Understanding IMS Locking

Rich Lewis
IBM

March 2, 2011
8566
**IMS Locking White Paper**

- “IMS Locking with Program Isolation or the IRLM”
  - Rich Lewis
  - Published in 2009
    - [www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101535](http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101535)
    - Contains detailed information about IMS locking

- This presentation is based on information in the white paper
  - The white paper should be used with this presentation
Agenda

- Lock managers
- Lock compatibility matrices
- Full function locks
- Fast Path locks
- Lock timeouts
- Deadlocks
- Design advice
- Space for lock control blocks
- PI vs. IRLM
- Locking Reports
**Lock Managers**

- IMS has three lock managers
  - Program Isolation (PI)
    - Does not support data sharing
    - Locks are managed by the IMS online system
  - IRLM
    - May be used with or without data sharing
    - IRLM is a separate address space
    - Multiple IRLMs are used with data sharing across LPARs
  - Fast Path lock manager
    - Used without data sharing
    - Fast Path also uses PI or IRLM
      - Required for deadlock detection
### Lock Compatibility

#### PI and FP lock compatibility matrix

<table>
<thead>
<tr>
<th>Lock Level</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 – read</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>2 – share</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>3 – update</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>4 - exclusive</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
</tbody>
</table>

#### IRLM lock compatibility matrix

<table>
<thead>
<tr>
<th>Lock Level</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>6</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 – read</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>3 – erase</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>4 – share</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>6 – update</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>8 - exclusive</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
</tbody>
</table>

#### Lock level names are often confusing
- Reads may require a read, share, update, or exclusive lock
- Locks for updates sometimes use read locks
**Full Function Locks**

- **Database Record Lock**
  - Requested when a database record is accessed
    - HDAM and PHDAM lock the RAP
  - Used to serialize access to database records
  - Level depends on the PROCOPT
    - PROCOPT=G PI level 2 IRLM level 4
      - Readers may be positioned in a database record concurrently
    - PROCOPT=update PI level 3 (or 1) IRLM level 6
      - Updaters have exclusive access to database record
  - Released
    - If not update, when PCB position is moved to another database record
    - If update, held until sync point
      - PI demotes level 3 to level 1 when positioned is moved off the record and root is not updated
**Full Function Locks**

- **Database Record Lock**
  - HDAM and PHDAM
    - Lock is on the RAP
    - Another reason to have more RAPs than roots
      - Rule of thumb: \# RAPs > 2 \times \# roots
  - Often, this is the most important lock
    - “Control” records often produce lock conflicts
Full Function Locks

- Segment Lock
  - Used only with PI
  - Segment lock is always requested for updates to dependent segments
    - Used to serialize access to updated dependent segments
    - Updates include updates to pointers in the segment (to other segments)
    - HISAM lock is for overflow logical records
  - Level: PI level 3
  - Released
    - At sync point

- If database record lock is held at level 1 (by another program)
  - Some dependent segment is locked at level 3
  - Segment lock is tested when dependent segment is accessed
    - Test waits if lock is held but does not get the lock
**IRLM vs. PI**

- PI may provide more concurrency
  - Allows access to non-updated segments in updated database record
**IRLM vs. PI**

- PI may provide more concurrency
  - Non-shared lock of root makes the entire database record inaccessible
**IRLM vs. PI**

- PI may provide more concurrency
  - Non-shared lock of dependent makes all of its children inaccessible
**IRLM vs. PI**

- PI may provide more concurrency
  - Non-shared lock of twin makes following twins inaccessible
**IRLM vs. PI**

- PI may provide more concurrency
  - Non-shared lock of twin does not make preceding twins inaccessible
**IRLM vs. PI**

- PI may provide more concurrency
  - This effect may be small
    - What is the probability of two transactions or BMPs accessing different branches in the same database record at the same time?

```
Root Key=5674748585
  ├── Dependent 1
  │    ├── Dependent 2
  │    │    ├── Dependent 3
  │    │    └── Dependent 3
  │    └── Dependent 4
  └── Dependent 5
       ├── Dependent 6
       └── Dependent 7
            └── Dependent 8
```
**Full Function Locks**

