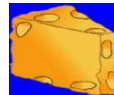


ORACLE®



ORACLE®

Who Ate My DB Time?



Techniques for Oracle DB Performance Analysis

Gongloor, Prabhaker (GP)

Product Management Director

Real Application Testing, DB Manageability & Diagnosability

Oracle America



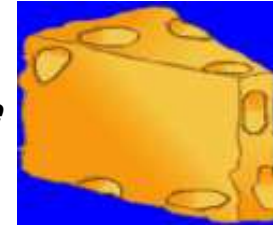
Agenda

- Motivation
- NEWS: DB Time
- DB Time Fundamentals
- DB Time Method for Performance Tuning
- DB Time Based Tools
 - ADDM
 - Enterprise Manager

Change is the Only Constant: Handwriting on the Wall



Change Happens
They Keep Moving the Cheese



Anticipate Change
Get Ready For The Cheese To Move



Monitor Change
*Smell The Cheese Often So You
Know When It Is Getting Old*
Adapt To Change Quickly
*The Quicker You Let Go Of Old Cheese,
The Sooner You Can Enjoy New Cheese*



Change
Move With The Cheese

Enjoy Change!
*Savor The Adventure And
Enjoy The Taste Of New Cheese!*

**Be Ready To Quickly Change
& Enjoy It, Again, And Again!**
They Keep Moving The Cheese



NEWS: DB Time is Not News



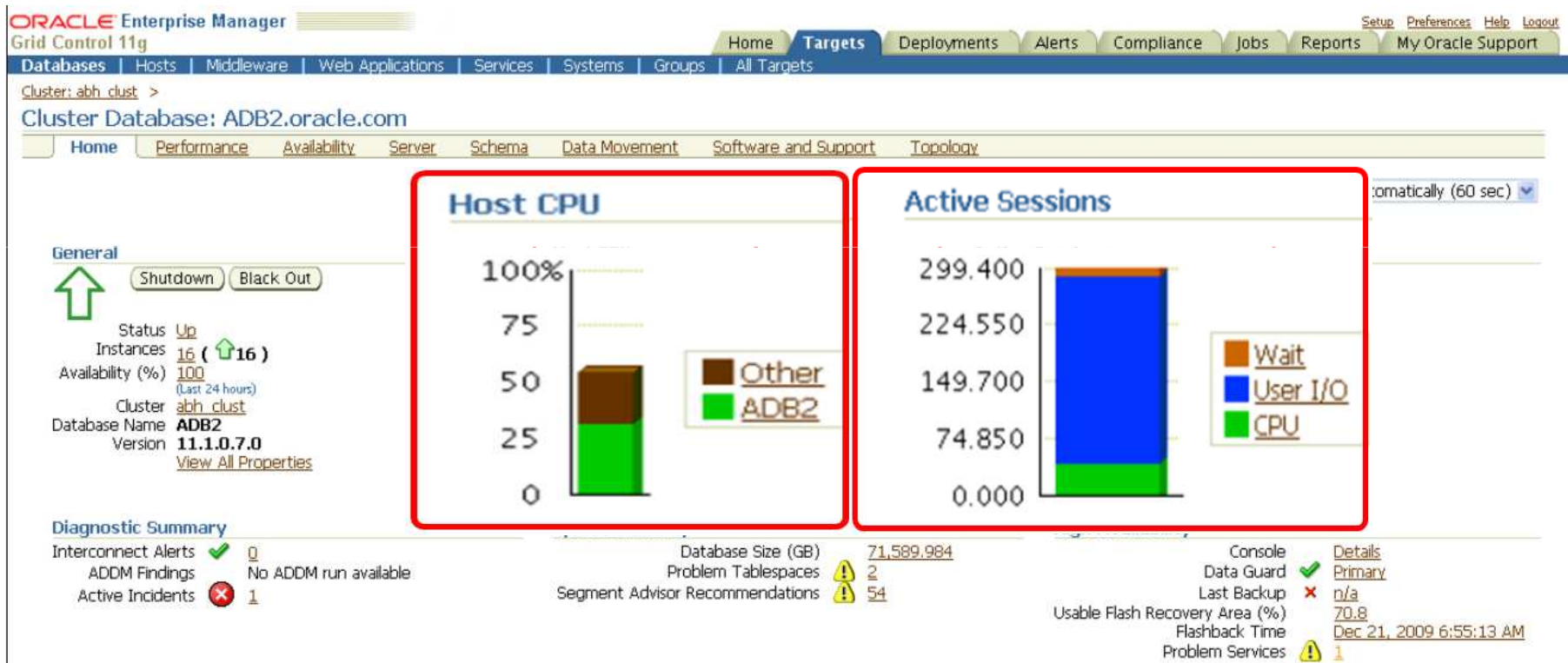


DB Time is Pervasive, Familiar, Evolutionary

- DB Time is the fundamental measure of Oracle performance
 - Arrived at progressively but designed-in from 10g
- You are likely using DB Time methods already
 - AWR and ASH reports
 - Enterprise Manager
- DB Time is performance analysis science matured
 - Wait event analysis, YAPP and tracing are all its ancestors

DB Time on the Database Home Page

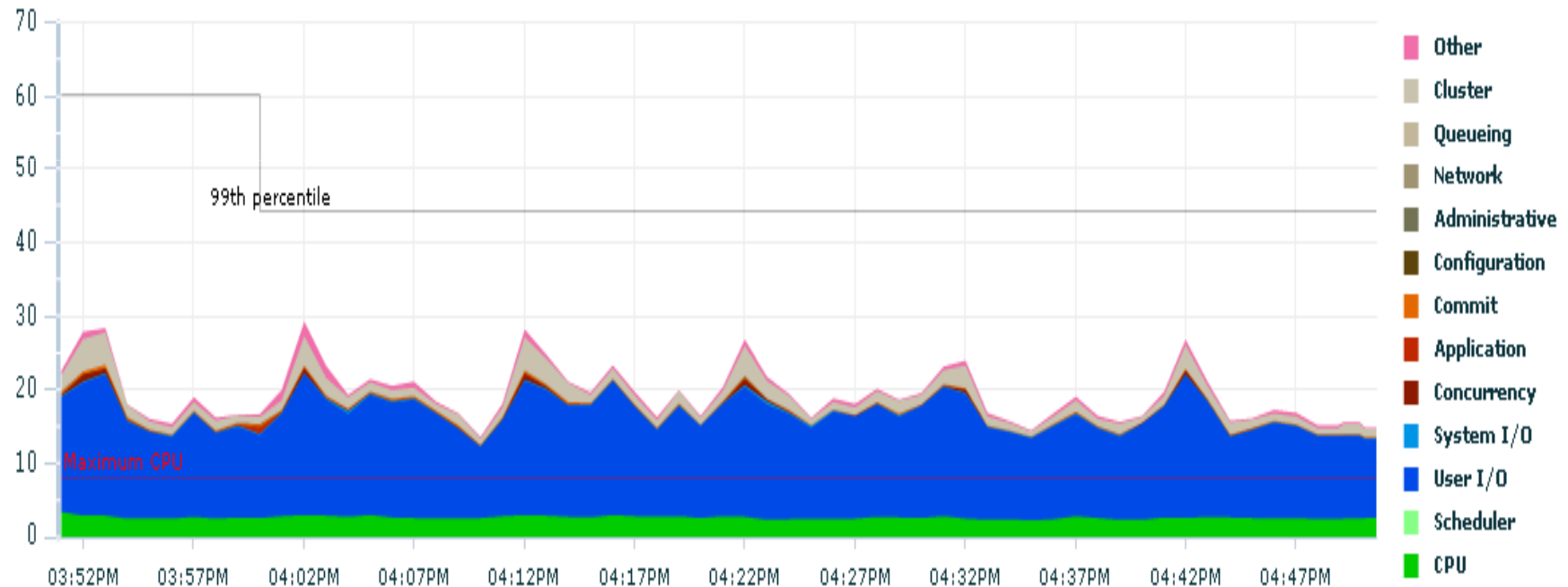
- Host CPU
 - Is host CPU-bound? Is this DB responsible?
- Active Sessions
 - DB Time accumulation by CPU, User I/O, Wait



DB Time on the Performance Page

- Area under curve is equal to DB time
- Wait class breakdown indicates solution space

