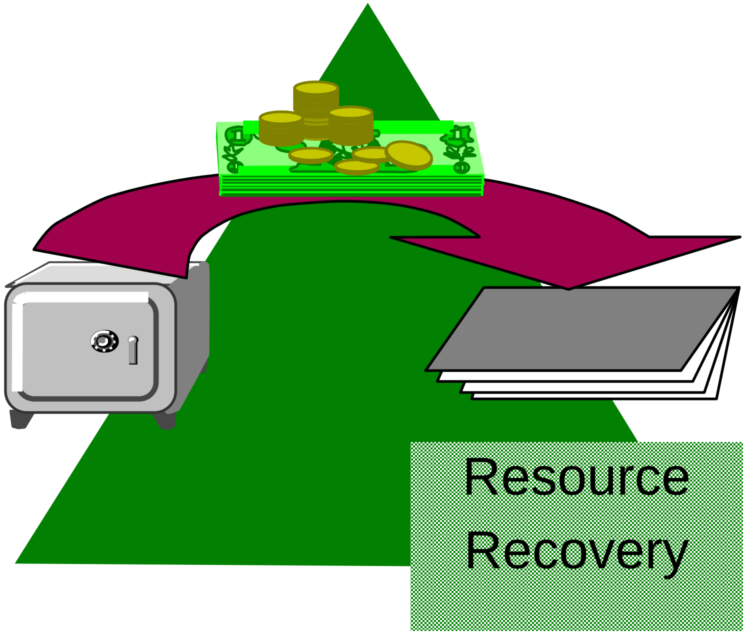
IBM INFORMATION ON DEMAND 2007
October 14 – 19, 2007
Mandalay Bay
Las Vegas, Nevada

Agenda

- Basic concepts of z/OS Resource Recovery
- Applications related to IMS and z/OS RRS
- Tips of running IMS with z/OS RRS
- Summary

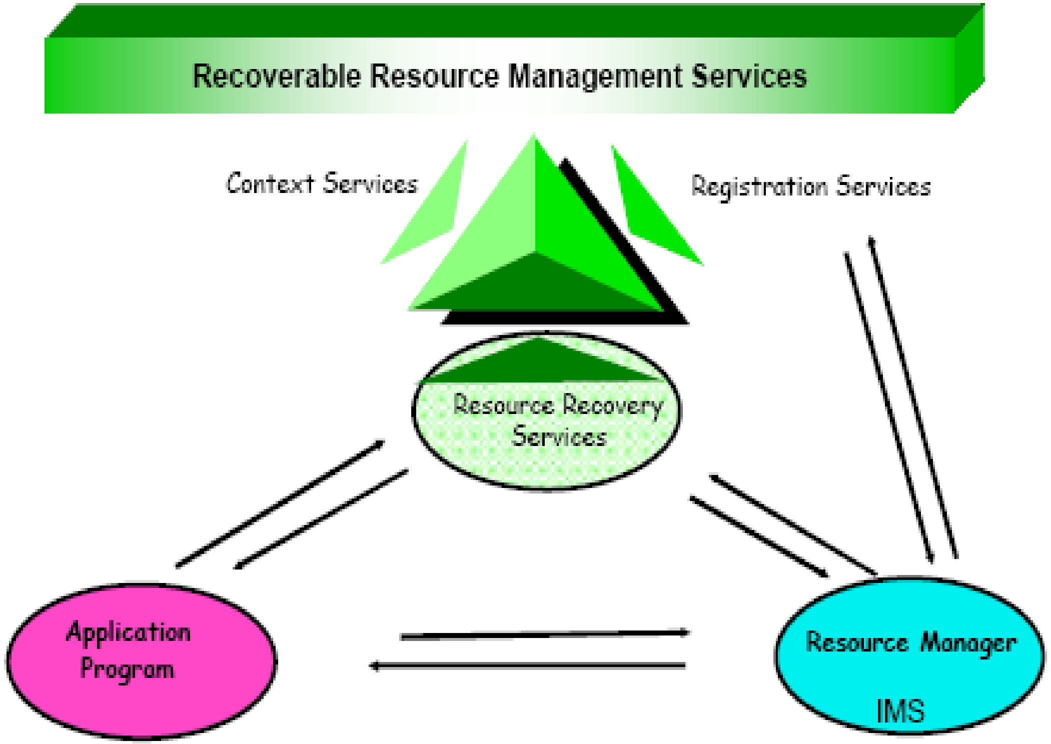
Your critical data or resource needs protection!

- Many resources are so critical to a company's work that the integrity of these resources must be guaranteed. These critical resources are called protected resources or recoverable resources.



z/OS Resource Recovery

- The z/OS Resource Recovery consists of the protocols and program interfaces that allow an application program to make consistent changes to multiple protected resources.
- Protected resources are sometimes called recoverable resources.

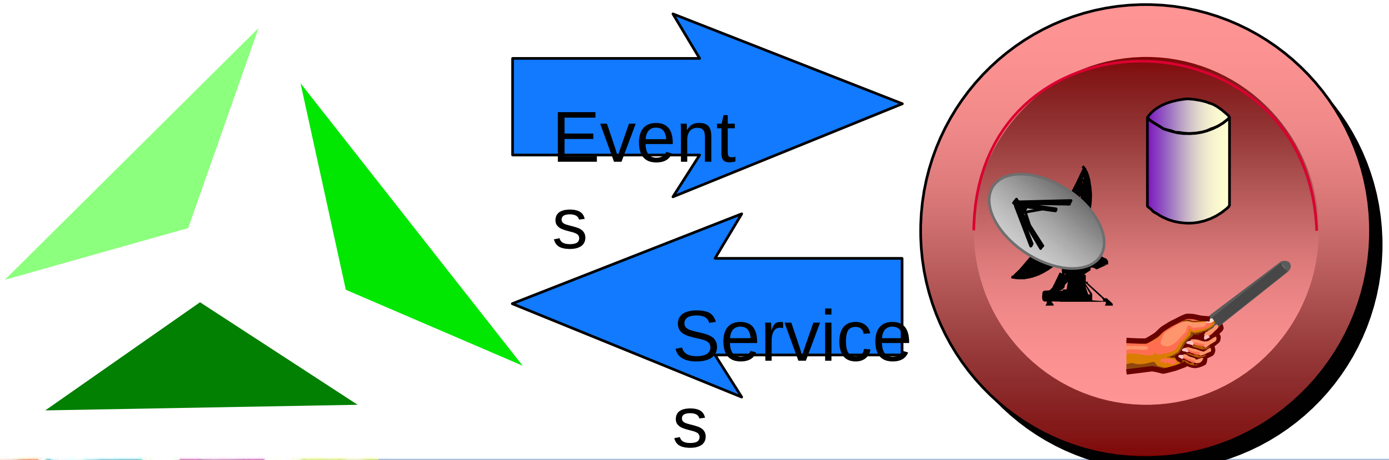


Resource Recovery Components

- Application Program
  - Accesses protected resources
- Resource Manager
  - Data Resource Manager (IMS DB)
  - Communication Resource Manager (IMS Connect)
  - Work Manager (IMS TM)
- Recoverable Resource Management Services
  - Resource Recovery Services (Syncpoint Manager)
  - Registration Services
  - Context Services
  - System Logger

Exit Managers - Providing Services to Resource Managers

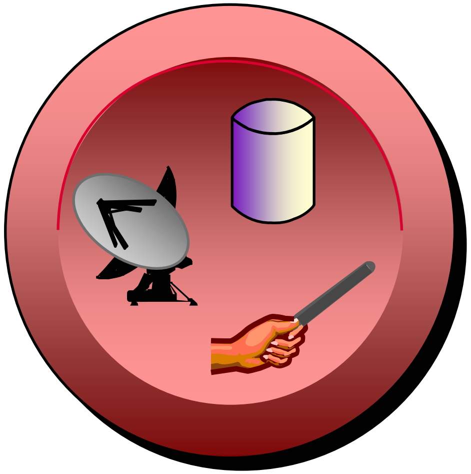
- **Exit Managers,**
  - Context Service and Resource Recovery Services
  - Provide **services** to RMs
  - Inform RMs of **events** via **exits**



Event
s

Service
s

Resource Managers



- **All resource managers (RM):**
  - ► Are **authorized**
  - ► **Register** with the system
  - ► Supply **exit routines**
  - ► Have a **system unique name**

- Additionally, RRS compliant resource managers:
  - ► Go through RRS **restart** processing
  - ► Have a **Sysplex unique name**
  - ► Respond to **syncpoint** events

