Session: 17901
MQ for z/OS – The Insider Story

Paul Kettley
PLM for Messaging on z
paulk@uk.ibm.com
Agenda

• Transactions
• Queue Managers
• Resource Managers
• What does a transaction look like?
• Scenario Walkthrough
• Summary
Agenda

• Transactions

• Queue Managers

• Resource Managers

• What does a transaction look like?

• Scenario Walkthrough

• Summary
Transactions

STATE 1

CHANGE

STATE 2

Message Transaction

Transaction Contract

ACID
Agenda

• Transactions

• **Queue Managers**

• Resource Managers

• What does a transaction look like?

• Scenario Walkthrough

• Summary
Core Technology - Logging and Buffering

Conveyor Belt Log

Pigeon Holes Pageset
Agenda

• Transactions

• Queue Managers

• **Resource Managers**

• What does a transaction look like?

• Scenario Walkthrough

• Summary
Building Blocks - Resource Managers

- Connection Manager
- Buffer Manager
- Data Manager
- Log Manager
- Recovery Manager
- Message Manager
- Lock Manager
- Redo Manager
- CF Manager
- MQI Manager
- MQOPEN Manager
- Data Manager
- Buffer Manager

- Commit
- Backout
- Rlse
- UR
- Ckpt
- MQOPEN
- UR
- Redo
- Undo
- Ckpt
Handling Applications - Connection Manager

MQCONN
MQOPEN
MQPUT
...

Control Blocks

Logs
Data
Other RMs

Application

SYNCPOINT
COORDINATOR

PHASE 1
PHASE 2

Connection manager

EOT
EOM
Controlling the MQI and MQSC - Message Manager

MQI
MQOPEN
Message Manager
Command, Runtime, Group servers

Application
MQOPEN
MQCLOSE
MQGET
MQPUT
MQPUT1
MQINQ
MQSET

MQSC
CONSOLE
QSG

SYSTEM.CLUSTER

Objects

BASE

Plus
Validation
Consistency
Triggering
Get-Wait

MQOPEN

...Locking
...Security
Controlling Messages and Objects - Data Manager

Log
Messages
Pagesets 1-99
Space group
Scavenger
SubQueue
Msg pointers
Pageset 0
Objects
Space group
Space group
REDO
UNDO
Pagesets 1-99
Messages
Space group
Space group
Bringing High Performance - Buffer Manager

**DEFINE BUFFPOOL(bpid)**

BUFFERS (nnnn)

** DEFINE PSID(psid) **

BUFFPOOL(bpid)

** NO FORCE **

** FORCE DIRTY PAGES > 3 CKPTS **

** CKPT, LRU, STEAL, WRITE AHEAD **

** NO FORCE **

** OLDEST **

---

** Buffers (nnnn) **

** Pageset **

** Read pageset **

** Write pageset **

** Checkpoint **

** Force dirty pages > 3 ckpts **
Local Message Queue Storage

- Head
- Start Scan
- Tail

Subqueue

M1 (del)    M2(del)
100MB
up to 100MB
very long message

M3
short messages
4K

M4
long message
4MB

M4
very long message
100MB

MD First part of text
Next part of text
Text up to 4MB

... up to 100MB

4MB + of text
Next part of text
Text up to 8MB
Handling Transactions - Recovery Manager

** TRANSACTION STATES **

1 PHASE
COMMIT
BEGIN
IN
FLIGHT
IN
COMMIT
COMMITTED

BACKOUT
BEGIN
IN
FLIGHT
IN
BACKOUT
BACKED OUT

2 PHASE
COMMIT
BEGIN
IN
FLIGHT
IN
COMMIT
COMMITTED

log
LOG LOAD
CKPT

LOG
LOAD

RESTART/RECOVERY

pagesets

log
Providing Logging Interfaces - Log Manager

• Log read and write functions

• Log Shunting

• Multiple active log data sets and archive log datasets

• Archive inventory management

• Duplexed for reliability

• “Bootstrap” file
  • End of log location
  • Archive inventory

• Various Utilities
Agenda

• Transactions
• Queue Managers
• Resource Managers
• What does a transaction look like?
• Scenario Walkthrough
• Summary
Examining Transactions - A Log Print

00000000D569  URID(00000000D569)  RM(RECOVERY)  TYPE( START UR )
      ****  00640024 00200001 03000000 0000D569 00000000 D545
0000  00240000 000D000 00000000 00000700 00000000 00000000 00000000 00000D6C4
0020  D6E6C4C1 4040B5B4 8FA08793 02864040 40404040 4040C2C1 E3C3C840 4040D6C4
0040  D6E6C4C1 40400000 00000000 0000

00000000D5CD  URID(00000000D569)  RM(DATA)  LRID(00000000.00000E01)  TYPE( UNDO REDO )
      ***  002A0064 0600000F C9000000 0000D569 00000000 D569
0000  00000000 00000E01 00040326 00000001 00000000 0000