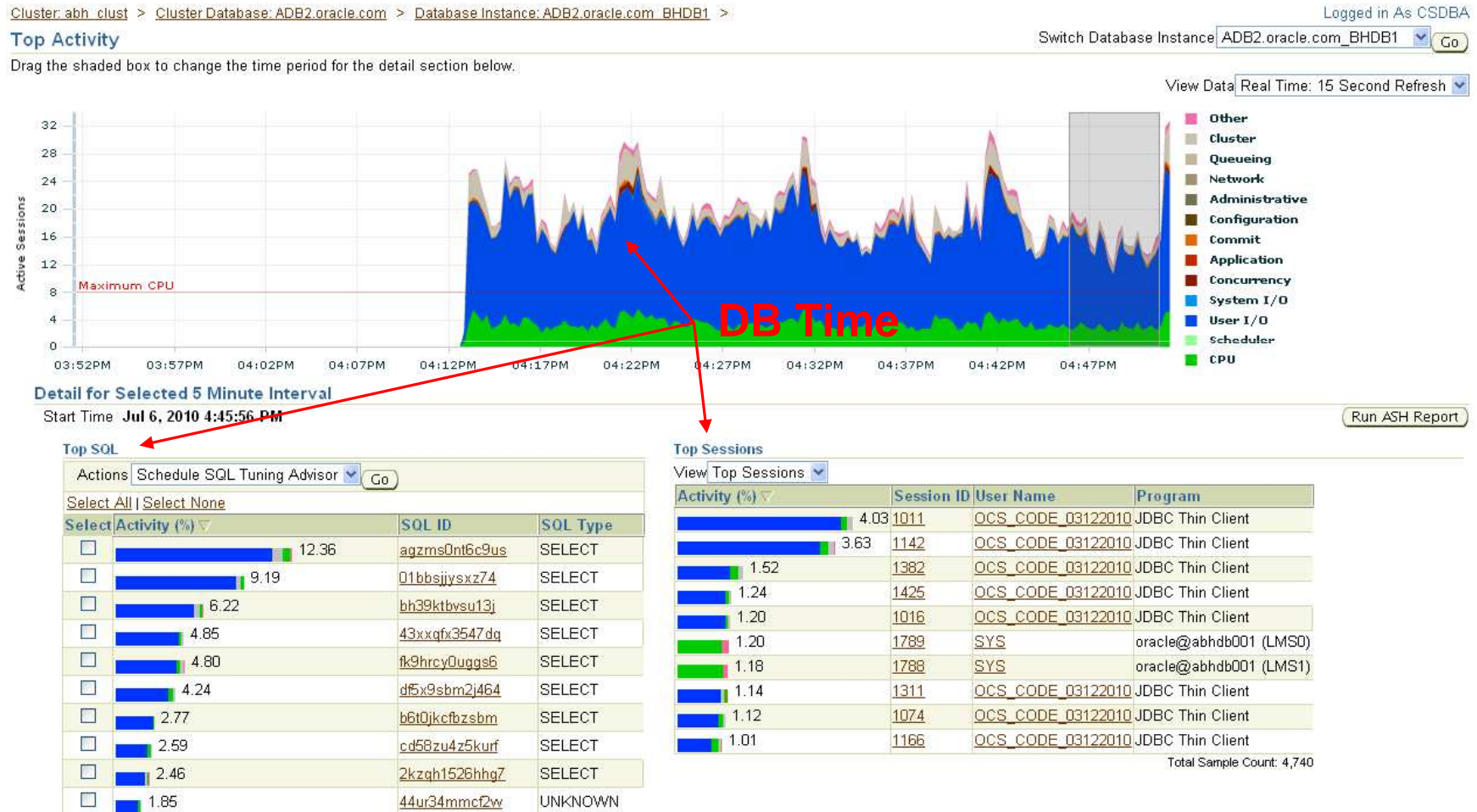
Average Active Sessions



Top Activity

DB Time on the Top Activity Page

- Plots DB Time from ASH data
- Shows top DB Time consumers by SQL and Session



DB Time in AWR Report

Time Model Statistics

- Time Model measures and accumulates DB Time through session instrumentation
- DB Time and DB CPU most important
- Also shows DB Time expended in various sub-operations

- Total time in database user-calls (DB Time): 2346.6s
- Ordered by % of DB time desc, Statistic name

Statistic Name	Time (s)	% of DB Time
sql execute elapsed time	2,282.41	97.26
DB CPU	601.12	25.62
parse time elapsed	107.84	4.60
hard parse elapsed time	94.50	4.03
PL/SQL compilation elapsed time	1.43	0.06
connection management call elapsed time	0.70	0.03
PL/SQL execution elapsed time	0.31	0.01
repeated bind elapsed time	0.22	0.01
hard parse (sharing criteria) elapsed time	0.12	0.01
failed parse elapsed time	0.04	0.00
sequence load elapsed time	0.02	0.00
hard parse (bind mismatch) elapsed time	0.01	0.00
DB time	2,346.61	
background elapsed time	14.43	
background cpu time	8.83	

DB Time in AWR Report

Top 5 Timed Foreground Events

- If you tune by Wait Events you are tuning by DB Time
- Top 5 Timed Events in the AWR report shows CPU time and non-idle waits for foregrounds

Event	Waits	Time(s)	Avg wait (ms)	% DB time	Wait Class
DB CPU		1,270		55.01	
direct path read	22,220	661	30	28.62	User I/O
direct path write temp	11,753	199	17	8.63	User I/O
direct path read temp	40,182	156	4	6.76	User I/O
log file sync	187	9	48	0.39	Commit

DB Time Fundamentals





Why Do We Care About Time?

- Human time is critical to the enterprise
- Systems performance affects business goals
 - Human time + technology resource time
- “Time is money”
- Performance improvement means doing things faster

Performance is always and only about time



DB Time Defined

- Total time in database calls by **foreground sessions**
- Includes **CPU** time, **IO** time and **non-idle wait** time
- DB Time is not called response time on purpose
- Common currency for Oracle performance analysis

Database time is total time spent by user processes either actively working or actively waiting in a database call.

Session-level Concepts

Database Time (DB Time) =

Total time session spends in all database calls

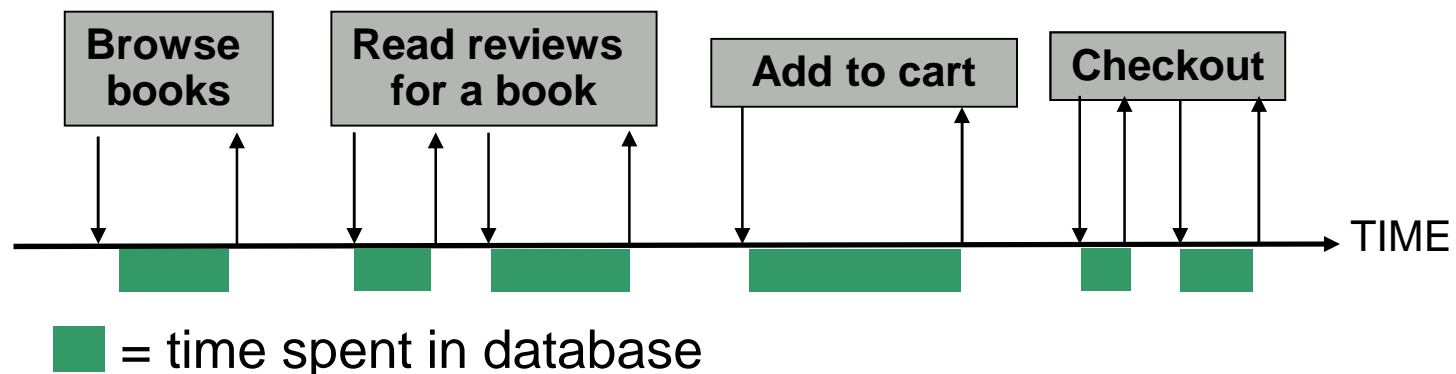
Active Session =

Session currently spending time in a database call

Average Activity of the Session (% Activity)