- **Block Lock**
  - Used only with block level data sharing (SHARELVL=2 or 3)
  - Requested when a block is updated
  - Used to serialize updates from different IMS systems
    - Requested with private attribute
      - Cannot be shared across different IMS systems (no matter what level)
  - Level for OSAM and ESDS is always IRLM level 4
  - Level for KSDS (primary and secondary indexes)
    - Inserts and replaces  IRLM level 4
    - Deletes  IRLM level 3
    - CI/CA splits  IRLM level 6
  - Released
    - At sync point

*Block locks are only for updates! (ISRT, DLET and REPL calls)*
**IMS Locking**

**Full Function Locks**

- **Block Lock**
  - Block locks are shared within an IMS system
    - Unless there is a delete with insert/replace of a KSDS record or a CI/CA split
  - Block lock conflicts typically occur for updates in a small database or small part of a database
    - Secondary index with high insert/delete activity to small range of records
      - Records in the same CI
      - Often due to keys based on current time
  - Small database with “control” records
    - Statistics maintenance, etc.
**Full Function Locks**

- **Busy Lock**
  - Requested to serialize activity to a data set
    - Update to KSDS with block level data sharing
      - Insert IRLM level 8
      - Non-insert IRLM level 2
    - Open and close of data set PI level 4 IRLM level 8
    - Creation of new block in data set PI level 4 IRLM level 8
  - Released
    - At end of operation (open, close, update, etc.)

- Lock waits are rarely a problem with busy locks

- The number of lock request may be important for data sharing
  - CF accesses for the lock structure for index updates
Fast Path Locks

- CI Lock
  - Similar to database record lock for full function
  - Requested when a CI is read into a buffer
  - Used to serialize access to segments in a CI
  - Level depends on the PROCOPT
    - PROCOPT=G  FP level 1  IRLM level 2
    - PROCOPT=update  FP level 4  IRLM level 8
  - Released
    - With update
      - By output thread (sync point with VSO)
    - Without update
      - By sync point or when buffer is stolen

No locks for SDEP CIs
Fast Path Locks

- **UOW Lock**
  - Only used when HSSP or High Speed Reorg (HSR) is active
    - Requested instead of a CI lock by HSSP and High Speed Reorg
    - Requested in addition to CI lock by others
  - Level depends on the PROCOPT
    - Non-HSSP or HSR request: FP level 1, IRLM level 2
    - HSSP or HSR request: FP level 4, IRLM level 8
  - Released
    - Non-HSSP, non-HSR request
      - When all locks on CIs in UOW are released
    - HSSP request
      - If update by output thread, if no update by sync point
    - HSR request
      - At end of reorg of UOW
**Lock Time Outs**

- PI and Fast Path lock managers *never* time out (i.e. end) a lock request.
- IRLM has capability to time out a lock request:
  - IRLM TIMEOUT parameter
    
    ```f
    F irlmproc,SET,TIMEOUT=seconds,imssubsystemname
    ```
    
    - Controls the reporting of “long locks” for an IMS system using the IRLM
    - It does **not** time out a lock request
    - It drives an IMS LOCKTIME process to check on time outs
  - IMS LOCKTIME parameter controls time outs of locks with IRLM
    - DFSVSMxx or DFSVSAMP parameter
      - LOCKTIME=(mtime,maction,btime,baction)
    - May be changed with UPDATE IMS SET(LOCKTIME(…) command
      
      ```sql
      UPDATE IMS SET(LOCKTIME(MSG(mtime),MSGOPT(maction),
          BMP(btime),BMPOPT(baction),TELLIRLM(Y|N))
      ```
Lock Time Outs

- If wait exceeds IMS LOCKTIME value, the waiter’s lock request ends
  - IMS “shoots the victim”
    - If ABEND is specified for ‘maction’ or ‘baction’
      - U3310 abend and IMS TM input message is discarded
    - If STATUS is specified for ‘maction’ or ‘baction’
      - ‘BD’ status code is returned for call which caused lock wait
  - The “bad guy” is probably the holder of the lock
Deadlock Detection

- Fast Path lock manager does not detect deadlocks
  - When a lock request waits, Fast Path passes information to the other lock manager (PI or IRLM)
    - Other lock manager does deadlock detection
- PI checks for deadlocks whenever a lock request waits
- IRLM checks for deadlocks on a timer basis
  - IRLM parameter: DEADLOK=(local,global)
    - Local is the time between deadlock detection cycles
    - Global value is ignored
      - Every local cycle is a global cycle
    - A wait must exist through two cycles before IRLM checks for a deadlock
      - With local value of 1 second, deadlock could last 2 seconds before detection
    - Reasonable values for local or 1 second or less
Deadlock Detection

