New York Oracle Users Group 2005

Performance Monitoring & Tuning for RAC (9i & 10g)

Rich Niemiec, TUSC

__Special Thanks:__ **Janet Bacon, Mike Ault, Madhu Tumma, Murali Vallath, Randy Swanson, Rick Stark , Sohan DeMel Erik Peterson and Kirk McGowan**

Audience Knowledge

Goals

- Overview of RAC & RAC Tuning
- Target RAC tips that are most useful

Non-Goals

- Learn ALL aspects of RAC

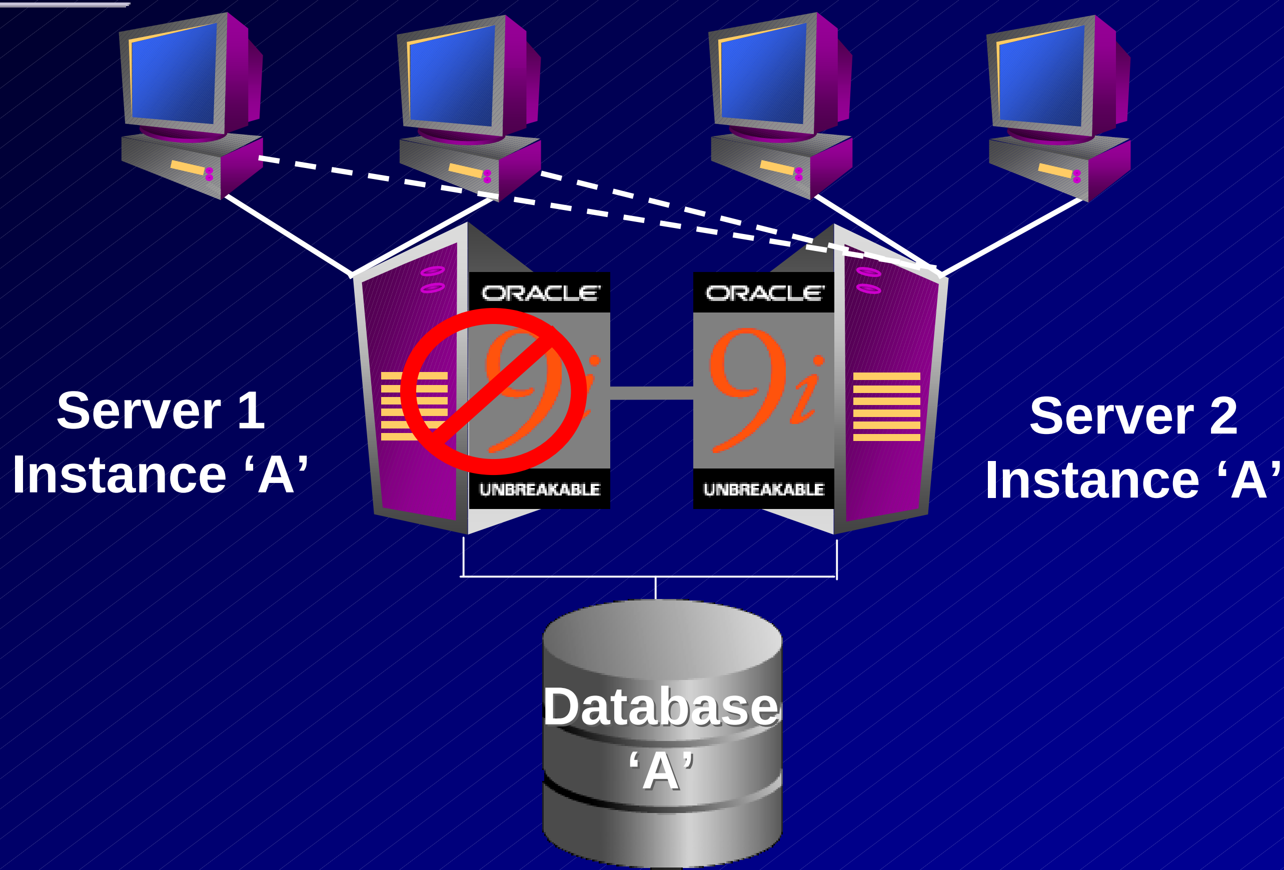
Overview

- RAC Overview
- Tuning the RAC Cluster Interconnect
- Monitoring the RAC Workload
- Monitoring RAC specific contention
- What's new in 10g
- Summary

Overview of Oracle9*i* RAC

- Many instances of Oracle running on many nodes
- All instances share a single physical database and have common data & control files
- Each instance has its own log files and rollback segments (UNDO Tablespace)
- All instances can simultaneously execute transactions against the single database
- Caches are synchronized using Oracle's Global Cache Management technology (Cache Fusion)

Real Application Clusters



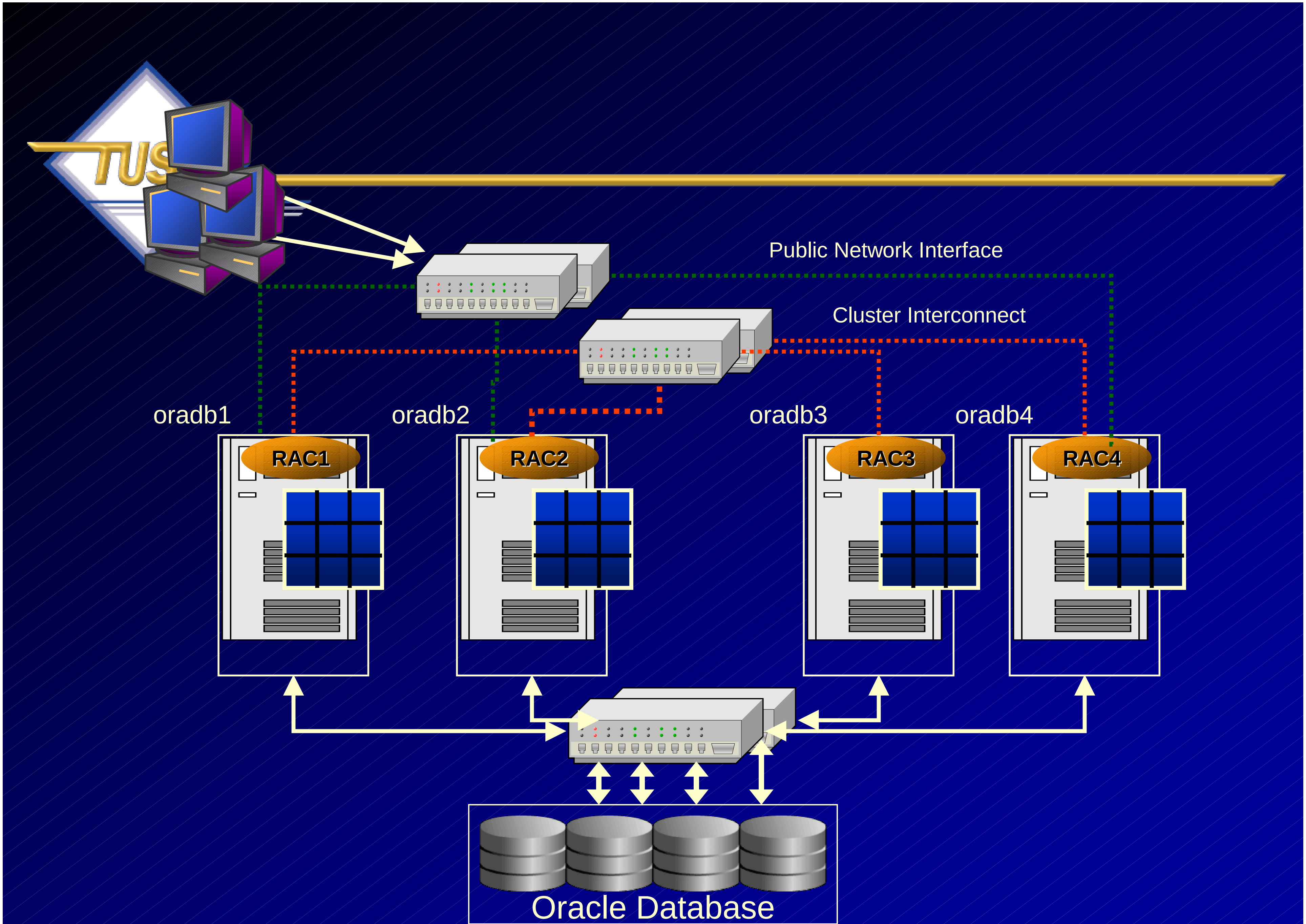
SERVER failure: your database remains available

Protect from SERVER failures

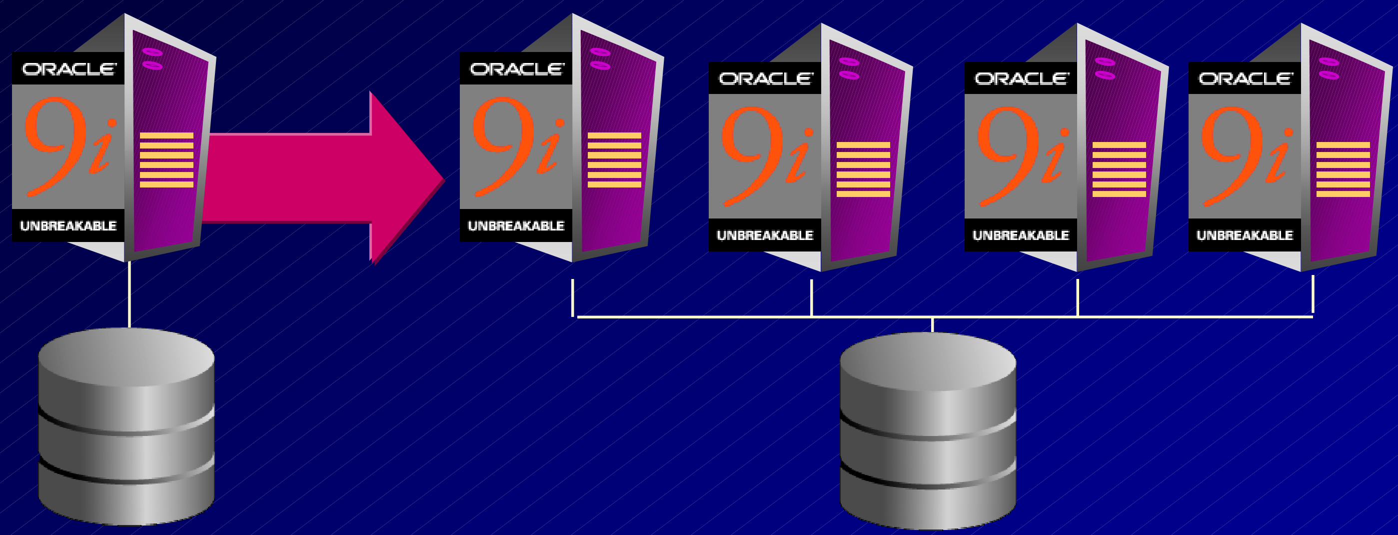
Availability

Identify all failure points

- Interconnect
- Public interface
- HBA’s
- Brocade switches
- Fiber Optics to Storage
- Node
- Instance


Public Network Interface
Cluster Interconnect
oradb1
oradb2
oradb3
oradb4
RAC1
RAC2
RAC3
RAC4
Oracle Database

Oracle9*i* Database Clusters



Start small, grow incrementally

Scalable AND highly available

NO downtime to add servers and disk

10g Grid Computing

Mainframe Model

Partitioning of one large server

Built with high quality, high cost parts

Complete, integrated software

High quality of service at high cost

Grid Computing Model

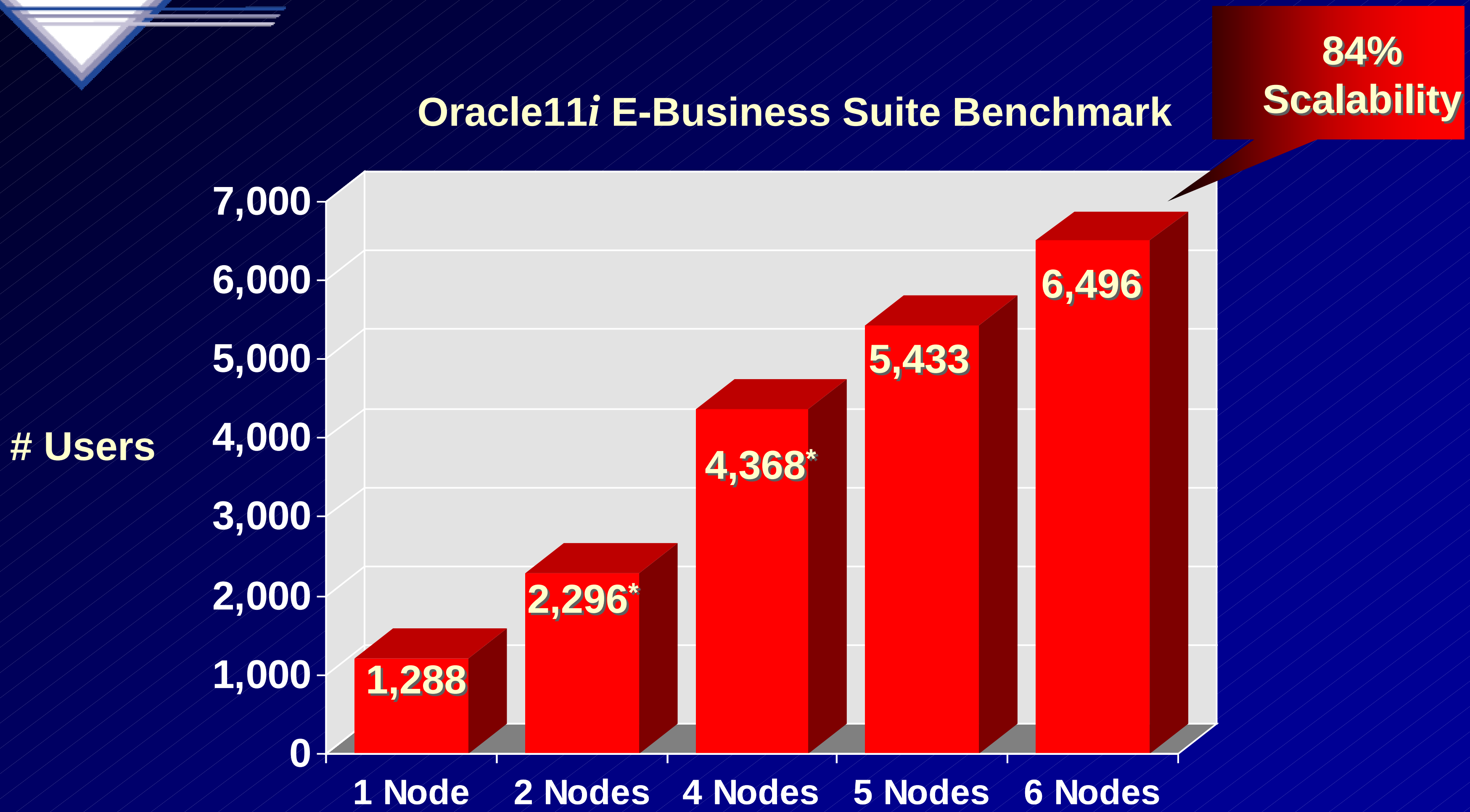
Coordinated use of many small servers

Built with low cost, standard, modular parts

Open, Complete, integrated software

High quality of service at low cost

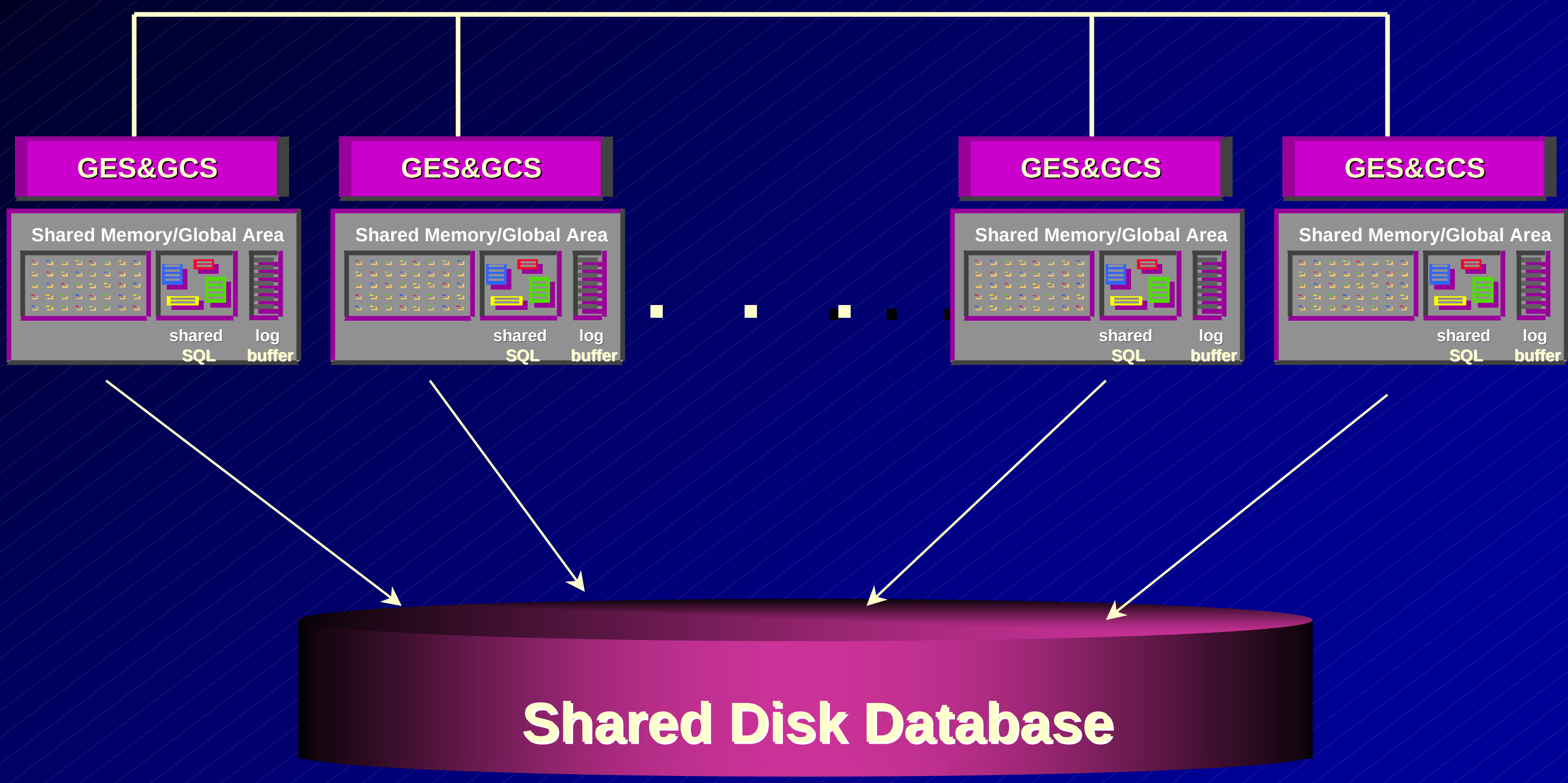
E-Business Suite Scalability with Oracle9*i* RAC



Running on HP Computers ***Audited**

RAC Interconnect & Block Coordination

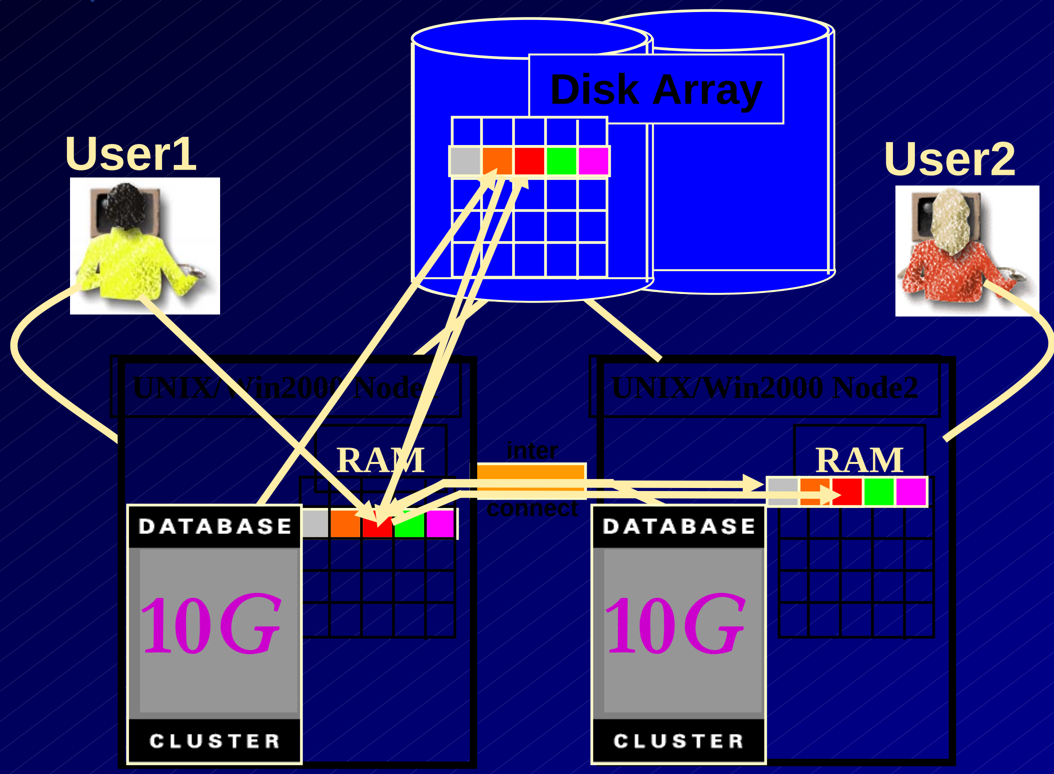
RAC Architecture
Shared Data Model



Cache Fusion

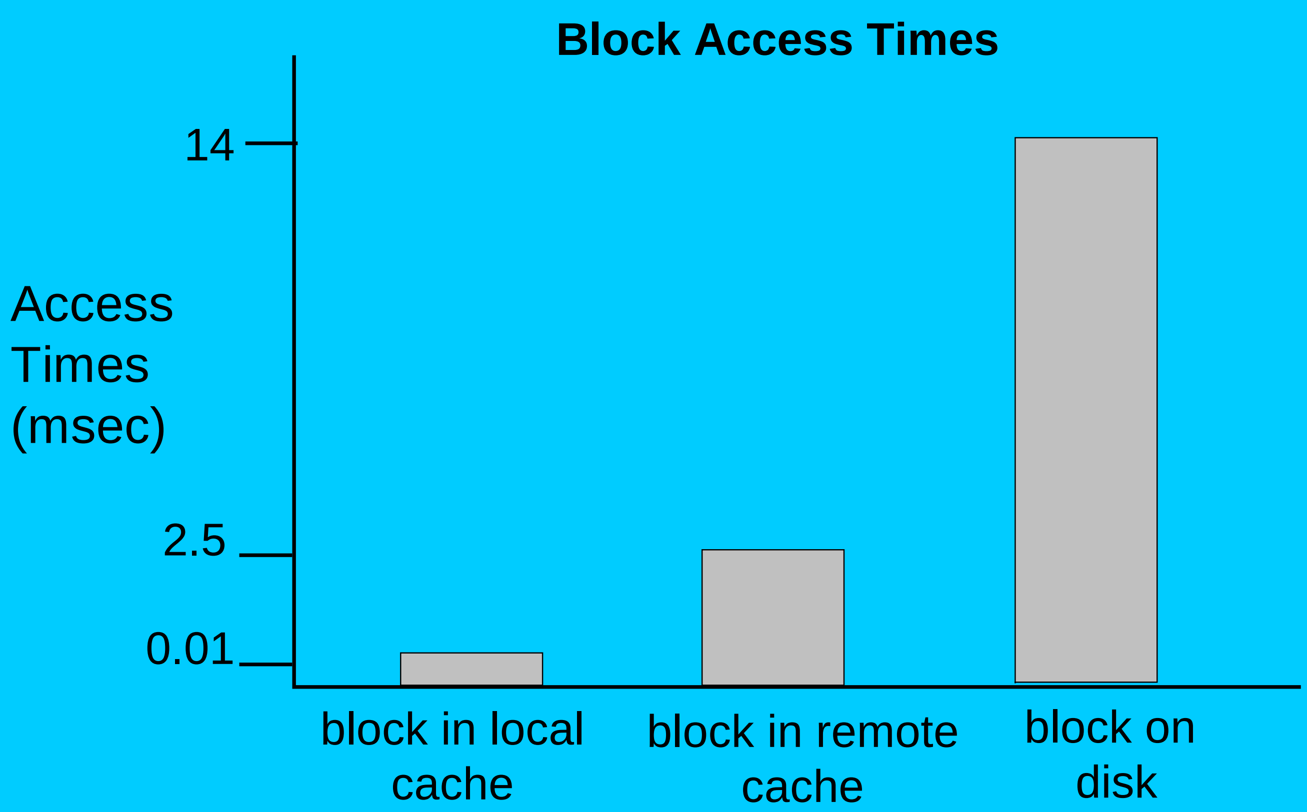
- Cache Fusion helps provide transparent scalability in a Real Application Clusters database
- The algorithms enable transportation of block images between instances
- The algorithms enable transportation of block images between instances
- Cache Fusion services track the current location and status of resources
- Directory structures within the SGA of each instance store the resource information

Real Applications Clusters - Cache Fusion



1. User1 queries data
2. User2 queries same data - via interconnect with no disc I/O
3. User1 updates a row of data and commits
4. User2 wants to update same block of data – 10g keeps data concurrency via interconnect