= Ratio of time active to total wall clock time

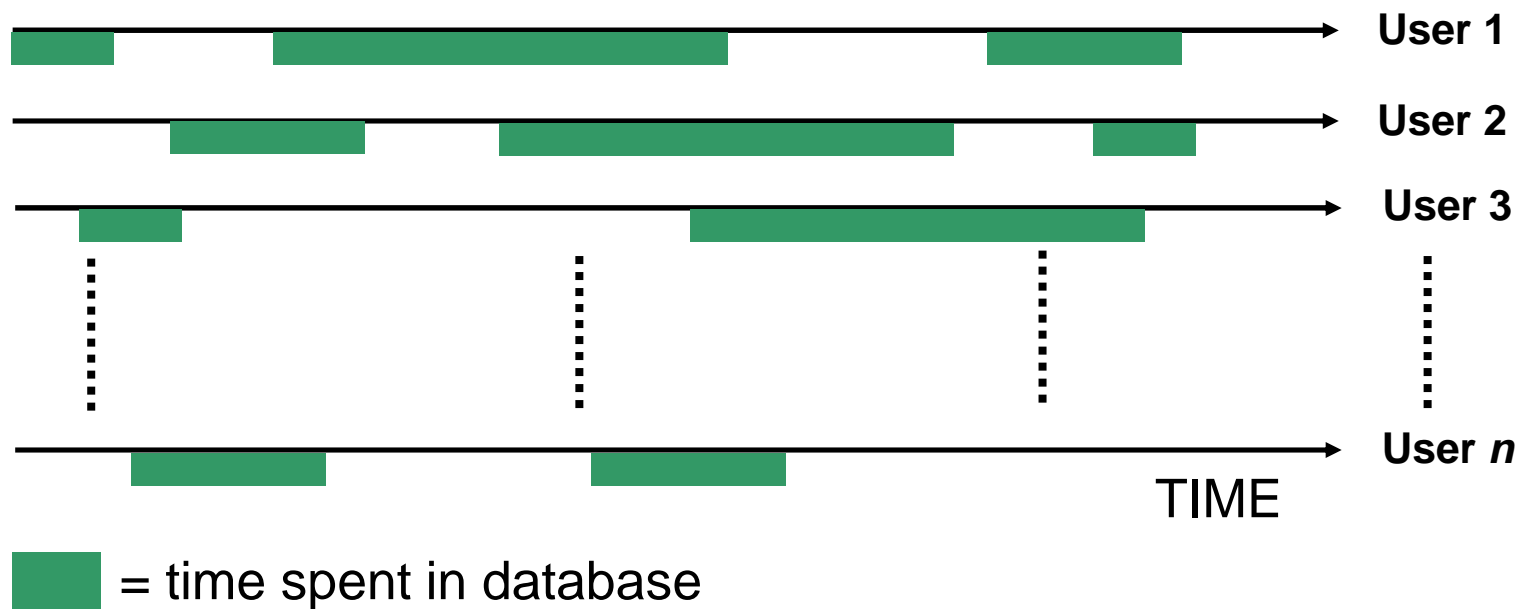
= $\text{Sum}(\text{Active Time}) / \text{Elapsed Time}$



Multiple Sessions

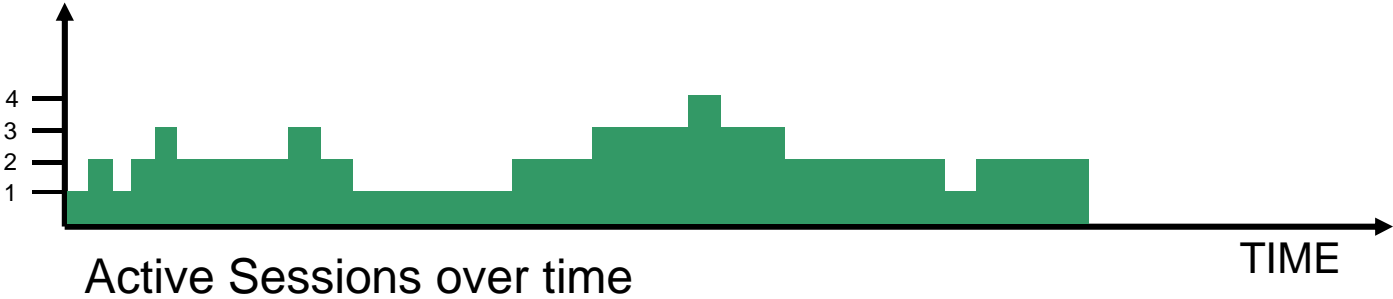
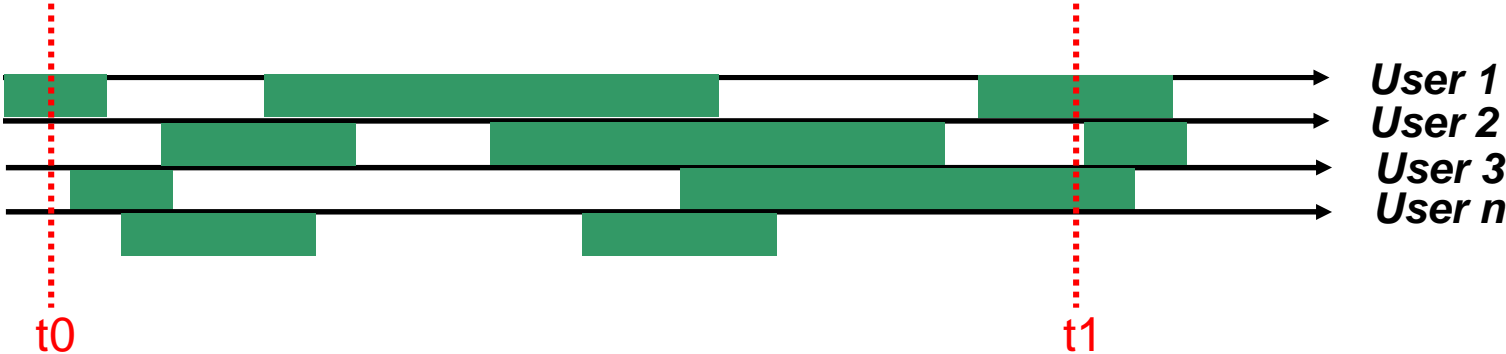
DB Time = Sum of DB Time over all sessions

Avg. Active Sessions = Sum of Avg. Activity over all sessions
= Sum(DB Time) / Elapsed Time



