

# ADVENTURES TUNING ORACLE APPLICATIONS ON RAC

*By Richard J. Niemiec, TUSC*

## ABSTRACT

Tuning the Oracle database is tough. Tuning the Oracle database is a lot tougher when you're running Oracle Applications since you're limited in what you can do. But, tuning Oracle Applications on RAC is definitely the hardest of all. The key to success is finding problems and then tuning them using the incredible Grid Control within Enterprise Manager. With Oracle 10g Enterprise Manager Grid Control, you now have a tremendous product at your side. This paper will look at a few of the screens that you can use to monitor the grid. BUT, I warn you, there are so many great screens and tools that I could never do the product justice with such a short paper. See the Oracle Database 10g Performance Tips and Techniques book for a detailed look at this product. Tuning Oracle Applications on RAC consists of:

- Start with the right stack & test everything under load
- Enterprise Manager is great – Grid Control extends it to RAC
- Grid Computing helps you find & show problems fast
- Grid Computing helps convince Oracle there is a problem
- Metalink is your friend to help find/fix bugs
- Some patches broke more than fixed at times – testing still a key
- Must separate the Apps issues with the Database issues
- Must separate the Database issues with the RAC issues
- Different product issues caused us to Start Over with that team
- Showed Oracle is still evolving with supporting products
- V\$/X\$ & logs are still helpful drilling deep into problems
- **Grid Control (Enterprise Manager for RAC) is the key product for success!**

## CLUSTERING AND AN ACCELERATION TOWARD GRID COMPUTING

In June 1970, IBM's Ted Codd published the 11 page paper "A Relational Model of Data for Large Shared Data Banks." This article would lead to the relational databases that would hold the world's data. Advances in information storage/extraction/analysis as well as predicting customer needs has driven us deep into the information age. With 64-bit processing and using grid control, it is theoretically possible to store all of information currently in every database into a single Oracle10g database (Oracle10g allows an 8E database) and load all of it in memory (64 bit allows 16E). The advances of the last decade will be dwarfed by those of the next 10 years. Welcome to the 21<sup>st</sup> century DBA! This paper is a light hearted look at things NOT to do.

The information age is about to take a drastic step forward. The power of 64-bit computing can be imagined when you consider the theoretical limit for addressable memory. In unsigned 32-bit computing, we could directly address 4G of memory (the sign will cost you 2G). For a standard Oracle database, this allowed a tremendously increased System Global Area (SGA). The SGA is where my most often used data can be stored and kept in memory for fast access. The move to 64-bit starts to accelerate the information age exponentially. With 64-bit, the theoretical limit of addressable memory ( $2$  to the power of  $64$ ) becomes 16E (Exabytes) or 18,446,744,073,709,551,616 bytes of memory. Since it is estimated that there is only 12 exabytes of information in the entire world, 16 exabytes is a pretty healthy amount memory (Larry now can run the entire world in a single Oracle database - WOL). Imagine storing every single piece of information on earth in one database and IN MEMORY. It is theoretically possible, although the physical architecture has not been needed (and hence, not built). Author Note: If you did store all databases on a single system, in memory, I predict there would be a major CPU bottleneck.

Now that we know that the future of hardware theoretically solves any amount of data we will ever need to store in our system, let's move to the database. How we will we quickly and securely access that information? The answer, of course, is Oracle. While competing databases have *some* Oracle features, the information age requires a database that is incredibly fast and tune-able while the system is running, completely available 24x7x52, completely recoverable at a

moments notice, allows maintenance on information that is being accessed, altered or even being recovered, and allows for test recoveries and resume-able recoveries of full or partial information. We also now require encrypted backups, encrypted data, and a way to manage it all with less resources. Welcome to the world of Oracle!

How do we scale the hardware that runs this database so that when we need more CPU power or want to service additional users. Welcome to the world of RAC (Real Application Clusters)! Real Applications allow us to share ONE database while having MULTIPLE System Global Areas (SGA) on multiple pieces of hardware. Using Oracle's "cache fusion," which is a process where we can move data from one SGA to another (saving costly disk I/O) when needed (via a high speed fiber interconnect), we get the most scalable Oracle to date. Imagine a single database running on a 8 machine cluster (8 machines hooked up by cache fusion to the same database) with 256G of memory on each instance. That's 2T of physical memory making 1+ terabyte of combined SGA not impossible to imagine along with 256 CPUs and 10T filesystems (required to get to an 8E database). I'm still anxiously awaiting the 16 Exabyte hardware that will run my 8E database. Consider the amount of directly addressable memory from 4-bit to 64-bit and you can see where we are headed now that most hardware is heading rapidly toward 64-bit computing.