RAC Internals



Source : Oracle World Presentation

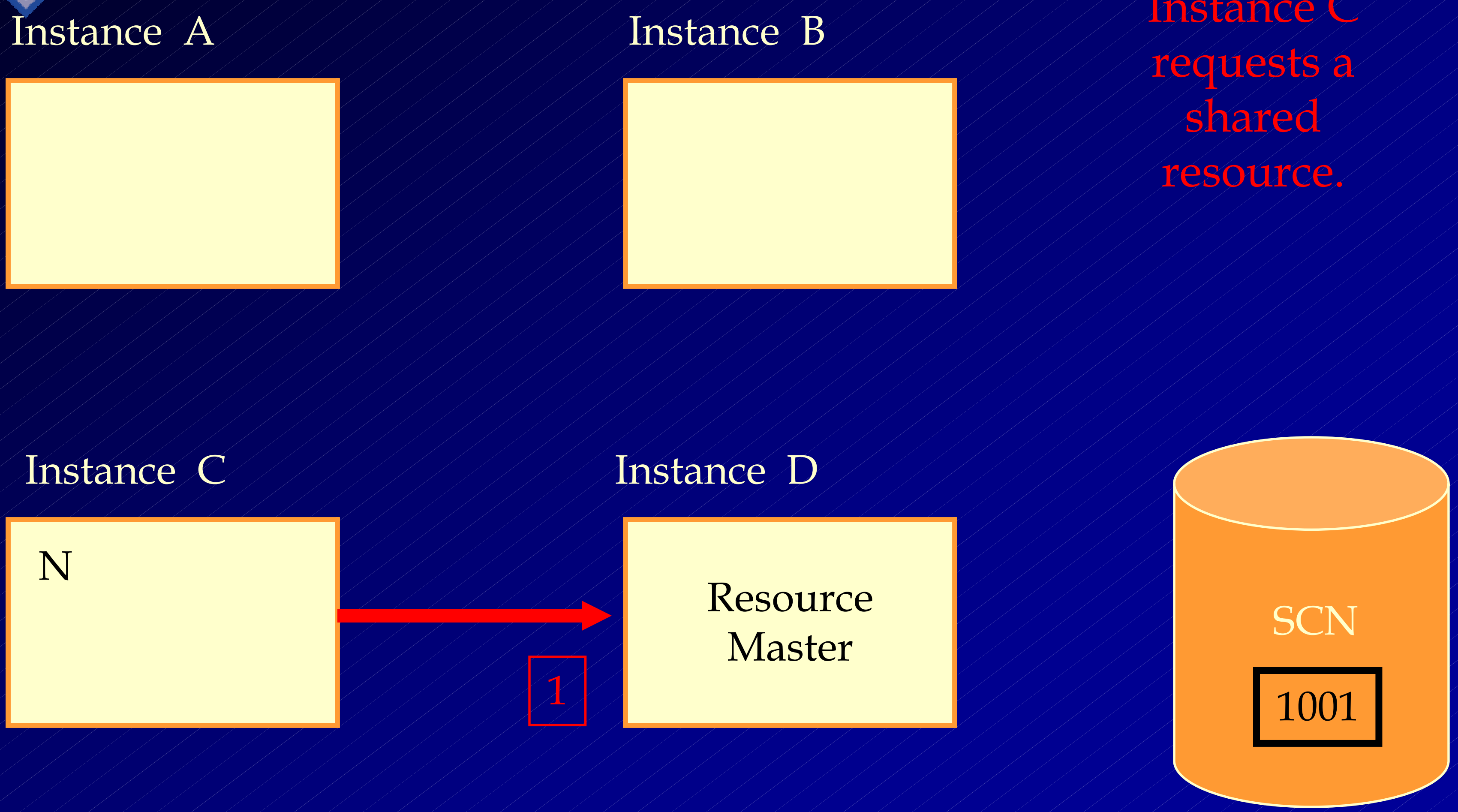
Interconnect Characteristics

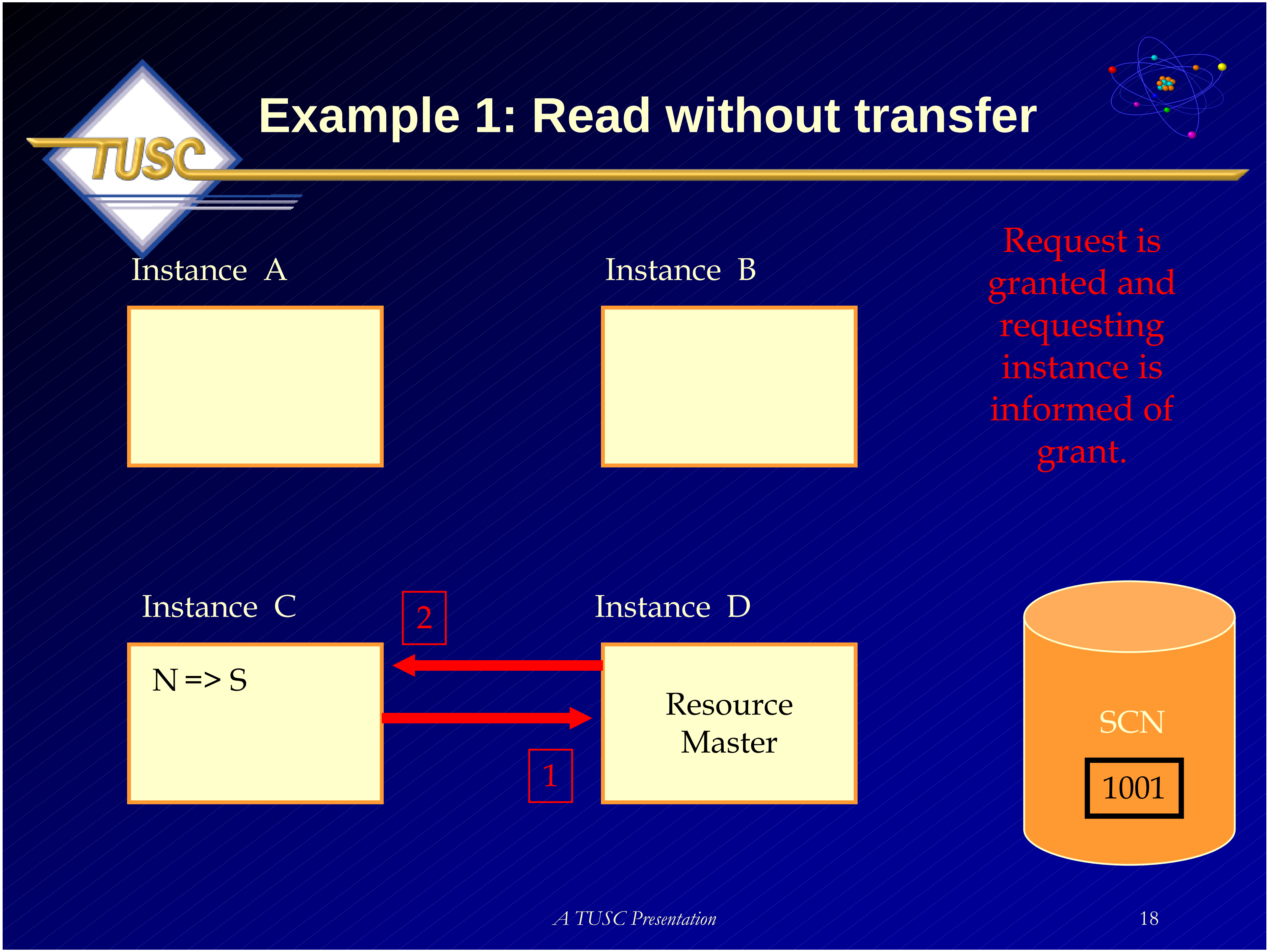
- Low latency for short messages
- High speed and sustained data rates for large messages
- Low host CPU utilization per message
- Flow control, error control and heartbeat continuity monitoring
- Host interfaces to interact directly with host processes (‘OS bypass’)
- Switch networks that scale well

Measurement	Typical SMP Bus	Memory Channel	Myrinet	Sun SCI	Gb Ether
Latency (μs)	0.5	3	7 to 9	10	100
CPU overhead (μs)	< 1	< 1	< 1	low	higher
Messages per sec (millions)	> 10	> 2			< 0.1
Hardware Bandwidth (MB/sec)	> 500	> 100	~ 250	~70	~ 50

Example 1: Read without transfer

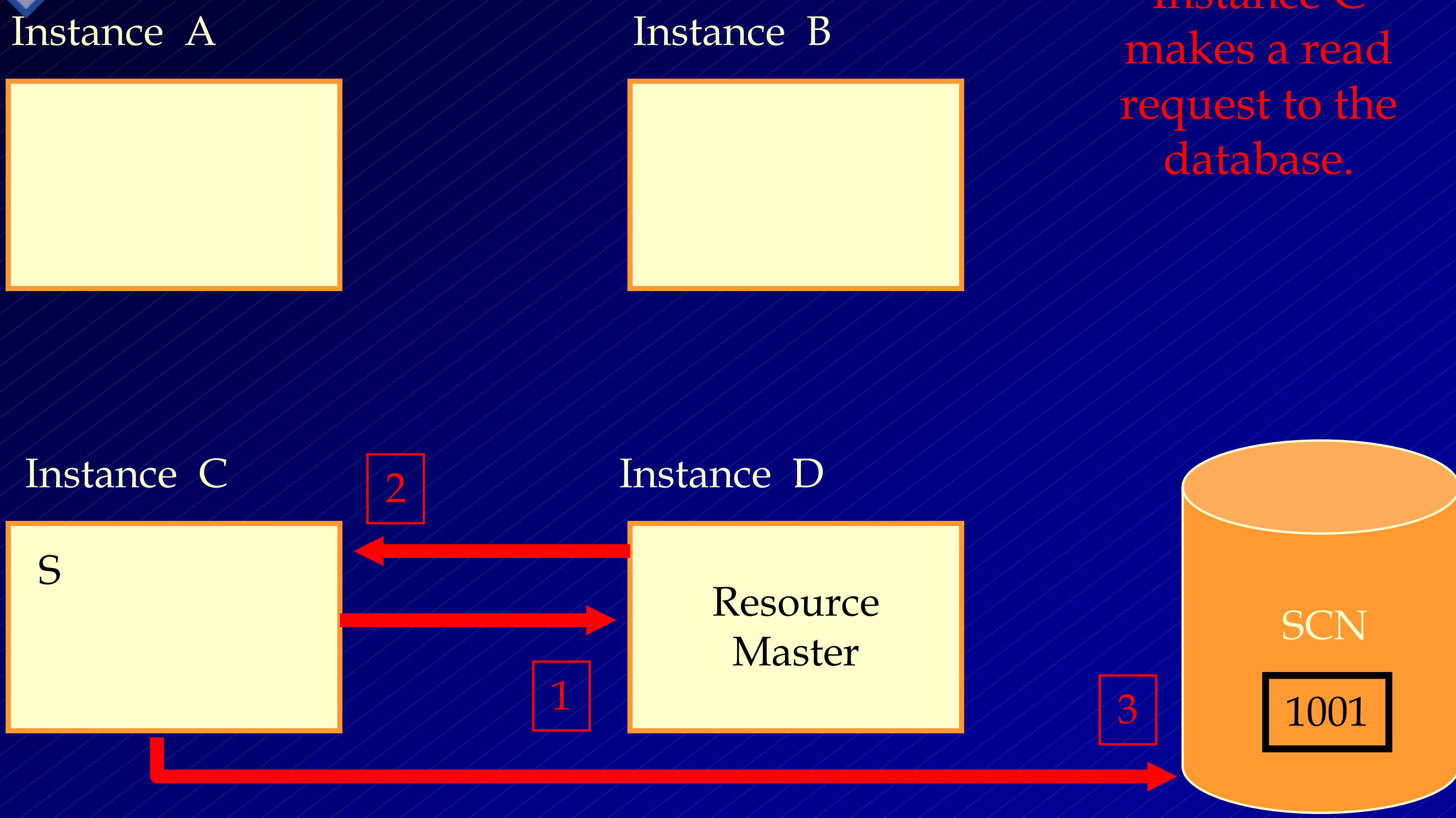
Instance C requests a shared resource.



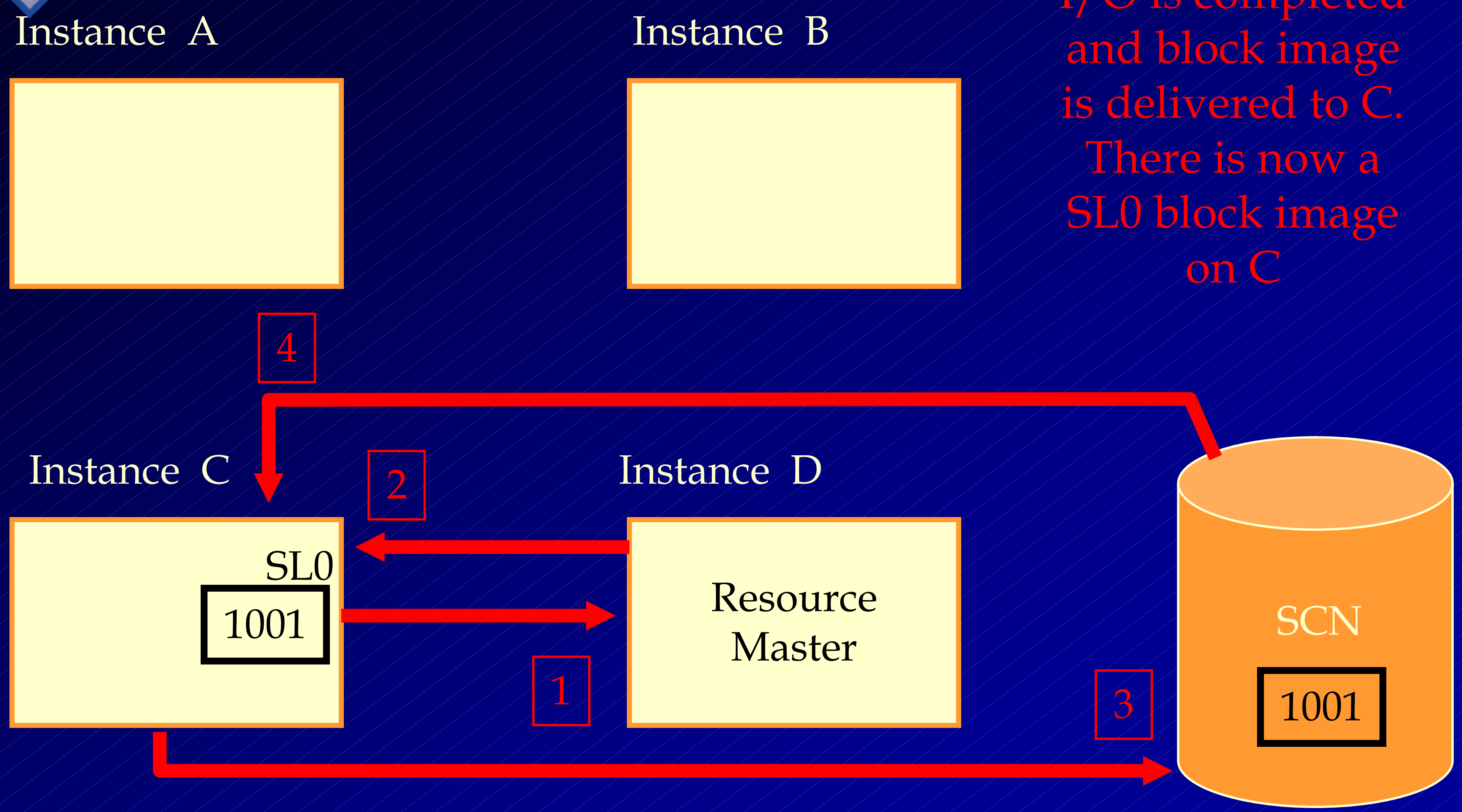

Example 1: Read without transfer
Instance A
Instance B
Request is granted and requesting instance is informed of grant.
Instance C
2
N => S
Instance D
Resource Master
1
SCN
1001
A TUSC Presentation
18

Example 1: Read without transfer

Instance C makes a read request to the database.

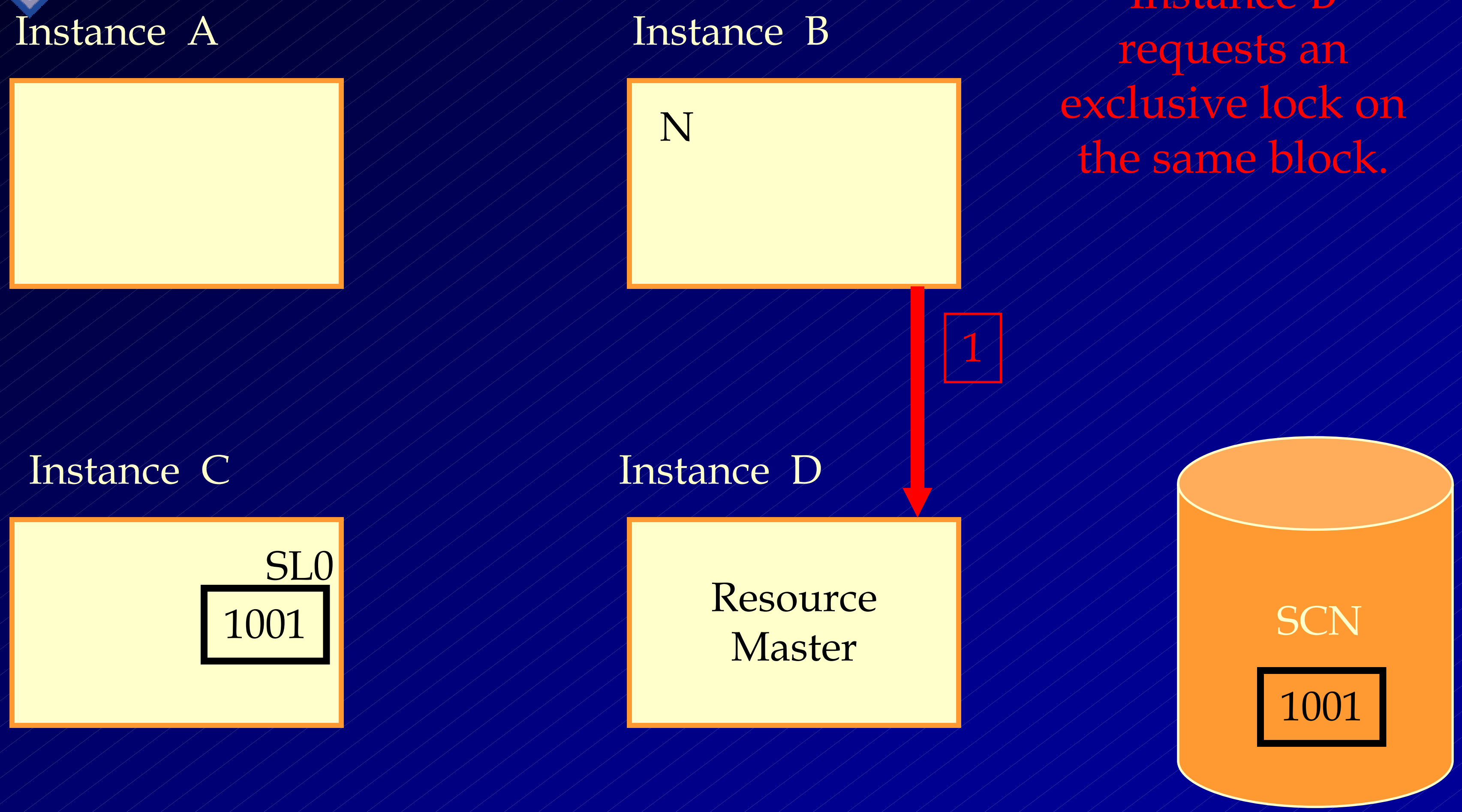


Example 1: Read without transfer



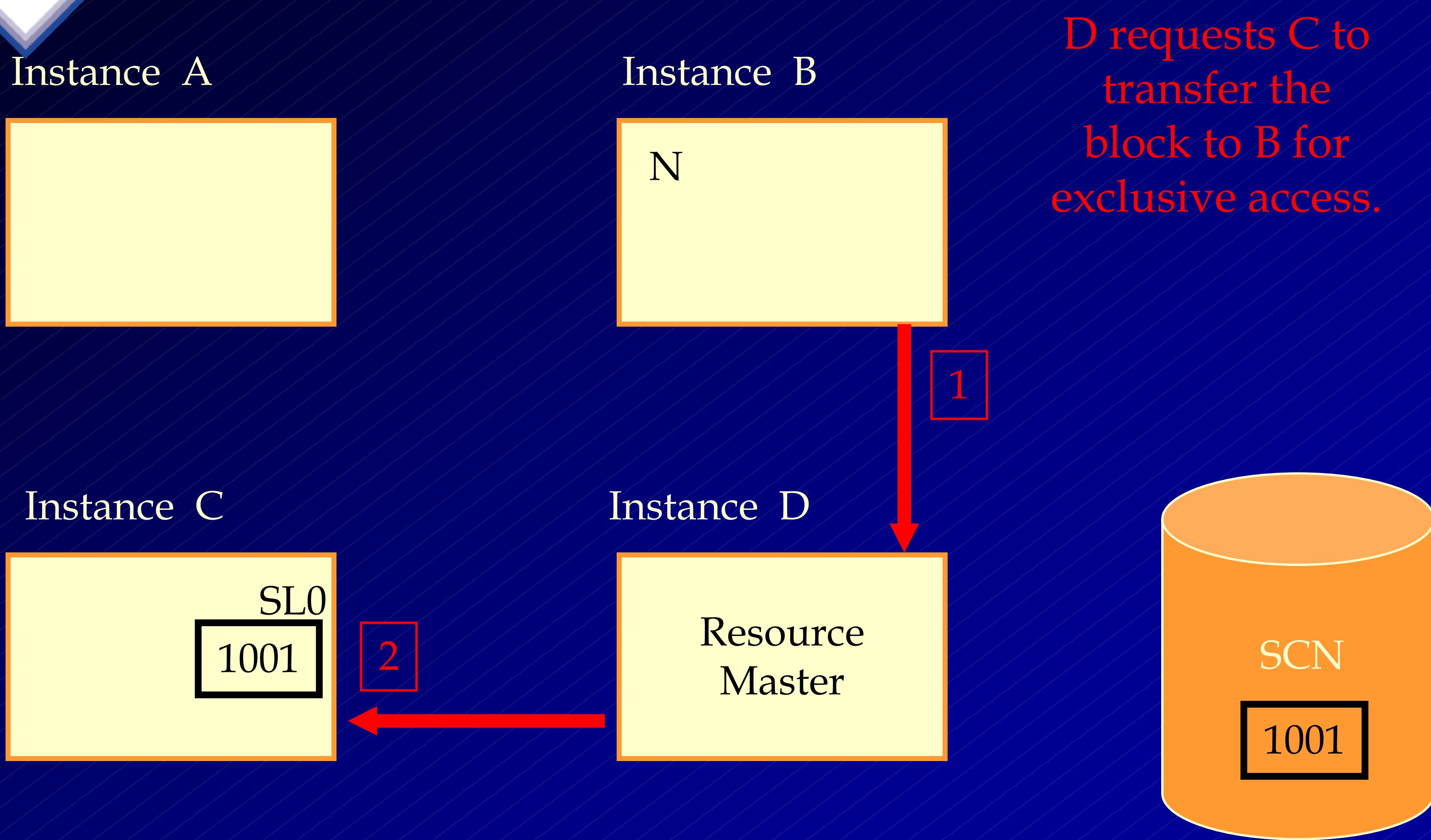
I/O is completed and block image is delivered to C. There is now a SL0 block image on C

Example 2: Read to Write Transfer

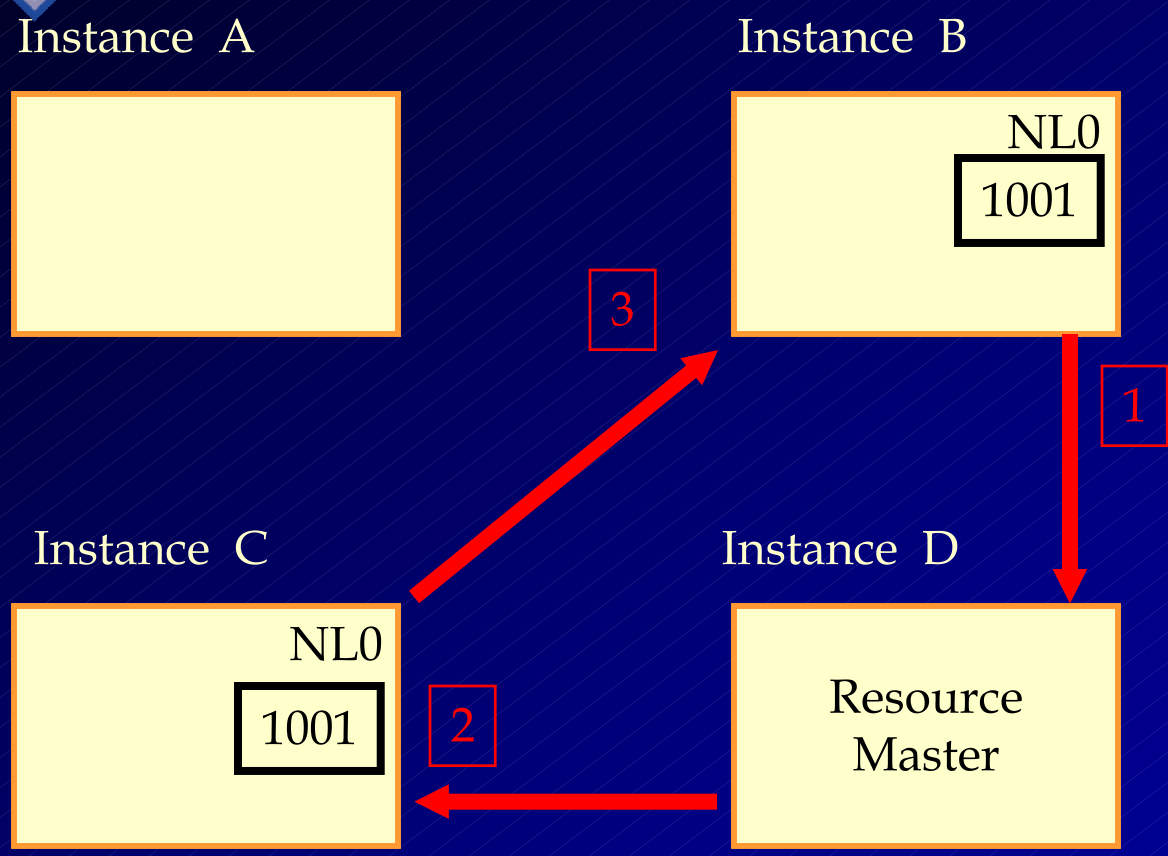


Instance B requests an exclusive lock on the same block.

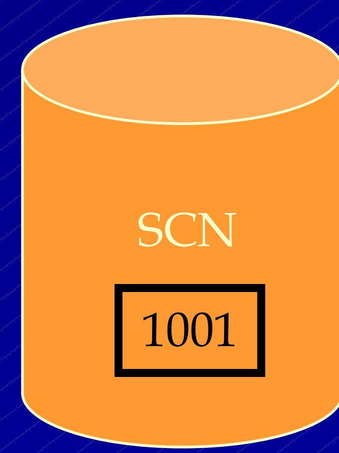
Example 2: Read to Write Transfer



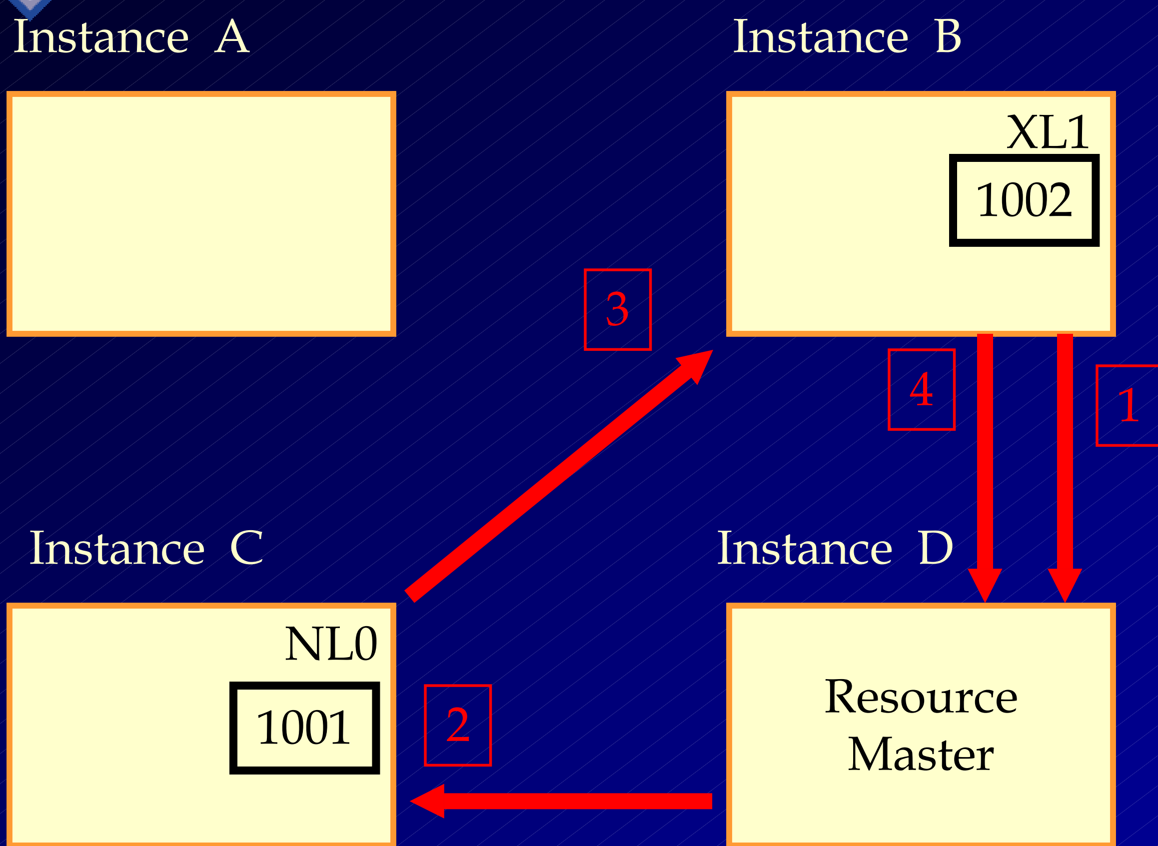
Example 2: Read to Write Transfer



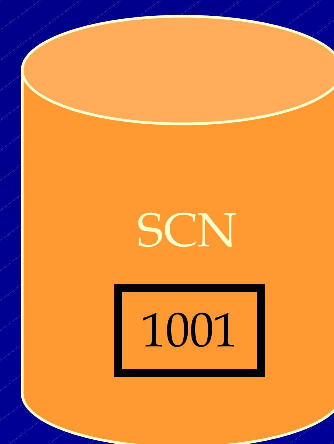
C sends the block to B indicating that C will close its own resource. Then C closes its own resource marking block image as Consistent Read



Example 2: Read to Write Transfer

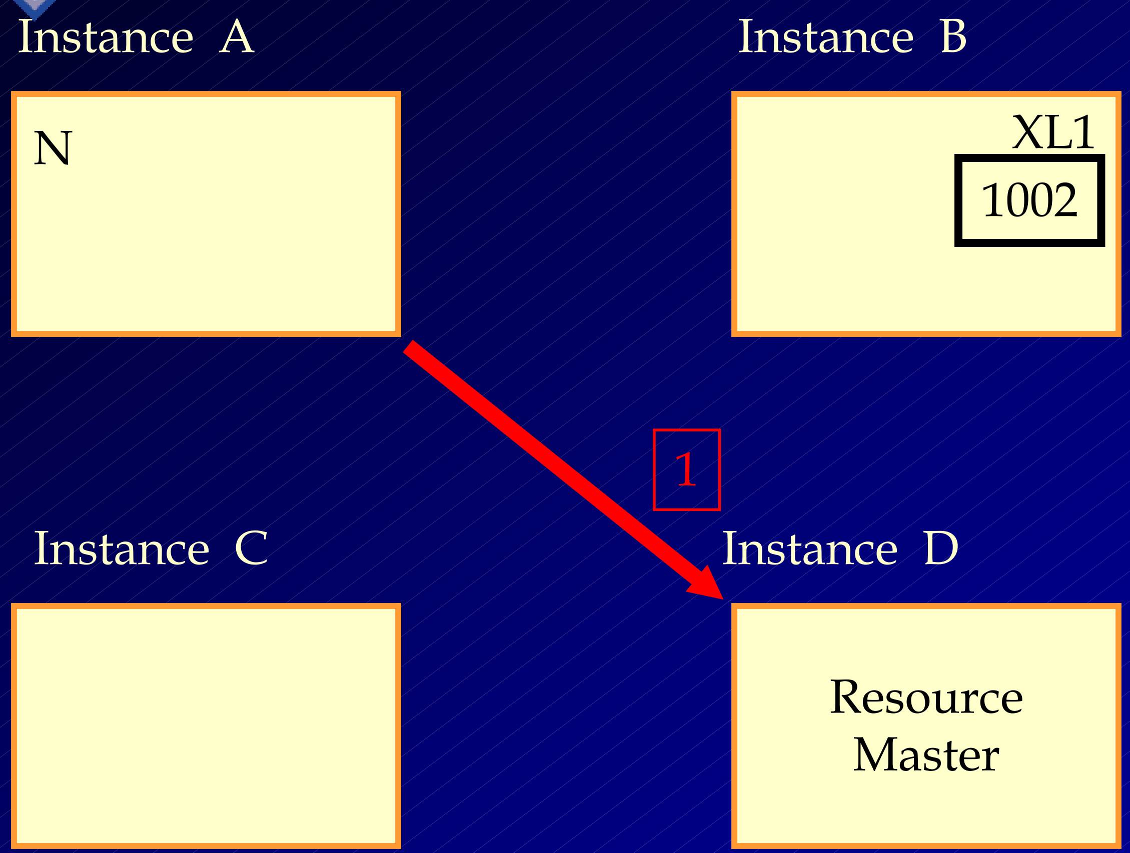


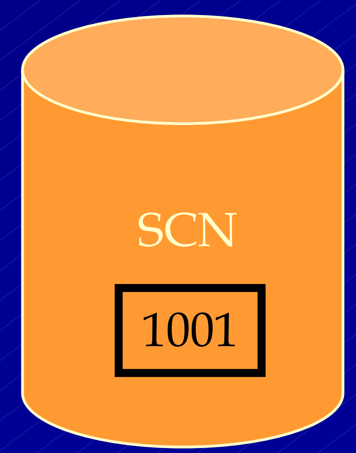
When B receives the message from C, it converts it's resource to X and sends a message to D informing the GCS of the assumption of X and the closure of resource on C.