- Deadlocks may be created with IMS and non-IMS resources
  - CICS applications with IMS and VSAM
  - IMS TM applications with IMS DB and DB2
  - DB2 stored procedures with IMS DB and DB2
  - Example:
    - Tran A holds IMS lock X
    - Tran B holds DB2 lock Y
    - Tran A requests DB2 lock Y and waits
    - Tran B requests IMS lock X and waits – DEADLOCK!
  - These deadlocks are only resolved by time outs
    - Usually, resolved by the “other” resource manager, not IMS
      - IMS only times out lock requests when LOCKTIME value for IMS is specified with IRLM
Handling Deadlock Victims

- **Actions for deadlock victims**
  - MPP, JMP, IFP, BMP, or JBP: Abend U0777
    - MPP, JMP, and IFP messages are rescheduled
  - APPC CPIC driven or modified standard application: Abend U0123
  - CICS task: CICS ADCD abend
  - ODBA thread: AIB “system failure” return code x’00000108’, reason code x’00000244’ and error extension code x’10000309’ and thread is terminated

- **Exceptions of abend for deadlock ‘victim’**
  - INIT STATUS GROUPB
    - Back out occurs and program receives a ‘BC’ status code
  - Non-message driven BMP or JBP with Fast Path PCB
    - Back out occurs and program receives an ‘FD’ status code
  - Deadlock during sync point processing with MSDBs
    - Back out and reprocessing occur
Design Advice

- Minimize PROCOPT values
  - PROCOPT=A produces “non-shared” level locks

- Take frequent checkpoints
  - But don’t create a logging problem by checkpointing too much user data
    - Such as all of working storage

- Be wary of communications during a sync interval
  - OTMA commit mode 1 with synclevel=syncpoint or synclevel=confirm
  - APPC with synclevel=syncpoint or synclevel=confirm
  - Synchronous callout (ICAL)
    - Default timeout for ICAL is 10 seconds
    - Application may set any value
  - Communications delays will likely cause locking problems
**Design Advice**

- **Try to limit high frequency updates to any record**
  - “Control” records can be a problem
    - For example, “next invoice number”
    - Possible solutions:
      - Delay calls to the record until the end of the transaction
      - Use multiple records, one for each series of numbers
      - Use non-sequential numbers, such as choosing numbers at random
    - Databases with only a few database records are often problems

- **Provide free space in (P)HIDAM with block level data sharing**
  - Without free space all inserts go to end of data set causing block lock conflicts
**Design Advice**

- **By wary of PROCOPT=E**
  - PROCOPT=E on root
    - Schedules program exclusively for the database in an IMS subsystem
      - Does not affect scheduling or locking in other IMS subsystems
    - If not data sharing
      - No locks are used for the database
    - If data sharing
      - All locks for database are held until sync point
  - PROCOPT=E on a dependent segment
    - Schedules program exclusively for the segment in an IMS subsystem
    - Locks are used for the database records
    - No PI locks are used for the segment
  - PROCOPT=E on root is sometimes used to allow BMPs with infrequent checkpoints to run
Design Advice

- Tune the system and applications
  - Use lots of database buffers
  - *The faster an application runs, the shorter the time it holds locks!*
Space for Lock Control Blocks

- **PI**
  - Each locked resource uses 24 byte control block
  - Each holder of a resource lock uses 24 byte control block
  - Rule of thumb: Each lock requires 48 bytes
  - PI lock control block storage location:
    - With Fast Path: ECSA
    - Without Fast Path: Extended private of DLI SAS address space
    - Without Fast Path or DLI SAS: Extended private of control region
  - PI storage is limited by PIMAX execution parameter
    - If PIMAX is not specified, limited by second subparameter of CORE= on IMSCTF macro
    - PIINCR specifies the increments in which storage is acquired
Space for Lock Control Blocks

- **IRLM**
  - Each lock requires about 540 bytes in 64-bit storage of IRLM address space
  - Space may be limited by the z/OS MEMLIMIT parameter on the job or job step