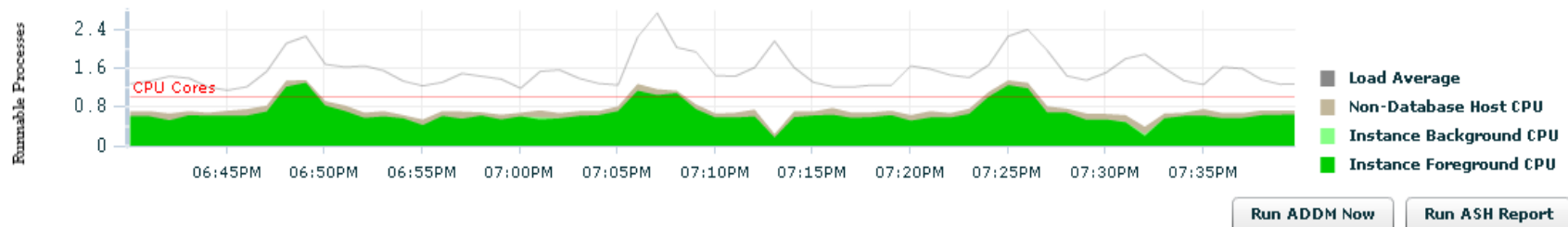


Visualizing DB Time

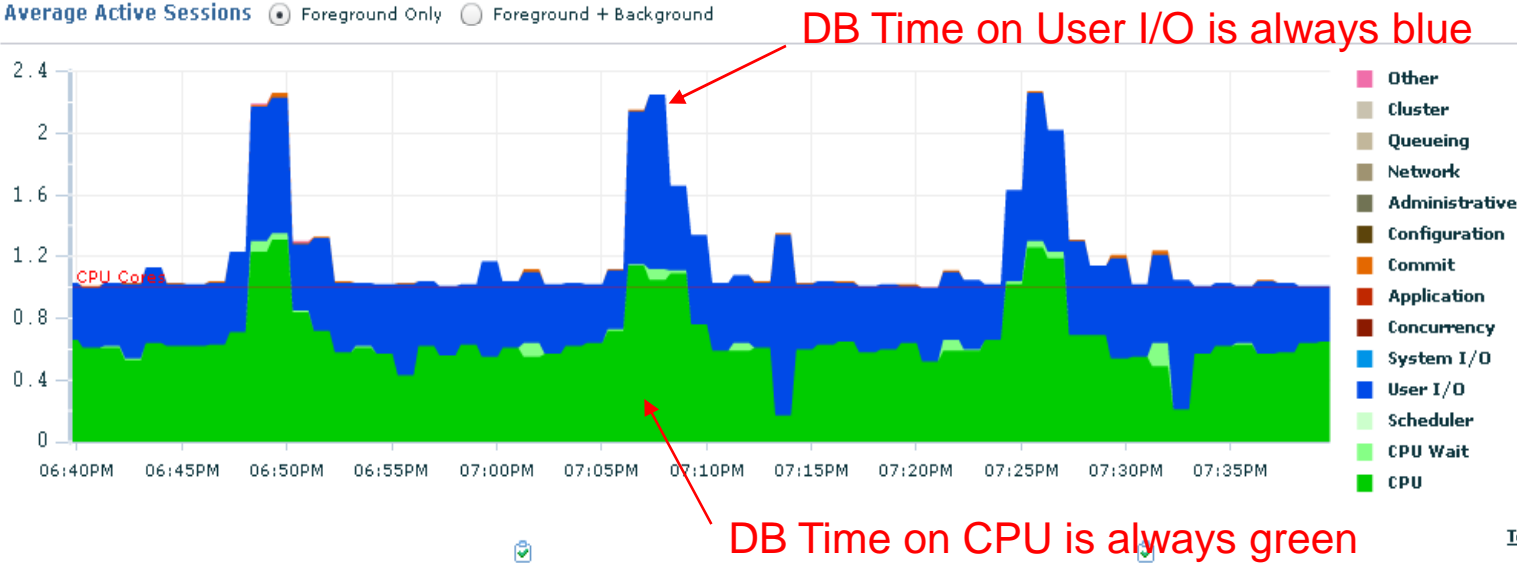


Enterprise Manager Performance Page

- Active Sessions by wait class over time
- Colored area is the amount of DB Time



Average Active Sessions Foreground Only Foreground + Background



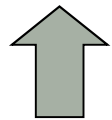
DB Time and System Performance





System Load and DB Time

- More users



More database calls



DB Time increases

- Larger transactions



More SQLs per call



DB Time increases

DB time increases as system load increases.



System Performance and DB Time

- I/O performance degrades

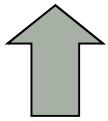


I/O wait time increases



DB Time increases

- Application performance degrades



Longer elapsed time per SQL



DB Time increases

DB time increases as performance degrades.

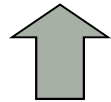


Host Performance and DB Time

- Host is CPU bound



Foregrounds accumulate active run-queue time



Wait event times are artificially inflated

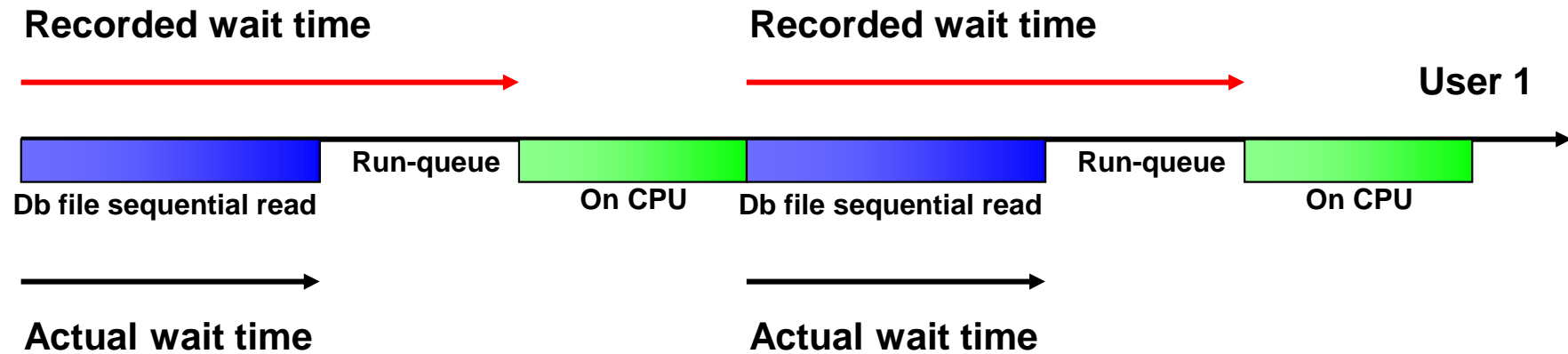


DB Time increases

Tune for CPU before waits when CPU constrained



CPU Run-Queue and DB Time



DB time is inflated when host is CPU-bound

The DB Time Method for Performance Tuning





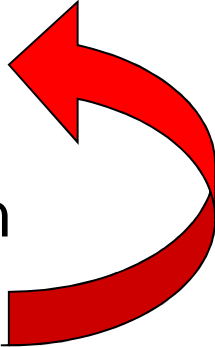
The DB Time Method: Short Course

or

just ask ADDM



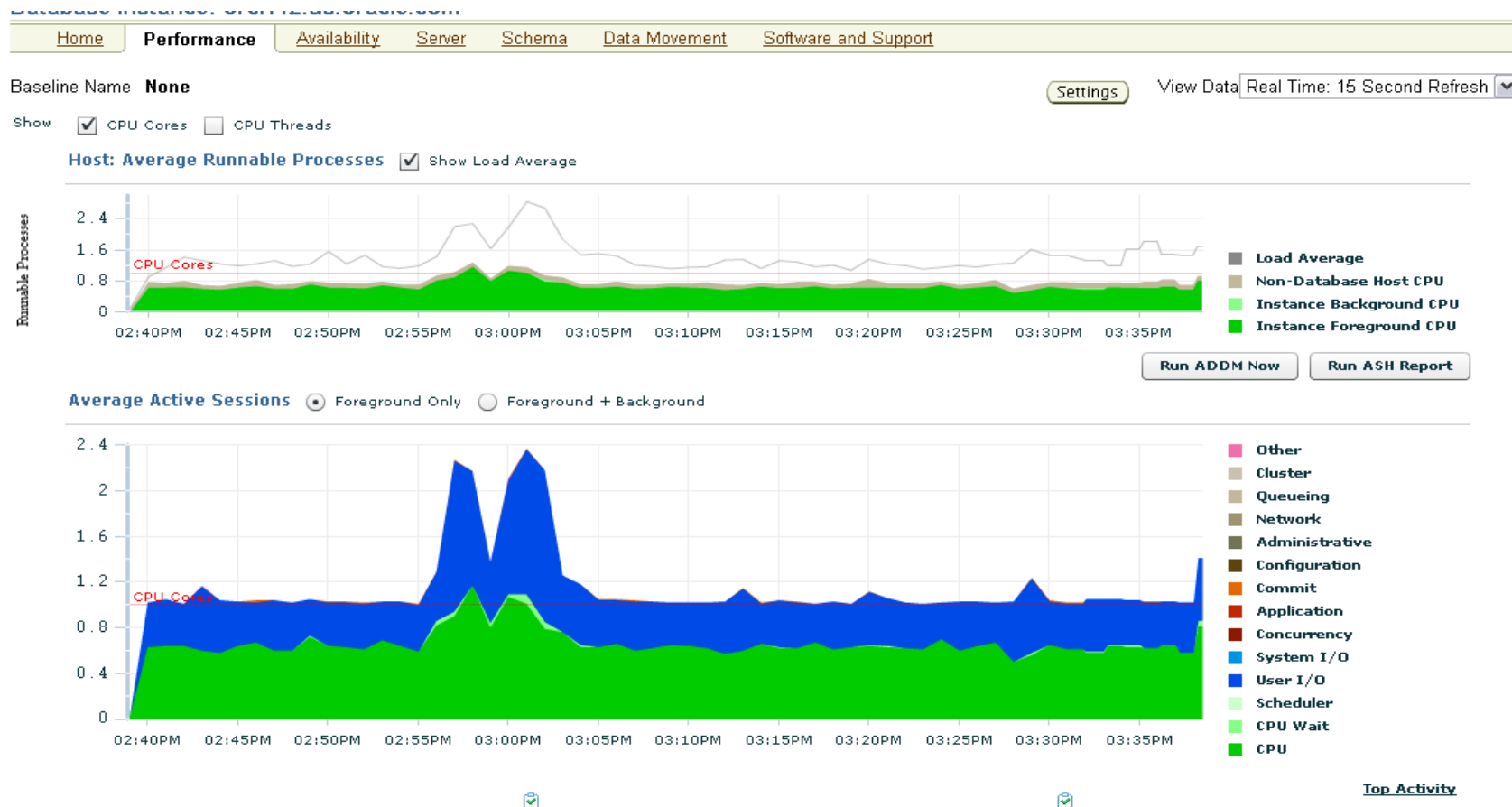
The DB Time Method: Process

1. Identify performance issue
 2. Scope the issue
 3. Set goals
 - ~~4. Data capture~~ → **AWR handles this**
 5. Investigate DB time distribution
 - Identify largest potential improvement
 6. Modify system to tune for largest gain
 7. Evaluate against goals
- 