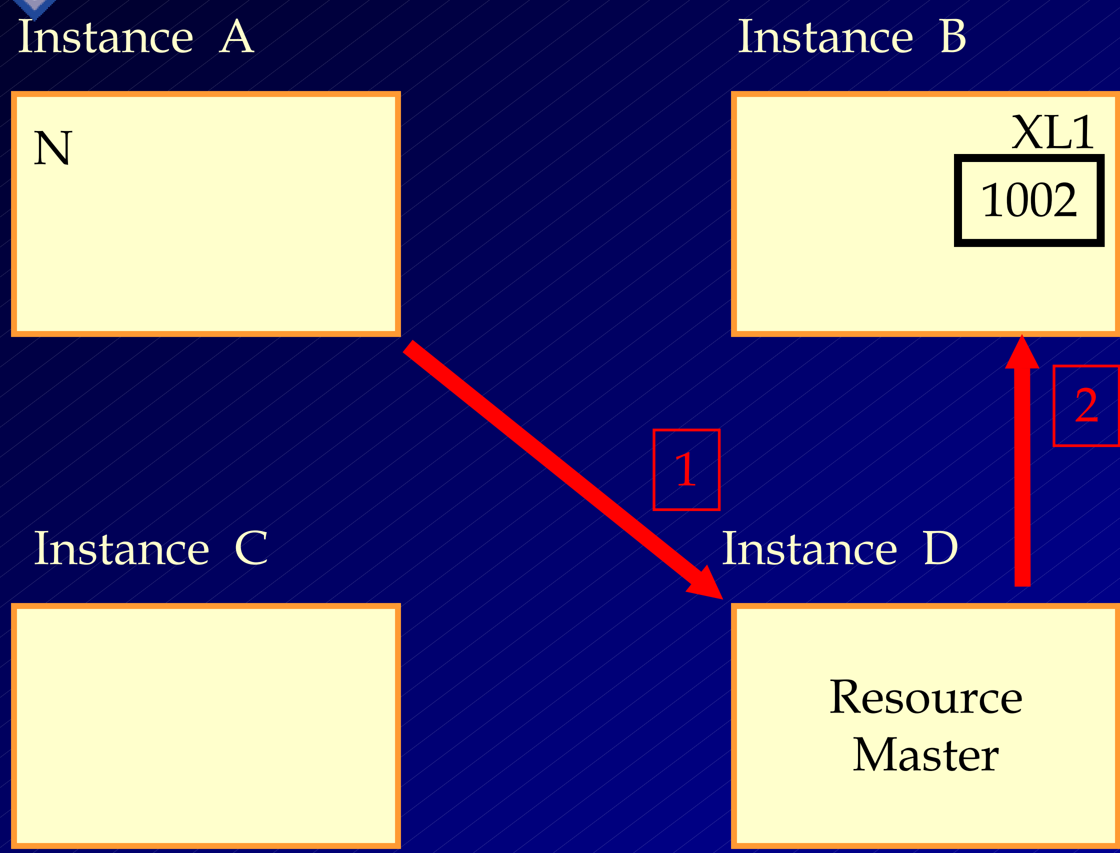
Example 3: Write to Write Transfer

Instance A requests to obtain the same block in exclusive mode.

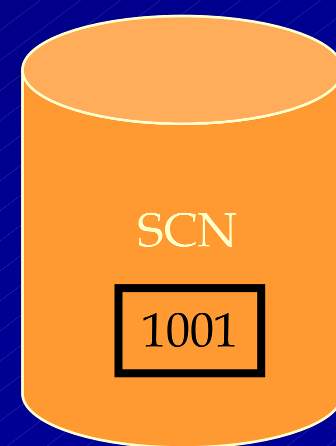




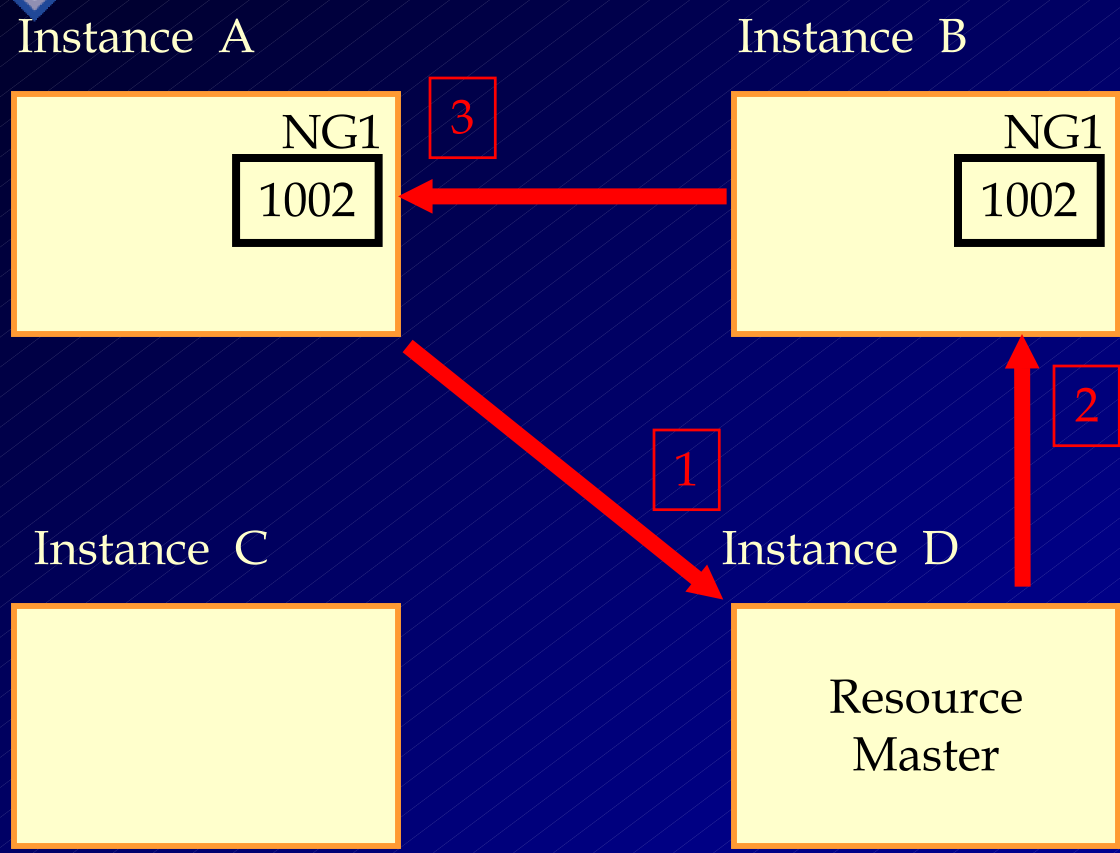
Example 3: Write to Write Transfer



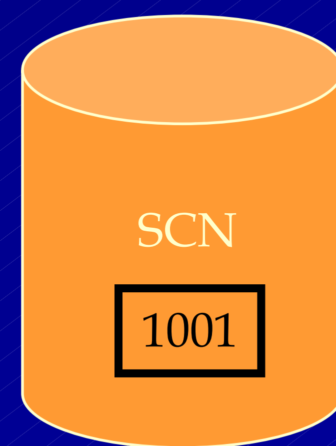
D instructs B to give up exclusive resource and transfer the current block with exclusive resource to instance A.



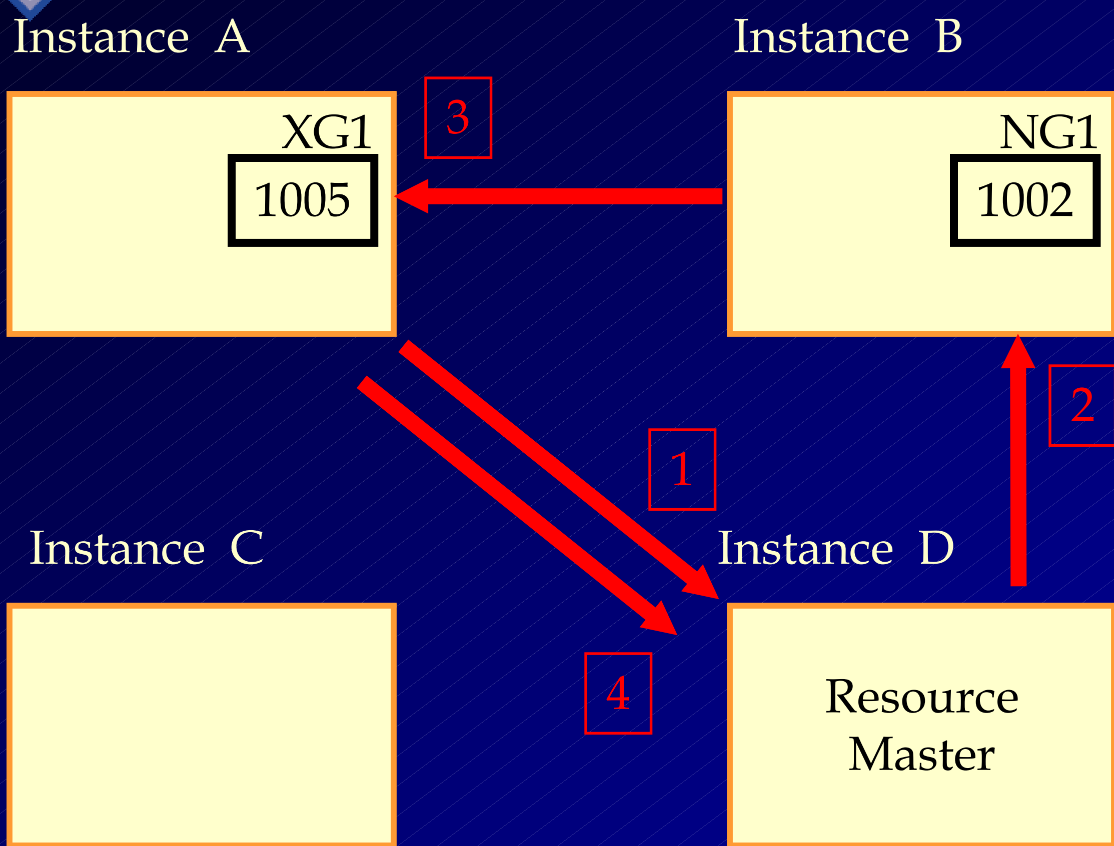
Example 3: Write to Write Transfer



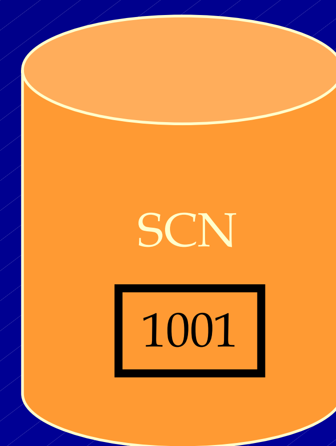
B transfers block and exclusive resources to A. B keeps a copy of the buffer and converts resource to N mode. Role is global because a dirty copy is on A & B



Example 3: Write to Write Transfer

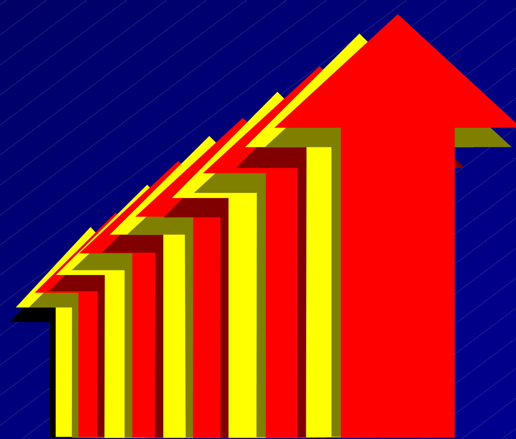


On receipt of the block and resource information, Instance A sends a resource assumption message to D and converts mode to X.



Analysis of Performance Issues

- Normal database Tuning and Monitoring
- RAC Cluster Interconnect Performance
- Monitoring Workload
- Monitoring RAC specific contention



Normal Database Tuning and Monitoring

- Prior to tuning RAC specific operations, each instance should be tuned separately.
  - APPLICATION Tuning
  - DATABASE Tuning
  - OS Tuning

THEN

- You can begin tuning RAC

What are you Waiting on?
(Single Instance Tuning - fyi only)

Tuning the RAC Cluster Interconnect

RAC issues are the same times TWO!

```
Top 5 Timed Events
~~~~~~~~~~~~~~~~~~                                                  % Total
Event                                               Waits    Time (s) Ela Time
-------------------------------------------- ------------ ----------- --------
global cache cr request                               820         154    72.50
CPU time                                                           54    25.34
global cache null to x                                478           1      .52
control file sequential read                          600           1      .52
control file parallel write                           141           1      .28
          -------------------------------------------------------------
```

- **Transfer times excessive from other instances in the cluster to this instance.**
- **Could be due to network problems or buffer transfer issues.**

Statspack - Top Wait Events Things to look for…

Wait Problem	Potential Fix
Sequential Read	Indicates many index reads – tune the code (especially joins); Faster I/O
Scattered Read	Indicates many full table scans – tune the code; cache small tables; Faster I/O
Free Buffer	Increase the DB_CACHE_SIZE; shorten the checkpoint; tune the code to get less dirty blocks, faster I/O, use multiple DBWR's.
Buffer Busy	Segment Header – Add freelists (if inserts) or freelist groups (esp. RAC). Use ASSM.

Statspack - Top Wait Events Things to look for…

Wait Problem	Potential Fix
Buffer Busy	Data Block – Separate 'hot' data; potentially use reverse key indexes; fix queries to reduce the blocks popularity, use smaller blocks, I/O, Increase initrans and/or maxtrans (this one's debatable) Reduce records per block.
Buffer Busy	Undo Header – Add rollback segments or increase size of segment area (auto undo)
Buffer Busy	Undo block – Commit more (not too much) Larger rollback segments/area. Try to fix the SQL.

Statspack - Top Wait Events Things to look for…

Wait Problem	Potential Fix
Enqueue - ST	Use LMT's or pre-allocate large extents
Enqueue - HW	Pre-allocate extents above HW (high water mark.)
Enqueue – TX	Increase initrans and/or maxtrans (TX4) on (transaction) the table or index. Fix locking issues if TX6. Bitmap (TX4) & Duplicates in Index (TX4).
Enqueue - TM (trans. mgmt.)	Index foreign keys; Check application locking of tables. DML Locks.

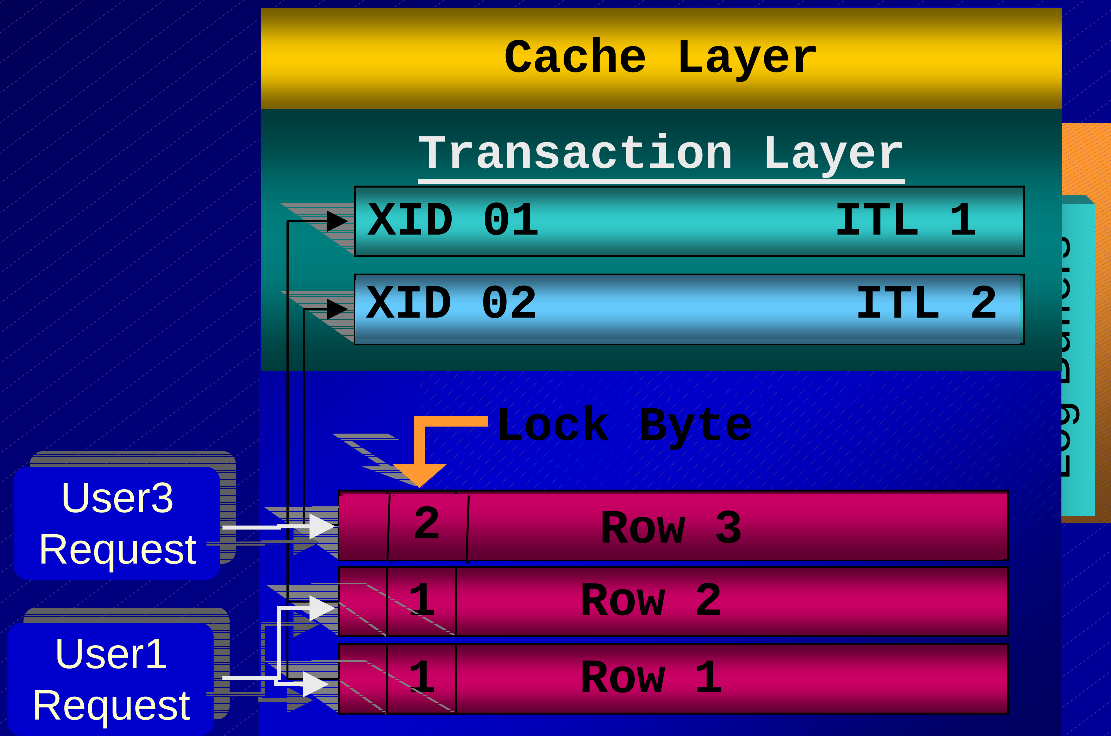
User 2 Updates Row# 3

User1 updates 2 rows with an insert/update/delete – an ITL is opened and xid tracks it in the data block (lock byte is set on row).

The xid ties to the UNDO header block which ties to the UNDO data block for undo.

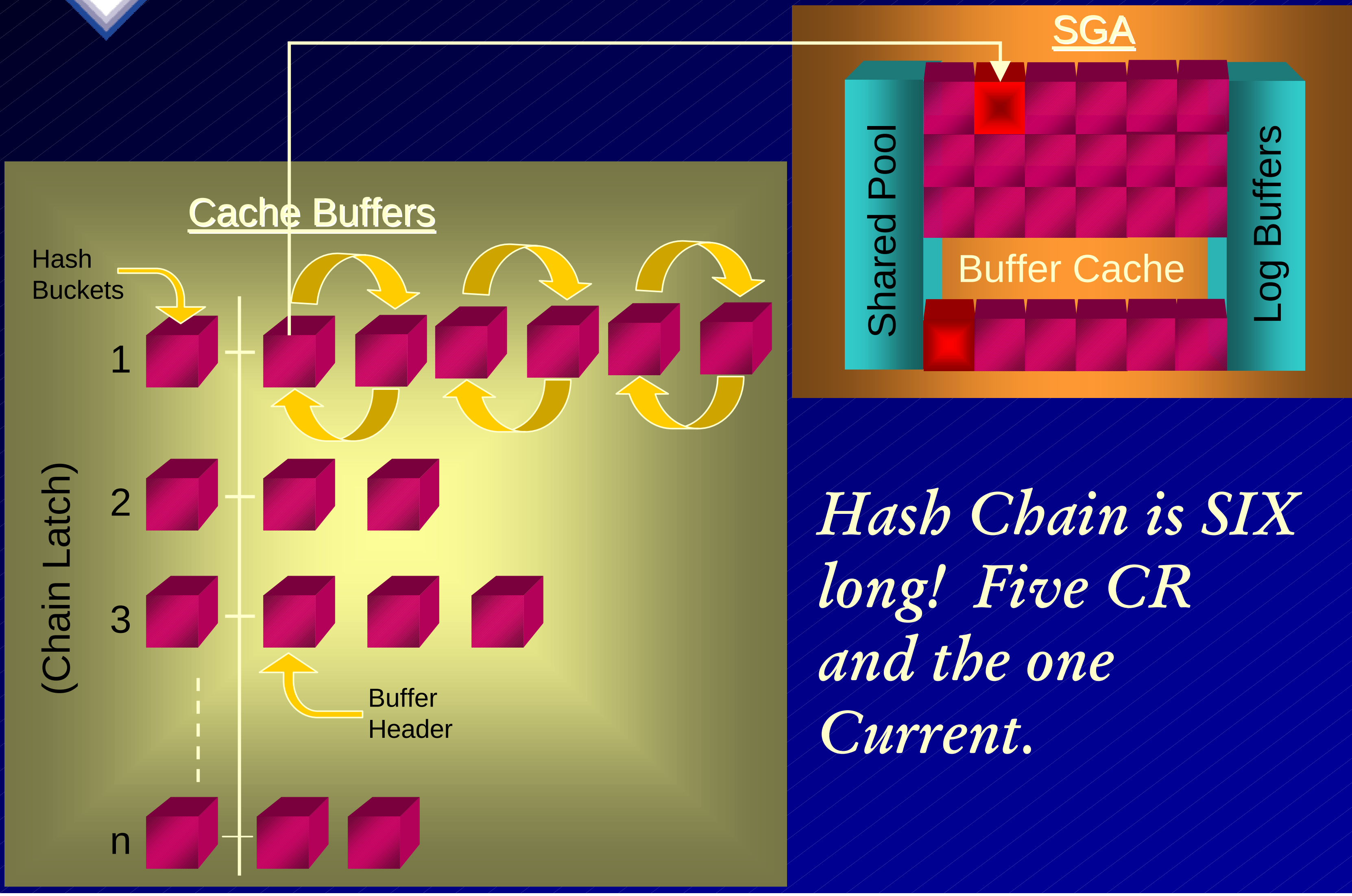
If user2 wants to query the row, they create a clone and rollback the transaction going to the undo header and undo block.

If user3 wants to update same row (they wait). If user 3 wants to update different row, then they open a second ITL with an xid that maps to an undo header & maps to an undo block.



Statspack - Top Wait Events Things to look for…

Wait Problem	Potential Fix
CBC Latches	Cache Buffers Chains Latches – Reduce the length of the hash chain (less copies) by reducing block's popularity. Increase the latches by increasing buffers. Use Oracle SQ generator.
LRU Chain Latch	This latch protects the LRU list when a user needs the latch to scan the LRU chain for a buffer. When a dirty buffer is encountered it is linked to the LRU-W. When adding, moving, or removing a buffer this latch is needed.



Hash Chain is SIX long! Five CR and the one Current.

Statspack - Latch Waits Things to look for…

Latch Problem	Potential Fix
Library Cache	Use bind variables; adjust the shared_pool_size
Shared Pool	Use bind variables; adjust the shared_pool_size
Redo allocation	Minimize redo generation and avoid unnecessary commits
Redo copy	Increase the _log_simultaneous_copies
Row cache objects	Increase the Shared Pool

Statspack - Top Wait Events Things to look for…

Wait Problem	Potential Fix
Session Logical Reads	All reads cached in memory. Includes both consistent gets and also the db block gets.
Consistent Gets	These are the reads of a block that are in the cache. They are NOT to be confused with consistent read (cr) version of a block in the buffer cache (usually the current version is read).
Db block gets	These are block gotten to be changed. MUST be the CURRENT block and not a cr block.
Db block changes	These are the db block gets (above) that were actually changed.
Physical Reads	Blocks not read from the cache. Either from disk, disk cache or O/S cache; there are also physical reads direct which bypass cache using Parallel Query (not in hit ratios).

Statspack – Instance Activity

Statistic	Total	per Second	per Trans
branch node splits	7,162	0.1	0.0
consistent gets	12,931,850,777	152,858.8	3,969.5
current blocks converted for CR	75,709	0.9	0.0
db block changes	343,632,442	4,061.9	105.5
db block gets	390,323,754	4,613.8	119.8
hot buffers moved to head of LRU	197,262,394	2,331.7	60.6
leaf node 90-10 splits	26,429	0.3	0.0
leaf node splits	840,436	9.9	0.3
logons cumulative	21,369	0.3	0.0
physical reads	504,643,275	5,965.1	154.9
physical writes	49,724,268	587.8	15.3
session logical reads	13,322,170,917	157,472.5	4,089.4
sorts (disk)	4,132	0.1	0.0
sorts (memory)	7,938,085	93.8	2.4
sorts (rows)	906,207,041	10,711.7	278.2
table fetch continued row	25,506,365	301.5	7.8
table scans (long tables)	111	0.0	0.0
table scans (short tables)	1,543,085	18.2	0.5

Normal Database Tuning and Monitoring

- Prior to tuning RAC specific operations, each instance should be tuned separately.
  - APPLICATION Tuning
  - DATABASE Tuning
  - OS Tuning

THEN
- You can begin tuning RAC

Tuning the RAC Cluster Interconnect

- Your primary tuning efforts after tuning each instance individually should focus on the processes that communicate through the cluster interconnect.

- Global Services Directory Processes
  - GES – Global Enqueue Services
  - GCS – Global Cache Services

Tuning the RAC Cluster Interconnect

- Global Cache Services (GCS) Waits
  - Indicates how efficiently data is being transferred over the cluster interconnect.
  - The critical RAC related waits are:

global cache busy	A wait event that occurs whenever a session has to wait for an ongoing operation on the resource to complete.
buffer busy global cache	A wait event that is signaled when a process has to wait for a block to become available because another process is obtaining a resource for this block.
buffer busy global CR	Waits on a consistent read via the global cache.

Tuning the RAC Cluster Interconnect

How:

- Query V$SESSION_WAIT to determine whether or not any sessions are experiencing RAC related waits.

- Identify the objects that are causing contention for these sessions.

- Modify the object to reduce contention.

Tuning the RAC Cluster Interconnect

- Query v$session_wait to determine whether or not any sessions are experiencing RAC related waits.

```
SELECT inst_id,
       event,
       p1 FILE_NUMBER,
       p2 BLOCK_NUMBER,
       WAIT_TIME
  FROM v$session_wait
 WHERE event in ('buffer busy global cr',
                 'global cache busy',
                 'buffer busy global cache');
```

Tuning the RAC Cluster Interconnect

- The output from this query should look something like this:

The block number and file number will indicate the object the requesting instance is waiting for.