Types of Resource Managers

There are 3 types of Resource Managers

Data Managers
- DB2 DB, IMS DL1, VSAM

Communications Managers
- APPC/PC, TRPC, MQ

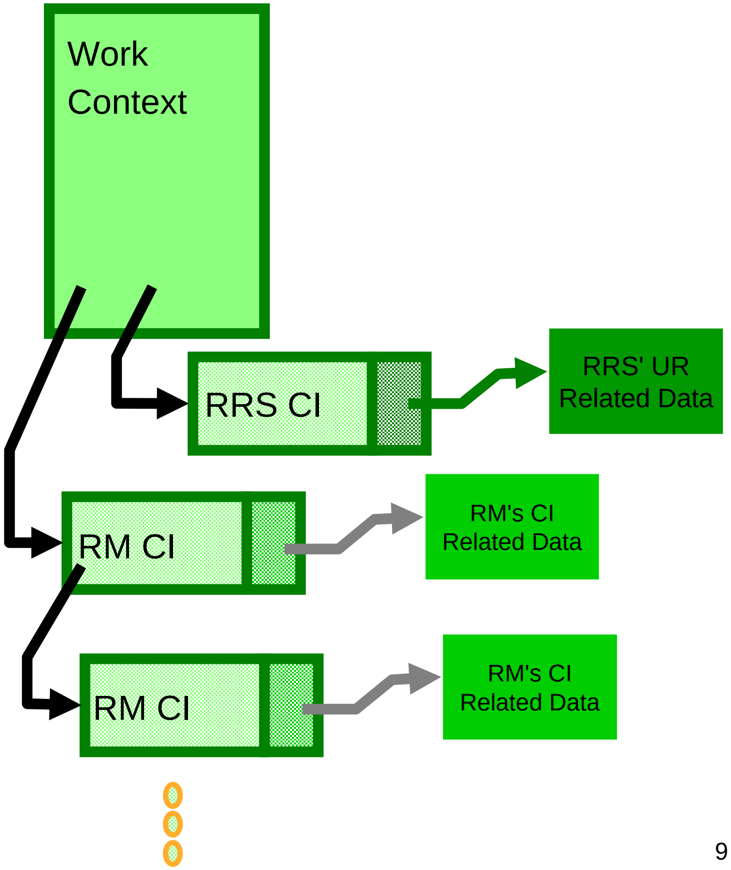
Work Managers
- IMS TM, CICS TS, DB2 Stored Procedures

Registration Services - Connecting RMs and EMs

- **Always available**
- **Is informed by EMs and RMs when they are available**
- **Informs EMs when RMs come and go**
- **Informs RMs when EMs come and go**
- **Provides global anchors for each**

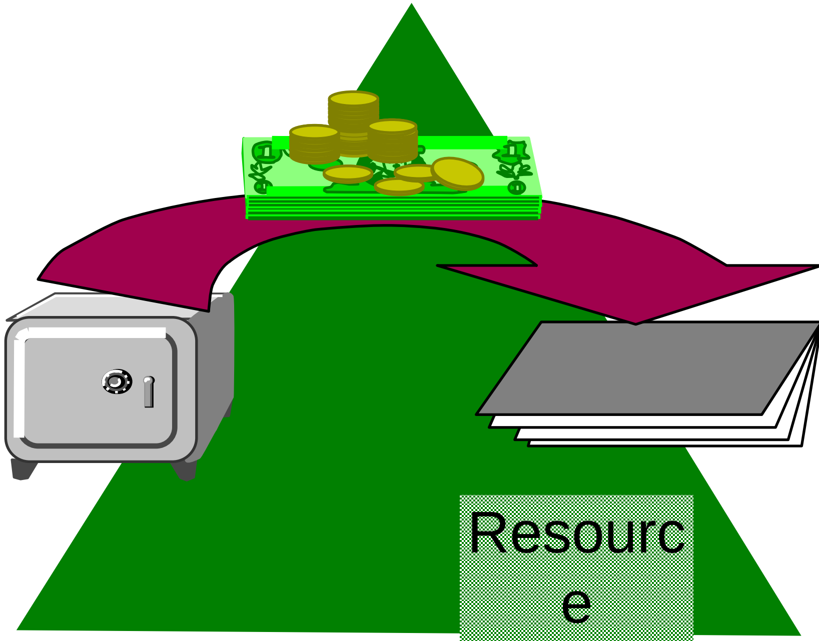
Context Services - Basic Structures

- The Context (or **Work Context**) is the base structure
- When a RM expresses interest in a Context, a CI is created for it.
- When RRS expresses interest in a context it gets a "special" CI
- Each CI has "user data" associated with it



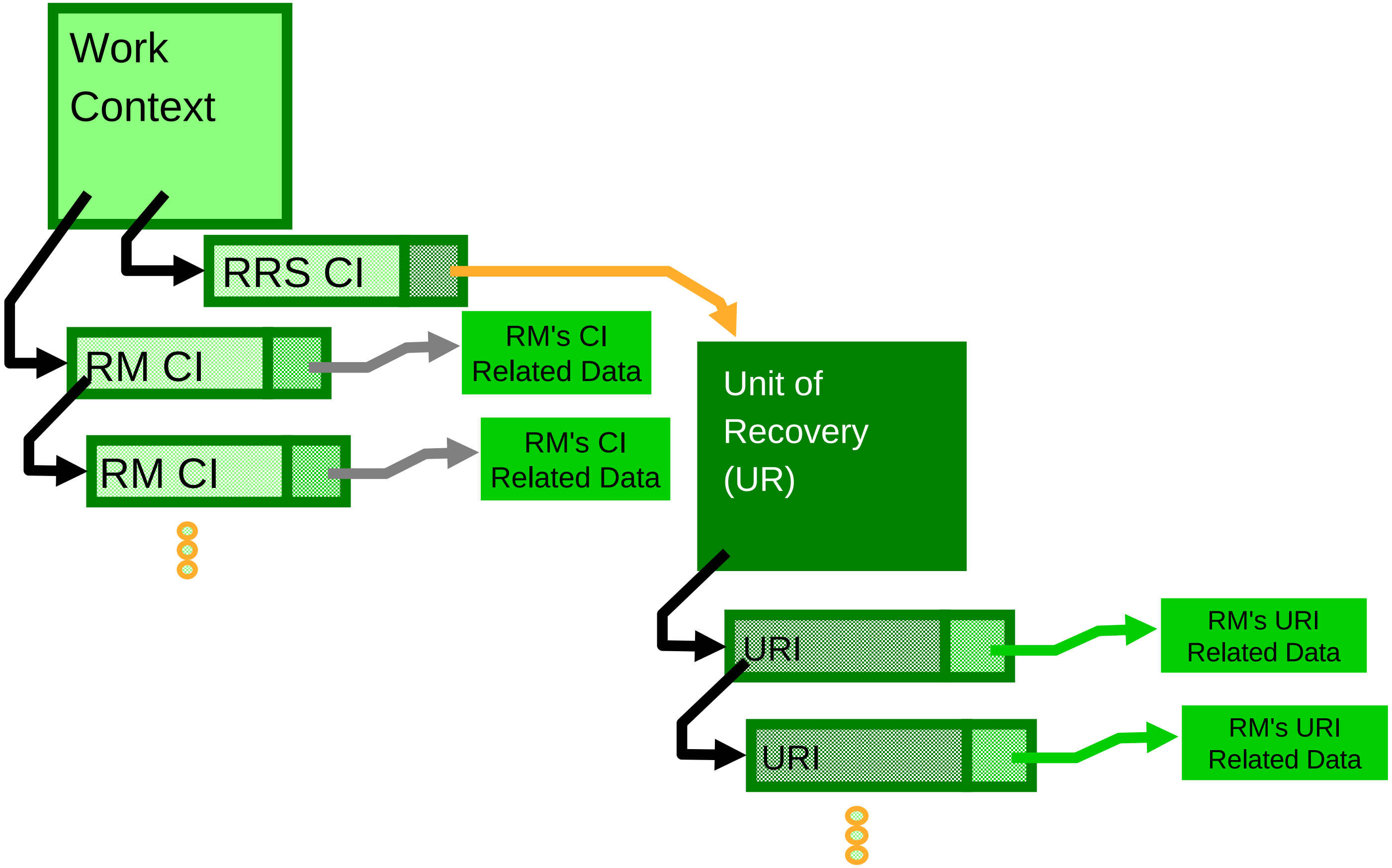
Resource Recovery Services - The Syncpoint Coordinator

- **Coordinates the two-phase commit process**
- **Creates an association between a Unit of Recovery (UR)** and a Work Context
- Preserves UR state across all failures



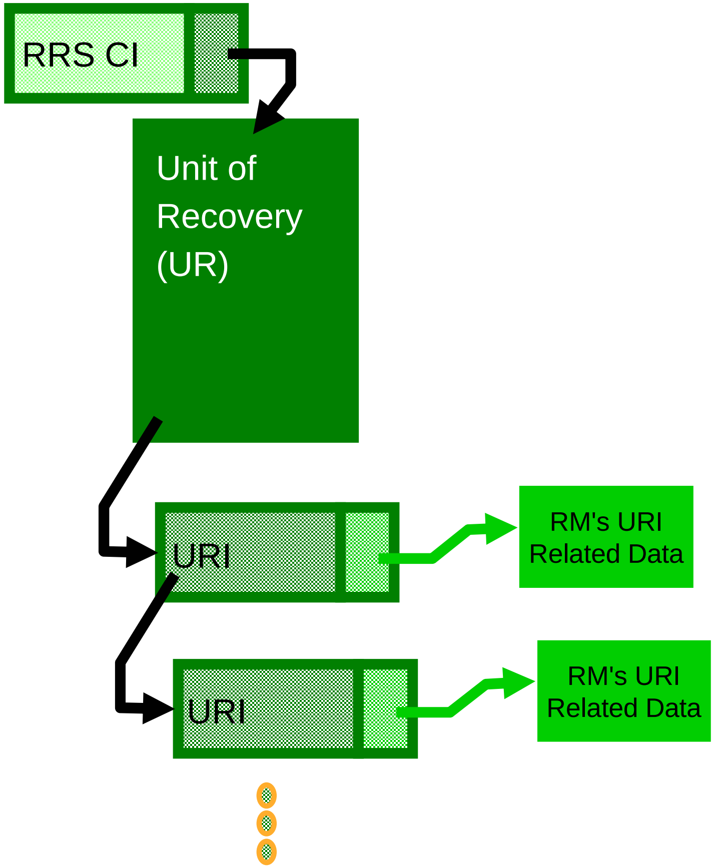
Resourc
e
Recover
y
Services

Context Services and RRS - Basic Structures Combined



Resource Recovery Services - Basic Structures

- The **Unit of Recovery (UR)** is the base structure
- Each time a RM expresses interest in a UR, a new **URI** is created.
- Each **URI** has persistent and nonpersistent "user data" associated with it



Required RRS Logstreams

- Main Unit of Recovery Logstream
  - *ATR.gname.MAIN.UR*
  - Contains the state of active URs. RRS periodically moves this information into RRS delayed UR state log when UR completion is delayed.
- Delayed Unit of Recovery Logstream
  - *ATR.gname.DELAYED.UR*
  - Contains the state of active URs when UR completion is delayed.
- Resource Manager Data Logstream
  - *ATR.gname.RM.DATA*
  - Contains information about the resource managers using RRS services.
- Restart Logstream
  - *ATR.gname.RESTART*
  - Contains information about incomplete URs needed during restart.