	<u>Memory to Directly Address</u>	<u>Indirect/Extended</u>
4 Bit:	16	(640)
8 Bit:	256	(65,536)
16 Bit:	65,536	(1,048,576)
32 Bit:	4,294,967,296	
64 Bit:	18,446,744,073,709,551,616	

The information age is about to take a drastic step forward. The power of 64-bit computing can be imagined when you consider the theoretical limit for addressable memory. In unsigned 32-bit computing, we could directly address 4G of memory (the sign will cost you 2G). For a standard Oracle database, this allowed a tremendously increased System Global Area (SGA). The SGA is where my most often used data can be stored and kept in memory for fast access. The move to 64-bit starts to accelerate the information age exponentially. With 64-bit, the theoretical limit of addressable memory ( $2$  to the power of  $64$ ) becomes 16E (Exabytes) or

18,446,744,073,709,551,616 bytes of memory. Since it is estimated that there is only 12 exabytes of information in the entire world, 16 exabytes is a pretty healthy amount memory (Larry now can run the entire world in a single Oracle database - WOL). Imagine storing every single piece of information on earth in one database and IN MEMORY. It is theoretically possible, although the physical architecture has not been needed (and hence, not built). Author Note: If you did store all databases on a single system, in memory, I predict there would be a major CPU bottleneck.

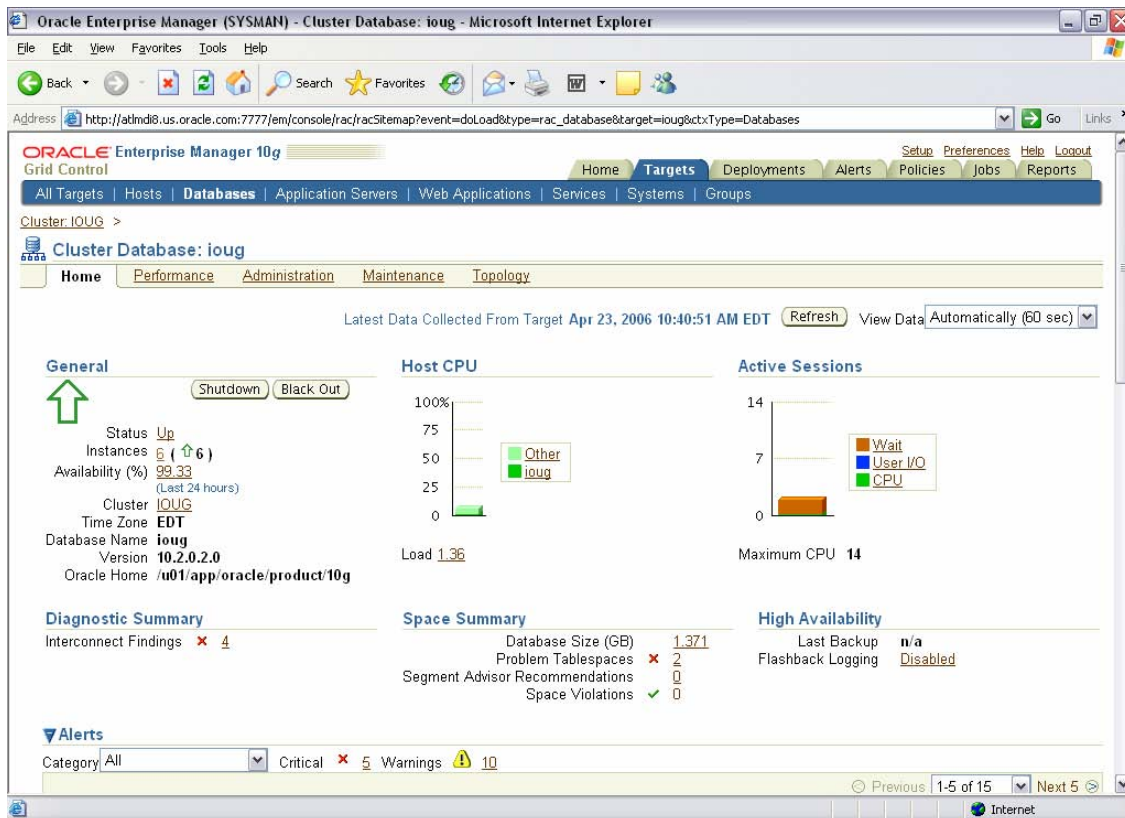
Now that we know that the future of hardware theoretically solves any amount of data we will ever need to store in our system, let's move to the database. How will we quickly and securely access that information? The answer, of course, is Oracle. While competing databases have *some* Oracle features, the information age requires a database that is incredibly fast and tune-able while the system is running, completely available 24x7x52, completely recoverable at a moments notice, allows maintenance on information that is being accessed, altered or even being recovered, and allows for test recoveries and resume-able recoveries of full or partial information. We also now require encrypted backups, encrypted data, and a way to manage it all with less resources. Welcome to the world of Oracle!