Performance tuning by removing excess DB time

Identify and Scope the Problem

- Big spike = big problem!
- Container (host) or database issue?



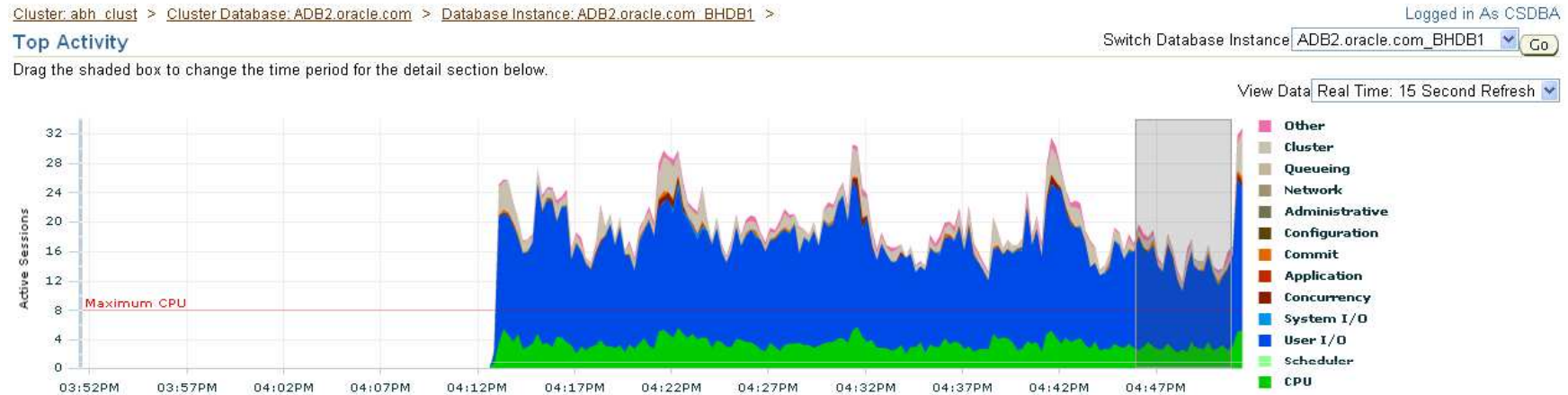


Investigate DB Time Distribution

- Identify excess DB time
 - Look for large skews in time that may represent inefficiencies
- System scope:
 - Resource limits – is problem outside the DB?
 - Resource contention (latches)
- Application scope:
 - Service, module, action
 - SQLID, rowsource
 - Resource contention (enqueuees)
- Session scope:
 - Long-running SQL
 - Resource contention (holding enqueuees in uncommitted txns)

Investigate DB Time distribution

- Which wait classes are dominant?
- “Top” lists show skew in key dimensions (SQL, session, file, etc.)



Detail for Selected 5 Minute Interval

Start Time Jul 6, 2010 4:45:56 PM

Top SQL

Actions:

Select All | Select None

Select	Activity (%)	SQL ID	SQL Type
<input type="checkbox"/>	12.36	agzms0nt6c9us	SELECT
<input type="checkbox"/>	9.19	01bbsjjysxz74	SELECT
<input type="checkbox"/>	6.22	bh39ktbvsu13j	SELECT
<input type="checkbox"/>	4.85	43xxqfx3547dq	SELECT
<input type="checkbox"/>	4.80	fk9hrcy0uggs6	SELECT
<input type="checkbox"/>	4.24	d5x9sbm2j464	SELECT
<input type="checkbox"/>	2.77	b6t0jkcfbzsbm	SELECT
<input type="checkbox"/>	2.59	cd58zu4z5kurf	SELECT
<input type="checkbox"/>	2.46	2kzqh1526hhg7	SELECT
<input type="checkbox"/>	1.85	44ur34mmcf2w	UNKNOWN

Top Sessions

View Top Sessions

Activity (%)	Session ID	User Name	Program
4.03	1011	OCS_CODE_03122010	JDBC Thin Client
3.63	1142	OCS_CODE_03122010	JDBC Thin Client
1.52	1382	OCS_CODE_03122010	JDBC Thin Client
1.24	1425	OCS_CODE_03122010	JDBC Thin Client
1.20	1016	OCS_CODE_03122010	JDBC Thin Client
1.20	1789	SYS	oracle@abhdb001 (LMS0)
1.18	1788	SYS	oracle@abhdb001 (LMS1)
1.14	1311	OCS_CODE_03122010	JDBC Thin Client
1.12	1074	OCS_CODE_03122010	JDBC Thin Client
1.01	1166	OCS_CODE_03122010	JDBC Thin Client

Total Sample Count: 4,740



Identify Time-saving Modifications

- System scope:
 - Initialization parameter changes
 - Add resources or capacity
- Application scope:
 - SQL Tuning Advisor: SQL profiles
 - Real-time SQL Monitoring
 - Access Advisor: Physical schema layout
 - Re-write SQL
- Session scope:
 - Kill session for immediate contention relief
 - Application design changes often long-term solution



Critical Success Factors

- Match solution scope to problem scope
 - Don't tweak optimizer parameters before tuning SQL
- Address root causes, not symptoms
- Proceed iteratively one fix at a time
 - Concurrent fixes should be orthogonal
- Measure and validate results at each successive step
- Stop when goals are met



Advantages of the DB Time Method

- Tunes exactly what affects users:
 - Time wasted in database
- Data capture not necessary:
 - ‘Always on’ instrumentation framework (AWR, ASH)
 - No requirement to reproduce problem (unlike tracing)
- Superior coverage of all scope levels
 - Works for concurrency problems such as locking
 - Tight integration with kernel-engineered tools
- Maturation and extension of well-proven methodologies



DB Time Based Tools:

ADDM

Enterprise Manager





Tools for Applying DB Time Method

Two use-cases, one method:

1. Tuning steady-state performance

- Improve overall workload throughput or response time
- Best practice: use ADDM

2. Diagnosing transient performance problems

- Confirm and investigate reported performance issues
- Best practice: use EM real-time screens

Example: Using ADDM

- Users complain about bad performance
 - AWR report indicates some unusual issues on system
 - What to do?