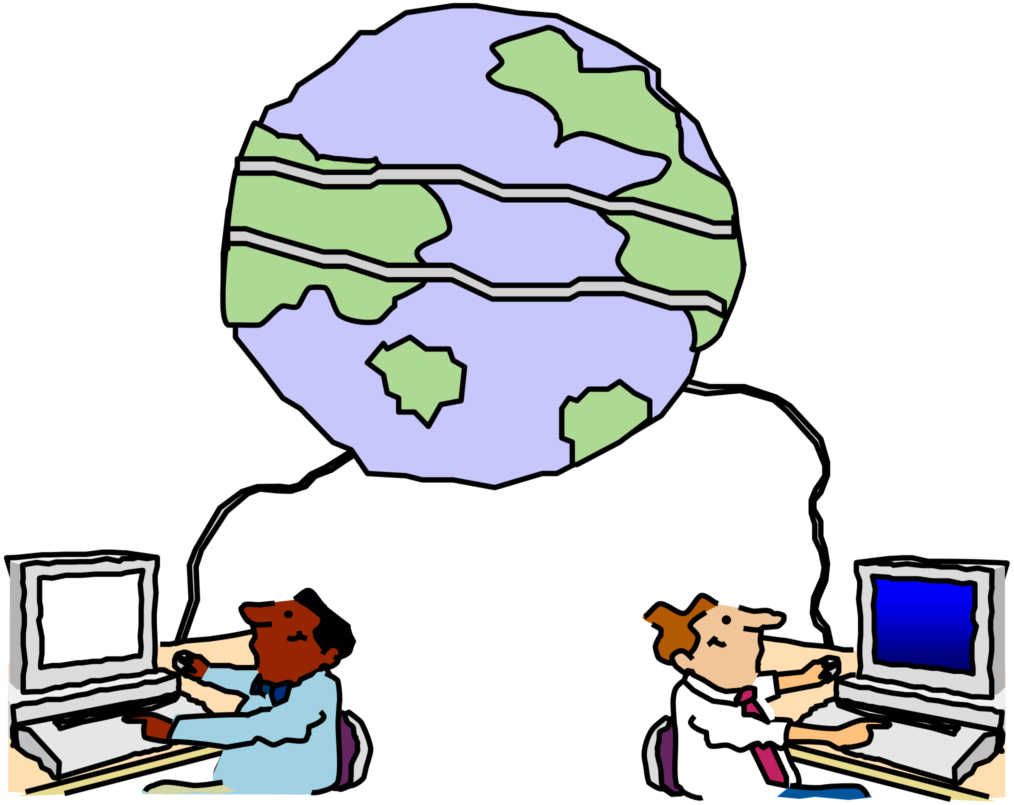
- RRS Archive Logstream (Optional)
  - *ATR.gname.ARCHIVE*
  - Contains information about completed URs

Sync Point Processing:

Application Program indicates commit point

Sync Point Phase 1 - "Prepare" : Each resourece manager votes

Sync Point Phase 2 - "Commit" : Changes are committed

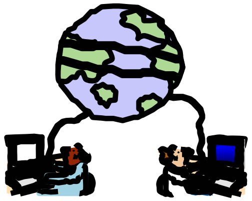


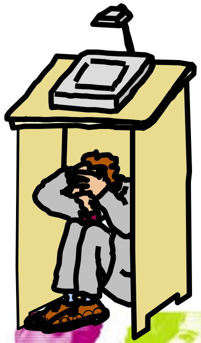
Vote

Vote

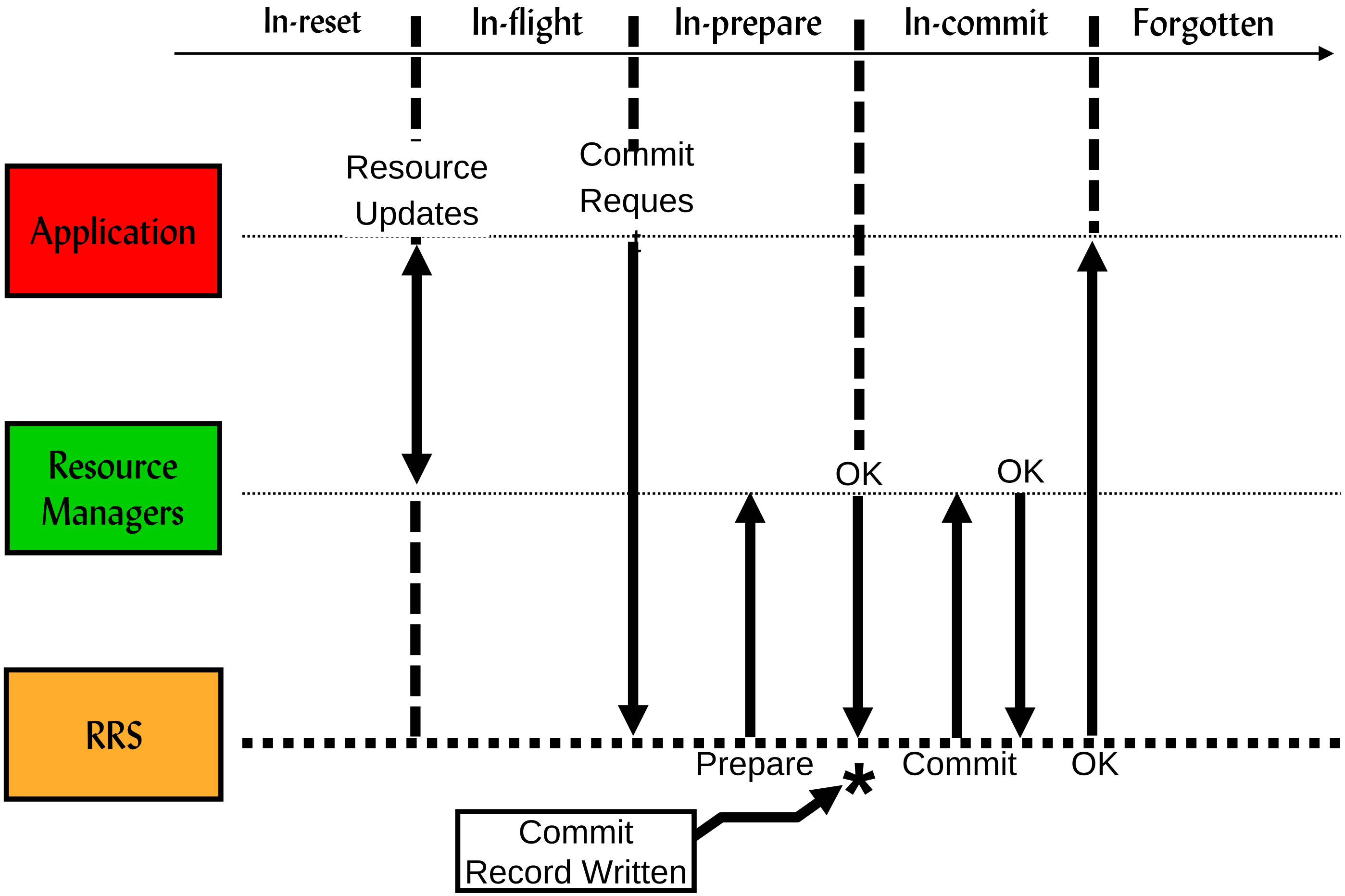
RRS Unit of Work States

- **IN-FLIGHT :**
  - ►Work changes in process
- **IN-DOUBT**
  - ►Work changes between phase 1 and phase 2
- **IN-COMMIT**
  - ►Work changes are committed
- **IN-BCKOUT**
  - ►Work changes are backed out