How do we scale the hardware that runs this database so that when we need more CPU power or want to service additional users. Welcome to the world of RAC (Real Application Clusters)! Real Applications allow us to share ONE database while having MULTIPLE System Global Areas (SGA) on multiple pieces of hardware. Using Oracle's "cache fusion," which is a process where we can move data from one SGA to another (saving costly disk I/O) when needed (via a high speed fiber interconnect), we get the most scalable Oracle to date. Imagine a single database running on a 8 machine cluster (8 machines hooked up by cache fusion to the same database) with 256G of memory on each instance. That's 2T of physical memory making 1+ terabyte of combined SGA not impossible to imagine along with 256 CPUs and 10T filesystems (required to get to an 8E database). I'm still anxiously awaiting the 16 Exabyte hardware that will run my 8E database. Consider the amount of directly addressable memory from 4-bit to 64-bit and you can see where we are headed now that most hardware is heading rapidly toward 64-bit computing.

	<u>Memory to Directly Address</u>	<u>Indirect/Extended</u>
4 Bit:	16	(640)
8 Bit:	256	(65,536)
16 Bit:	65,536	(1,048,576)
32 Bit:	4,294,967,296	
64 Bit:	18,446,744,073,709,551,616	

## MANAGING THE GRID

One of the best screens to manage the grid is displayed below. It's the screen to click on a cluster and see whether the nodes are up or down as well as see the individual nodes. Here is the cluster "ioug" showing six nodes that are all up. To get to this screen, I just went to the Targets tab and clicked on the ioug cluster.



**Figure 1: Looking at the IOUG Cluster Database under Targets/Databases**

If you move down the page a bit, you can see the instances (all using ASM) that are associated with this cluster as shown in Figure 2.

The screenshot shows the Oracle Enterprise Manager (SYSMAN) interface in a Microsoft Internet Explorer browser. The address bar indicates the URL: [http://atmd18.us.oracle.com:7777/em/console/rac/racSiteMap?event=dLoad&type=rac\\_database&target=ioug&ctxType=Databases](http://atmd18.us.oracle.com:7777/em/console/rac/racSiteMap?event=dLoad&type=rac_database&target=ioug&ctxType=Databases). The page displays job execution statistics and a table of database instances.

**Job Executions Summary:**

Status	Submitted to the Cluster Database	Submitted to any member
Scheduled	0	0
Running	0	0
Suspended	0	0
Problem	0	0

**Instances Table:**

Name	Status	Alerts	Policy Violations	Compliance Score (%)	ADDM Findings	ASM	Sessions: CPU	Sessions: I/O	Sessions: Other	Instance CPU (%)		
ioug_ioug1	⊕	1	2	4	9	1	91	2	+ASM1_atmd1.us.oracle.com_02	0	1.84	.8
ioug_ioug2	⊕	1	1	4	9	1	91	0	+ASM2_atmd2.us.oracle.com_01	0	0	.41
ioug_ioug3	⊕	0	2	4	9	1	91	0	+ASM3_atmd3.us.oracle.com_01	0	.07	.65
ioug_ioug4	⊕	0	2	4	9	1	91	0	+ASM4_atmd4.us.oracle.com_01	0	.03	.28
ioug_ioug5	⊕	1	1	4	9	1	91	0	+ASM5_atmd5.us.oracle.com_01	0	0	.33
ioug_ioug6	⊕	1	1	4	9	1	91	0	+ASM6_atmd7.us.oracle.com_01	0	.04	.14