```
INST_ID EVENT                                   FILE_NUMBER BLOCK_NUMBER  WAIT_TIME
------- ------------------------------------ ----------- ------------ ----------
      1 global cache busy                              9          150         15
      2 global cache busy                              9          150         10
```

Tuning the RAC Cluster Interconnect

- Identify objects that are causing contention for these sessions by identifying the object that corresponds to the file and block for each file_number/block_number combination returned:

```
SELECT owner,
       segment_name,
       segment_type
  FROM dba_extents
 WHERE file_id = 9
   AND 150 between block_id AND block_id+blocks-1;
```

Tuning the RAC Cluster Interconnect

- The output will be similar to:

```
OWNER      SEGMENT_NAME                   SEGMENT_TYPE
---------- ------------------------------ ---------------
SYSTEM     MOD_TEST_IND                   INDEX
```

Tuning the RAC Cluster Interconnect

- Modify the object to reduce the chances for application contention.

Modify what?

- Reduce the number of rows per block
- Adjust the block size to a smaller block size
- Modify INITRANS and FREELISTS

Tuning the RAC Cluster Interconnect
CR Block Transfer Time

- **Block contention can be measured by using block transfer time, calculated by:**

Accumulated round-trip time for all requests for consistent read blocks.

$$\frac{\texttt{global cache cr block receive time}}{\texttt{global cache cr blocks received}}$$

Total number of consistent read blocks successfully received from another instance.

Tuning the RAC Cluster Interconnect
CR Block Transfer Time

- Use a self-join query on GV$SYSSTAT to compute this ratio:

```
COLUMN "AVG RECEIVE TIME (ms)" FORMAT 9999999.9
COLUMN inst_id format 9999
PROMPT GCS CR BLOCKS
SELECT b1.inst_id, b2.value "RECEIVED",
       b1.value "RECEIVE TIME",
       ((b1.value / b2.value) * 10) "AVG RECEIVE TIME (ms)"
  FROM gv$sysstat b1, gv$sysstat b2
 WHERE b1.name = 'global cache cr block receive time'
   AND b2.name = 'global cache cr blocks received'
   AND b1.inst_id = b2.inst_id;
```

```
INST_ID   RECEIVED RECEIVE TIME AVG RECEIVE TIME (ms)
------- ---------- ------------ ---------------------
      1       2791         3287                  11.8
      2       3760         7482                  19.9
```

Tuning the RAC Cluster Interconnect
CR Block Transfer Time

- **Problem Indicators:**
  - **High Transfer Time**
  - **One node showing excessive transfer time**

- **Use OS commands to verify cluster interconnects are functioning correctly.**

Tuning the RAC Cluster Interconnect
CR Block Service Time

- Measure the latency of service times.
- Service time is comprised of consistent read build time, log flush time, and send time.

global cache cr blocks served	Number of requests for a CR block served by LMS
global cache cr block build time	The time that the LMS process requires to create a CR block on the holding instance.
global cache cr block flush time	Time waited for a log flush when a CR request is served (part of the serve time)
global cache cr block send time	Time required by LMS to initiate a send of the CR block.

Tuning the RAC Cluster Interconnect
CR Block Service Time

- Query GV$SYSSTAT to determine average service times by instance:

```
SELECT a.inst_id "Instance",
       (a.value+b.value+c.value)/decode(d.value,0,1, d.value) "LMS Service
   Time"
  FROM gv$sysstat A,
       gv$sysstat B,
       gv$sysstat C,
       gv$sysstat D
 WHERE A.name = 'global cache cr block build time'
   AND B.name = 'global cache cr block flush time'
   AND C.name = 'global cache cr block send time'
   AND D.name = 'global cache cr blocks served'
   AND B.inst_id = A.inst_id
   AND C.inst_id = A.inst_id
   AND D.inst_id = A.inst_id
ORDER
   BY a.inst_id;
```

Tuning the RAC Cluster Interconnect
CR Block Service Time

- Result:

```
Instance LMS Service Time
--------- -----------------
        1        1.07933923
        2        .636687318
```

Instance 1 is on a faster node and serves blocks faster.

The service time TO Instance 2 is shorter!

Instance 2 is on the slower node.

Tuning the RAC Cluster Interconnect
CR Block Service Time – Component Level

- **Query GV$SYSSTAT to drill-down into service time for individual components:**

```
SELECT A.inst_id "Instance", (A.value/D.value) "Consistent Read Build",
      (B.value/D.value) "Log Flush Wait",(C.value/D.value) "Send Time"
 FROM GV$SYSSTAT A, GV$SYSSTAT B,
      GV$SYSSTAT C, GV$SYSSTAT D
WHERE A.name = 'global cache cr block build time'
  AND B.name = 'global cache cr block flush time'
  AND C.name = 'global cache cr block send time'
  AND D.name = 'global cache cr blocks served'
  AND B.inst_id=a.inst_id
  AND C.inst_id=a.inst_id
  AND D.inst_id=a.inst_id
ORDER
   BY A.inst_id;
```

```
 Instance Consistent Read Build Log Flush Wait  Send Time
--------- --------------------- -------------- ----------
        1             .00737234     1.05059755  .02203942
        2             .04645529      .51214820  .07844674
```

Tuning the RAC Cluster Interconnect
OS Troubleshooting Commands

- Monitor cluster interconnects using OS commands to find:
  - Large number of processes in the run state waiting for cpu or scheduling
  - Platform specific OS parameter settings that affect IPC buffering or process scheduling
  - Slow, busy, faulty interconnects. Look for dropped packets, retransmits, CRC errors.
  - Ensure you have a private network
  - Ensure inter-instance traffic is not routed through a public network

Tuning the RAC Cluster Interconnect
OS Troubleshooting Commands

- Useful OS commands in a Sun Solaris environment to aid in your research are:

```
$ netstat –l
$ netstat –s
$ sar –c
$ sar –q
$ vmstat
$ iostat
```

Tuning the RAC Cluster Interconnect
Global Performance Views for RAC

- Global Dynamic Performance view names for Real Application Clusters are prefixed with *GV$*.

- GV$SYSSTAT, GV$DML_MISC, GV$SYSTEM_EVENT, GV$SESSION_WAIT, GV$SYSTEM_EVENT contain numerous statistics that are of interest to the DBA when tuning RAC.

- The CLASS column of GV$SYSSTAT tells you the type of statistic. RAC related statistics are in classes 8, 32 and 40.

Tuning the RAC Cluster Interconnect
Your Analysis Toolkit

- **RACDIAG.SQL is an Oracle-provided script that can also help with troubleshooting. RACDIAG.SQL can be downloaded from Metalink or the TUSC website, www.tusc.com (Metalink Note: 135714.1).**

- **STATSPACK combined with the queries illustrated in this presentation will provide you with the tools you need to effectively address RAC system performance issues.**

- **Oracle 10g OEM and ADDM (later in the presentation).**

- **Third Party tools can also be used.**

Tuning the RAC Cluster Interconnect
Undesirable Statistics

- As mentioned, the view GV$SYSSTAT will contain statistics that indicate the performance of your RAC system.
- The following statistics should always be as near to zero as possible:

global cache blocks lost	Block losses during transfers. May indicate network problems.
global cache blocks corrupt	Blocks that were corrupted during transfer. High values indicate an IPC, network, or hardware problem.

Tuning the RAC Cluster Interconnect
Undesirable Statistics

```
SELECT A.VALUE "GC BLOCKS LOST 1",
       B.VALUE "GC BLOCKS CORRUPT 1",
       C.VALUE "GC BLOCKS LOST 2",
       D.VALUE "GC BLOCKS CORRUPT 2"
  FROM GV$SYSSTAT A, GV$SYSSTAT B, GV$SYSSTAT C, GV$SYSSTAT D
 WHERE A.INST_ID=1 AND A.NAME='global cache blocks lost'
   AND B.INST_ID=1 AND B.NAME='global cache blocks corrupt'
   AND C.INST_ID=2 AND C.NAME='global cache blocks lost'
   AND D.INST_ID=2 AND D.NAME='global cache blocks corrupt'

GC BLOCKS LOST 1 GC BLOCKS CORRUPT 1 GC BLOCKS LOST 2 GC BLOCKS CORRUPT 2
---------------- ------------------- ---------------- -------------------
               0                   0              652                   0
```

Instance 2 is experiencing some lost blocks.

Tuning the RAC Cluster Interconnect
Undesirable Statistics

- Take a closer look to see what is causing the lost blocks:

```
SELECT A.INST_ID "INSTANCE", A.VALUE "GC BLOCKS LOST",
B.VALUE "GC CUR BLOCKS SERVED",
C.VALUE "GC CR BLOCKS SERVED",
A.VALUE/(B.VALUE+C.VALUE) RATIO
FROM GV$SYSSTAT A, GV$SYSSTAT B, GV$SYSSTAT C
WHERE A.NAME='global cache blocks lost' AND
      B.NAME='global cache current blocks served' AND
      C.NAME='global cache cr blocks served' and
      B.INST_ID=a.inst_id AND
      C.INST_ID = a.inst_id;
```

In this case the database Instance 1 takes 22 seconds to perform a series of tests, Instance 2 takes 25 minutes.

```
  Instance gc blocks lost gc cur blocks served gc cr blocks served      RATIO
---------- -------------- -------------------- ------------------- ----------
         1              0                 3923                2734          0
         2            652                 3008                4380 .088251218
```

Tuning the RAC Cluster Interconnect
Undesirable Statistics

- The TCP receive and send buffers on Instance 2 were set at 64K
- This is a 8k block size instance with a db_file_multiblock_read_count of 16. This was causing excessive network traffic since the system was using full table scans resulting in a read of 128K.
- In addition the actual TCP buffer area was set to a small number.

Tuning the RAC Cluster Interconnect

Current Block Transfer Statistics

- In addition to monitoring consistent read blocks, we also need to be concerned with processing current mode blocks.
- Calculate the average receive time for current mode blocks:

Accumulated round trip time for all requests for current blocks

$$\frac{\texttt{Global cache current block receive time}}{\texttt{Global cache current blocks received}}$$

Number of current blocks received from the holding instance over the interconnect

Tuning the RAC Cluster Interconnect

Current Block Transfer Statistics

- **Query the GV$SYSSTAT view to obtain this ratio for each instance:**

```
COLUMN "AVG RECEIVE TIME (ms)" format 9999999.9
COLUMN inst_id FORMAT 9999
PROMPT GCS CURRENT BLOCKS
SELECT b1.inst_id,
       b2.value "RECEIVED",
       b1.value "RECEIVE TIME",
      ((b1.value / b2.value) * 10) "AVG RECEIVE TIME (ms)"
 FROM gv$sysstat b1, gv$sysstat b2
WHERE b1.name = 'global cache current block receive time'
  AND b2.name = 'global cache current blocks received'
  AND b1.inst_id = b2.inst_id;
```

```
INST_ID   RECEIVED RECEIVE TIME AVG RECEIVE TIME (ms)
------ ---------- ------------ ---------------------
     1      22694        68999                  30.4
     2      23931        42090                  17.6
```

Tuning the RAC Cluster Interconnect
Current Block Service Time Statistics

- Service time for current blocks is comprised of pin time, log flush time, and send time.

global cache current blocks served	The number of current blocks shipped to the requesting instance over the interconnect.
global cache block pin time	The time it takes to pin the current block before shipping it to the requesting instance. Pinning is necessary to disallow changes to the block while it is prepared to be shipped to another instance.
global cache block flush time	The time it takes to flush the changes to a block to disk (forced log flush), before the block is shipped to the requesting instance.
global cache block send time	The time it takes to send the current block to the requesting instance over the interconnect.

Tuning the RAC Cluster Interconnect
Current Block Service Time Statistics

- To calculate current block service time, query GV$SYSSTAT:

```
SELECT a.inst_id "Instance",
      (a.value+b.value+c.value)/d.value "Current Blk Service Time"
 FROM GV$SYSSTAT A,
      GV$SYSSTAT B,
      GV$SYSSTAT C,
      GV$SYSSTAT D
WHERE A.name = 'global cache current block pin time'
  AND B.name = 'global cache current block flush time'
  AND C.name = 'global cache current block send time'
  AND D.name = 'global cache current blocks served'
  AND B.inst_id = A.inst_id
  AND C.inst_id = A.inst_id
  AND D.inst_id = A.inst_id
ORDER
   BY a.inst_id;
```

Tuning the RAC Cluster Interconnect
Current Block Service Time Statistics

- The output from the query may look like this:

```
 Instance Current Blk Service Time
--------- ------------------------
        1               1.18461603
        2               1.63126376
```

Instance 2 requires 38% more time to service current blocks than Instance 1.

- Drill-down to service time components will help identify the cause of the problem.

Tuning the RAC Cluster Interconnect

Current Block Service Time Statistics

```
SELECT A.inst_id "Instance",
      (A.value/D.value) "Current Block Pin",
      (B.value/D.value) "Log Flush Wait",
      (C.value/D.value) "Send Time"
FROM GV$SYSSTAT A,
     GV$SYSSTAT B,
     GV$SYSSTAT C,
     GV$SYSSTAT D
WHERE A.name = 'global cache current block build time'
  AND B.name = 'global cache current block flush time'
  AND C.name = 'global cache current block send time'
  AND D.name = 'global cache current blocks served'
  AND B.inst_id=a.inst_id
  AND C.inst_id=a.inst_id
  AND D.inst_id=a.inst_id
ORDER
   BY A.inst_id;
```

Where is the problem?

High pin times could indicate problems at the IO interface level.

Instance	Current Block Pin	Log Flush Wait	Send Time
1	.69366887	.472058762	.018196236
2	1.07740715	.480549199	.072346418

Tuning the RAC Cluster Interconnect
Global Cache Convert and Get Times

GCS: **Global Cache Services.** A process that communicates through the cluster interconnect

- A final set of statistics can be useful in identifying RAC interconnect performance issues.

global cache convert time	The accumulated time that all sessions require to perform global conversion on GCS resources
global cache converts	Resource converts on buffer cache blocks. Incremented whenever GCS resources are converted from Null to Exclusive, shared to Exclusive, or Null to Shared.
global cache get time	The accumulated time of all sessions needed to open a GCS resource for a local buffer.
Global cache gets	The number of buffer gets that result in opening a new resource with the GCS.

Tuning the RAC Cluster Interconnect
Global Cache Convert and Get Times

- To measure these times, query the GV$SYSSTAT view:

```
SELECT A.inst_id "Instance", A.value/B.value "Avg Cache Conv. Time",
       C.value/D.value "Avg Cache Get Time", E.value "GC Convert Timeouts"
  FROM GV$SYSSTAT A, GV$SYSSTAT B,
       GV$SYSSTAT C, GV$SYSSTAT D,
       GV$SYSSTAT
 WHERE A.name = 'global cache convert time'
   AND B.name = 'global cache converts'
   AND c.name = 'global cache get time'
   AND D.name = 'global cache gets'
   AND E.name = 'global cache convert timeouts'
   AND B.inst_id = A.inst_id
   AND C.inst_id = A.inst_id
   AND D.inst_id = A.inst_id
   AND E.inst_id = A.inst_id
ORDER
   BY A.inst_id;
```

Neither convert time is excessive. Excessive would be > 10-20 ms. Instance 1 has higher convert times because it is getting and converting from instance 2, which is running on slower CPUs

```
Instance Avg Cache Conv. Time Avg Cache Get Time GC Convert Timeouts
--------- ------------------- ------------------ -------------------
        1           1.85812072         .981296356                   0
        2           1.65947528         .627444273                   0
```

Tuning the RAC Cluster Interconnect
Global Cache Convert and Get Times

- Use the GV$SYSTEM_EVENT view to review TIME_WAITED statistics for various GCS events if the get or convert times become significant.
- STATSPACK can be used for this to view these events.

Tuning the RAC Cluster Interconnect
Global Cache Convert and Get Times

- Interpreting the *convert* and *get* statistics

High convert times	Instances swapping a lot of blocks over the interconnect
Large values or rapid increases in gets, converts or average times	GCS contention
High latencies for resource operations	Excessive system loads
Non-zero GC converts Timeouts	System contention or congestion. Could indicate serious performance problems.

Tuning the RAC Cluster Interconnect
Other Wait Events

- **If these RAC-specific wait events show up in the TOP-5 wait events on the STATSPACK report, you need to determine the cause of the waits:**
  - **global cache open s**
  - **global cache open x**
  - **global cache null to s**
  - **global cache null to x**
  - **global cache cr request**
  - **Global Cache Service Utilization for Logical Reads**

Tuning the RAC Cluster Interconnect
Other Wait Events

Event:

global cache open s

global cache open x

Description:

- A session has to wait for receiving permission for shared(s) or exclusive(x) access to the requested resource
- Wait duration should be short.
- Wait followed by a read from disk.
- Blocks requested are not cached in any instance.

Action:

- Not much can be done
- When associated with high totals or high per-transaction wait time, data blocks are not cached.
- Will cause sub-optimal buffer cache hit ratios
- Consider preloading heavily used tables

Tuning the RAC Cluster Interconnect
Other Wait Events

Event:

global cache null to s

global cache null to x

Description:

- Happens when two instances exchange the same block back and forth over the network.
- These events will consume a greater proportion of total wait time if one instance requests cached data blocks from other instances.

Action:

- Reduce the number of rows per block to eliminate the need for block swapping between instances.

Tuning the RAC Cluster Interconnect
Other Wait Events

Event:

global cache cr request

Description:

• **Happens when an instance requests a CR data block and the block to be transferred hasn't arrived at the requesting instance.**

Action:

•**Examine the cluster interconnects for possible problems.**

•**Modify objects to reduce possibility of contention.**

Tuning the RAC Cluster Interconnect
High GCS Time Per Request

- **Examine the Cluster Statistics page of your STATSPACK report when:**
    - **Global cache waits constitute a large proportion of the wait time listed on the first page of your STATSPACK report.**

AND

    - **Response times or throughput does not conform to your service level requirements.**
- **STATSPACK report should be taken during heavy RAC workloads.**

Tuning the RAC Cluster Interconnect
High GCS Time Per Request

- **Causes of a high GCS time per request are:**

- **Contention for blocks**
- **System Load**
- **Network issues**

That's nice to know, but how can I fix it?

Tuning the RAC Cluster Interconnect
High GCS Time Per Request

- **System Load**

  - If processes are queuing for the CPU, raise the priority of the GCS processes (LMS*n*) to have priority over other processes to lower GCS times.

  - Reduce the load on the server by reducing the number of processes on the database server.

  - Increase capacity by adding CPUs to the server

  - Add nodes to the cluster database

Tuning the RAC Cluster Interconnect
High GCS Time Per Request

- **Network issues**

  • OS logs and OS stats will indicate if a network link is congested.

  • Ensure packets are being routed through the private interconnect, not the public network.

Cluster Interconnect verification

```
SQL>CONNECT SYS/<> AS SYSDBA
SQL>ORADEBUG SETMYPID
SQL>ORADEBUG IPC
SQL>EXIT
```

Is this private IP address?

Output generated in udump directory

```
SSKGXPT 0x3671e28 flags SSKGXPT_READPENDING info for
network 0
        socket no 9    IP 142.23.153.1  UDP 59084
        sflags SSKGXPT_WRITESSKGXPT_UP
        info for network 1
        socket no 0     IP 0.0.0.0      UDP 0
        sflags SSKGXPT_DOWN
context timestamp 0x4402d
        no ports
```

Interconnect Best Practices (Metalink Note: 278132.1)

- Have at least a gigabit ethernet for optimal performance
- Do not use crossover cables (use a switch)
- Increase the UDP buffer sizes at the OS maximum
- Turn on UDP checksumming

GV$CACHE_TRANSFER & GV$BH

- Displays block types and classes that Oracle has transferred at least once over the cluster interconnect
- XNC column records the number of lock conversions (potential pings)
  - used to identify the blocks that are being frequently transferred (pinged) between instances
  - only shows buffers with a nonzero XNC count
  - If NAME column is blank - buffer is associated with a temporary segment

Tuning the Cluster Interconnect (Hot Blocks)

```
SELECT INST_ID,
       NAME,
       FILE#,
       CLASS#,
       MAX(XNC)
FROM   GV$CACHE_TRANSFER
GROUP BY INST_ID,
         NAME,
         FILE#,
         CLASS#
```

```
INST_ID NAME            FILE#     CLASS#   MAX(XNC)
------ ----------  ---------- ---------- ----------
     1 IDL_UB2$             1          4        231
     1 PK_USPRL             4          1         47
     1 PK_COMP              4          1         39
     1 COMPANY            171          1       2849
```

Tuning the Cluster Interconnect (Hot Blocks)

```
SELECT FILE#,
       BLOCK#,
       CLASS#,
       STATUS,
       XNC
FROM   GV$CACHE_TRANSFER
WHERE NAME   = 'COMPANY'
AND   FILE# = 171;
```

FILE#	BLOCK#	CLASS#	STAT	XNC
171	898	1	XCUR	1321
171	1945	1	XCUR	27
171	1976	1	XCUR	19
171	2039	1	XCUR	849

Tuning the Cluster Interconnect (Hot Blocks)

```
SELECT COMP_ID,
       COMP_NAME
FROM   COMPANY
WHERE  DBMS_ROWID.ROWID_BLOCK_NUMBER(ROWID) = 898;
```

```
           COMP_ID  NAME
-----------------  -------------------------
              3949  ORACLE SOFTWARE
              3952  CATAMARAN INC.
              3957  PARIYAR BROTHERS INC.
              3961  DIGITAL BROADCASTING INC.
```

Tuning the RAC Cluster Interconnect
Using STATSPACK Reports

- The STATSPACK report shows statistics ONLY for the node or instance on which it was run.

- Run `statspack.snap` procedure and `spreport.sql` script on each node you want to monitor to compare to other instances.

Tuning the RAC Cluster Interconnect
Using STATSPACK Reports

Top 5 Timed Events

CPU time (processing time) should be the predominant event

Exceeds CPU time, therefore needs investigation.

```
Top 5 Timed Events
~~~~~~~~~~~~~~~~~~                                                  % [illegible]tal
Event                                          Waits    Time (s) El[illegible] Time
-------------------------------------------- ------------ ----------- --------
global cache cr request                          820         154     72.50
CPU time                                                      54     25.34
global cache null to x                           478           1       .52
control file sequential read                     600           1       .52
control file parallel write                      141           1       .28
          -------------------------------------------------------------
```

- Transfer times are excessive from other instances in this cluster to this instance.
- Could be due to network problems or buffer cache sizing issues.

Tuning the RAC Cluster Interconnect
Using STATSPACK Reports

- Network changes were made
- An index was added
- STATSPACK report now looks like this:

CPU time is now the predominant event

```
Top 5 Timed Events
~~~~~~~~~~~~~~~~~~                                                   % Total
Event                                               Waits    Time (s) Ela Time
-------------------------------------------- ------------ ----------- --------
CPU time                                                           99    64.87
global cache null to x                              1,655          28    18.43
enqueue                                                46           8     5.12
global cache busy                                     104           7     4.73
DFS lock handle                                        38           2     1.64
```

Tuning the RAC Cluster Interconnect
Using STATSPACK Reports

After network and index changes.

- **Workload characteristics for this instance:**

```
Cluster Statistics for DB: DB2  Instance: INST1  Snaps: 105 -106  Snaps: 25 -26

Global Cache Service - Workload Characteristics
-----------------------------------------------
Ave global cache get time (ms):                        3.1      8.2
Ave global cache convert time (ms):                    3.2     16.5
Ave build time for CR block (ms):                      0.2      1.5
Ave flush time for CR block (ms):                      0.0      6.0
Ave send time for CR block (ms):                       1.0      0.9
Ave time to process CR block request (ms):             1.3      8.5
Ave receive time for CR block (ms):                   17.2     18.3
Ave pin time for current block (ms):                   0.2     13.7
Ave flush time for current block (ms):                 0.0      3.9
Ave send time for current block (ms):                  0.9      0.8
Ave time to process current block request (ms):        1.1     18.4
Ave receive time for current block (ms):               3.1     17.4
Global cache hit ratio:                                1.7      2.5
Ratio of current block defers:                         0.0      0.2
% of messages sent for buffer gets:                    1.4      2.2
% of remote buffer gets:                               1.1      1.6
Ratio of I/O for coherence:                            8.7      2.9
Ratio of local vs remote work:                         0.6      0.5
Ratio of fusion vs physical writes:                    1.0      0.0
```

Tuning the RAC Cluster Interconnect
Using STATSPACK Reports

- Global Enqueue Services (GES) control the inter-instance locks in Oracle 9i RAC.
- The STATSPACK report contains a special section for these statistics.

```
Global Enqueue Service Statistics
---------------------------------
Ave global lock get time (ms):                          0.9
Ave global lock convert time (ms):                      1.3
Ratio of global lock gets vs global lock releases:      1.1
```

Tuning the RAC Cluster Interconnect
Using STATSPACK Reports

- Guidelines for GES Statistics:
  - All times should be < 15ms
  - Ratio of global lock gets vs global lock releases should be near 1.0
- High values could indicate possible network or memory problems
- Could also be caused by application locking issues
- May need to review the enqueue section of STATSPACK report for further analysis.

Tuning the RAC Cluster Interconnect
Using STATSPACK Reports

- GCS AND GES messaging
  - Watch for excessive queue times (>20-30ms)

```
GCS and GES Messaging statistics
--------------------------------
Ave message sent queue time (ms):                            1.8
Ave message sent queue time on ksxp (ms):                    2.6
Ave message received queue time (ms):                        1.2
Ave GCS message process time (ms):                           1.2
Ave GES message process time (ms):                           0.2
% of direct sent messages:                                  58.4
% of indirect sent messages:                                 4.9
% of flow controlled messages:                              36.7
```

Tuning the RAC Cluster Interconnect
Using STATSPACK Reports

Statistic:

gcs blocked converts

gcs blocked cr converts

Description:

- Instance requested a block from another instance and was unable to obtain the conversion of the block.
- Indicates users on different instances need to access the same blocks.

Action:

- Prevent users on different instances from needing access to the same blocks.
- Ensure sufficient freelists (not an issue if using automated freelists)
- Reduce block contention through freelists, initrans.
- Limit rows per block.

Tuning the RAC Cluster Interconnect
Using STATSPACK Reports

- The Library Cache Activity report shows statistics regarding the GES.
- Watch for GES Invalid Requests and GES Invalidations.
- Could indicate insufficient sizing of the shared pool resulting in GES contention.

```
Library Cache Activity for DB: DB2  Instance: INST2  Snaps: 25 -26
->"Pct Misses"  should be very low


                    GES Lock      GES Pin      GES Pin   GES Inval GES Invali-
Namespace           Requests     Requests     Releases    Requests     dations
--------------- ------------ ------------ ------------ ----------- -----------
BODY                       1            0            0           0           0
CLUSTER                    4            0            0           0           0
INDEX                     84            0            0           0           0
SQL AREA                   0            0            0           0           0
TABLE/PROCEDURE          617          192            0          77           0
TRIGGER                    0            0            0           0           0
```

Tuning the RAC Cluster Interconnect
GES Monitoring

- **Oracle makes a Global Cache Service request whenever a user accesses a buffer cache to read or modify a data block and the block is not in the local cache.**

… and the crowd gasps!

- **RATIOS can be used to give an indication of how hard your Global Services Directory processes are working.**

Tuning the RAC Cluster Interconnect

GES Monitoring

- **To estimate the use of the GCS relative to the number of logical reads:**

Sum of GCS requests

global cache gets + global cache converts + global cache cr blocks rcvd + global cache current blocks rcvd

consistent gets + db block gets

Number of logical reads

Tuning the RAC Cluster Interconnect
GES Monitoring

- This information can be found by querying the GV$SYSSTAT view

```
SELECT a.inst_id "Instance",
      (A.VALUE+B.VALUE+C.VALUE+D.VALUE)/(E.VALUE+F.VALUE) "GLOBAL CACHE HIT RATIO"
  FROM GV$SYSSTAT A, GV$SYSSTAT B,
       GV$SYSSTAT C, GV$SYSSTAT D,
       GV$SYSSTAT E, GV$SYSSTAT F
WHERE  A.NAME='global cache gets'
  AND B.NAME='global cache converts'
  AND C.NAME='global cache cr blocks received'
  AND D.NAME='global cache current blocks received'
  AND E.NAME='consistent gets'
  AND F.NAME='db block gets'
  AND B.INST_ID=A.INST_ID AND C.INST_ID=A.INST_ID
  AND D.INST_ID=A.INST_ID AND E.INST_ID=A.INST_ID
  AND F.INST_ID=A.INST_ID;
```

SINCE INSTANCE STARTUP

```
  Instance GLOBAL CACHE HIT RATIO
---------- ----------------------
         1             .02403656
         2            .014798887
```

Tuning the RAC Cluster Interconnect
GES Monitoring

- Some blocks, those frequently requested by local and remote users, will be hot.
- If a block is hot, its transfer is delayed for a few milliseconds to allow the local users to complete their work.
- The following ratio provides a rough estimate of how prevalent this is:

$$\frac{\texttt{global cache defers}}{\texttt{global cache current blocks served}}$$

- A ratio of higher than 0.3 indicates that you have some pretty hot blocks in your database.

Tuning the RAC Cluster Interconnect
GES Monitoring

- **To find the blocks involved in busy waits, query the columns NAME, KIND, FORCED_READS, FORCED_WRITES.**

```
 Select INST_ID "Instance", name, kind,
        sum(forced_reads) "Forced Reads",
        sum(forced_writes) "Forced Writes"
   FROM gv$cache_transfer
  WHERE owner# != 0
  GROUP BY inst_id, name, kind
  ORDER BY 1,4 desc,2;
```

```
Instance NAME                 KIND       Forced Reads Forced Writes
-------- -------------------- ---------- ------------ -------------
       1 MOD_TEST_IND          INDEX              308             0
       1 TEST2                 TABLE               64             0
       1 AQ$_QUEUE_TABLES      TABLE                5             0
       2 TEST2                 TABLE              473             0
       2 MOD_TEST_IND          INDEX              221             0
```

Tuning the RAC Cluster Interconnect
GES Monitoring

- If you discover a problem, you may be able to alleviate contention by:
  – Reducing hot spots or spreading the accesses to index blocks or data blocks.
  – Use Oracle has or range partitions wherever applicable, just as you would in single-instance Oracle databases
  – Reduce concurrency on the object by implementing resource management or load balancing.
  – Decrease the rate of modifications on the object (use fewer database processes).

Tuning the RAC Cluster Interconnect
GES Monitoring

- Fusion writes occur when a block previously changed by another instance needs to be written to disk in response to a checkpoint or cache aging.
- Oracle sends a message to notify the other instance that a fusion write will be performed to move the data block to disk.
- Fusion writes do not require an additional write to disk.
- Fusion writes are a subset of all physical writes incurred by an instance.

Tuning the RAC Cluster Interconnect
GES Monitoring

- The following ratio shows the proportion of writes that Oracle manages with fusion writes:

$$\frac{\textit{DBWR fusion writes}}{\textit{physical writes}}$$

```
SELECT A.inst_id "Instance",
       A.VALUE/B.VALUE "Cache Fusion Writes Ratio"
  FROM GV$SYSSTAT A,
       GV$SYSSTAT B
 WHERE A.name='DBWR fusion writes'
   AND B.name='physical writes'
   AND B.inst_id=a.inst_id
ORDER
   BY A.INST_ID;