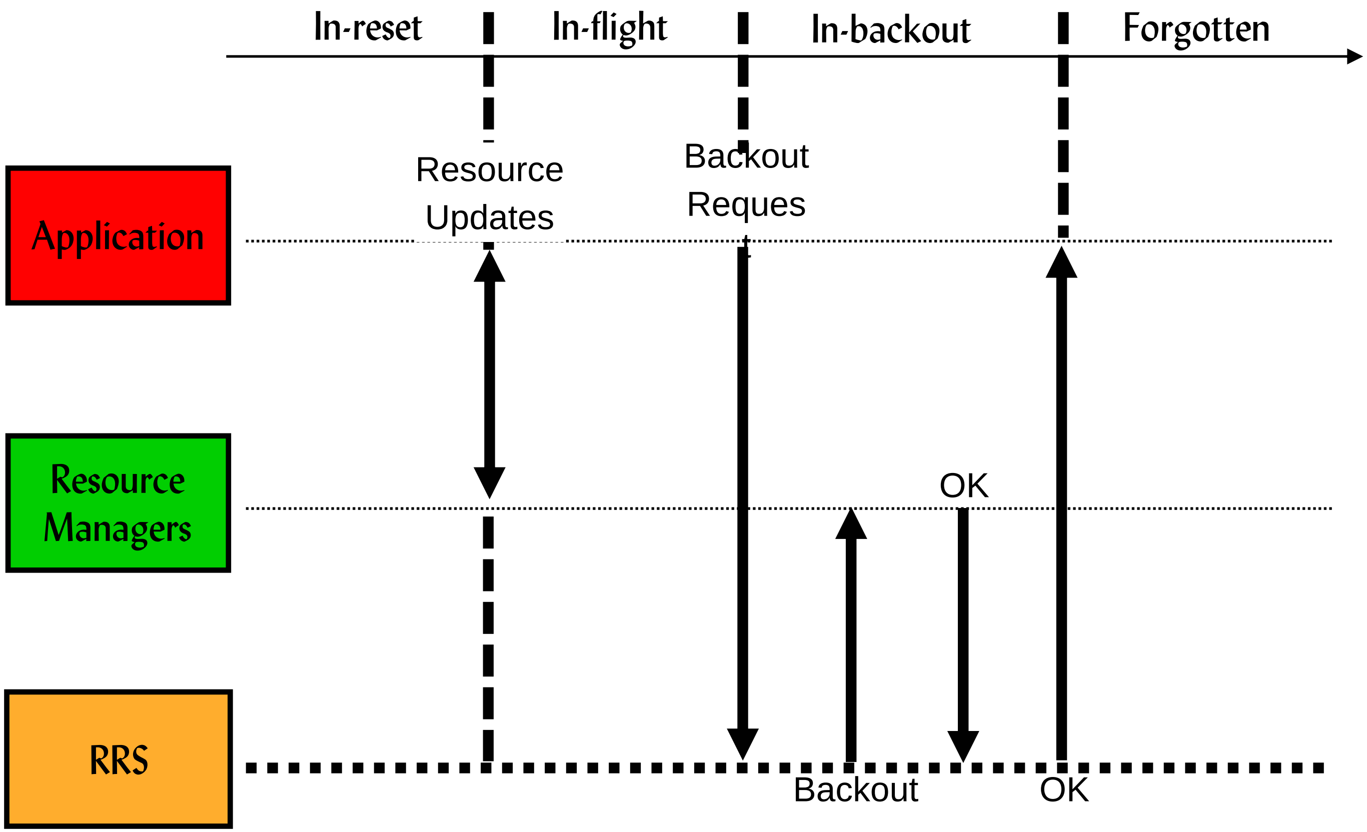




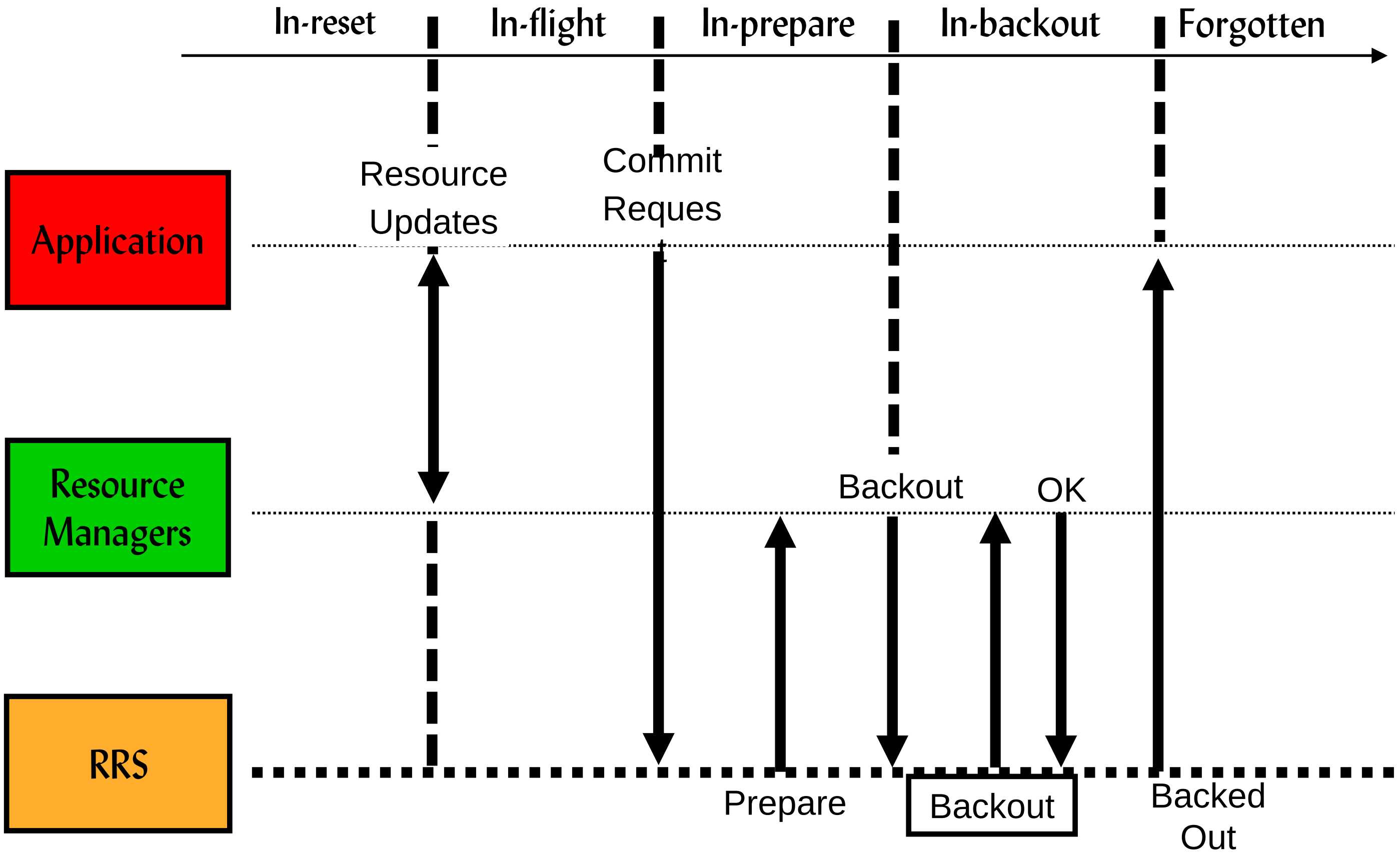
Basic Two-Phase Commit



Application Initiated Backout



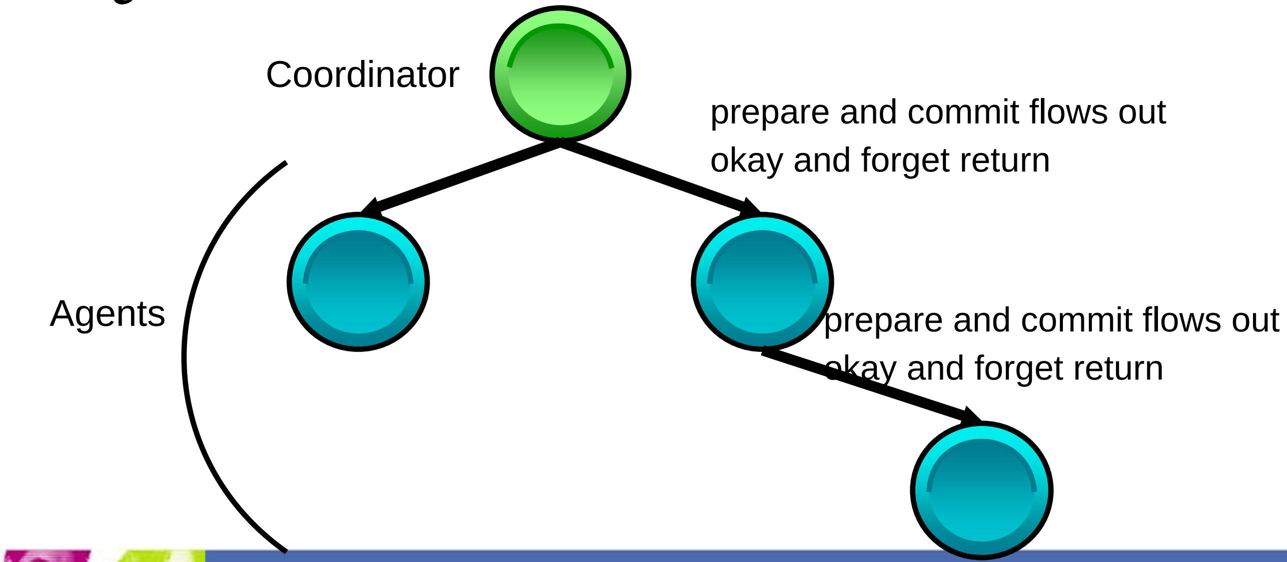
RM Causes Backout



Backout

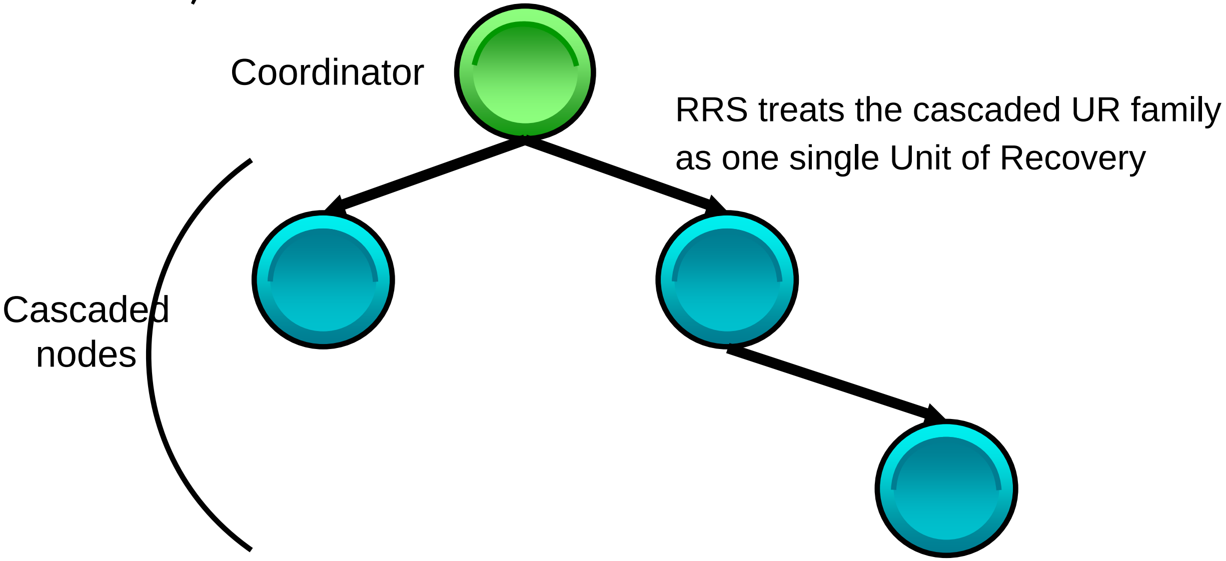
Distributed Transactions

- A distributed transaction occurs when an application updates resources on more then one system.
- The syncpoint manager on each system goes through a two-phase commit process with communication between syncpoint managers handled by a communications manager.