00000000D5F7  URID(00000000D569)  RM(DATA)  LRID(00000001.00000201)  TYPE( UNDO REDO )
      ***  0026002A 06000008 C9000000 0000D569 00000000 D5CD
0000  00000001 00000201 00000000 00000001 00000000 0000

00000000D61D  URID(00000000D569)  RM(RECOVERY)  TYPE( START COMMIT1 )
      ****  007C0026 02000002 03000000 0000D569 00000000 D5F7
0000  00240000 000D000 00000000 00000700 00000000 00000000 00000000 00000404
0020  40400404 40400000 00000000 00000000 00000000 00000000 00000000 00000000
0040  00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
0060  00000000 0000

00000000D699  URID(00000000D569)  RM(RECOVERY)  TYPE( PHASE 1 TO 2 )
      ****  00240024 02000010 03000000 0000D569 00000000 D61D
0000  00240000 000D000 00000000 00000000 00000000 0000

00000000D6BD  URID(00000000D569)  RM(RECOVERY)  TYPE( END COMMIT2 )
      ****  00240024 02000010 03000000 0000D569 00000000 D699
0000  00240000 000D000 00000000 00000000 0000
Concurrency and Isolation - Lock Manager

MQOO_INPUT_SHARED  
MQOO_INPUT_EXCLUSIVE

MQOPEN vs DELETE QLOCAL

API Request
Transaction
Lifetime

MQOPEN

Exclusive  Exclusive
Managing Shared Messages - Coupling Facility Manager

DEFINE QL(queue)
CFSTRUCT(structure)
QSGDISP(SHARED)
Shared Message Queue Storage Using CF List Structures

List Structure

MQGET

MQCMIT

MQPUT

MQGET

MQCMIT

keys

Data

State | Priority | Time | Qmgr
--- | --- | --- | ---
MsgId/CorrelId

WRITE
READ
MOVE
DELETE
MONITOR

committed puts

uncommitted puts

queue details

uncommitted gets

expired messages
Agenda

- Transactions
- Queue Managers
- Resource Managers
- What does a transaction look like?
- Scenario Walkthrough
- Summary
### Scenario - MQPut to a Triggered Queue

<table>
<thead>
<tr>
<th>Application</th>
<th>Message Manager</th>
<th>Data Manager</th>
<th>Buffer Manager</th>
<th>Recovery Manager</th>
<th>Log Manager</th>
<th>Lock Manager</th>
</tr>
</thead>
<tbody>
<tr>
<td>MQOPEN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>ACQUIRE LOCK</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOCATE QUEUE IN HASH TABLE</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SECURITY</td>
<td>BASE NAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACQUIRE HANDLE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MQPUT</td>
<td>USE HANDLE</td>
<td>LOCATE PAGE TO HOLD MSG</td>
<td>BUFFER PAGE</td>
<td>START UR</td>
<td>LOG RECORDS</td>
<td></td>
</tr>
<tr>
<td>CHECK TRIGGER RULES</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>LOG RECORDS</td>
<td></td>
</tr>
<tr>
<td>MQCMIT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>FORCE LOG</td>
<td>RELEASE LOCKS</td>
</tr>
</tbody>
</table>
Scenario - MQGet from a Queue

<table>
<thead>
<tr>
<th>Application</th>
<th>Message Manager</th>
<th>Data Manager</th>
<th>Buffer Manager</th>
<th>Recovery Manager</th>
<th>Log Manager</th>
<th>Lock Manager</th>
</tr>
</thead>
<tbody>
<tr>
<td>MQOPEN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOCATE QUEUE</td>
<td></td>
<td>ACQUIRE LOCK</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SECURITY</td>
<td>IN HASH TABLE</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>BASE NAME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ACQUIRE HANDLE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MQGET</td>
<td>USE HANDLE</td>
<td>FIND MSG</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(INDEX / NEXT)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>BUFFER PAGE</td>
<td></td>
<td>START UR</td>
<td>LOG RECORDS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>LOG RECORDS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MQCMIT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>FORCE LOG</td>
<td>RELEASE LOCKS</td>
</tr>
</tbody>
</table>

MQOPEN

LOCATE QUEUE IN HASH TABLE

SECURITY BASE NAME ACQUIRE HANDLE

MQGET

USE HANDLE FIND MSG (INDEX / NEXT)

BUFFER PAGE START UR LOG RECORDS

MQCMIT

FORCE LOG RELEASE LOCKS
Summary

• Delivers transactional messaging
  • Enables robust business applications

• Complex, but well organised
  • Adapters, Address spaces, Resource Managers

• Designed for throughput, availability and scalability
  • Logging, Buffering, Locking, Communications
Session: 17901
MQ for z/OS – The Insider Story

Paul Kettley
PLM for Messaging on z
paulk@uk.ibm.com