- **Coupling Facility Lock Structure**
  - Each lock protecting an update uses an entry in the lock record list
    - All block locks
    - Level 6 database record locks
    - Level 8 Fast Path CI and UOW locks
  - Record list entries are about 250 bytes
  - Goal for lock table: 1000 entries per held lock
    - Provides false contention rate of 0.1%
    - Entries are typically 2 bytes
    - Therefore, about 2000 bytes per held lock
Space for Lock Control Blocks

- When lock space is exhausted
  - PI: U0775 abend of requestor
  - IRLM: U3300 abend of requestor
  - Lock structure record list: U3307 of requestor

- Excessive space for locks
  - Usually caused by BMPs
    - Usually a very small subset of BMPs
LOCKMAX Usage

- LOCKMAX parameter limits the number of locks held by a dependent region or batch job at any time
  - Specified in 1000s
  - Specified in PSBGEN statement of PSB
  - Specified as region parameter
    - Overrides PSB specification
- U3301 of program when LOCKMAX reached
- Log records contain the maximum number actually used
  - x’37’ and x’5937’ for online systems
  - x’41’ for batch data sharing
- Recommendation:
  - Specify LOCKMAX in all dependent regions
  - Specify it in test systems
**PI vs. IRLM**

- IRLM required for block level data sharing
- PI has shorter path length
  - May not be significant in total application path length
- PI has maximum of 63 waiters
  - 64th waiter receives U2478 abend
    - MPP or JMP is rescheduled
    - IRLM has no limit on the number of waiters
- IRLM has “long locks” capability
  - Reports locks which wait for a long time
- Lock timeout capability requires IRLM
Locking Reports
**IMS Monitor**

- PROGRAM I/O Report

<table>
<thead>
<tr>
<th>PSBNAME</th>
<th>PCB NAME</th>
<th>IWAITS</th>
<th>TOTAL</th>
<th>MEAN</th>
<th>MAXIMUM</th>
<th>DDN/FUNC</th>
<th>MODULE</th>
</tr>
</thead>
<tbody>
<tr>
<td>AZLACL</td>
<td>RZCMA001</td>
<td>2</td>
<td>3419</td>
<td>1709</td>
<td>1991</td>
<td>PI</td>
<td>RZCMA001...1</td>
</tr>
</tbody>
</table>

- REGION IWAIT Report

<table>
<thead>
<tr>
<th><strong>REGION</strong></th>
<th>45 OCCURRENCES</th>
<th>..........IWAIT TIME......</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TOTAL</td>
<td>MEAN</td>
</tr>
<tr>
<td>16</td>
<td>20959</td>
<td>1309</td>
</tr>
<tr>
<td>19</td>
<td>48901</td>
<td>2573</td>
</tr>
</tbody>
</table>

- Notes:
  - “PI” appears for both PI and IRLM
  - Segment code is “1” except for PI segment locks

- You can examine these reports to see if you have a lot of locks and to determine their average wait times
**KBLA IRLM Lock Trace Analysis Utilities (DFSKLTx0)**

- Report produced from IRLM lock trace
  - Excellent source of overall information on lock waits

### Suspended IRLM Lock Requests Summary Report - Wait Time Order Page 001

<table>
<thead>
<tr>
<th>Database DS Name</th>
<th>Lock Req Id</th>
<th>Wait Count</th>
<th>Not Int Count</th>
<th>Total Time</th>
<th>Average Time</th>
<th>Maximum Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>BFLMSGY3 01</td>
<td>8628</td>
<td>115</td>
<td>110</td>
<td>9.198</td>
<td>0.079</td>
<td>2.76</td>
</tr>
<tr>
<td>BFLMSGY7 01</td>
<td>8452</td>
<td>102</td>
<td>98</td>
<td>4.813</td>
<td>0.047</td>
<td>4.36</td>
</tr>
<tr>
<td>BFLMSGP 01</td>
<td>15862</td>
<td>181</td>
<td>169</td>
<td>4.401</td>
<td>0.024</td>
<td>0.64</td>
</tr>
<tr>
<td>BFLSUMP 01</td>
<td>3929</td>
<td>40</td>
<td>37</td>
<td>3.703</td>
<td>0.092</td>
<td>2.39</td>
</tr>
<tr>
<td>BCMTLIRD 09</td>
<td>1153</td>
<td>1</td>
<td>1</td>
<td>3.400</td>
<td>3.400</td>
<td>3.40</td>
</tr>
</tbody>
</table>

**Wait Count:** Includes internal latch waits and lock waits

**Not Int Count:** Not including internal latch waits
**KBLA Lock Trace Detailed Print Program (DFSKLTC0)**

- Report produced from IRLM lock trace
  - Detailed information about each wait
    - Voluminous!