Instance Cache Fusion Writes Ratio
--------- -------------------------
        1                .216290958
        2                .131862042
```

Tuning the RAC Cluster Interconnect
GES Monitoring

- **A high large value for Cache Fusion Writes ratio may indicate:**
  - **Insufficiently sized caches**
  - **Insufficient checkpoints**
  - **Large numbers of buffers written due to cache replacement or checkpointing.**

Tuning the RAC Cluster Interconnect
CACHE_TRANSFER views

- FORCED_READS and FORCED_WRITES column are used to determine which types of objects your RAC instances are sharing.
- Values in FORCED_WRITES column provide counts of how often a certain block type experiences a transfer out of a local buffer cache due to a request for current version by another instance.
- The NAME column shows the name of the object containing blocks being transferred.

V$CACHE_TRANSFER	
FILE#	NUMBER
BLOCK#	NUMBER
CLASS#	NUMBER
STATUS	VARCHAR2(5)
XNC	NUMBER
FORCED_READS	NUMBER
FORCED_WRITES	NUMBER
NAME	VARCHAR2(30)
PARTITION_NAME	VARCHAR2(30)
KIND	VARCHAR2(15)
OWNER#	NUMBER
GC_ELEMENT_ADDR	RAW(4)
GC_ELEMENT_NAME	NUMBER

Deprecated views in 10g

These views were deprecated in 10g:

GV$/V$CLASS_CACHE_TRANSFER
GV$/V$CACHE_LOCK
GV$/V$FALSE_PING
GV$/V$FILE_CACHE_TRANSFER
GV$/V$GC_ELEMENTS_WITH_COLLISIONS
GV$/V$LOCK_ACTIVITY
GV$/V$TEMP_CACHE_TRANSFER

The useful info was incorporated into:

GV$/V$INSTANCE_CACHE_TRANSFER
GV$/V$SEGMENT_STATISTICS"

Tuning the RAC Cluster Interconnect

Monitoring the GES Processes

- To monitor the Global Enqueue Service Processes, use the GV$ENQUEUE_STAT view.

GV$ENQUEUE_STAT

INST_ID	NUMBER
EQ_TYPE	VARCHAR2(2)
TOTAL_REQ#	NUMBER
TOTAL_WAIT	NUMBER
SUCC_REQ#	NUMBER
FAILED_REQ#	NUMBER
CUM_WAIT_TIME	NUMBER

Tuning the RAC Cluster Interconnect

Monitoring the GES Processes

- Retrieve all of the enqueues with a total_wait# value greater than zero:

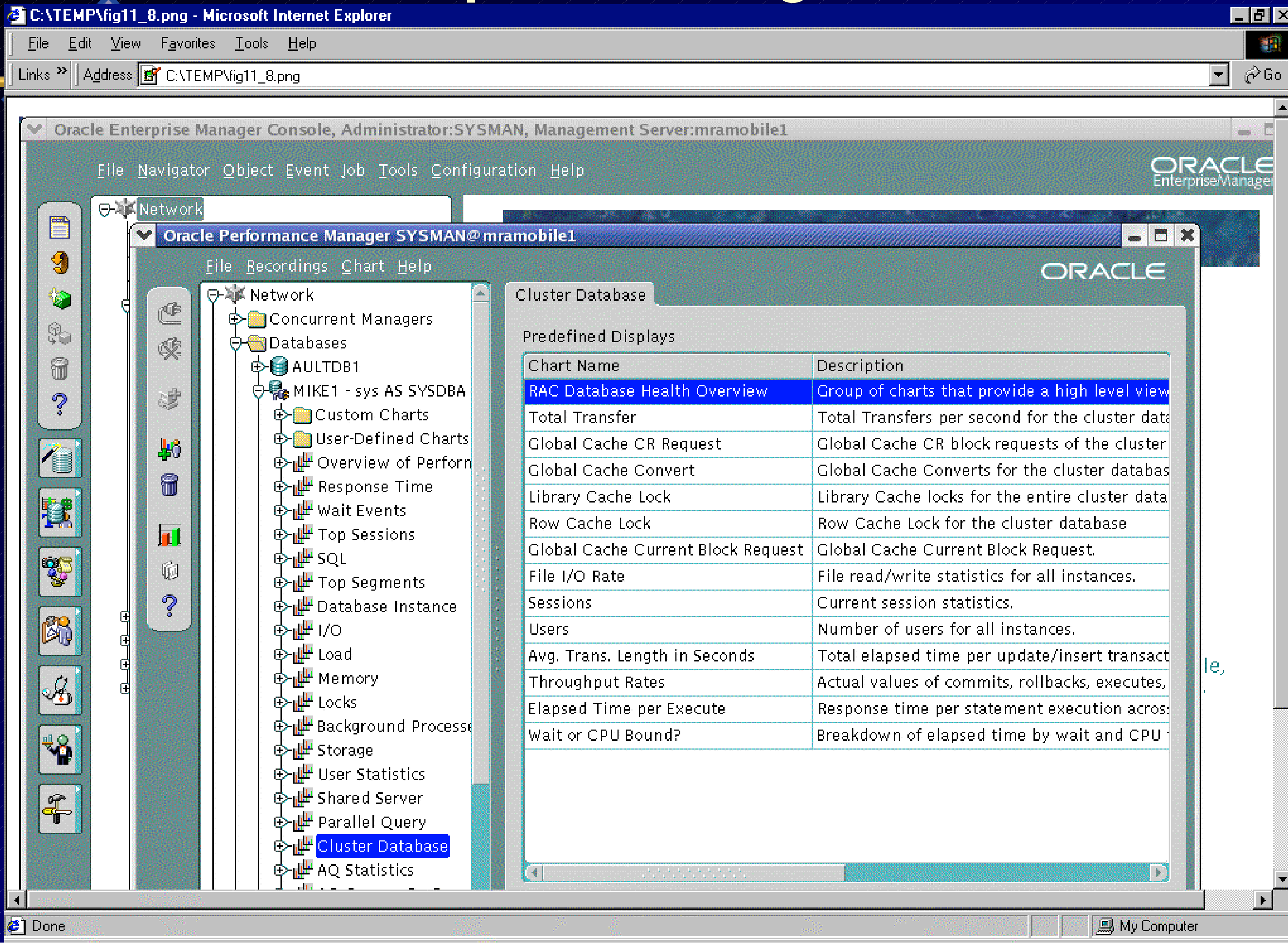
```
SELECT *
  FROM gv$enqueue_stat
 WHERE total_wait# > 0
ORDER BY inst_id, cum_wait_time desc;