Top 5 Timed Foreground Events

~~~~~

| Event                | Waits  | Time (s) | Avg<br>wait<br>(ms) | % DB<br>time | Wait Class |
|----------------------|--------|----------|---------------------|--------------|------------|
| eng: HW - contention | 72,518 | 5,310    | 73                  | 41.0         | Configurat |
| buffer busy waits    | 81,124 | 4,693    | 58                  | 36.3         | Concurrenc |
| eng: TX - contention | 133    | 1,699    | 12776               | 13.1         | Other      |
| Data file init write | 6,839  | 300      | 44                  | 2.3          | User I/O   |
| free buffer waits    | 14,733 | 144      | 10                  | 1.1          | Configurat |

# What does the Internet advise?

- Let's search on 'enq: HW contention'
  - 9,000 hits and LOTS of discussion

---

About 9,090 results (0.27 seconds) [Advanced search](#)

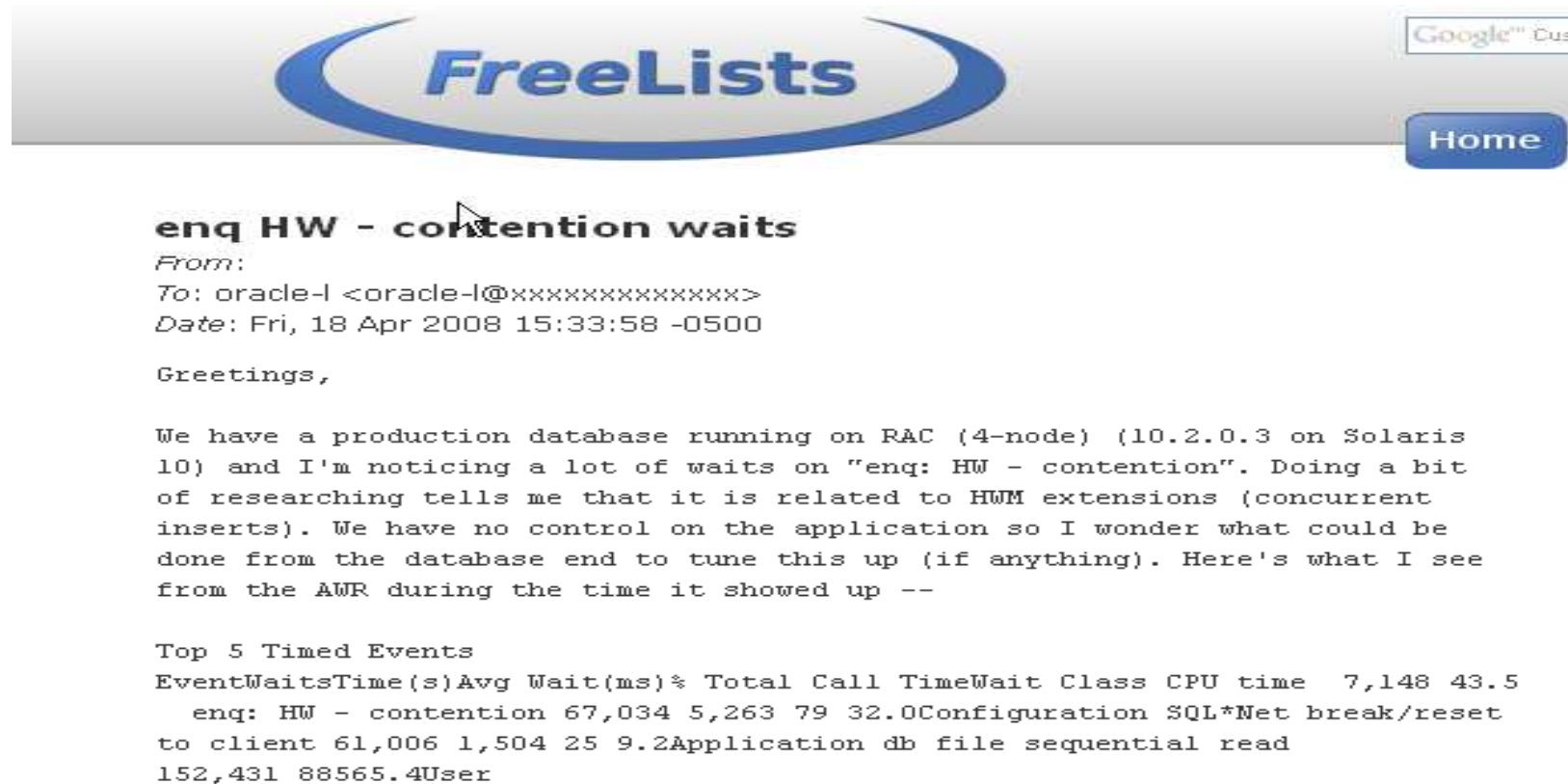
[OTN Discussion Forums : enq: HW - contention waits ...](#)  
May 13, 2008 ... We have a production database running on RAC (4-node) (10.2.0.3 on Solaris 10) and I'm noticing a lot of waits on "**enq: HW - contention**". ...  
[forums.oracle.com/forums/thread.jspa?messageID...tstart...](#) - [Cached](#) - [Similar](#)  
[Enq: TX - index contention ...](#) - 2 posts - Aug 18, 2009  
[Performance issue ...](#) - 11 posts - May 17, 2007  
[More results from forums.oracle.com »](#)

[enq HW - contention waits - oracle-l - FreeLists](#)  
Apr 18, 2008 ... RE: **enq HW - contention** waits. From: RE: **enq HW - contention** waits. From: Re: **enq HW - contention** waits ...  
[www.freelists.org/post/oracle-l/enq-HW-contention-waits](#) - [Cached](#) - [Similar](#)  
[Re: enq HW - contention waits - oracle-l - FreeLists](#)  
Re: **enq HW - contention** waits. From: To: Date: Fri, 18 Apr 2008 21:12:00 -0500 ...  
[www.freelists.org/post/oracle-l/enq-HW-contention-waits,5](#) - [Cached](#)  
[+ Show more results from www.freelists.org](#)

[enq: HW - contention](#)  
Jan 17, 2006 ... AWR/ADDM shows the reason for the slow down to be HWM **contention**

## Example: Following the links

- Many hits are other people asking the same question – what do I do about this wait event?



**FreeLists**  [Home](#)

**enq HW - contention waits**

*From:*  
*To:* oracle-l <oracle-l@xxxxxxxxxxxxxxxx>  
*Date:* Fri, 18 Apr 2008 15:33:58 -0500

Greetings,

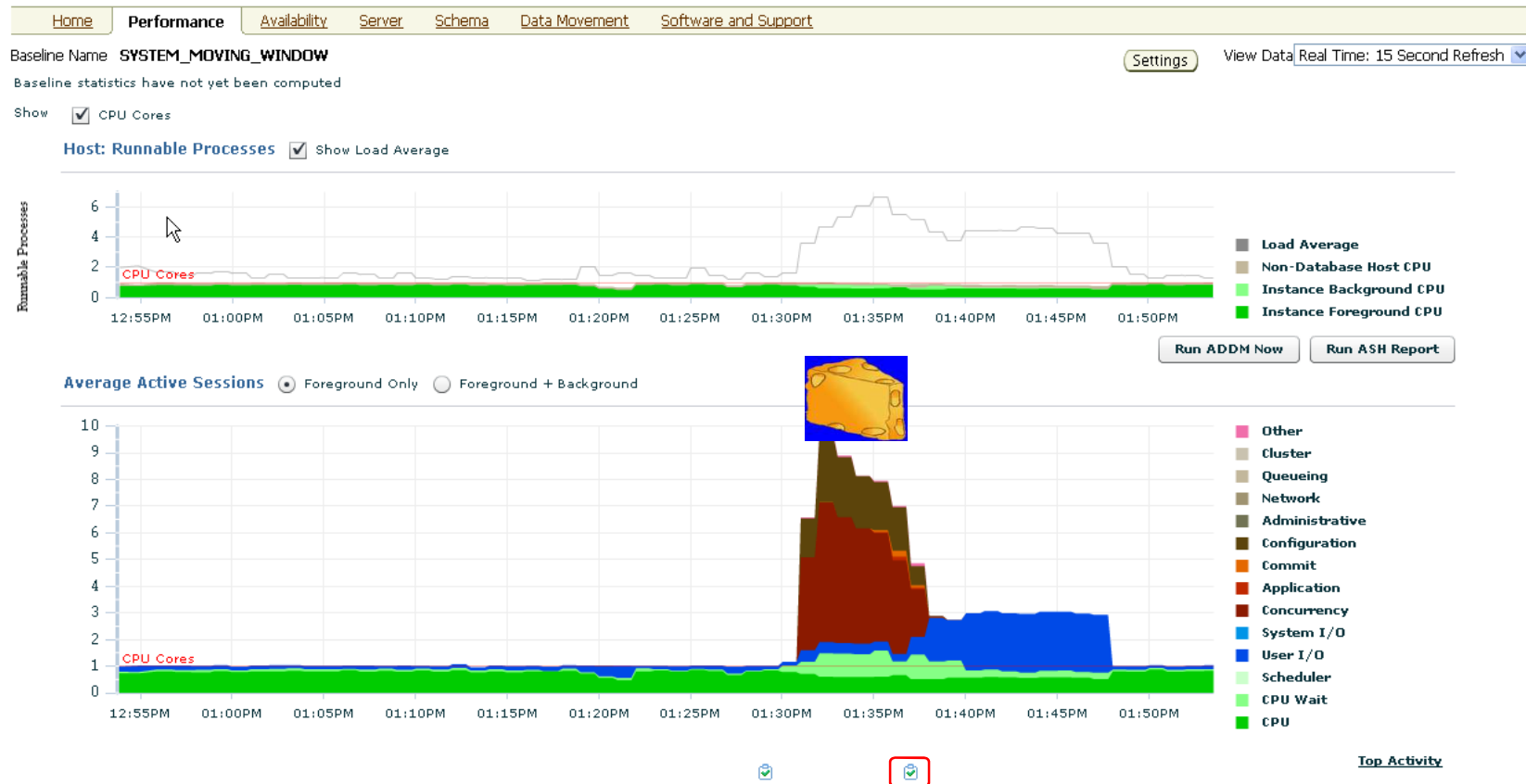
We have a production database running on RAC (4-node) (10.2.0.3 on Solaris 10) and I'm noticing a lot of waits on "enq: HW - contention". Doing a bit of researching tells me that it is related to HWM extensions (concurrent inserts). We have no control on the application so I wonder what could be done from the database end to tune this up (if anything). Here's what I see from the AWR during the time it showed up --

Top 5 Timed Events