<table>
<thead>
<tr>
<th>Start Time</th>
<th>End Time</th>
<th>Elapsed Time</th>
<th>Type</th>
<th>Num</th>
<th>Type</th>
<th>Lvl</th>
<th>DB</th>
<th>RBA/HASH</th>
<th>S</th>
<th>RCFB</th>
<th>TRAC</th>
<th>Type</th>
<th>Num</th>
<th>Time</th>
<th>Seq#</th>
</tr>
</thead>
<tbody>
<tr>
<td>16:06:09.723</td>
<td>16:06:09.724</td>
<td>0.000</td>
<td>F</td>
<td>100</td>
<td>BIDP</td>
<td>4</td>
<td>BCICINY</td>
<td>01</td>
<td>099DE001</td>
<td>P</td>
<td>CPR</td>
<td>0000</td>
<td>08C0</td>
<td>ISRT</td>
<td>001</td>
</tr>
<tr>
<td>16:06:09.727</td>
<td>16:06:09.727</td>
<td>0.004</td>
<td>F</td>
<td>100</td>
<td>BIDP</td>
<td>4</td>
<td>BCICINY</td>
<td>01</td>
<td>099DE001</td>
<td>P</td>
<td>CPKF</td>
<td>0000</td>
<td>08C0</td>
<td>ISRT</td>
<td>001</td>
</tr>
<tr>
<td>16:06:09.567</td>
<td>16:06:09.952</td>
<td>0.385</td>
<td>G</td>
<td>067</td>
<td>FPCI</td>
<td>8</td>
<td>BCMTRMD</td>
<td>08</td>
<td>00024CE0</td>
<td>F</td>
<td>K</td>
<td>0440</td>
<td>08F0</td>
<td>9F73</td>
<td></td>
</tr>
<tr>
<td>16:06:10.170</td>
<td>16:06:10.170</td>
<td>0.004</td>
<td>G</td>
<td>067</td>
<td>BIDP</td>
<td>4</td>
<td>BAGTX1P</td>
<td>01</td>
<td>32117800</td>
<td>P</td>
<td>CPKF</td>
<td>0840</td>
<td>08F0</td>
<td>ISRT</td>
<td>001</td>
</tr>
<tr>
<td>16:06:10.209</td>
<td>16:06:10.242</td>
<td>0.032</td>
<td>G</td>
<td>100</td>
<td>FPCI</td>
<td>8</td>
<td>BGLACAD</td>
<td>06</td>
<td>005203A0</td>
<td>F</td>
<td>K</td>
<td>0440</td>
<td>08F0</td>
<td>9A67</td>
<td></td>
</tr>
<tr>
<td>16:06:10.354</td>
<td>16:06:10.354</td>
<td>0.004</td>
<td>L</td>
<td>122</td>
<td>FPCI</td>
<td>8</td>
<td>BCMRDAD</td>
<td>10</td>
<td>00053AEO</td>
<td>F</td>
<td>K</td>
<td>0440</td>
<td>2080</td>
<td>D030</td>
<td></td>
</tr>
<tr>
<td>16:06:10.397</td>
<td>16:06:10.398</td>
<td>0.001</td>
<td>L</td>
<td>122</td>
<td>FPCI</td>
<td>8</td>
<td>BCMRDAD</td>
<td>11</td>
<td>00143820</td>
<td>F</td>
<td>K</td>
<td>0440</td>
<td>2080</td>
<td>DFDE</td>
<td></td>
</tr>
<tr>
<td>16:06:10.438</td>
<td>16:06:10.438</td>
<td>0.000</td>
<td>L</td>
<td>122</td>
<td>FPCI</td>
<td>8</td>
<td>BCMRDAD</td>
<td>13</td>
<td>0009E000</td>
<td>F</td>
<td>K</td>
<td>0440</td>
<td>2080</td>
<td>EB9D</td>
<td></td>
</tr>
<tr>
<td>16:06:10.959</td>
<td>16:06:10.992</td>
<td>0.032</td>
<td>L</td>
<td>038</td>
<td>BIDP</td>
<td>6</td>
<td>BCMTRPP</td>
<td>01</td>
<td>0412E8804</td>
<td>P</td>
<td>PKF</td>
<td>0000</td>
<td>2080</td>
<td>ISRT</td>
<td>001</td>
</tr>
<tr>
<td>16:06:11.011</td>
<td>16:06:11.012</td>
<td>0.001</td>
<td>L</td>
<td>122</td>
<td>FPCI</td>
<td>8</td>
<td>BCMRDAD</td>
<td>11</td>
<td>00168360</td>
<td>F</td>
<td>K</td>
<td>0440</td>
<td>2080</td>
<td>D79D</td>
<td></td>
</tr>
</tbody>
</table>