   INST_ID EQ TOTAL_REQ# TOTAL_WAIT#  SUCC_REQ# FAILED_REQ# CUM_WAIT_TIME
---------- -- ---------- ----------- ---------- ----------- -------------
         1 TX      31928          26      31928           0        293303
         1 PS        995         571        994           1         55658
         1 TA       1067         874       1067           0         10466
         1 TD        974         974        974           0          2980
         1 DR        176         176        176           0           406
         1 US        190         189        190           0           404
         1 PI         47          27         47           0           104
         :
         :
```

Tuning the RAC Cluster Interconnect
Monitoring the GES Processes

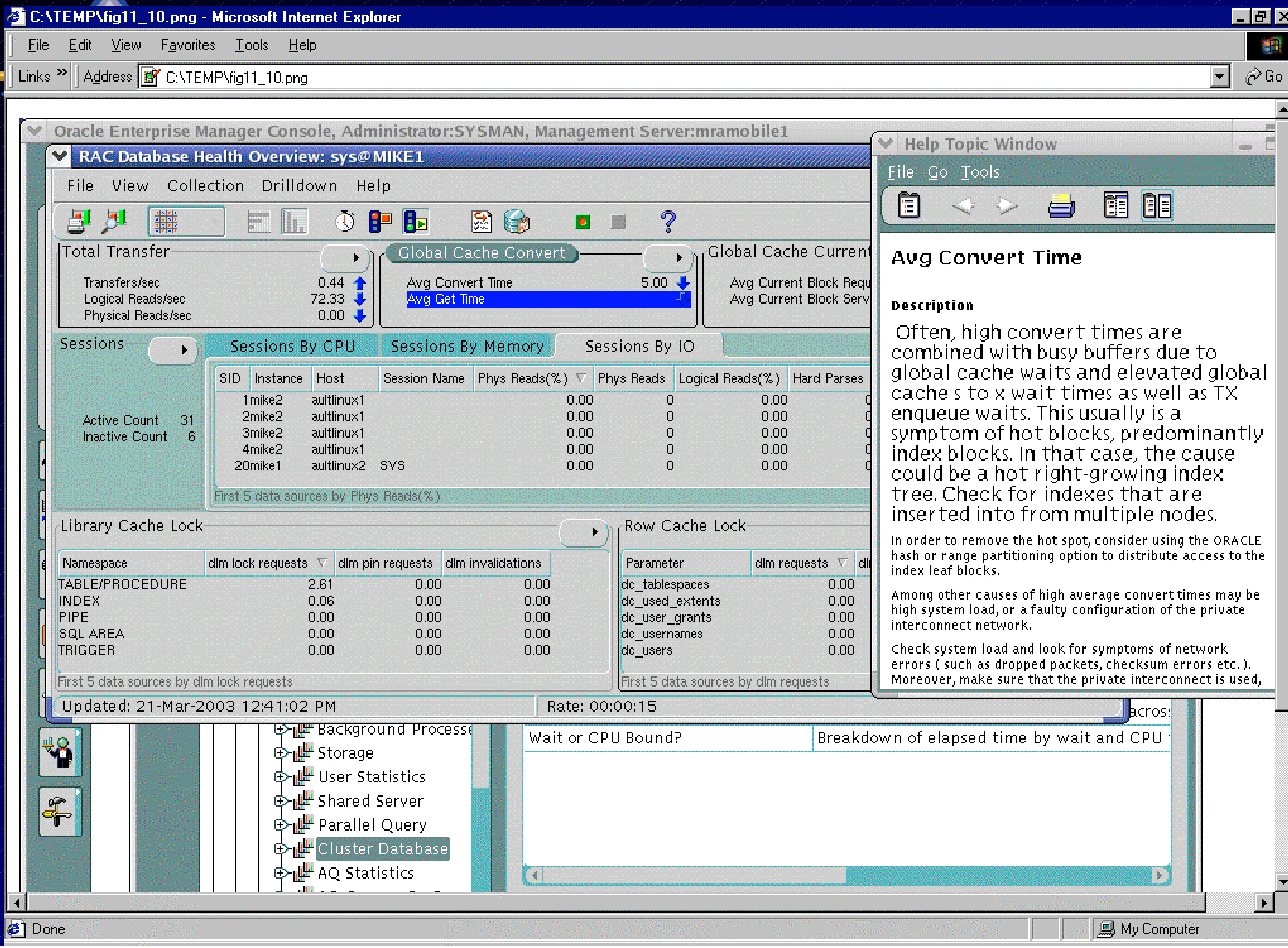
- Oracle says enqueues of interest in the RAC environment are:

SQ Enqueue	Indicates there is contention for sequences. Increase the cache size of sequences using ALTER SEQUENCES. Also, when creating sequences, the NOORDER keyword should be used to avoid forced ordering of queued sequence values.
TX Enqueue	Application-related row locking usually causes problems here. RAC processing can magnify the effect of TX enqueue waits. Index block splits can cause TX enqueue waits. Some TX enqueue performance issues can be resolved by setting the value of INITRANS parameter for a table or index (have also heard setting it to be equal to the number of CPUs per node multiplied by the number of nodes in a cluster multiplied by 0.75). I prefer setting INITTRANS to the number of concurrent processes issuing DML against the block.

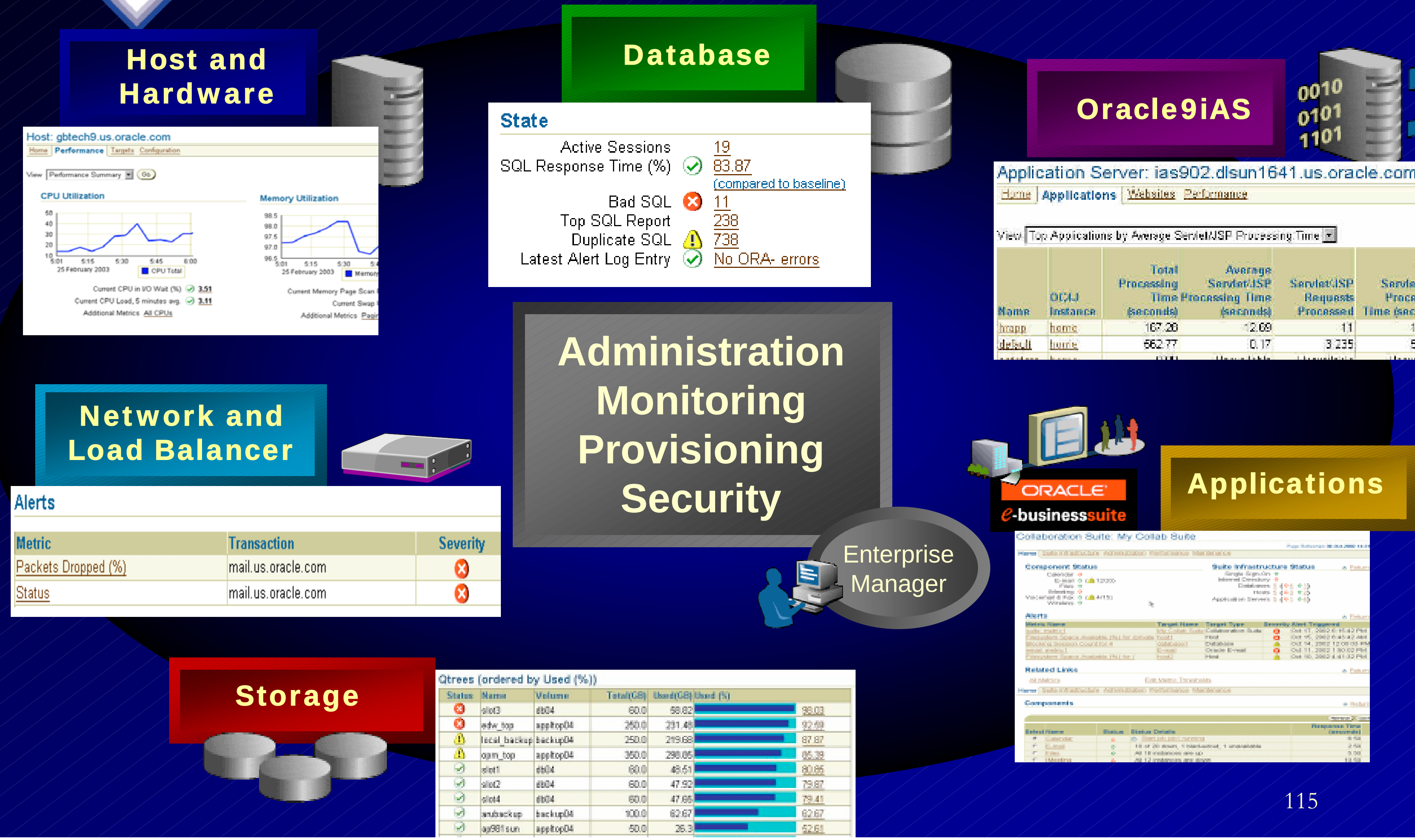
Enterprise Manager – 9i



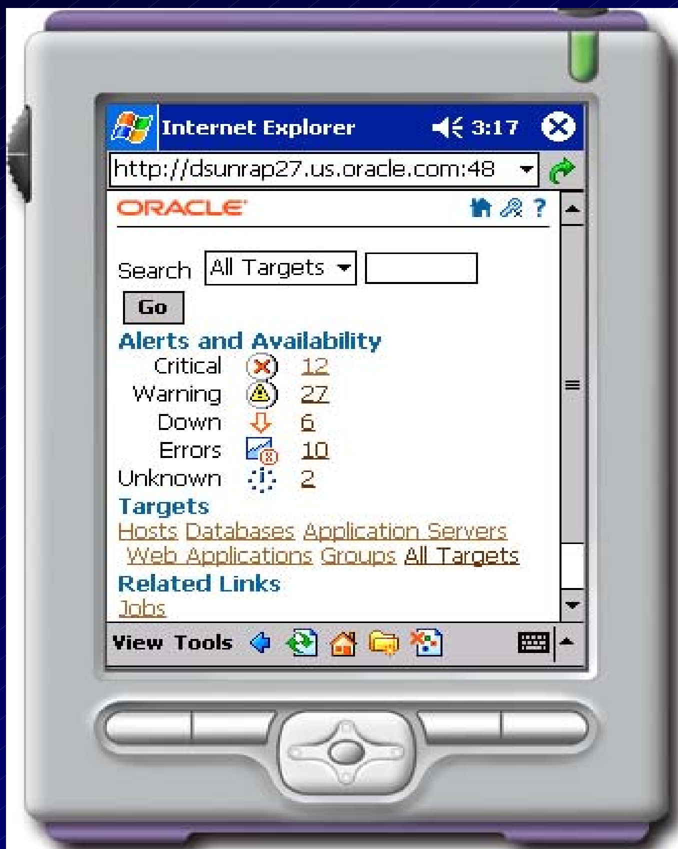
Enterprise Manager – 9i

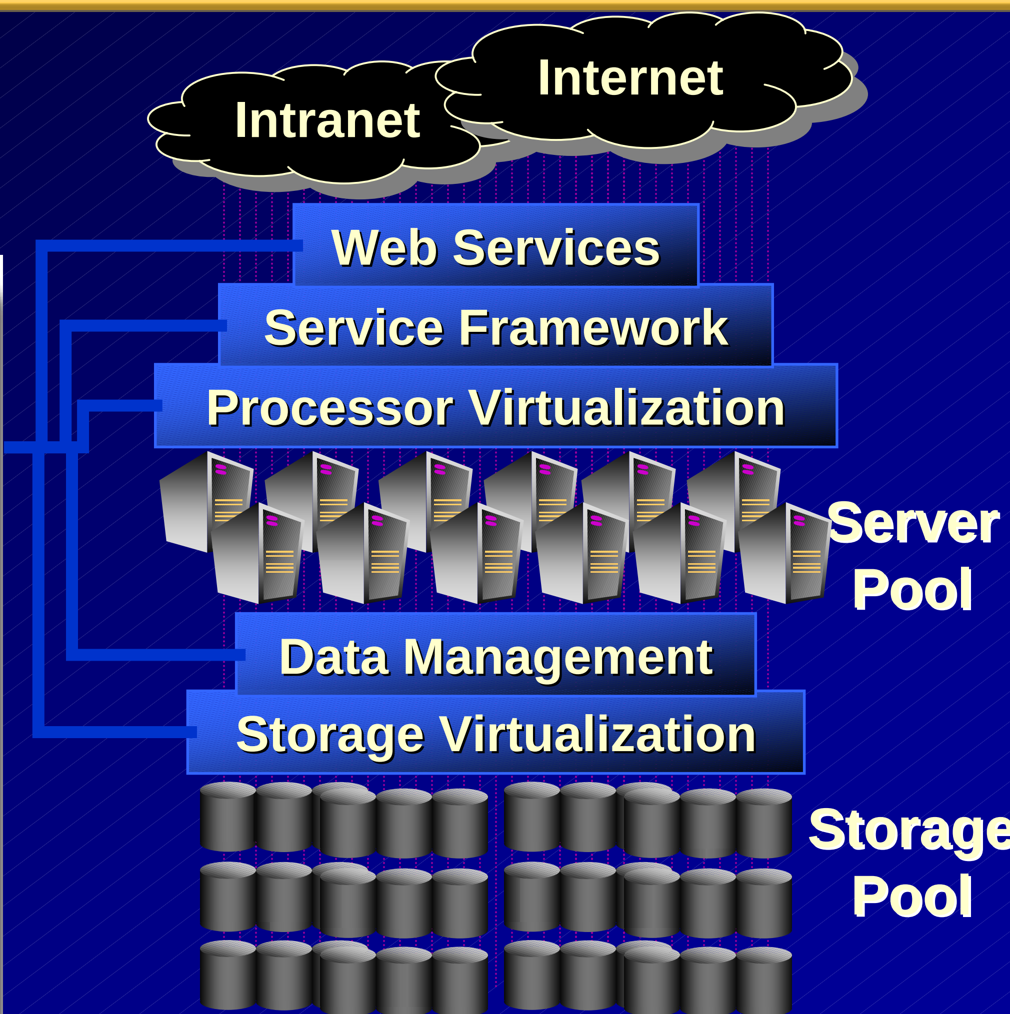


Enterprise Manager 10g for the Grid



The Future ….
Manage end to end

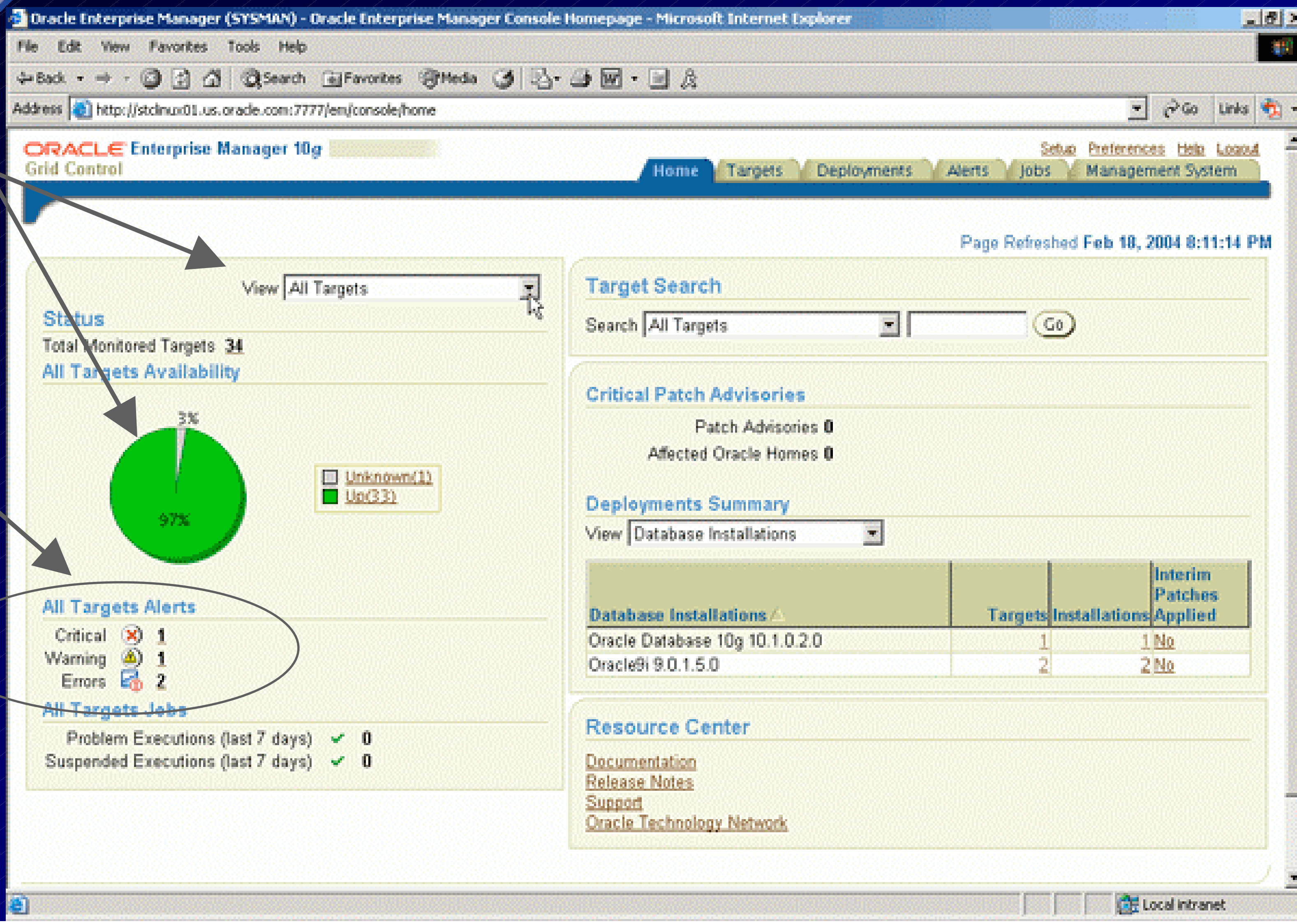




Monitor All Targets

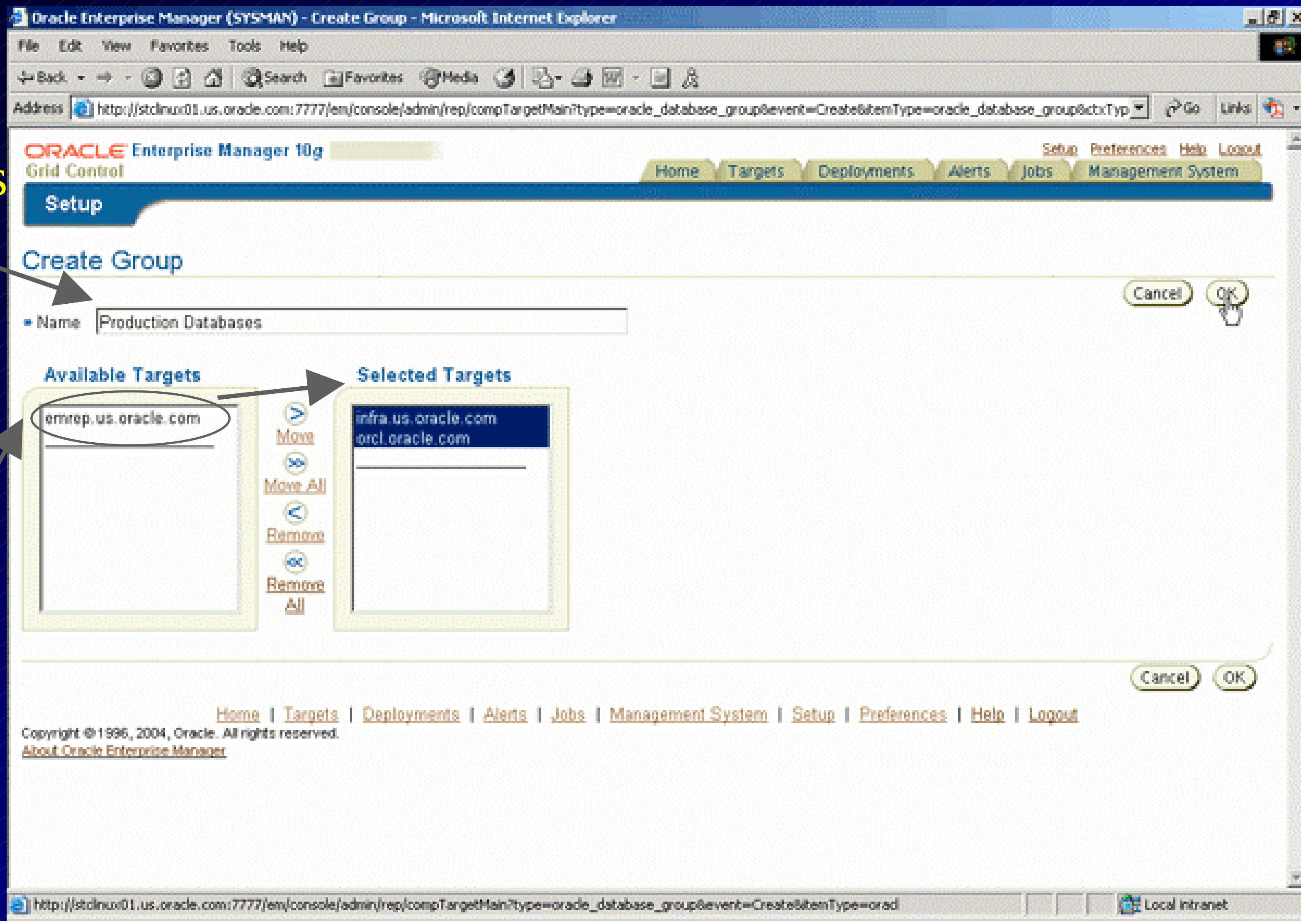
Monitor Targets… are they up?

Target Alerts



Database Performance

Create a Prod DB's Group

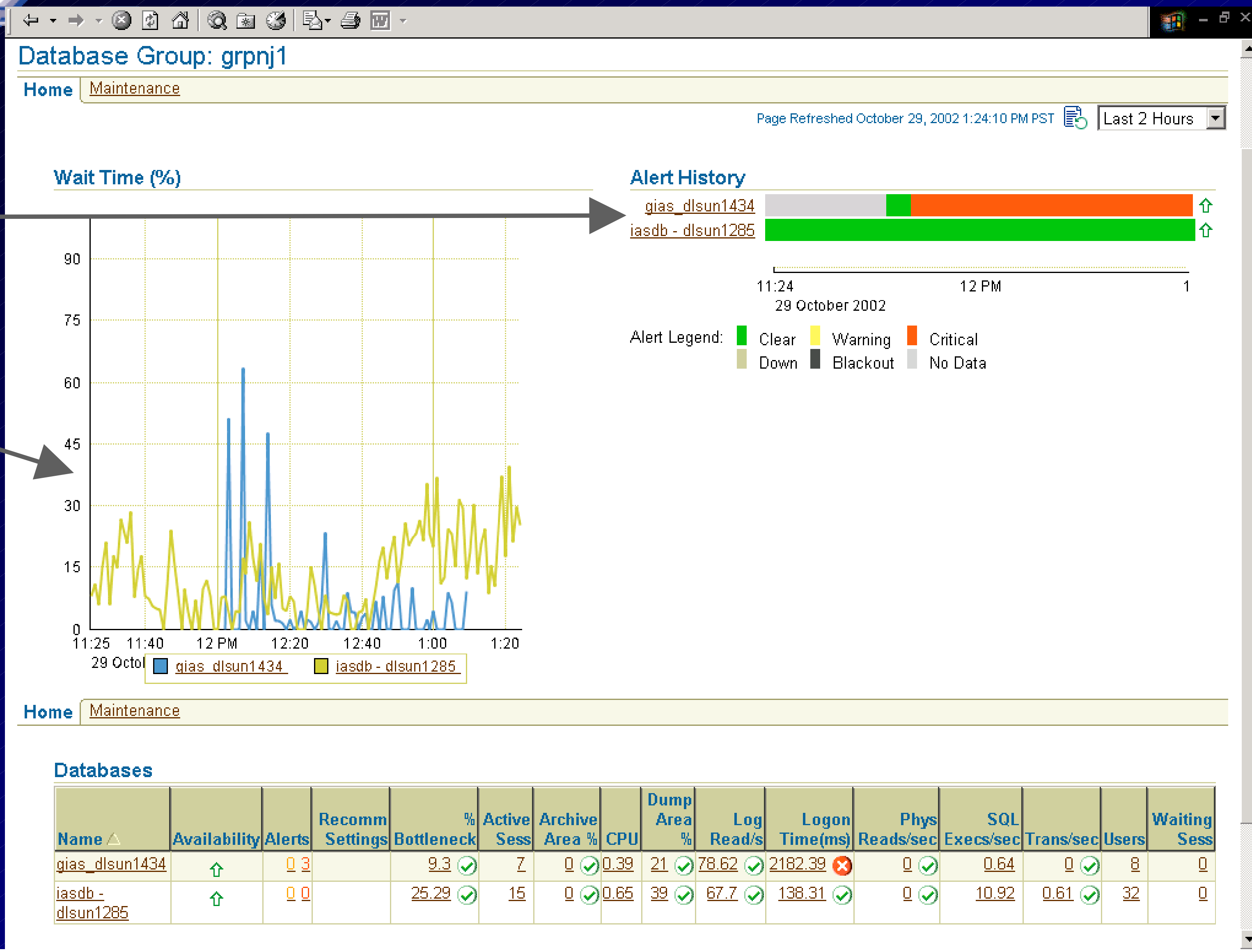


Pick Items for the Group

Database Performance

Alert Issues

Major Waits



Database Performance

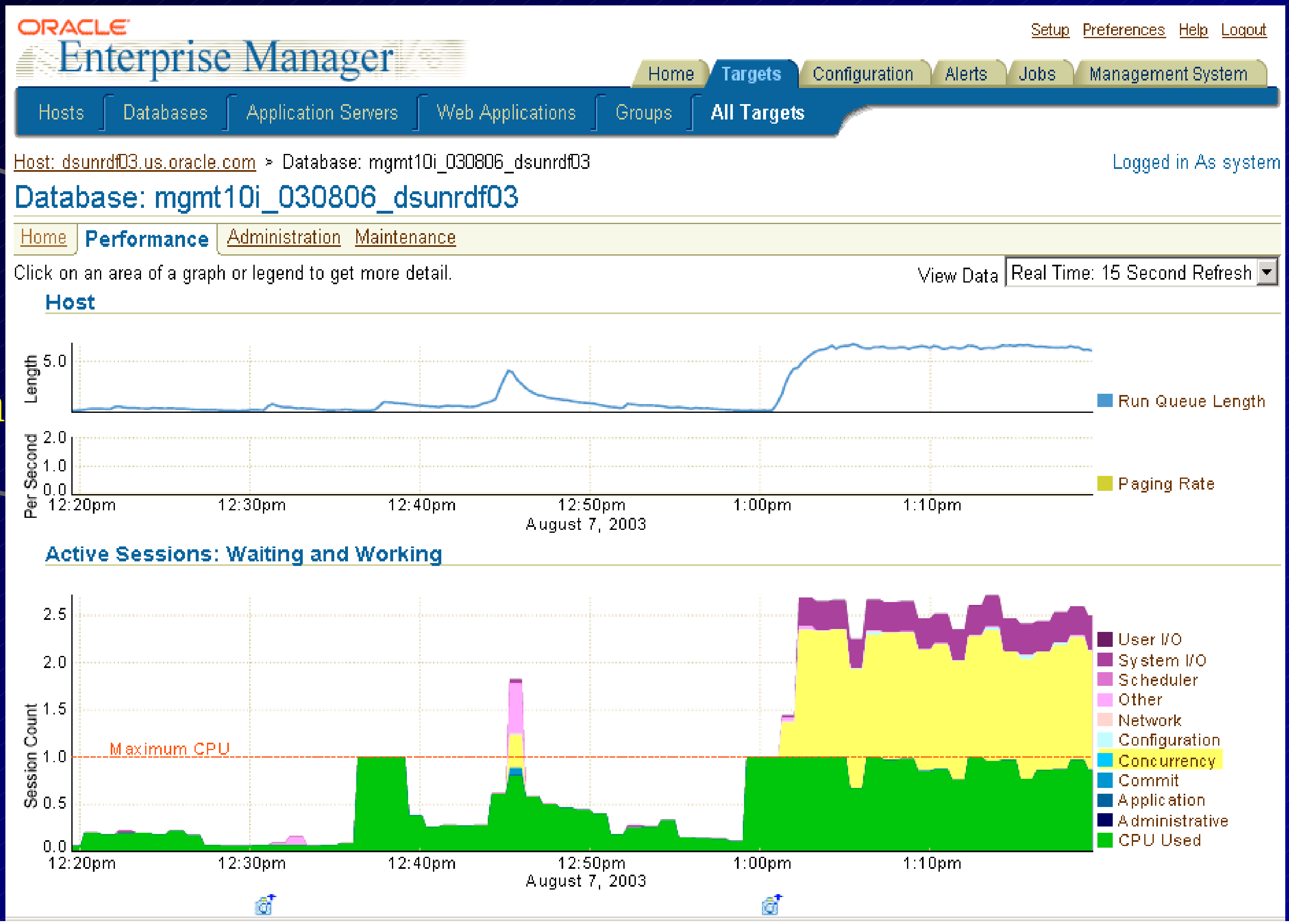
Monitor Database

We have a CPU issue!

Database Performance

Monitor Perform.

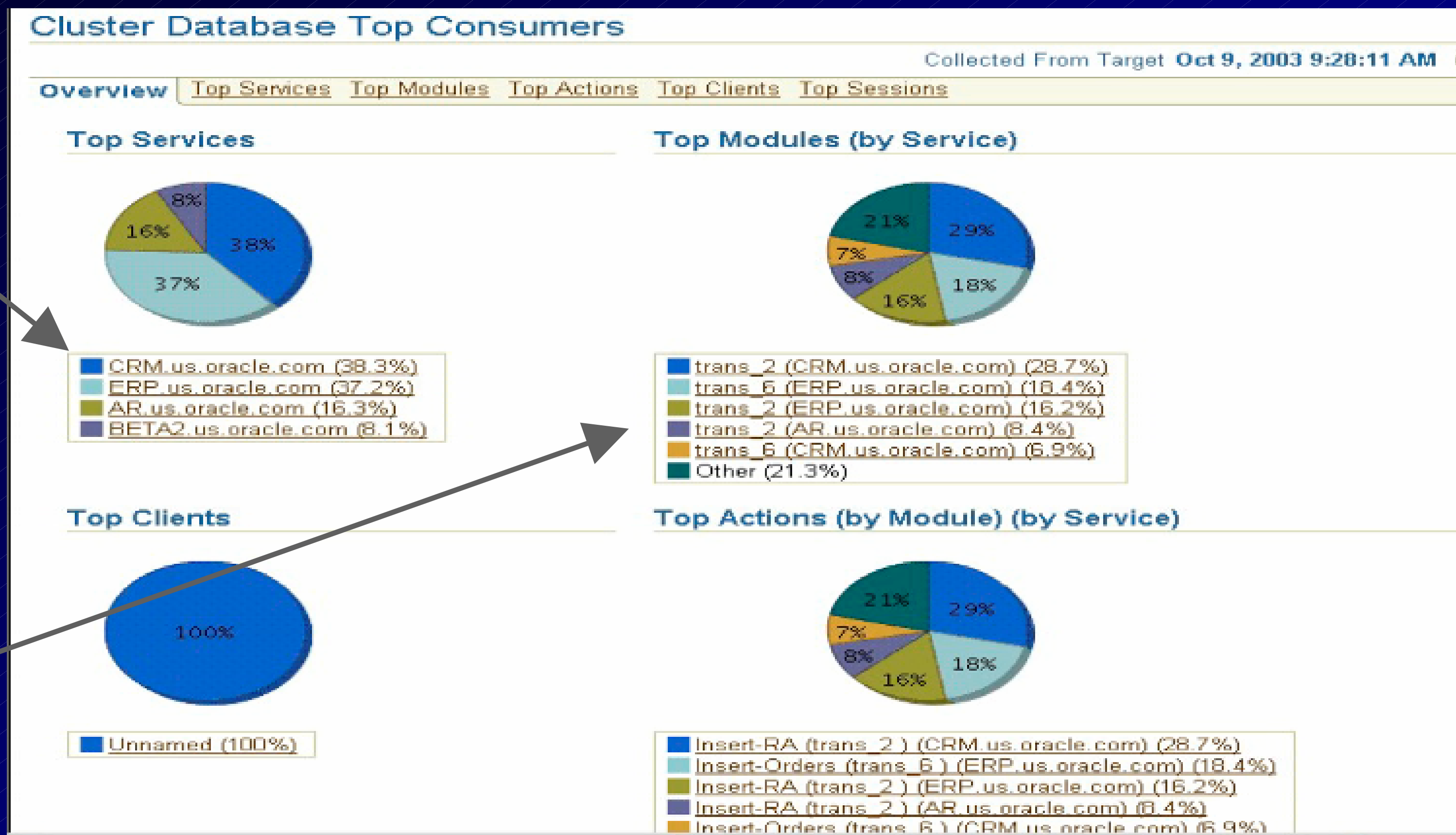
We have a Concurr. issue!



Grid Services - Automatic Workload Management