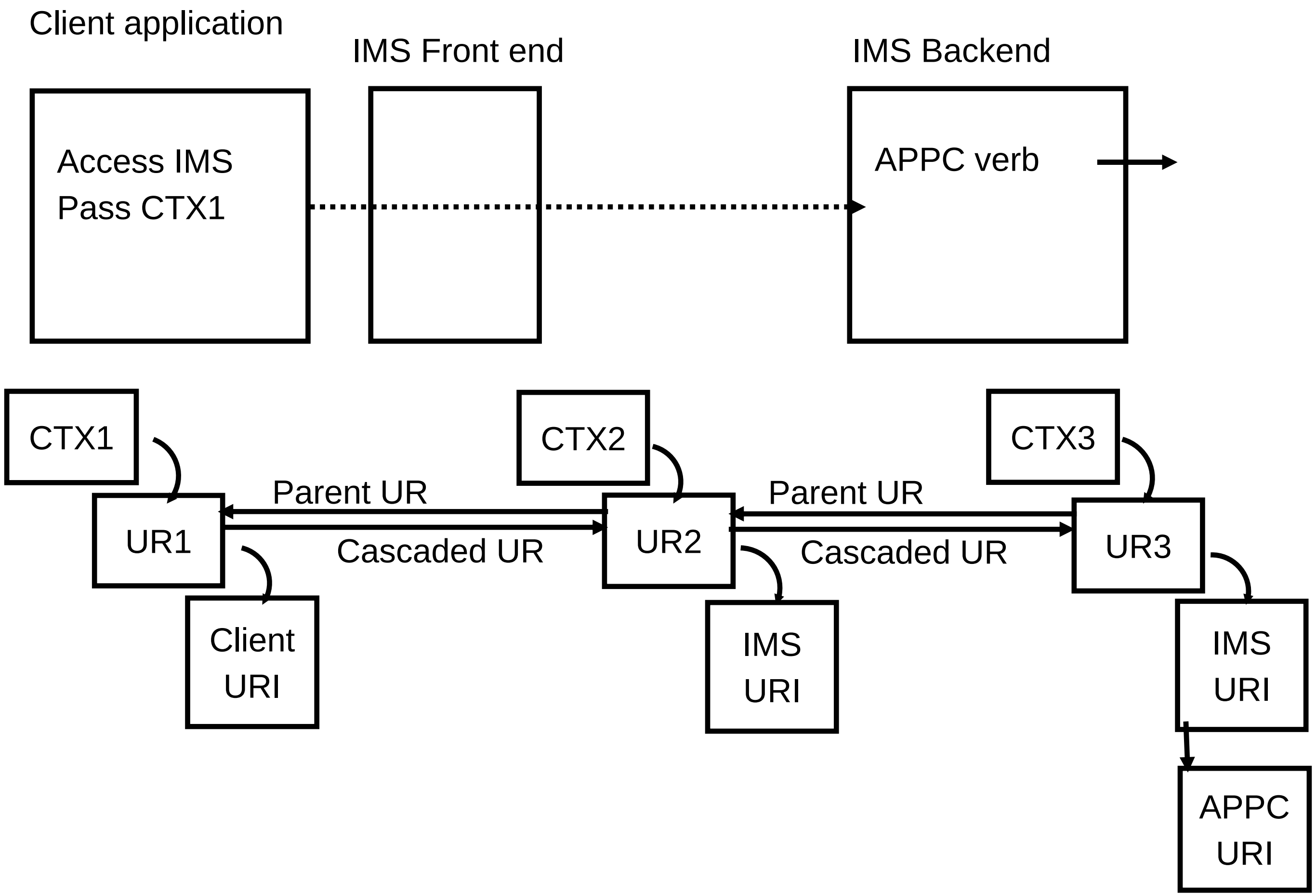


Cascaded Transactions

- A type of distributed transaction where RRS coordinates the separate parts of the transaction.
- Nodes can be on a single system (local cascaded) or on any system in an RRS logging group (multisystem cascaded).


Coordinator
RRS treats the cascaded UR family as one single Unit of Recovery
Cascaded nodes

Cascaded Transactions Example



IBM Information On Demand 2007

IBM

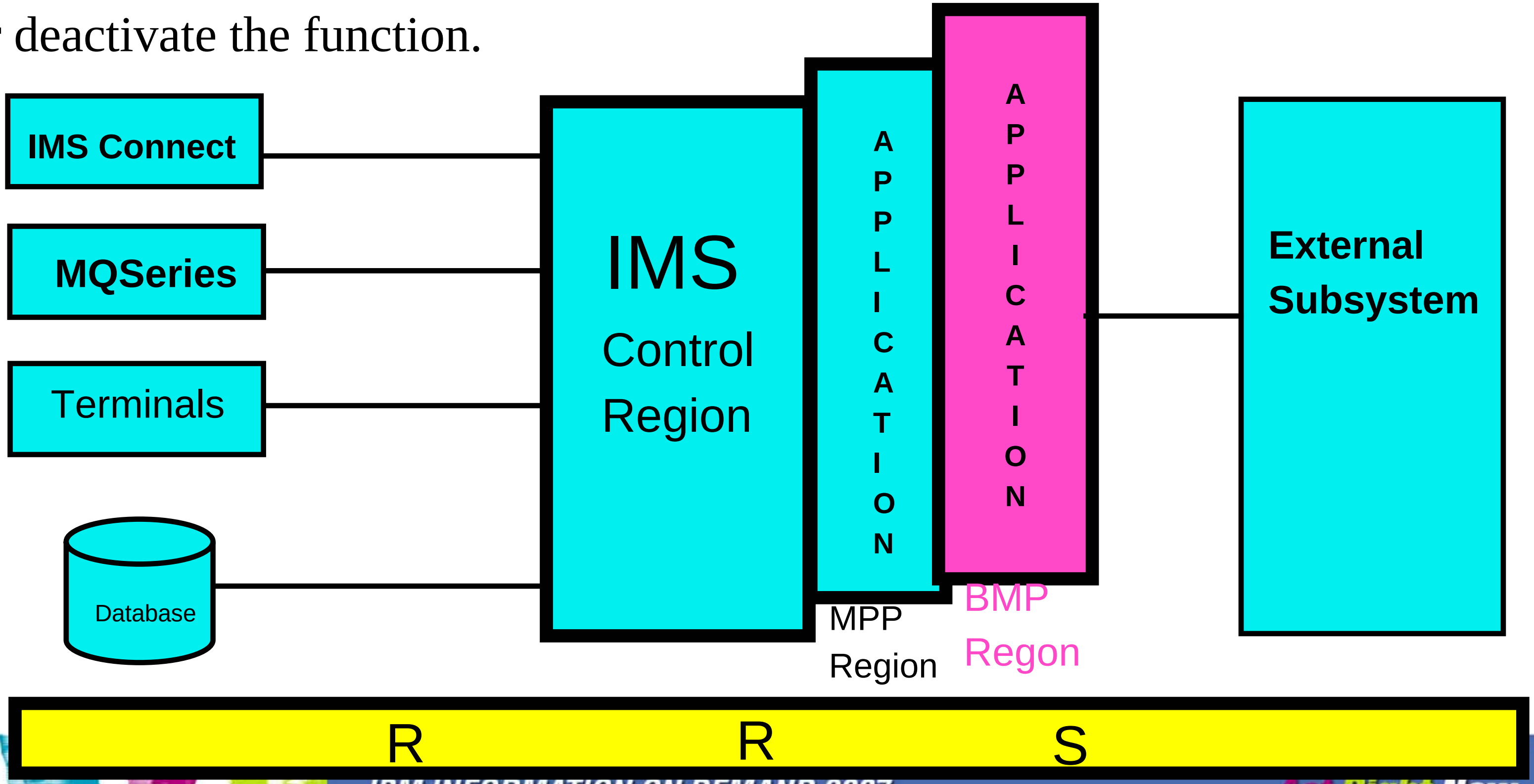
Application related to IMS and RRS

Act.Right.Now.