F – false contention
G – global contention
L – local contention
**RMF II - IRLM Long Lock Detection Report**

- Shows lock waits greater than IRLM LOCKTIME value
  - Also shows holders of lock and other waiters for lock

<table>
<thead>
<tr>
<th>State</th>
<th>Type</th>
<th>Lock_Name</th>
<th>IMS_ID</th>
<th>Recovery_Token</th>
<th>PSB_Name</th>
<th>Elap_Time</th>
<th>CICS_ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>CF</td>
<td>Structure</td>
<td>ACOXLOCK</td>
<td>at 07/28/2006 13:02:10 Deadlock Cycle 00002EC7</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TOP</td>
<td>BMP</td>
<td>09C943CFA7800101D70000000000000000</td>
<td>DFSSAM</td>
<td>06:04</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BLOCKER</td>
<td>ACO3</td>
<td>ACO3</td>
<td>00000000300000000000</td>
<td>IRLMTOP</td>
<td>06:09</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TOP</td>
<td>BMP</td>
<td>09C3614505800101D70000000000000000</td>
<td>DFSSAM</td>
<td>06:09</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BLOCKER</td>
<td>ACO1</td>
<td>ACO1</td>
<td>00000000600000000000</td>
<td>IRLMTOP</td>
<td>06:09</td>
<td></td>
<td></td>
</tr>
<tr>
<td>WAITER</td>
<td>BMP</td>
<td>09C3614505800101D70000000000000000</td>
<td>DFSSAM</td>
<td>05:52</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACO2</td>
<td>ACO2</td>
<td>ACO2</td>
<td>00000000800000000000</td>
<td>IRLMWTA</td>
<td>05:52</td>
<td></td>
<td></td>
</tr>
<tr>
<td>WAITER</td>
<td>BMP</td>
<td>09C943CFA7800101D70000000000000000</td>
<td>DFSSAM</td>
<td>05:42</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACO2</td>
<td>ACO2</td>
<td>ACO2</td>
<td>00000000900000000000</td>
<td>IRLMWTA</td>
<td>05:42</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**DFSERA30 Deadlock Report**

- Provides detailed information about each deadlock

<table>
<thead>
<tr>
<th>RESOURCE DMB-NAME</th>
<th>LOCK-LEN</th>
<th>LOCK-NAME</th>
<th>WAITER FOR THIS RESOURCE IS VICTIM</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMLDDCDB</td>
<td>08</td>
<td>7EB22000834A01D7</td>
<td>KEY FOR RESOURCE IS FROM DELETE WORK AREA</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IMS-NAME</th>
<th>TRAN/JOB</th>
<th>PSB-NAME</th>
<th>PCB--DBD</th>
<th>PST#</th>
<th>RGN</th>
<th>CALL</th>
<th>LOCK</th>
<th>LOCKFUNC</th>
<th>STATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMS2</td>
<td>TRLDDC1</td>
<td>CMLDDCDB</td>
<td>CMLDDCDB</td>
<td>00003</td>
<td>MPP</td>
<td>DLET</td>
<td>GBIDP</td>
<td>22400318</td>
<td>04-P</td>
</tr>
<tr>
<td>IMS1</td>
<td>USMEED2</td>
<td>CMLDDCDB</td>
<td>--------</td>
<td>00007</td>
<td>MPP</td>
<td>----</td>
<td>-----</td>
<td>--------</td>
<td>04-P</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RESOURCE DMB-NAME</th>
<th>LOCK-LEN</th>
<th>LOCK-NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMLDDCDB</td>
<td>08</td>
<td>7EB22000843A01D7</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IMS-NAME</th>
<th>TRAN/JOB</th>
<th>PSB-NAME</th>
<th>PCB--DBD</th>
<th>PST#</th>
<th>RGN</th>
<th>CALL</th>
<th>LOCK</th>
<th>LOCKFUNC</th>
<th>STATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMS1</td>
<td>USMEED2</td>
<td>CMLDDCDB</td>
<td>--------</td>
<td>00007</td>
<td>MPP</td>
<td>GET</td>
<td>GRIDX</td>
<td>30400358</td>
<td>06-P</td>
</tr>
<tr>
<td>IMS2</td>
<td>TRLDDC1</td>
<td>CMLDDCDB</td>
<td>--------</td>
<td>00003</td>
<td>MPP</td>
<td>----</td>
<td>-----</td>
<td>--------</td>
<td>06-P</td>
</tr>
</tbody>
</table>