```
EventWaitsTime(s)Avg Wait(ms)% Total Call TimeWait Class CPU time 7,148 43.5
  enq: HW - contention 67,034 5,263 79 32.0Configuration SQL*Net break/reset
to client 61,006 1,504 25 9.2Application db file sequential read
152,431 88565.4User
```

# Solution: ADDM in Enterprise Manager

- Performance Page shows a spike of unusual activity
  - Click on the ADDM analysis icon



# Solution: Using ADDM

- ADDM performance findings are DB Time based
  - Ranked by impact = percent of DB Time affected
  - Click on the issue with the highest impact

## Automatic Database Diagnostic Monitor (ADDM)

Page Refreshed Jul 12, 2010 6:17:07 PM PHT [Refresh](#)

### Database Activity

The icon selected below the graph identifies the ADDM analysis period. Click on a different icon to select a different analysis period.

### ADDM Performance Analysis

Task Name **ADDM:570420430\_1\_2**

[Filters](#) [View Snapshots](#) [View Report](#)

Task Owner **SYS** Average Active Sessions **19** Period Start Time **Jul 12, 2010 5:50:28 PM** End Time **Jul 12, 2010 6:01:48 PM**

| Impact (%) | Finding                                  | Occurrences | Icon                                                                                 |
|------------|------------------------------------------|-------------|--------------------------------------------------------------------------------------|
| 41         | <b>High Watermark Waits</b>              | 1 of 2      |  |
| 36.3       |                                          | 1 of 2      |                                                                                      |
| 13.1       | Unusual "Other" Wait Event               | 1 of 2      |                                                                                      |
| 2.3        | Unusual "User I/O" Wait Event            | 1 of 2      |                                                                                      |
| 2.1        | Top Segments by "User I/O" and "Cluster" | 1 of 2      |                                                                                      |

### Informational Findings

[Home](#) | [Targets](#) | [Deployments](#) | [Alerts](#) | [Compliance](#) | [Jobs](#) | [Reports](#) | [My Oracle Support](#) | [Setup](#) | [Preferences](#) | [Help](#) | [Logout](#)

Copyright © 1996, 2010, Oracle and/or its affiliates. All rights reserved.  
Oracle is a registered trademark of Oracle Corporation and/or its affiliates.  
Other names may be trademarks of their respective owners.  
[About Oracle Enterprise Manager](#)

# Solution: Using ADDM

- ADDM recommendations describe solutions for findings
- EM presents 'implement' actions where possible

ORACLE Enterprise Manager  
Grid Control 11g

Home **Targets** Deployments Alerts Compliance Jobs Reports My Oracle Support


All Targets | Databases | Hosts | Middleware | Web Applications | Services | Systems | Groups | Virtual Servers

Database Instance: krelease > Automatic Database Diagnostic Monitor (ADDM):SYS.ADDM:570420430\_1\_6 > Logged in As SYS

### Performance Finding Details: High Watermark Waits

Finding **Contention on the high watermark (HW) enqueue was consuming significant database time.**

Impact (Active Sessions) **.96**

Percentage of Finding's Impact (%)  **13**

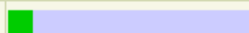
Period Start Time **Jul 14, 2010 1:29:11 PM**

End Time **Jul 14, 2010 1:36:23 PM**

Filtered **No**

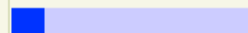
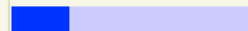
**Consider using ORACLE's recommended solution of automatic segment space management in a locally managed tablespace for the tablespace "WORK" containing the TABLE "SYS.TEMP" with object ID 62750. Alternatively, you can move this object to a different tablespace that is locally managed with automatic segment space management.**

Database Object **SYS.TEMP**

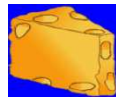
Show Schema  10.3

Findings Path

Expand All | Collapse All

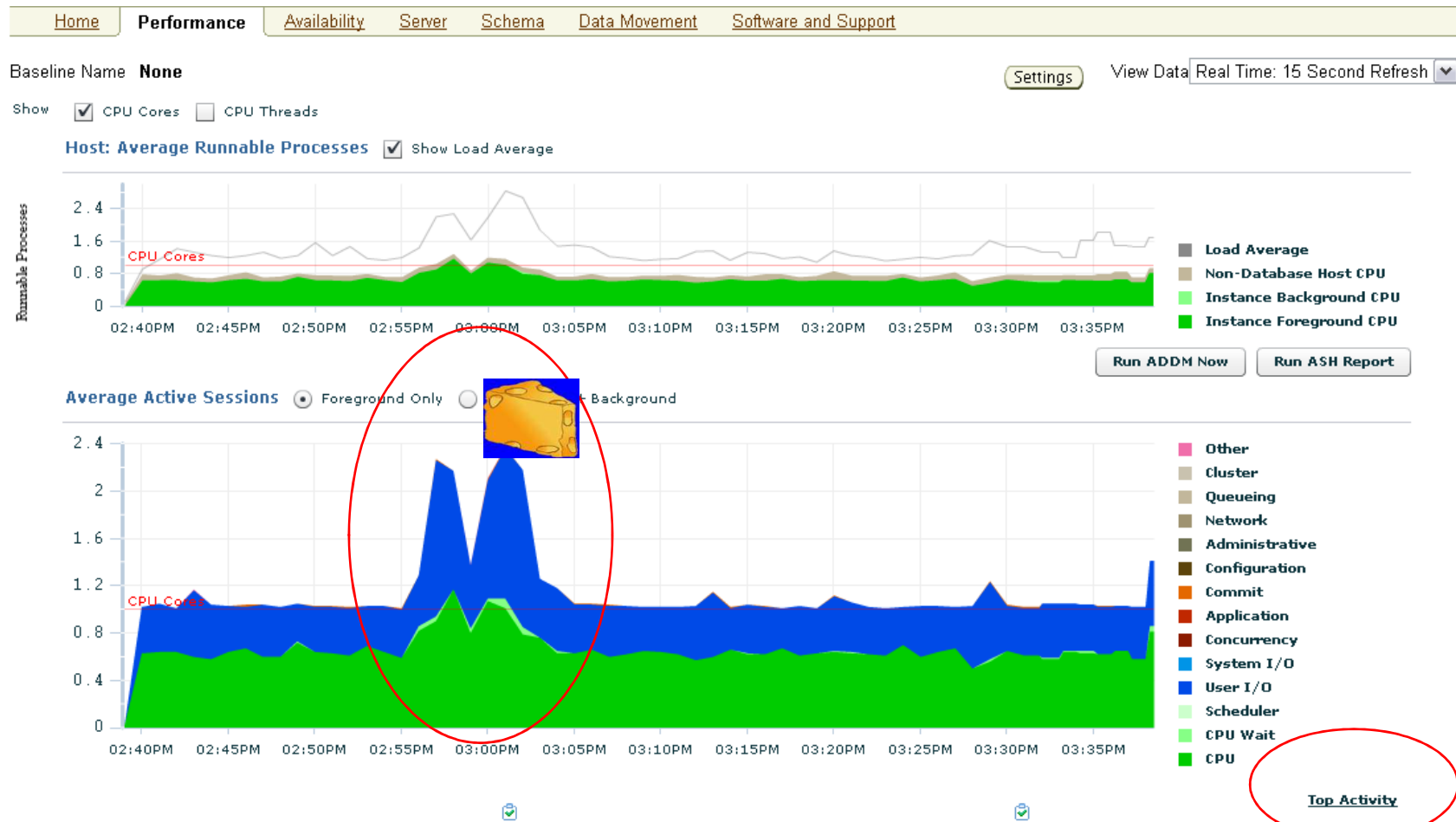
| Findings                                                                               | Percentage of Finding's Impact (%)                                                         | Additional Information |
|----------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------|------------------------|
| Contention on the high watermark (HW) enqueue was consuming significant database time. |  13   |                        |
| Wait class "Configuration" was consuming significant database time.                    |  23.3 |                        |

## Best Practice: Enterprise Manager Real-time Interface

- Transient (sub-hour) or immediate time scope
  - Requires interactivity of UI
- ‘Click on the big stuff’ 
  - Data visualizations display skew directly
- Takes some expertise to separate symptoms from root causes

# Click on the Big Stuff – Follow the DB Time

- Spike in DB Time signals possible performance issue
- Click through to Top Activity for more details



# Follow the DB Time – Top Activity Drilldown

- Top Activity Page confirms problem and shows highly skewed DB Time by SQL ID. Click on top SQL ID for further details.

ORACLE Enterprise Manager 11g Database Control

Database Instance: orcl112.us.oracle.com > Setup Preferences Help Logout Database

Database Instance: orcl112.us.oracle.com > Logged in As SYS

### Top Activity

Drag the shaded box to change the time period for the detail section below.

View Data: Real Time: Manual Refresh

**Detail for Selected 5 Minute Interval**  
Start Time: Jan 29, 2010 2:58:15 PM PST Run ASH Report

#### Top SQL

Actions: Schedule SQL Tuning Advisor

Select All | Select None

| Select                   | Activity (%) | SQL ID        | SQL Type |
|--------------------------|--------------|---------------|----------|
| <input type="checkbox"/> | 95.94        | 1a7j4qkhvatyw | SELECT   |
| <input type="checkbox"/> | 1.11         | d15cdr0zt3vtp | SELECT   |
| <input type="checkbox"/> | .92          | 1au8jkpu5gr3n | SELECT   |

#### Top Sessions

View: Top Sessions

| Activity (%) | Session ID | QC Session ID | User Name | Program                                 |
|--------------|------------|---------------|-----------|-----------------------------------------|
| 24.49        | 23         | 140           | OOW       | oracle@rsarwal-pc4.us.oracle.com (P002) |
| 24.14        | 152        | 140           | OOW       | oracle@rsarwal-pc4.us.oracle.com (P003) |
| 9.08         | 28         | 140           | OOW       | oracle@rsarwal-pc4.us.oracle.com (P001) |

# Click on the Big Stuff – Follow the DB Time

- SQL Details confirms this SQL produced the spike in DB Time and shows parallel execution. Click on 'SQL Monitoring' to see DB Time by execution plan steps.

ORACLE Enterprise Manager 11g Database Control

Database Instance: orcl112.us.oracle.com > Top Activity > SQL Details: 1a7j4qkhvatyw

Switch to SQL ID  Go View Data Real Time: Manual Refresh Refresh SQL Worksheet Schedule SQL Tuning Advisor SQL Repair Advisor