IBM INFORMATION ON DEMAND 2007
October 14 – 19, 2007
Mandalay Bay
Las Vegas, Nevada

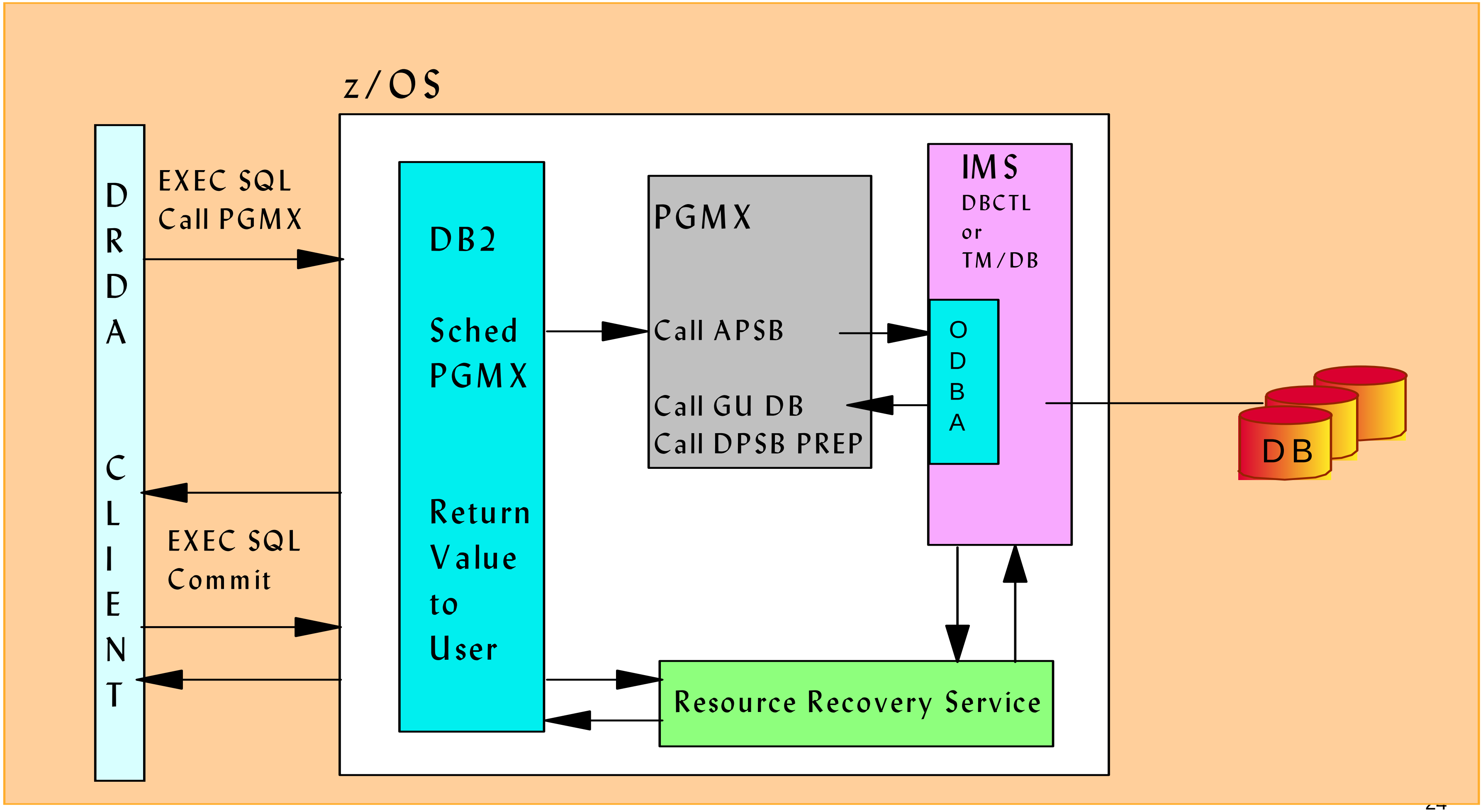
IMS Batch Support

- IMS supports the 2-phase commit process for batch application programs.
- Users have batch applications which can access and update the resources of more than one resourece managers, like IMS or DB2.
- The "RRS=" in DBBBATCH and DLIBATCH can be used to activate or deactivate the function.



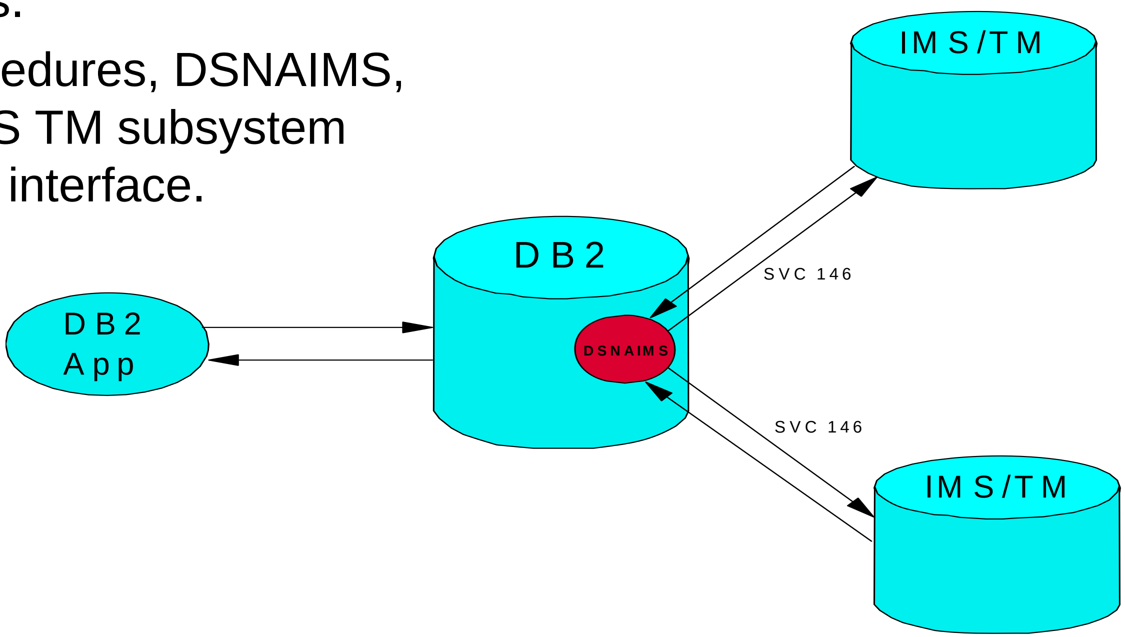
IMS ODBA Support

- IMS ODBA is an interface that allows z/OS application to access IMS DL/I full-function databases and data-entry databases (DEDBs).



DB2 Stored Procedures

- DB2 stored procedures access to an IMS DB subsystem using ODBA interface.
- DB2 stored procedures access to an IMS TM subsystem using MVS/APPC calls.
- DB2 stored procedures, DSNAIMS, access to an IMS TM subsystem using OTMA C/I interface.