**DEADLOCK ANALYSIS REPORT - END OF REPORT**
### Coupling Facility Usage Summary – Structure Summary

<table>
<thead>
<tr>
<th>STRUCTURE TYPE</th>
<th>STRUCTURE NAME</th>
<th>STATUS</th>
<th>CHG</th>
<th>SIZE</th>
<th>STOR</th>
<th>REQ</th>
<th>#</th>
<th>REQ</th>
<th>UTIL</th>
<th>SEC</th>
<th>TOT/CUR</th>
<th>TOT/CUR</th>
<th>TOT/CUR</th>
<th>LOCK</th>
<th>XI'S</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOCK</td>
<td>MMHL_IMSIRLM</td>
<td>ACTIVE</td>
<td></td>
<td>34M</td>
<td>71551</td>
<td>0.1</td>
<td>0.1</td>
<td>59.63</td>
<td></td>
<td></td>
<td>62K</td>
<td>0</td>
<td>8389K</td>
<td>N/A</td>
<td></td>
</tr>
</tbody>
</table>

**COUPLING FACILITY NAME = CF01**

TOTAL SAMPLES (AVG) = 240 (MAX) = 240 (MIN) = 240
Coupling Facility Usage Summary – Structure Summary

<table>
<thead>
<tr>
<th>Structure Type</th>
<th>Name</th>
<th>Status</th>
<th>CHG</th>
<th>Alloc Size</th>
<th>% Alloc</th>
<th>% CF</th>
<th>% All</th>
<th>Avg Size</th>
<th>LST/DIR Entries</th>
<th>DATA Entries</th>
<th>LOCK Entries</th>
<th>DIR REC Entries</th>
<th>XI's</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOCK</td>
<td>MMHL_IMSIRLM</td>
<td>ACTIVE</td>
<td>34M</td>
<td>34M</td>
<td>100.00</td>
<td></td>
<td></td>
<td>34M</td>
<td>28</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Record List:
- AVG: 59.63
- LST/DIR: 62K
- DATA: 0
- LOCK: 8389K

Lock Table:
- AVG: 28
- DATA: 0
- LOCK: 170
- XI's: N/A
### RMF Coupling Facility Reports

#### Coupling Facility Structure Activity

<table>
<thead>
<tr>
<th>Structure Name = MMHL_IMSIRLM</th>
<th>Type = LOCK</th>
<th>Status = ACTIVE</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>System Name</th>
<th>TOTAL</th>
<th>AVG/SEC</th>
<th># REQ</th>
<th>% OF REQ</th>
<th>-SERV TIME (MIC)</th>
<th>CUR TIME (MIC)</th>
<th># OF</th>
<th>% OF</th>
<th>REASON</th>
<th># DELAYED REQUESTS</th>
<th>% OF DELAYED REQUESTS</th>
<th>DELAYED TIME (MIC)</th>
<th>AVG TIME (MIC)</th>
<th>STD_DEV</th>
<th>CONTENTIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSY</td>
<td>69K</td>
<td>57.96</td>
<td>0.1</td>
<td>108.6</td>
<td>387.3</td>
<td>97.1</td>
<td>0.0</td>
<td>9.3</td>
<td>PR WT</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>5.4</td>
<td>15</td>
</tr>
<tr>
<td>CHNGD</td>
<td>1</td>
<td>0.0</td>
<td>0.0</td>
<td>51.1</td>
<td>9.3</td>
<td>0.0</td>
<td>0.0</td>
<td>PR CMP</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>SYSN</td>
<td>406</td>
<td>0.34</td>
<td>0.0</td>
<td>51.1</td>
<td>9.3</td>
<td>0.0</td>
<td>0.0</td>
<td>PR CMP</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>CHNGD</td>
<td>0</td>
<td>0.0</td>
<td>0.0</td>
<td>51.1</td>
<td>9.3</td>
<td>0.0</td>
<td>0.0</td>
<td>PR CMP</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>SYSO</td>
<td>1014</td>
<td>0.84</td>
<td>1.4</td>
<td>18.0</td>
<td>7.8</td>
<td>0.0</td>
<td>0.0</td>
<td>PR SCH</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>CHNGD</td>
<td>0</td>
<td>0.0</td>
<td>0.0</td>
<td>51.1</td>
<td>9.3</td>
<td>0.0</td>
<td>0.0</td>
<td>PR CMP</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>TOTAL</td>
<td>71551</td>
<td>59.63</td>
<td>1.4</td>
<td>18.0</td>
<td>7.8</td>
<td>0.0</td>
<td>0.0</td>
<td>PR SCH</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>1</td>
</tr>
</tbody>
</table>