Top Services

Top Modules



Complete Presentation by Oracle's Erik Peterson at: http://www.oracleracsig.org

10g RAC Enhancements

GRID Control

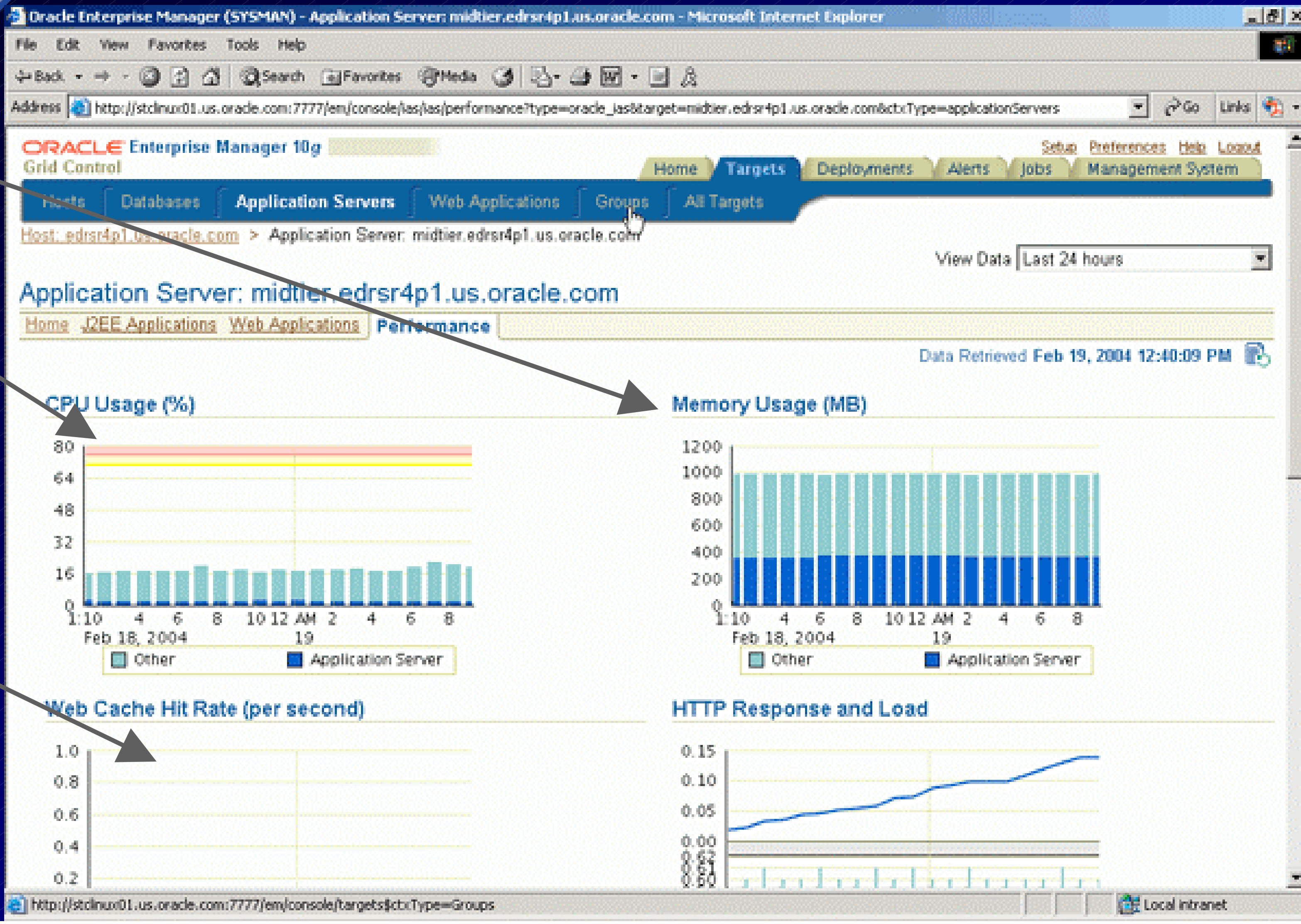
- Allows for RAC instance startup, shutdown
- Allows for RAC instance creation
- Allows for resource reallocation based on SLAs
- Allows for automatic provisioning when used with RAC, ASM and Linux

App. Server Performance

Memory

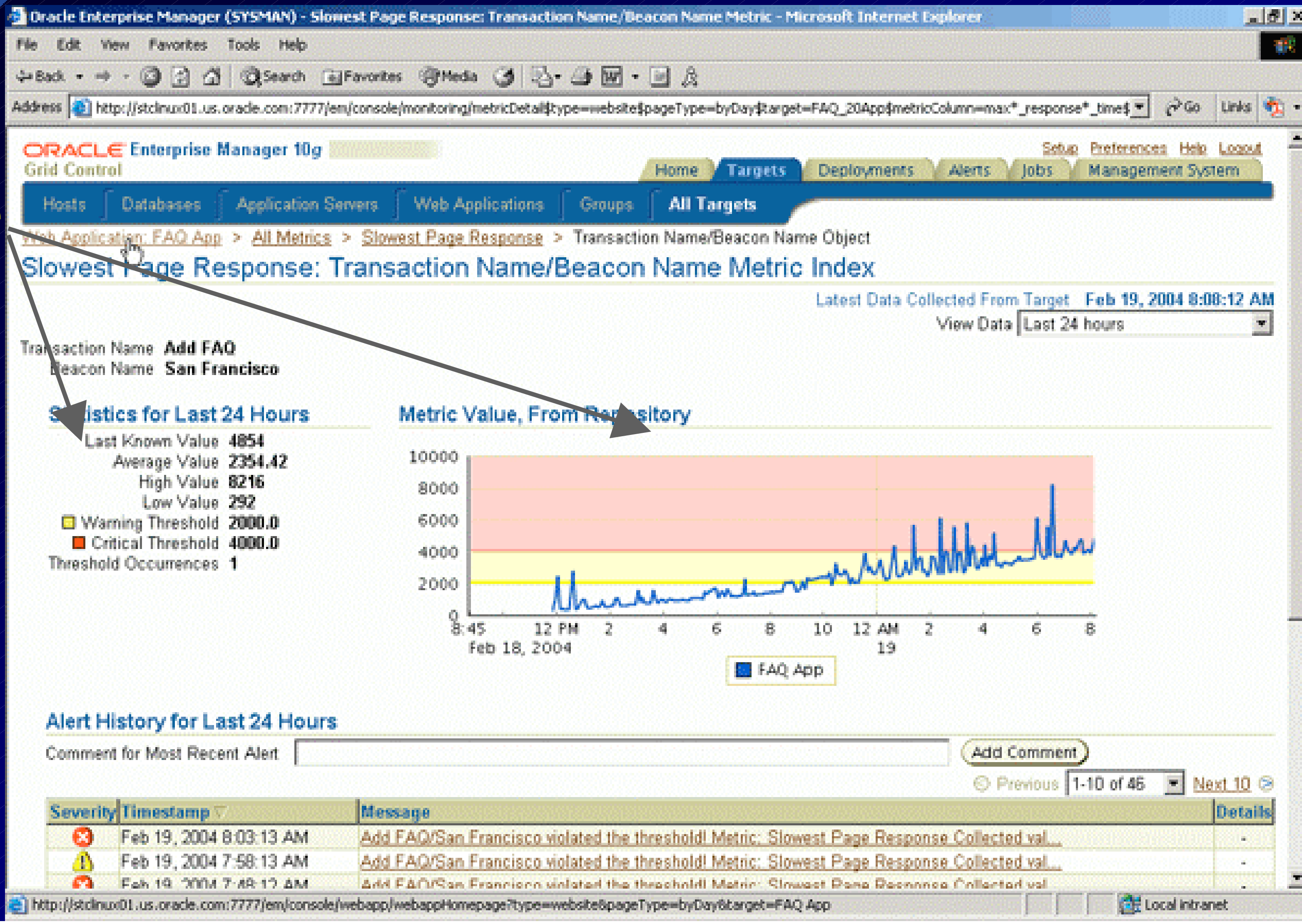
CPU

Web
Cache
Hits



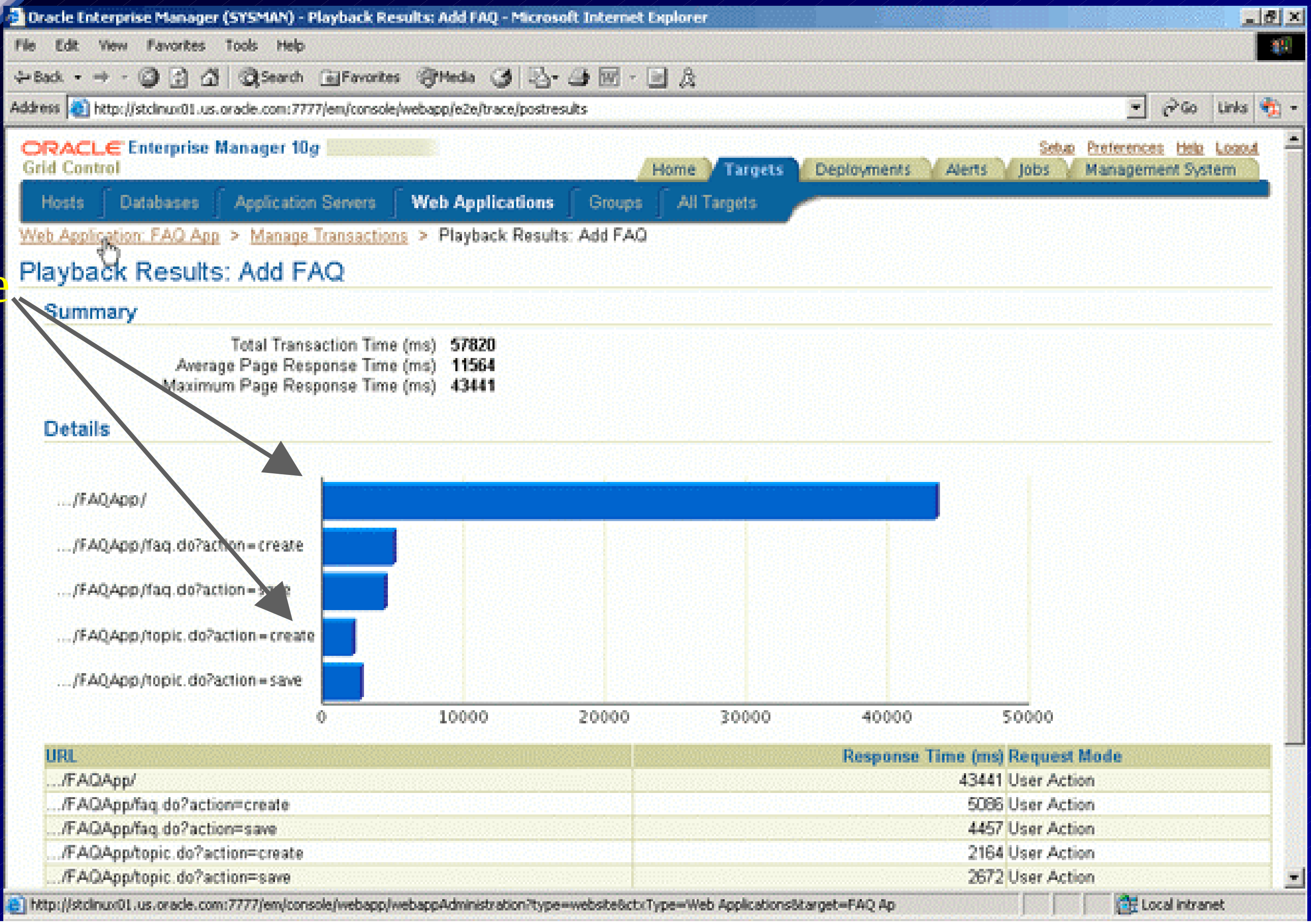
View the Web Application

Page Response Time Detail



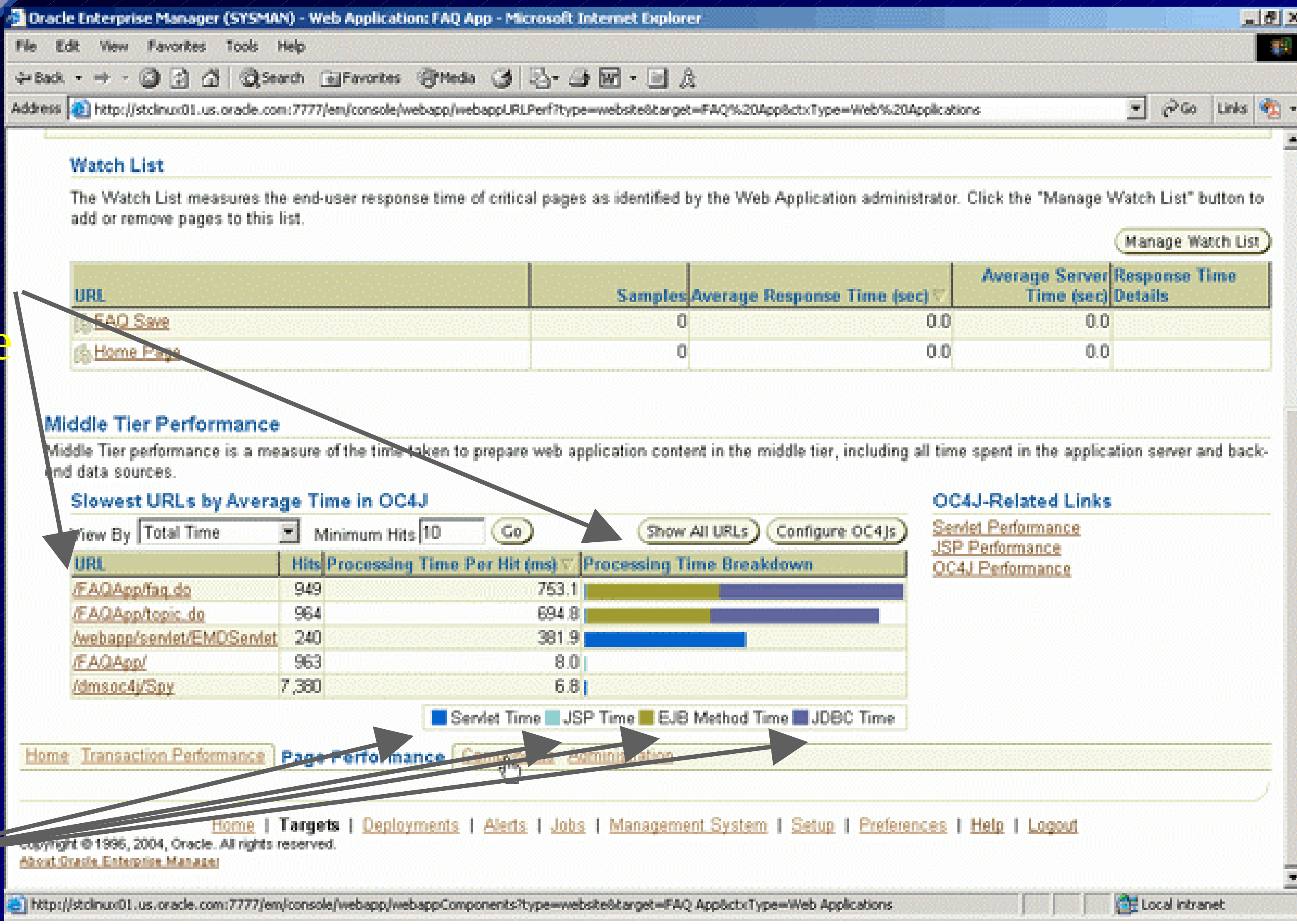
URL Response Times

Web Page Response Time all URL's



Middle Tier Performance

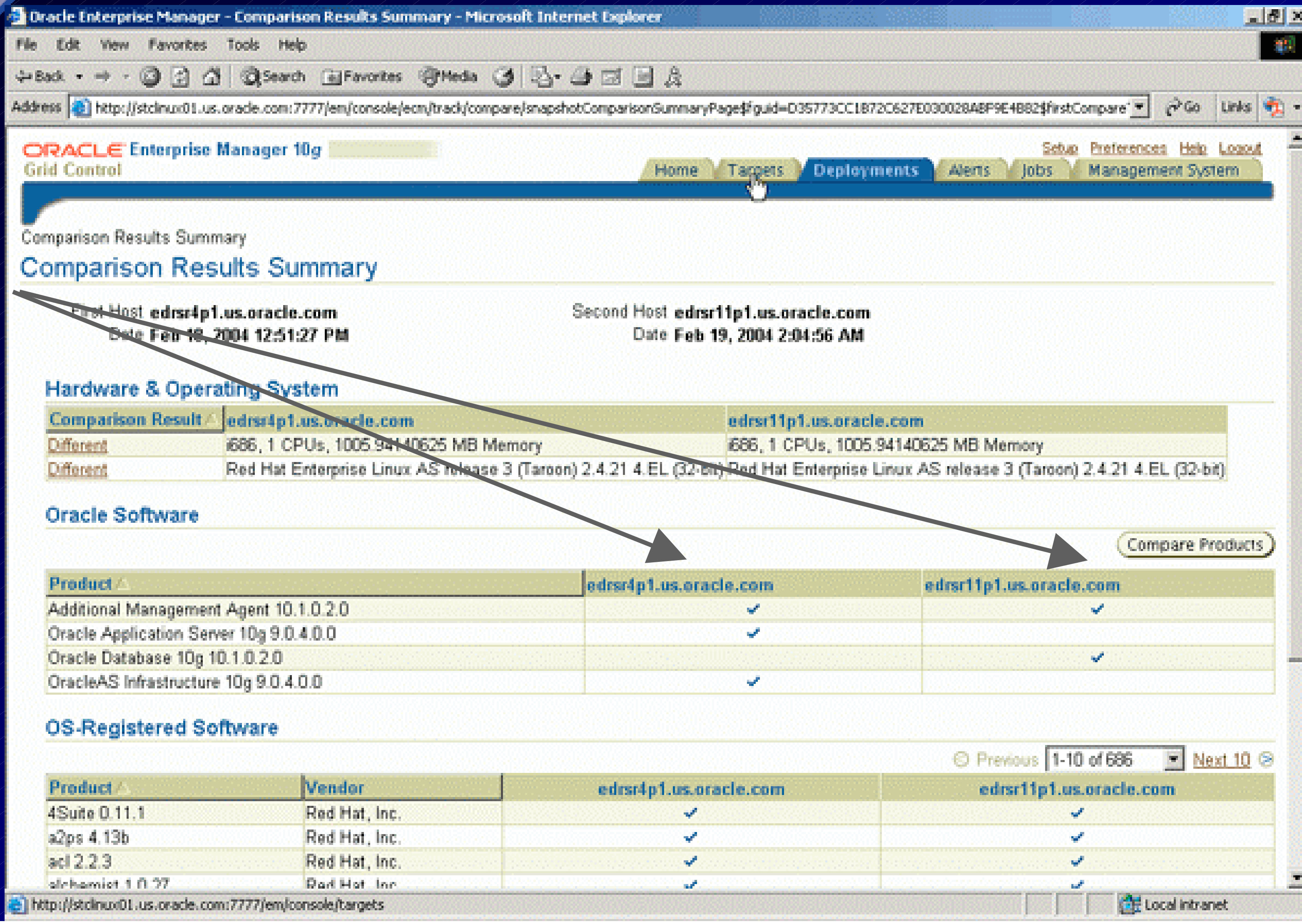
Web Page Detail Response Time all URL's



Splits Time into Parts

Host Performance

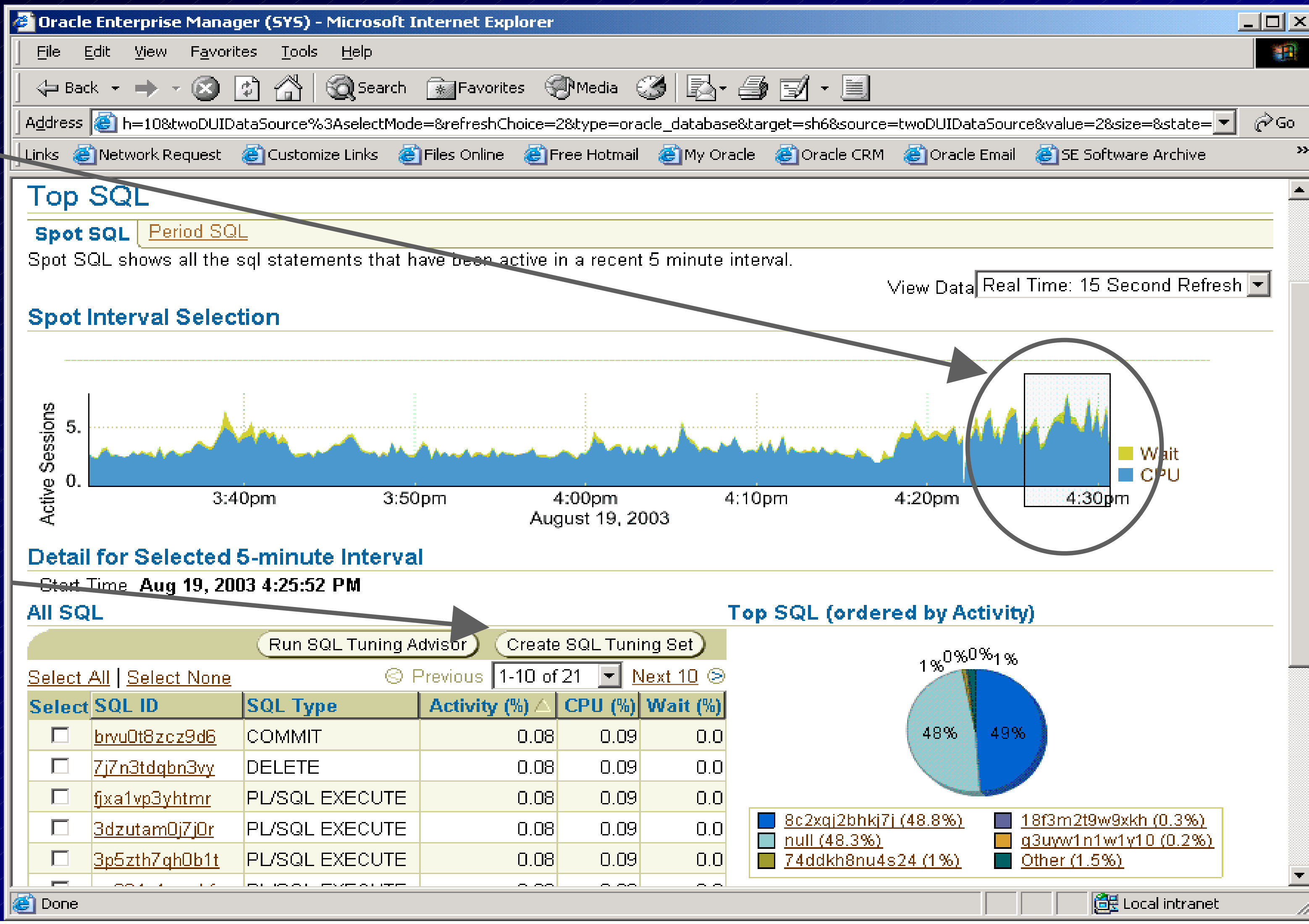
Side by Side Compare



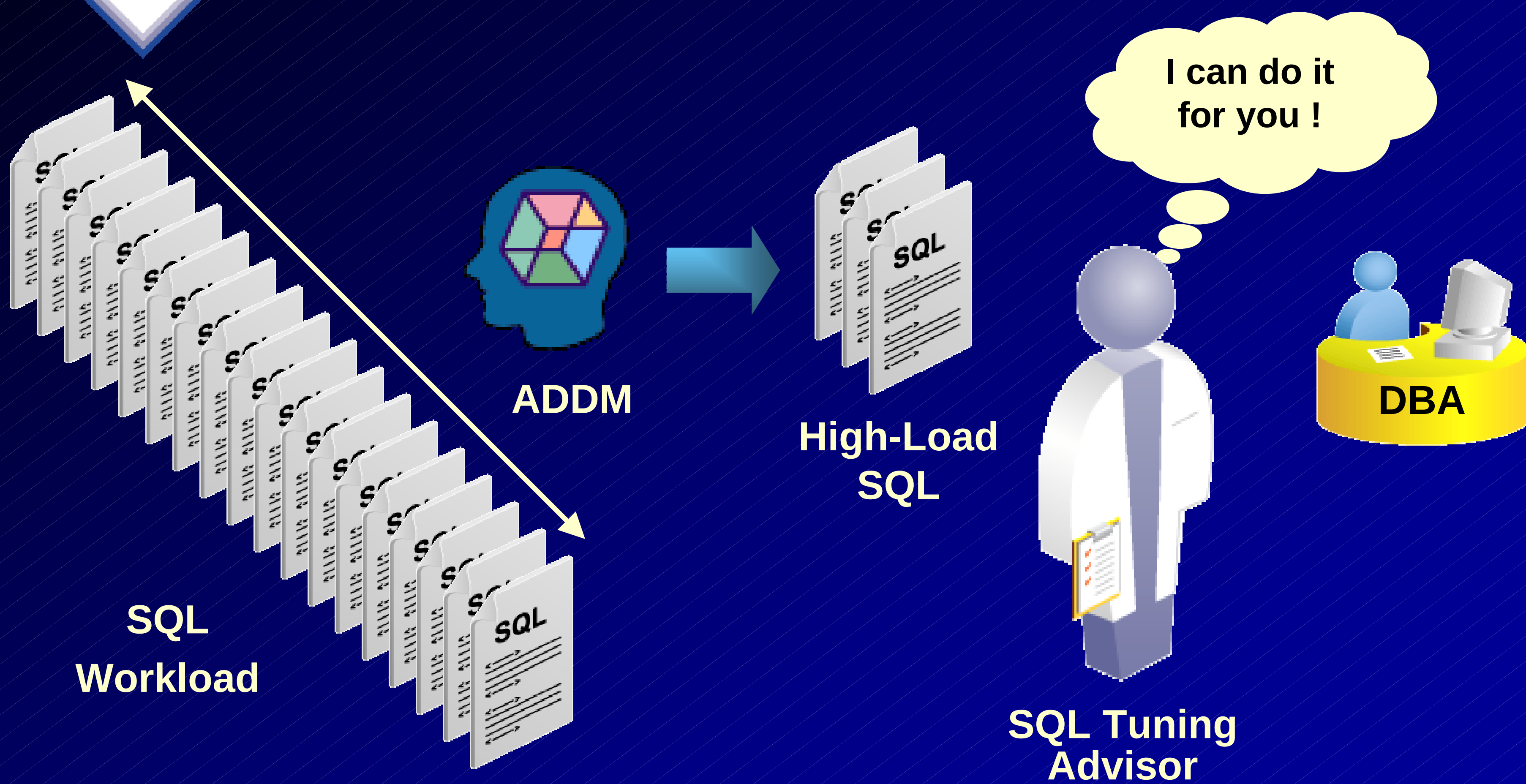
ADDM SQL Tuning Advisor

Drill into Top SQL for worst time period

Get Help!



Use Automatic Database Diagnostics Monitor (ADDM)



Automatic Workload Repository

Like a better statspack!

Repository of performance information

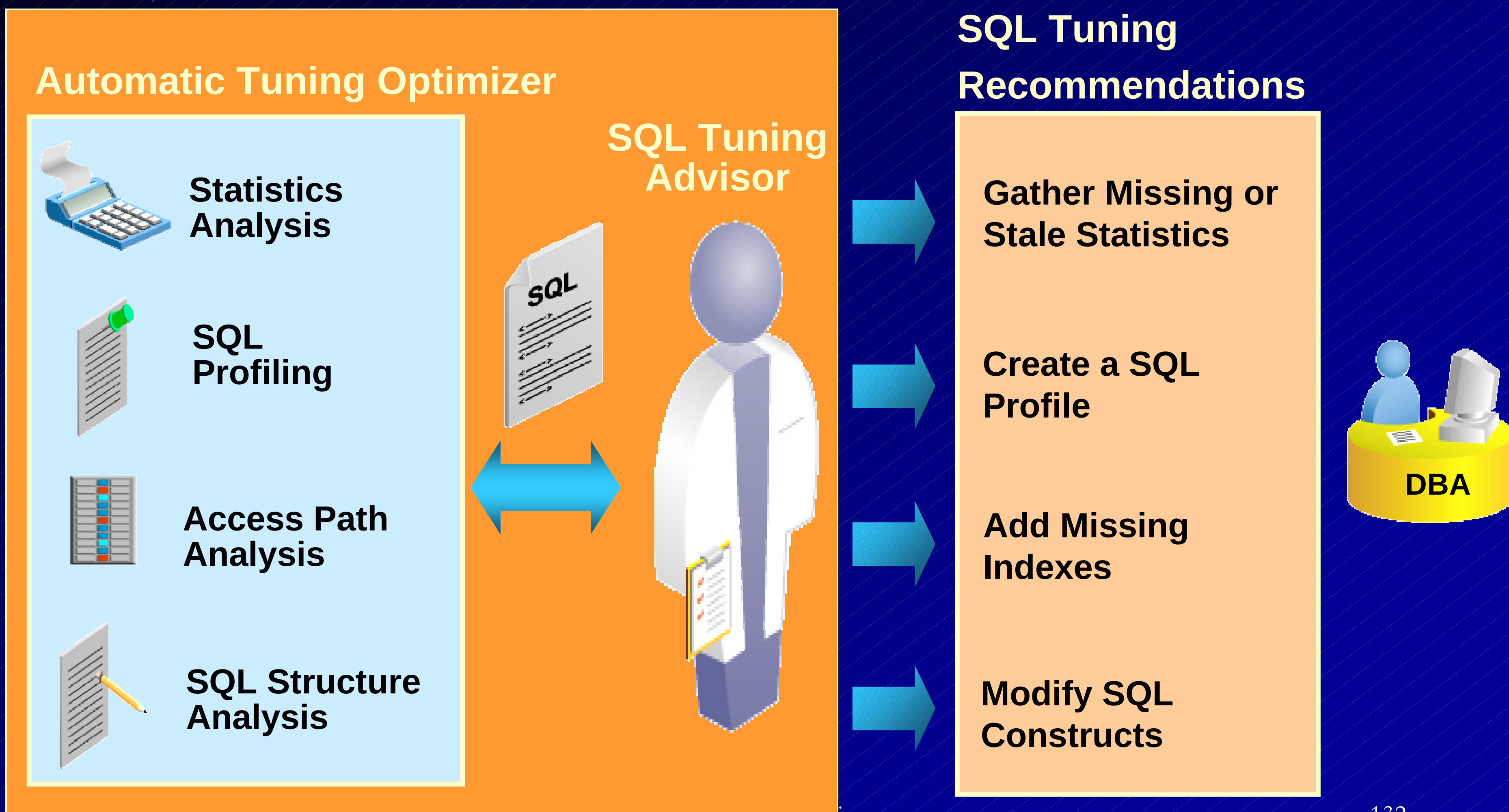
- Base statistics
- SQL statistics
- Metrics
- ACTIVE SESSION HISTORY

Workload data is captured every 30 minutes or manually and saved for 7 days by default

Self-manages its space requirements

Provides the basis for improved performance diagnostic facilities

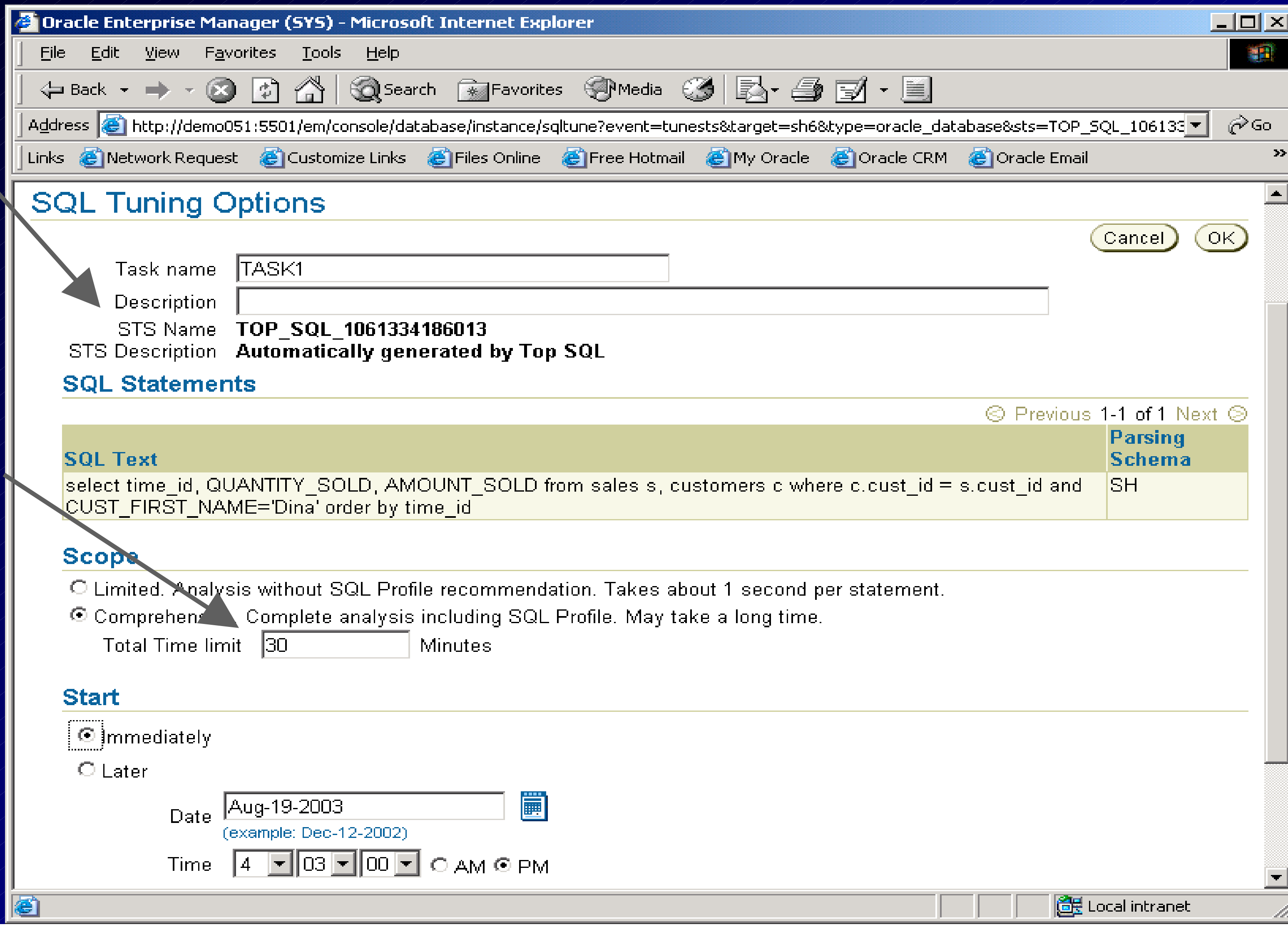
Automatic Tuning Optimizer (ATO)



Using SQL Tuning Sets

SQL Tuning Sets Info

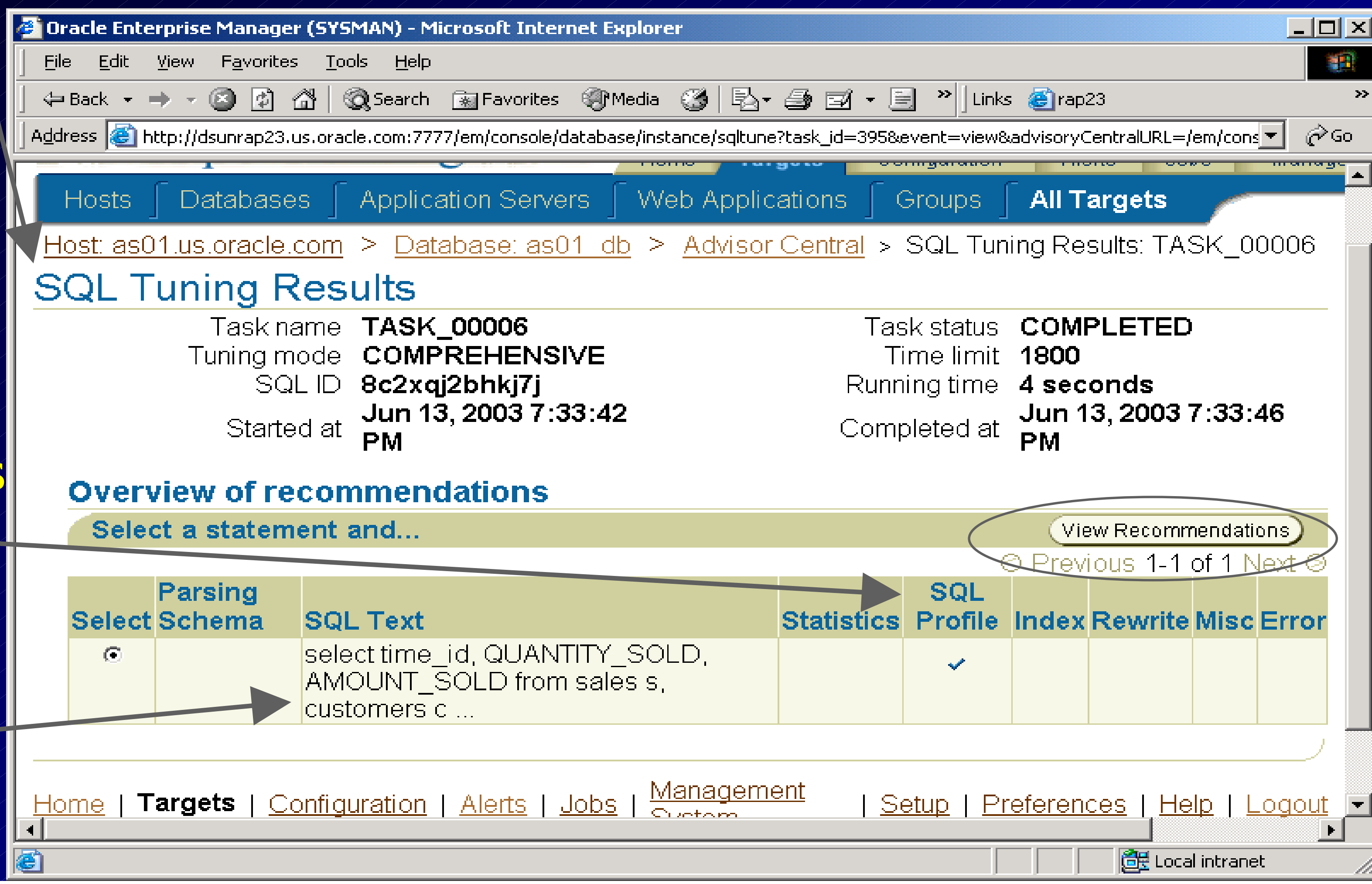
How long to work on this



Using SQL Tuning Sets

Tuning Results for this job

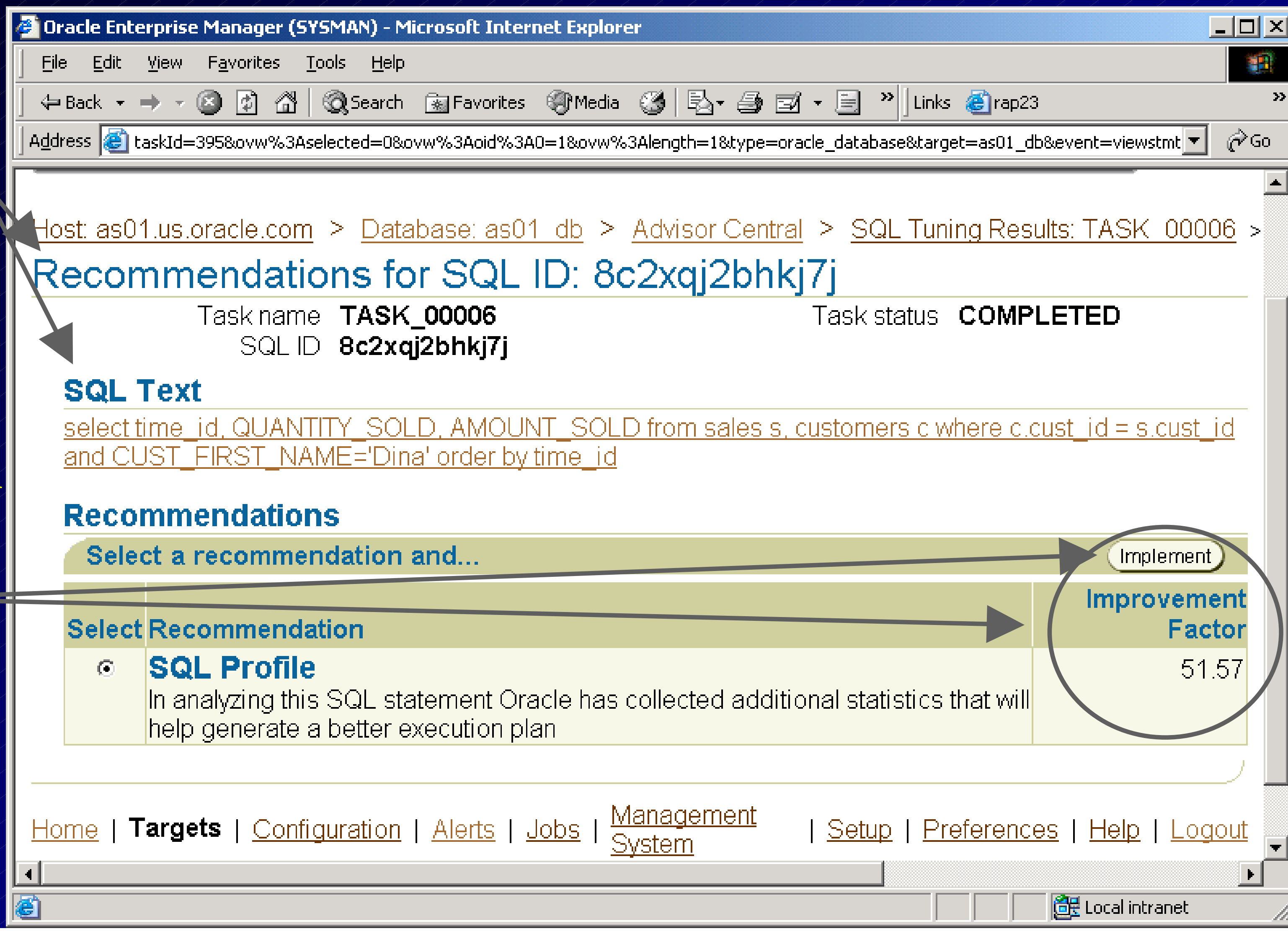
Suggests using a profile For this SQL



Using SQL Tuning Sets

Detailed info

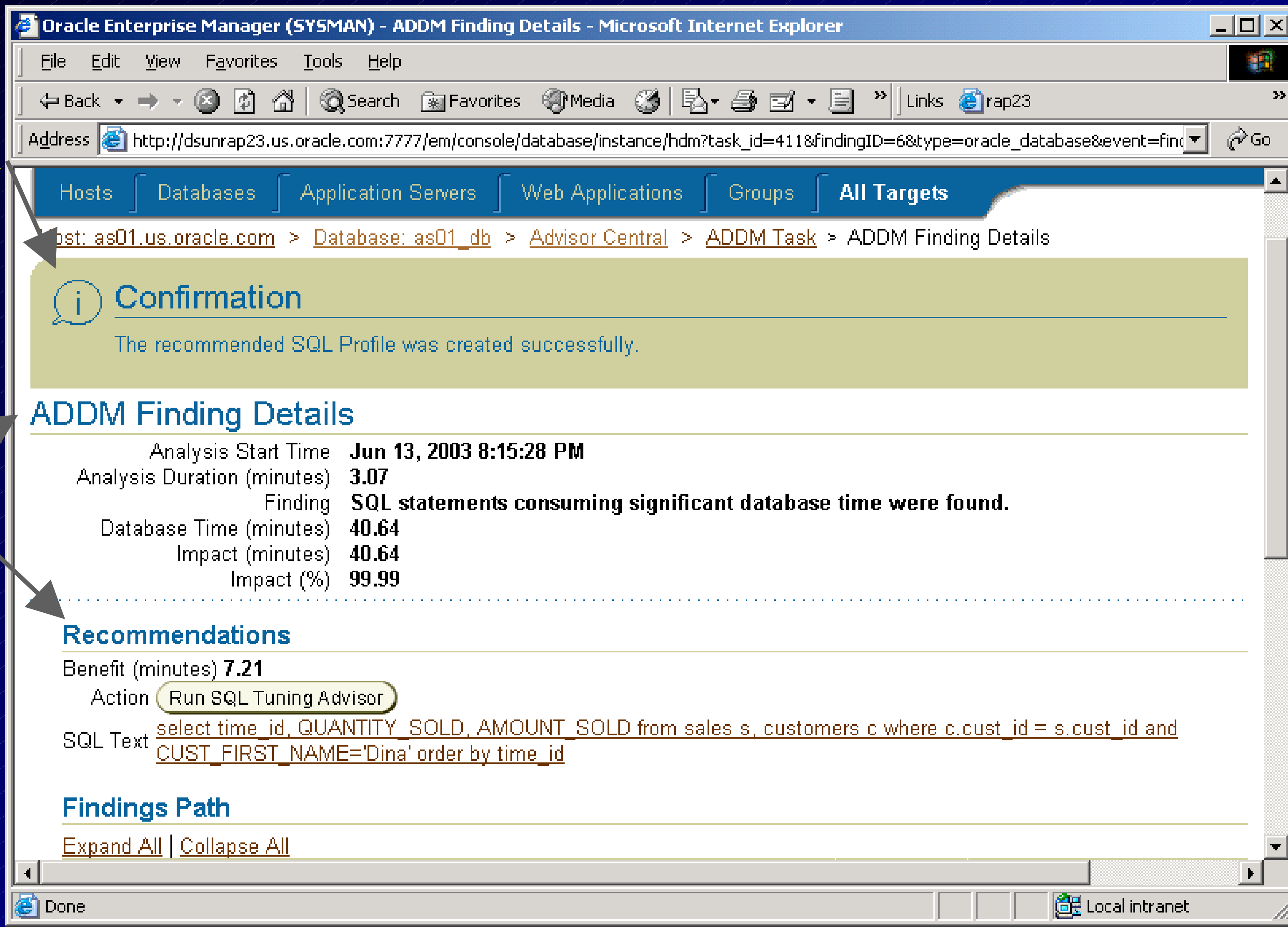
Improve Factor and click to Use



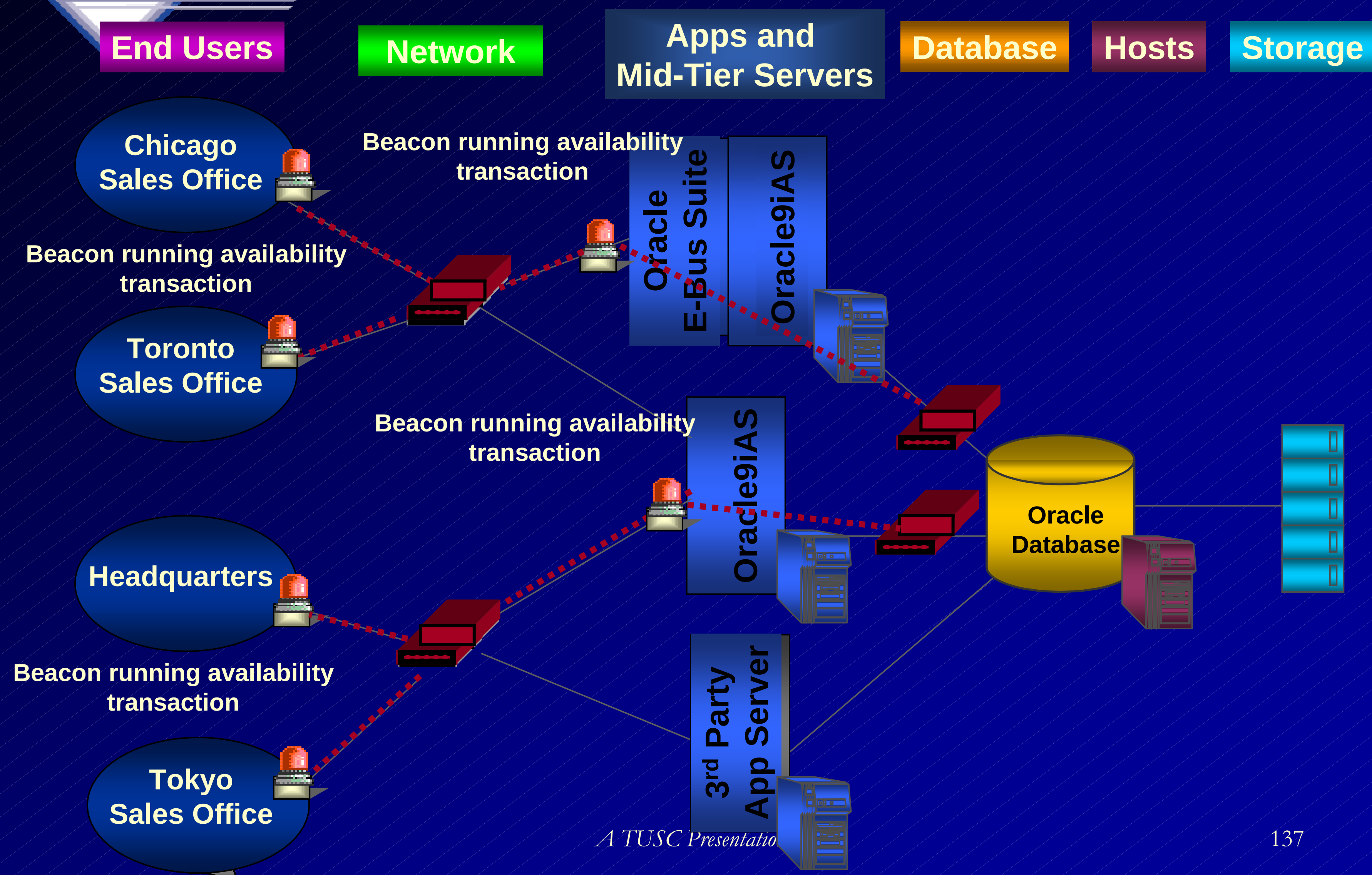
Using SQL Tuning Sets

Confirmed to use suggestion

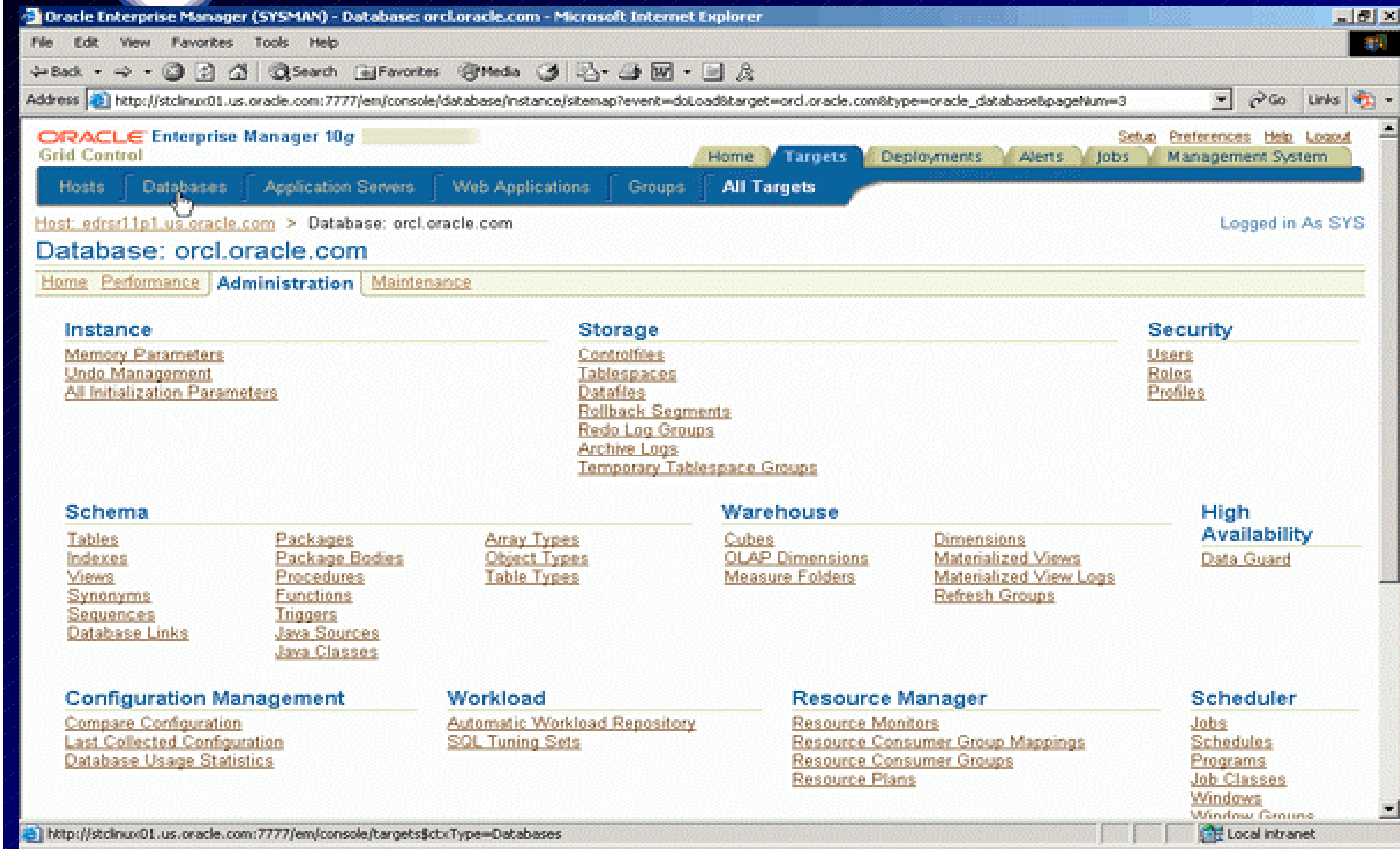
Detail



Availability Monitoring Topology



Enterprise Manager for the Grid; Many more Options!



Oracle Database 10g Release 1

Helpful Tuning Features (FYI Only)

DBMS_ADVANCED_REWRITE

Allows one SQL statement to be replaced by another behind the scenes every time someone runs it.

How to easily change that code so that it goes to your views instead of the original tables.