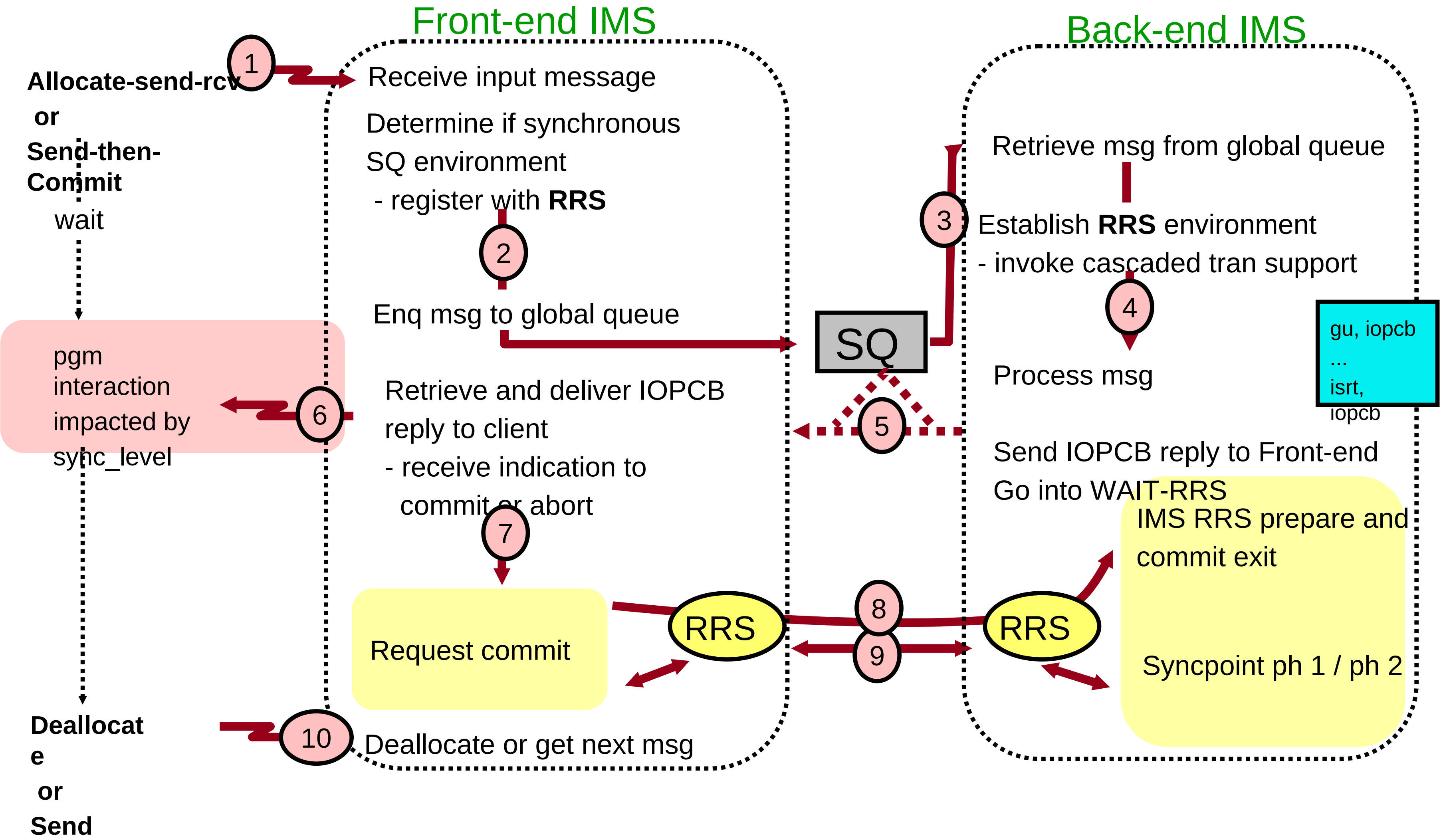
IMS Protected Transaction Support

- IMS supports protected transactions in the APPC/IMS and OTMA environments, so database resources are protected by a two-phase commit processing.
  - – APPC/IMS, a component of IMS TM, allows distributed and cooperative between IMS and systems that have implemented APPC.
  - – OTMA, a component of IMS TM, allows protected transaction specified with synclevel=syncpt in the input message prefix to be processed in IMS with RRS.

IMS OTMA/APPC Synchronous Shared Queues support

- IMS V8 exploits the z/OS RRS cascaded transaction function to provide the IMS shared queues enhancement for protected and non-protected transaction for both OTMA and APPC so that the synchronous shared queues transaction can be processed by any shared queues back-end IMS.

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



Front-end IMS

Back-end IMS

Allocate-send-rcv
or
Send-then-Commit
wait

1 Receive input message

Determine if synchronous SQ environment
- register with **RRS**

2 Enq msg to global queue

SQ

3 Retrieve msg from global queue

Establish **RRS** environment
- invoke cascaded tran support

4 Process msg

gu, iopcb
...
isrt, iopcb

pgm interaction impacted by sync_level

6 Retrieve and deliver IOPCB reply to client
- receive indication to commit or abort

5 Send IOPCB reply to Front-end
Go into WAIT-RRS

7 Request commit

RRS 8 9 RRS

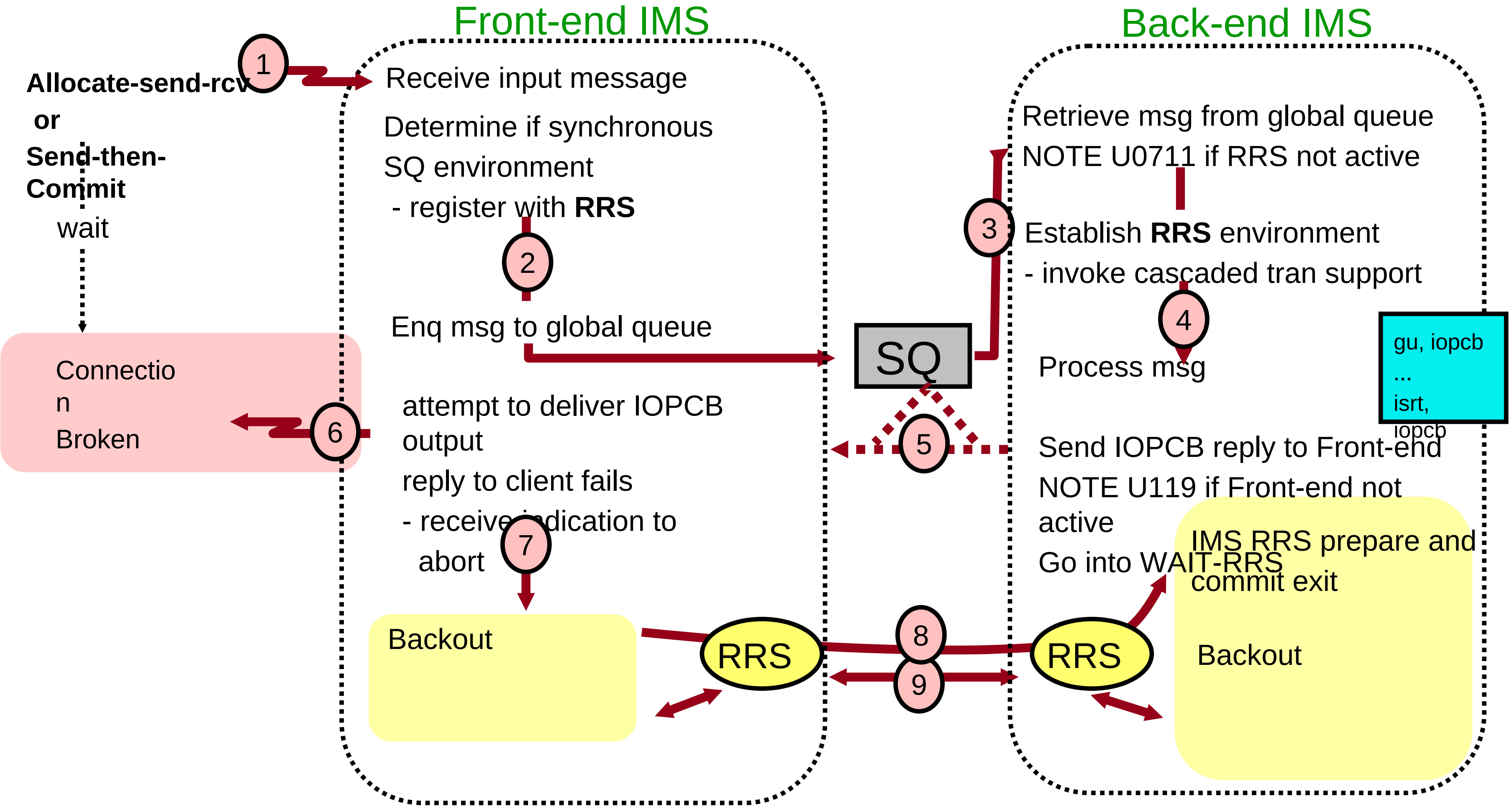
IMS RRS prepare and commit exit

Syncpoint ph 1 / ph 2

Deallocate
or
Send

10 Deallocate or get next msg

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



Front-end IMS

Back-end IMS

Allocate-send-rcv
or
Send-then-Commit
wait

Connection Broken

Receive input message

Determine if synchronous SQ environment
- register with **RRS**

Enq msg to global queue

attempt to deliver IOPCB output
reply to client fails
- receive indication to abort