**Request Details**

- **SYSL**
  - Sync: 584 (97.1% of total, avg 8.2 mic)
  - Async: 0 (0.0% of total, avg 0.0 mic)
- **SYSN**
  - Sync: 406 (88.2% of total, avg 7.5 mic)
  - Async: 12 (0.0% of total, avg 9.3 mic)
- **SYSO**
  - Sync: 1014 (88.2% of total, avg 7.8 mic)
  - Async: 0 (0.0% of total, avg 0.0 mic)
- **SYSM**
  - Sync: 69547 (97.1% of total, avg 9.7 mic)
  - Async: 103 (1.4% of total, avg 9.3 mic)

**External Requests**

- **SYSL**
  - Sync: 0 (0.0% of total, avg 0.0 mic)
  - Async: 0 (0.0% of total, avg 0.0 mic)
- **SYSN**
  - Sync: 0 (0.0% of total, avg 0.0 mic)
  - Async: 0 (0.0% of total, avg 0.0 mic)
- **SYSO**
  - Sync: 0 (0.0% of total, avg 0.0 mic)
  - Async: 0 (0.0% of total, avg 0.0 mic)
- **SYSM**
  - Sync: 103 (0.1% of total, avg 387.3 mic)
  - Async: 0 (0.0% of total, avg 0.0 mic)

**Total**

- Sync: 71551 (97.1% of total, avg 7.8 mic)
- Async: 115 (1.6% of total, avg 366.8 mic)

**Contention Details**

- **SYSL**
  - Included in Async: 0 (0.0% of total, avg 0.0 mic)
  - CHNGD: 0 (0.0% of total, avg 0.0 mic)
- **SYSN**
  - Included in Async: 0 (0.0% of total, avg 0.0 mic)
  - CHNGD: 0 (0.0% of total, avg 0.0 mic)
- **SYSO**
  - Included in Async: 0 (0.0% of total, avg 0.0 mic)
  - CHNGD: 0 (0.0% of total, avg 0.0 mic)
- **SYSM**
  - Included in Async: 15 (0.2% of total, avg 387.3 mic)
  - CHNGD: 1 (0.0% of total, avg 0.0 mic)

**Total**

- CHNGD: 16 (0.2% of total, avg 0.0 mic)

---

**Request Statistics**

- **Req Total**: 723
  - Deferred: 7
  - Cont: 7
  - False Cont: 0

- **Req Total**: 723
  - Deferred: 7
  - Cont: 7
  - False Cont: 0

**Total**

- **Req Total**: 82K
  - Deferred: 88
  - Cont: 87
  - False Cont: 16
# RMF Coupling Facility Reports

## Coupling Facility Structure Activity

<table>
<thead>
<tr>
<th>STRUCTURE NAME = MMHL_IMSIRLM</th>
<th>TYPE = LOCK</th>
<th>STATUS = ACTIVE</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>SYSTEM NAME</th>
<th>TOTAL</th>
<th># REQ</th>
<th>% OF ALL</th>
<th>SERV TIME (MIC) -</th>
<th>REQ DEFERRED</th>
<th>EXTERNAL REQUEST</th>
</tr>
</thead>
</table>
Summary

- Locking affects IMS performance
- Locking is influenced by
  - Database design
  - Application program design
  - Syncpoint frequencies
- There are multiple sources of information about locking
  - These may be used to discover and address locking problems