```
connect /
set echo off
alter session set query_rewrite_enabled = true;
alter session set query_rewrite_integrity = trusted;
set echo on
```

DBMS_ADVANCED_REWRITE

```
create table dept as
select        deptno, upper(dname) dname, loc
from  scott.dept;
```

```
select * from dept;
```

```
 DEPTNO DNAME                LOC
--------- -------------- --------------
       10 ACCOUNTING       NEW YORK
       20 RESEARCH         DALLAS
       30 SALES            CHICAGO
       40 OPERATIONS       BOSTON
```

DBMS_ADVANCED_REWRITE

```
create or replace view dept_v as
select    deptno, initcap(dname) dname, loc
from  dept
order by loc;
```

```
select * from dept_v;
```

DEPTNO	DNAME	LOC
40	Operations	BOSTON
30	Sales	CHICAGO
20	Research	DALLAS
10	Accounting	NEW YORK

DBMS_ADVANCED_REWRITE

(Note: system needs a grant on the package from sys)

```
begin

sys.dbms_advanced_rewrite.declare_rewrite_equivalence
   ( name             => 'DEMO_TIME',
     source_stmt      => 'select * from dept',
     destination_stmt => 'select * from dept_v',
     validate         => FALSE,
     rewrite_mode     => 'TEXT_MATCH' );
end;
/
```

DBMS_ADVANCED_REWRITE

select * from dept; (dept works like dept_v)

DEPTNO	DNAME	LOC
40	Operations	BOSTON
30	Sales	CHICAGO
20	Research	DALLAS
10	Accounting	NEW YORK

To remove it:

exec sys.dbms_advanced_rewrite.drop_rewrite_equivalence('DEMO_TIME');

Flush Buffer Cache

The new 10g feature allows the flush of the buffer cache. It is NOT intended for production use, but rather for system testing purposes.

This can help you in your tuning needs or as a band-aid if you have 'free buffer' waits (there are better ways to fix this like writing more often or increasing the DB_CACHE_SIZE)

Note that any Oracle I/O not done in the SGA counts as a physical I/O. If your system has O/S caching or disk caching, the actual I/O that shows up as physical may indeed be a memory read outside of Oracle.

To flush the buffer cache perform the following:

SQL> ALTER SYSTEM FLUSH BUFFER_CACHE;

Flush Buffer Cache – Example

```
select count(*) from tab1;

 COUNT(*)
----------------
     1147

Execution Plan
----------------------------------------------------------
   0      SELECT STATEMENT Optimizer=CHOOSE (Cost=4 Card=1)
   1    0   SORT (AGGREGATE)
   2    1     TABLE ACCESS (FULL) OF 'TAB1' (TABLE) (Cost=4 Card=1147)

Statistics
----------------------------------------------------------
          0  db block gets
          7  consistent gets
          6  physical reads
```

Flush Buffer Cache – Example

select count(*) from tab1; (Run it again and the physical reads go away)

```
 COUNT(*)
----------------
     1147

Execution Plan
----------------------------------------------------------
   0      SELECT STATEMENT Optimizer=CHOOSE (Cost=4 Card=1)
   1    0   SORT (AGGREGATE)
   2    1     TABLE ACCESS (FULL) OF 'TAB1' (TABLE) (Cost=4 Card=1147)

Statistics
----------------------------------------------------------
          0  db block gets
          7  consistent gets
          0  physical reads
```

Flush Buffer Cache – Example

ALTER SYSTEM FLUSH BUFFER_CACHE;

System altered.

select count(*) from tab1; (Flush the cache and the physical reads are back)

```
 COUNT(*)
----------------
     1147

Execution Plan
----------------------------------------------------------------------------------------------------------------------
   0      SELECT STATEMENT Optimizer=CHOOSE (Cost=4 Card=1)
   1    0   SORT (AGGREGATE)
   2    1     TABLE ACCESS (FULL) OF 'TAB1' (TABLE) (Cost=4 Card=1147)

Statistics
----------------------------------------------------------
        0  db block gets
        7  consistent gets
        6  physical reads
```

Flush Buffer Cache - Internal

What about V$/X$ information?

select name, value
from v$parameter (this internally accesses x$ksppcv & x$ksppi)
where name like '%compatible%';

```
NAME                           VALUE
------------------------------ ----------------------------------------------------------
compatible                     10.1.0.1.0
plsql_v2_compatibility         FALSE

Statistics
----------------------------------------------------------
      283  recursive calls
        0  db block gets
       69  consistent gets
       31  physical reads
```

Flush Buffer Cache - Internal

Run it a second time and you get:

Statistics

----

0 recursive calls

0 db block gets

0 consistent gets

0 physical reads

2 rows processed

ALTER SYSTEM FLUSH BUFFER CACHE and you get the same:

Statistics

----

0 recursive calls

0 db block gets

0 consistent gets

0 physical reads

2 rows processed

Oracle Database 10g Release 2

Helpful Tuning Features (FYI Only)

Oracle Database 10g Release 2 – Improved Data Warehousing

Performance

- Up to 5x improvement in sort performance
- Up to 3x improvement in aggregate performance

Partitioning

- Increase maximum number of partitions per table from 64k to 1024K-1
- Support 'multidimensional' partition-pruning

Analytics

- Support standard linear algebra libraries within PL/SQL
- Enhancements to the SQL 'model' clause
- Decision trees in Oracle Data Mining

ETL

- DML Error-logging

New in-Memory Sort Algorithm

New improved sort implementation

- Hash-based implementation

Dramatic transparent performance improvements

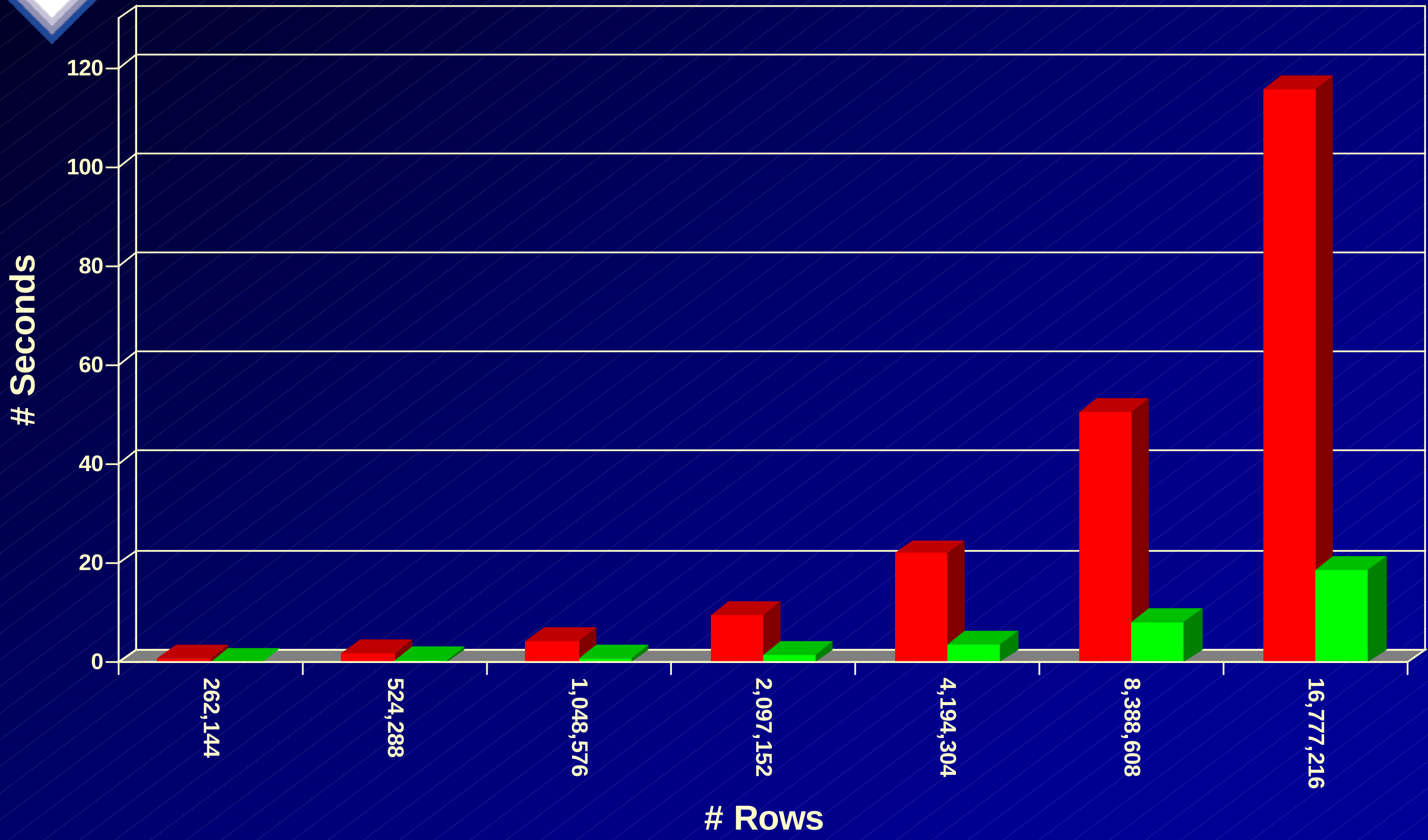
- **Fully leverages large amounts of memory**
- **Sort operation can be up to 5 times faster (*)**

Improvements depending on sort characteristics

- Higher cardinality, more improvements
- Faster CPU, more improvements
- Select fewer columns, more improvements

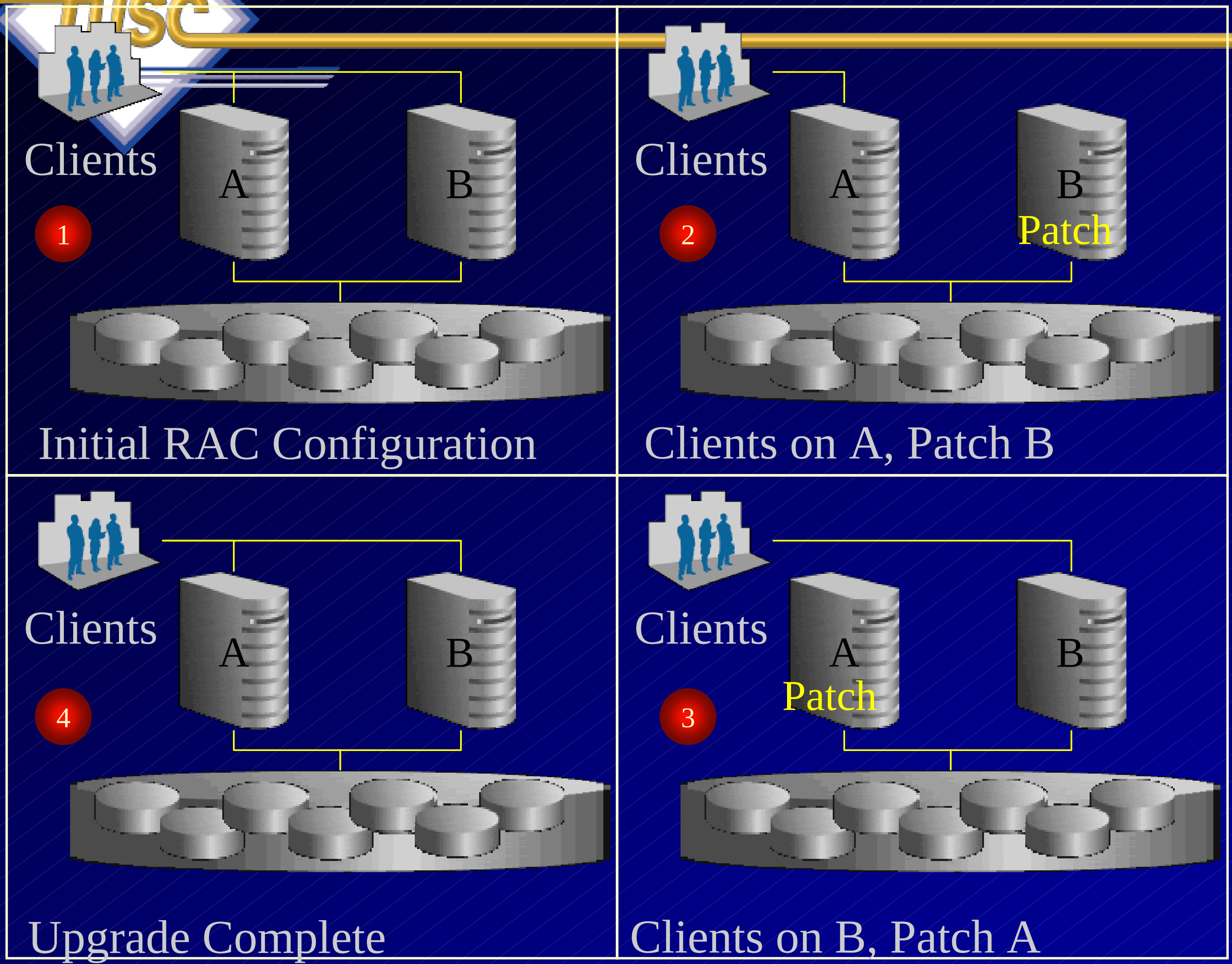
(*) Total improvement depends on the weight of the sort in the overall operation

Sort Performance Improvements



Test details in “Sort Performance Improvements in Oracle Database 10*g* Release 2” by Mark van de Wiel, June 2005

Rolling Patch Upgrade using RAC



Oracle Patch Upgrades

Operating System Upgrades

Hardware Upgrades

Oracle Database 10g Release 2 - Summary

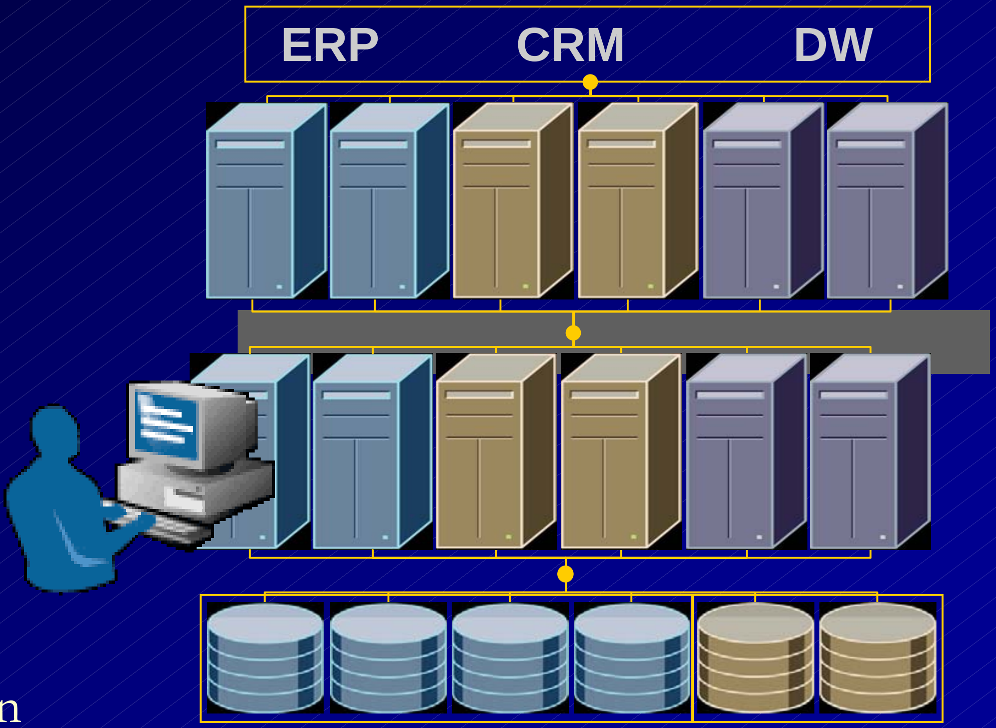
Lowest Cost

- Oracle Backup
- Improved Sort
- Streams Performance
- Oracle Clusterware

Highest Quality of Service

- Rolling Upgrades
- Fast-Start Failover
- Transparent Data Encryption

Easier to Manage



Summary

- Tune each instance in a database cluster independently prior to tuning RAC.
- Reduce contention for blocks to improve service time.
- Operate on a well-tuned network.
- Monitor system load.
- Use V$ views to monitor RAC systems.
- STATSPACK contains vital information for RAC systems.
- New Features of Oracle10g will ease administration even further.

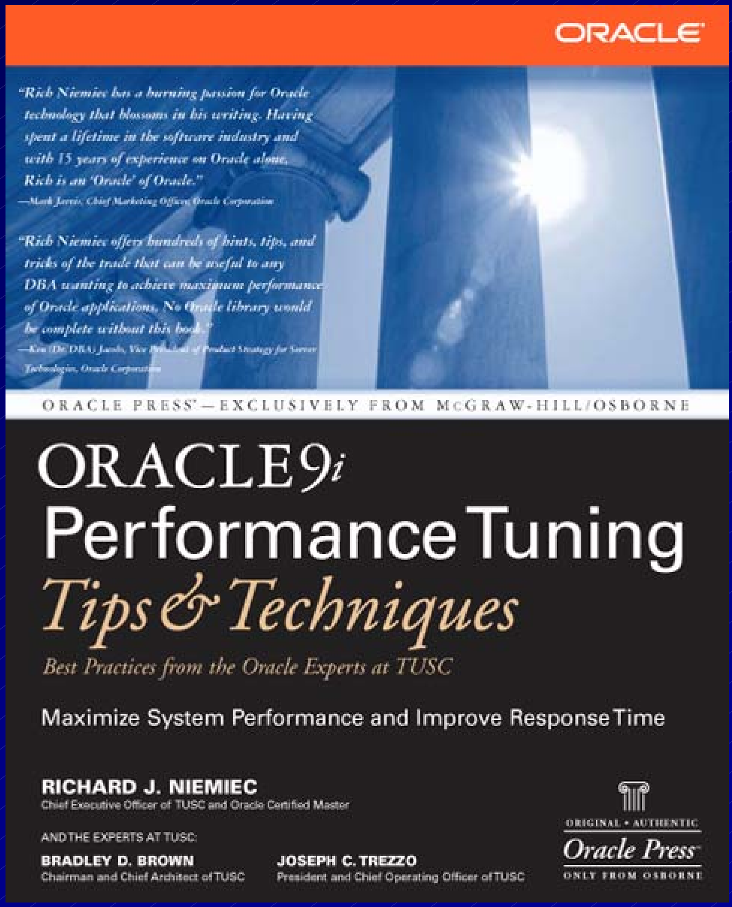
"You must be the change you wish to see in the world."

--Mahatma Gandhi

For More Information

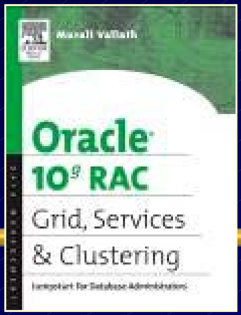
www.tusc.com

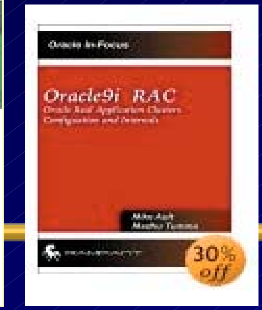
Oracle9i Performance Tuning Tips & Techniques; Richard J. Niemiec; Oracle Press (May 2003)

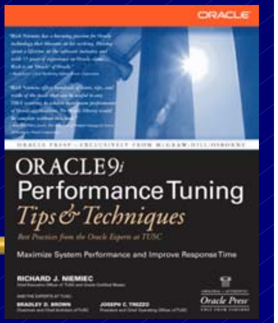


"Excellence is the result of caring more than others think is wise; risking more than others think is safe. Dreaming more than others think is practical and expecting more than others think is possible."

References







- *Special thanks to Murali Vallath, Steve Adams, Mike Ault, Brad Brown, Kevin Gilpin, Herve Lejeune, Randy Swanson and Joe Trezzo.*
- *Oracle9i Performance Tuning Tips & Techniques*, Rich Niemiec
- *The Self-managing Database: Automatic Performance Diagnosis;* Karl Dias & Mark Ramacher, Oracle Corporation
- *EM Grid Control 10g;* otn.oracle.com, Oracle Corporation
- *Oracle Enterprise Manager 10g: Making the Grid a Reality*; Jay Rossiter, Oracle Corporation
- *The Self-Managing Database: Guided Application and SQL Tuning;* Benoit Dageville, Oracle Corporation
- *The New Enterprise Manager: End to End Performance Management of Oracle*; Julie Wong & Arsalan Farooq, Oracle Corporation
- *Enterprise Manager : Scalable Oracle Management*; John Kennedy, Oracle Corporation
- *Performance Tuning 10g RAC on Linux, www.moug.org by Muralli Vallath.*
- Oracle 10g documentation & Oracle 9i Concepts manual

References

- *Oracle Database 10g Performance Overview*; **Hervé Lejeune, Oracle Corporation**
- *Oracle 10g; Penny Avril,*, **Oracle Corporation**
- *Internals of Real Application Cluster*, **Madhu Tumma, Credit Suisse First Boston**
- *Oracle9i RAC; Real Application Clusters Configuration and Internals*, **Mike Ault & Madhu Tumma**
- **Oracle Database 10g *A*utomated Features , Mike Ault, TUSC**
- **Oracle Database 10g New Features, Mike Ault, Daniel Liu, Madhu Tumma, Rampant Technical Press, 2003, www.rampant.cc**
- **Oracle Database 10g - The World's First Self-Managing, Grid-Ready Database Arrives, Kelli Wiseth, Oracle Technology Network, 2003, otn.oracle.com**
- *Oracle Tuning Presentation, Oracle Corporation*
- **www.tusc.com, www.oracle.com, www.ixora.com, www.laoug.org, www.ioug.org, technet.oracle.com**
- *Real Application Clusters, Real Customers Real Results*, **Erik Peterson, Technical Manager, RAC, Oracle Corp.**
- **Oracle 9i RAC class & instructor's comments**

References

- *Oracle9iAS Clusters: Solutions for Scalability and Availability, Chet Fryjoff, Product Manager, Oracle Corporation*
- *Oracle RAC and Linux in the real enterprise*, **Mark Clark, Director, Merrill Lynch Europe PLC, Global Database Technologies**
- *Tips for Tuning Oracle9i RAC on Linux*, **Kurt Engeleiter, Van Okamura, Oracle**
- *Leveraging Oracle9i RAC on Intel-based servers to build an "Adaptive Architecture*, **Stephen White, Cap Gemini Ernst & Young, Dr Don Mowbray, Oracle, Werner Schueler, Intel**
- *Running YOUR Applications on Real Application Clusters (RAC); RAC Deployment Best Practices*, **Kirk McGowan, Oracle Corporation**
- *The Present, The Future but not Science Fiction; Real Application Clusters Development*, **Angelo Pruscino, Oracle**
- *Building the Adaptive Enterprise; Adaptive Architecture and Oracle*, **Malcolm Carnegie, Cap Gemini Ernst & Young**
- *Deploying a Highly Manageable Oracle9i Real Applications Database*, **Bill Kehoe, Oracle**
- *Getting the most out of your database*, **Andy Mendelsohn, SVP Server Technologies, Oracle Corporation**

TUSC Services

Oracle Technical Solutions
- **Full-Life Cycle Development Projects**
- **Enterprise Architecture**
- **Database Services**

Oracle Application Solutions
- **Oracle Applications Implementations/Upgrades**
- **Oracle Applications Tuning**

Managed Services
- **24x7x365 Remote Monitoring & Management**
- **Functional & Technical Support**

Training & Mentoring

Oracle Authorized Reseller

Copyright Information

Neither TUSC, NYOUG, Oracle nor the authors guarantee this document to be error-free. Please provide comments/questions to rich@tusc.com.

Enjoy the Day!

Call with questions: (800) 755-TUSC; rich@tusc.com

www.tusc.com