```
select /*+ parallel(1, 2) parallel(o, 1) use_hash(o 1) leading(1) */ count(*)
from oow.orders o, oow.lineitem l
where l.l_orderkey = o.o_orderkey and ora_hash(l.l_comment || l.l_comment || dump(l.l_comment)) > 0 and
ora_hash(o.o_comment || o.o_comment || dump(o.o_comment)) > 0 and...
```

Details

Select the plan hash value to see the details below. Plan Hash Value 2167399637

Statistics Activity Plan Plan Control Tuning History **SQL Monitoring**

Summary

Drag the shaded box to change the time period for the detail section below.

Detail for Selected 5 Minute Interval

Start Time Jan 29, 2010 2:58:15 PM

| Activity (%) | SID | QC SID | User | Program                                 | Service    | Plan Hash Value |
|--------------|-----|--------|------|-----------------------------------------|------------|-----------------|
| 27.50        | 23  | 140    | OOOW | oracle@rsarwal-pc4.us.oracle.com (P002) | SYS\$USERS | 2167399637      |
| 27.12        | 152 | 140    | OOOW | oracle@rsarwal-pc4.us.oracle.com (P003) | SYS\$USERS | 2167399637      |

# Follow the DB Time – SQL Execution Details

Database Instance: orcl112.us.oracle.com > Monitored SQL Executions >

Logged in As SYS

Monitored SQL Execution Details

Save Mail View Report

**Overview**

SQL ID: 1a7j4qkhvtyw  
 Parallel: 2  
 Execution Started: Wed Sep 15, 2010 11:02:22 AM  
 Last Refresh Time: Wed Sep 15, 2010 11:07:17 AM  
 Execution ID: 16777226  
 User: OOW  
 Fetch Calls: 1

**Time & Wait Statistics**

Duration: 4.9m  
 Database Time: 9.7m  
 Wait Activity %: 100  
 PL/SQL & Java: 0.0s

**IO Statistics**

IO Requests: 20K  
 IO Bytes: 2093M  
 Buffer Gets: 156K

**Details**

Plan Statistics Parallel Activity

Plan Hash Value: 2167399637

TIP: Right mouse click on the table allows to toggle between IO Requests and IO Bytes

| Operation           | Name     | Estima... | Cost | Timeline(295s) | Ex... | Actu... | Mem... | Tem... | IO Requests | CPU Activity % | Wait Activity % |
|---------------------|----------|-----------|------|----------------|-------|---------|--------|--------|-------------|----------------|-----------------|
| SELECT STATEMENT    |          |           |      |                | 1     | 1       |        |        |             |                |                 |
| SORT AGGREGATE      |          |           | 1    |                | 1     | 1       |        |        |             |                |                 |
| PX COORDINATOR      |          |           |      |                | 5     | 2       |        |        |             |                | .27             |
| PX SEND QC (RANDOM) | :TQ10002 |           | 1    |                | 2     | 2       |        |        |             |                |                 |
| SORT AGGREGATE      |          |           | 1    |                | 2     | 2       |        |        |             |                |                 |
| HASH JOIN           |          | 389K      | 41K  |                | 2     | 7702K   | 120M   | 294M   | 1846        | 11             | 18              |
| PX RECEIVE          |          | 7702K     | 19K  |                | 2     | 7702K   |        |        |             | .52            |                 |
| PX SEND HASH        | :TQ10001 | 7702K     | 19K  |                | 2     | 7702K   |        |        |             | 4.69           |                 |
| PX BLOCK ITERATOR   |          | 7702K     | 19K  |                | 2     | 7702K   |        |        |             |                |                 |
| TABLE ACCESS FULL   | LINEITEM | 7702K     | 19K  |                | 26    | 7702K   |        |        | 2024        | 65             | 75              |
| BUFFER SORT         |          |           |      |                | 2     | 1920K   | 116M   | 126M   | 16K         | 1.56           | 2.66            |
| PX RECEIVE          |          | 1920K     | 780! |                | 2     | 1920K   |        |        |             |                |                 |
| PX SEND HASH        | :TQ10000 | 1920K     | 780! |                | 1     | 1920K   |        |        |             |                |                 |
| TABLE ACCESS FULL   | ORDERS   | 1920K     | 780! |                | 1     | 1920K   |        |        | 448         | 17             | 3.46            |



## DB Time Method Summary

- DB time is the fundamental performance metric
- The method allows DB time analysis at many scopes
  - Proper scoping of problems and solutions is critical to success
- DB time based diagnosis removes value judgments
  - Scientific method, not sorcerer's magic

***Performance improvement means doing the same work in less DB Time***

*O* & *A*

**ORACLE®**