Backout

SQ

Retrieve msg from global queue
NOTE U0711 if RRS not active

Establish **RRS** environment
- invoke cascaded tran support

Process msg

gu, iopcb
...
isrt, iopcb

Send IOPCB reply to Front-end
NOTE U119 if Front-end not active
Go into WAIT-RRS

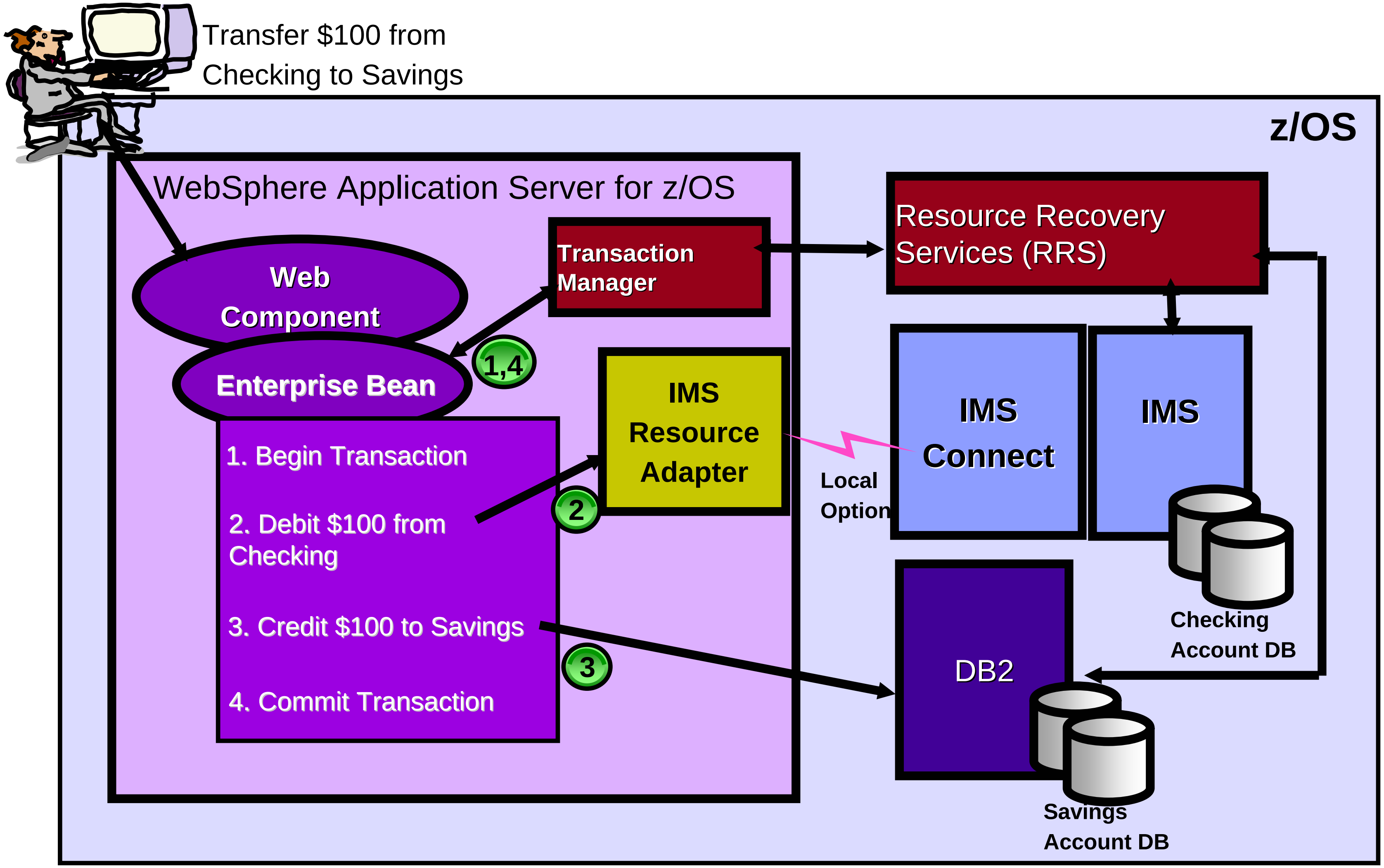
IMS RRS prepare and commit exit

Backout

IMS TM Resource Adapter and IMS Connect

- IMS TM Resource Adapter works with RRS, IMS Connect and the J2EE transaction manager to make consistent changes to IMS databases and other critical resources.
- To participate in the two-phase commit processing with IMS/RRS, IMS TM Resource Adapter uses the IMS OTMA send-then-commit (synclevel-SYNCPT) protocol to send protected transactions. To enable global transactions in J2EE applications, TCP/IP interface or local-option interface of IMS Connect can be used by the IMS TM Resource Adapter.

Transaction Support with WAS for z/OS with Local Option



IBM Information On Demand 2007

TIPS of Running IMS with RRS

Act.Right.Now.

IBM INFORMATION ON DEMAND 2007
October 14 – 19, 2007
Mandalay Bay
Las Vegas, Nevada

Time out values for J2EE client applications

- If a resource manager takes too long to implement the changes requested in the transaction submitted by the client application, or if the transaction involves a large number of resource managers, the client application can appear hung. To prevent excessive wait times or hung client applications, set a timeout value for your client applications that is acceptable to your installation. Keep in mind, however, that if your time-out value is to short, and IMS processes a message on its queues from a client that has already timed out, the IMS application running in the dependent region could abend.

Create more IMS dependent regions when using RRS

- If an IMS application program is updating a database at the request of an RRS-managed transaction, then the application program remains scheduled in the dependent region for the duration of the RRS two-phase commit process, which may or may not be delayed by other participating subsystems. While an application program remains scheduled in a dependent region, the dependent region has a WAIT-RRS status and the application program cannot process other transactions.
- Defining additional IMS dependent regions can increase the transaction throughput rate by reducing the likelihood that subsequent transactions will have to wait for an application program to be scheduled in a free dependent region.

May need to process transactions in one IMS

- A global transaction can contain multiple IMS transactions with the **same** IMS transaction code. Due to restrictions related to the transfer of database locks and buffers, all of the IMS transactions contained in a global transaction must be executed in one IMS system.

Minimize the impact on performance of offloading log data to DASD

- RRS logs many activities related to the RRS events and services. The RRS data is written to a log stream. When the storage used for the log stream reaches a user-defined high threshold, a percentage of the total storage used, the z/OS system logger offloads the data to DASD. The System logger offloads only enough data to bring the total storage used down to the low threshold and then stops offloading.
- Correctly setting the high and low thresholds for offloading in the z/OS LOGR policy can minimize the performance impact of offloading. Setting the high threshold too low or the low threshold too high can result in offloading too frequently, which can slow performance. On the other hand, setting the high threshold too high could lead to the log stream storage becoming full before the system logger can offload to create more space. If a coupling facility or staging data set space for a log stream becomes completely full, the system logger rejects all write requests until the log data can be offloaded to DASD log data sets.

Minimize the impact on performance of offloading log data to DASD…

- Also, if you are using a coupling facility log stream, determining the optimum size of the coupling facility structure helps in handling the offload processing rate and in keeping more log data available on the coupling facility.
- If you are using a DASD-only log stream, the size of the staging data set is another factor affecting the RRS performance. In general, make the size of the staging data set large enough to hold all of the log data in the coupling facility.
- Finally, consider not using the RRS Archive log stream, which will reduce the RRS logging activities.

Diagnosing problems related to RRS and IMS

- The first step in resolving any problem is to determine where the problem actually lies. For example, if an IMS transaction managed by RRS hangs, the IMS Connect client application might time out the transaction request, but the UR might remain hung in IMS, RRS, or both. To resolve this situation, you will want to determine where the UR is hung. IMS Connect, IMS, and RRS together provide a variety of methods for determining where the problem lies.

Diagnosing problems related to RRS and IMS…

- In IMS Connect you can issue the IMS Connect MVS command VIEWUOR to display the status of a specific UR or of all URs currently being managed by IMS Connect.
- In IMS itself, you can issue the command /DISPLAY UOR ALL to display the status of all URs in IMS. Be aware that IMS may have any number of different types of URs other than those associated with IMS.
- From the RRS ISPF panels you can view even more comprehensive RRS information for resource managers, work managers, URs, and log streams.
- If needed, take an SVC dump of the IMS control regions and dependent regions, the IMS Connect address space, the RRS address space, and the z/OS system logger address space.

Diagnosing problems related to RRS and IMS…

- You can also generate valuable diagnostic information by using the trace functions of IMS Connect recorder, the RRS component, and other associated components such as the IMS RRST table trace and IMS OTMT table trace.

Diagnosing problems related to RRS and IMS…

- Running RRS and IMS with the RRS CTRACE and the IMS OTMT and RRST traces is recommended. Also of particular use in diagnosing problems are the IMS Connect internal traces HWSTRACE and BPETRACE, which can be viewed in a dump of IMS Connect in IMS Version 9. In IMS Version 10, you do not have to create a dump to view the HWSTRACE and BPETRACE traces.
- For IMS Connect traces, the default size of the IMS Connect trace tables can be small for diagnosing some problems, including those related to RRS. To ensure that your trace tables capture as much useful information as possible, increase their sizes in the IMS BPECFG PROCLIB member. For example, the following size specifications, which define 200 4 K pages, should be sufficient:
- TRCLEV=(*,HIGH,BPE,PAGES=200)
- TRCLEV=(*,HIGH,HWS,PAGES=200)

Diagnosing problems related to RRS and IMS…

- An additional source of diagnostic information is the IMS online log data set (OLDS), which can also provide detailed RRS diagnostic information.
- If an IMS dependent region hangs for a long time with a WAIT-RRS status, you can try issuing the IMS command “/STOP REGION reg# ABDUMP transaction-name” to rollback the transaction and free the dependent region or you can use the RRS ISPF panels to abort the UR and to get out of the wait state.

More tips…

- Upgrade to z/OS 1.6 to have the “RRS Restart anytime/anywhere enhancement”.
- If RRS needs to be shutdown, please try first to soft stop before to force or MVS cancel.
  - SETRRS CANCEL command prior to a system IPL result in a cleaner system recovery
- The “RRS=“ startup parameter in IMS procedure could be considered to activate/deactivate the RRS function for IMS.

Summary

- z/OS Resource Recovery allows an application program to make consistent changes to multiple protected resources**.**
- OTMA/APPC synchronous synclevel = syncpt message, "protected transaction", is supported by using RRS.
- IMS batch application programs support 2-phase commit syncpoint processing using RRS.
- IMS V8 OTMA/APPC synchronous messages are allowed to be processed by any backend IMS using RRS.
- You may need to tune the system for running IMS with RRS so that the IMS transactions can be processed efficiently.