**Related Links:**

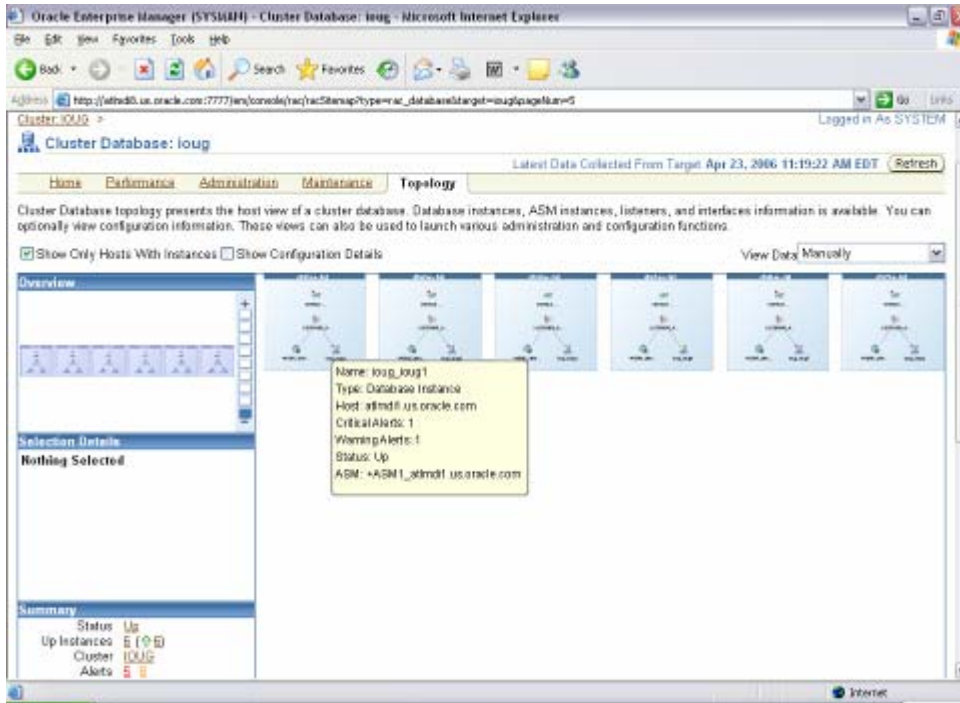
- Access
- All Metrics
- Execute SQL
- Metric and Policy Settings
- Reports
- Target Properties
- Advisor Central
- Blackouts
- iSQL\*Plus
- Metric Collection Errors
- Rules Manager
- Alert History
- Deployments
- Jobs
- Monitoring Configuration
- SQL History

Navigation: Home | **Targets** | Deployments | Alerts | Policies | Jobs | Reports | Setup | Preferences | Help | Logout

Copyright © 1996, 2005, Oracle. All rights reserved.  
 Oracle, JD Edwards, PeopleSoft, and Retek are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.  
 About Oracle Enterprise Manager

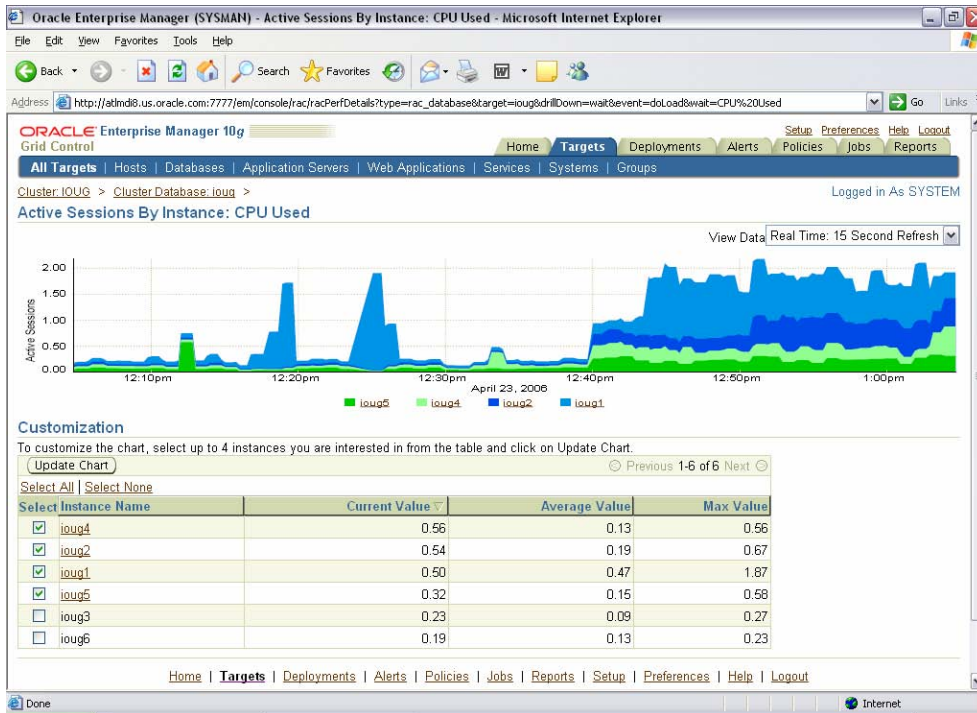
**Figure 2: Further down the page of Figure 1 we see the individual Nodes 1-6**

If I click on the topology tab (see Figure 3), we can see the topology for all six instances (each instance is on a separate node, so there are also six separate nodes. Notice that my mouse is over one of the instances and additional information about the instance is provided.



**Figure 3: Looking at the Topology of the 6 Nodes in the IOUG Cluster**

If I click on the Performance tab and then click onto the CPU Used chart (see Figure 4), I can see performance all nodes in the “ioug” cluster, each in a different color.

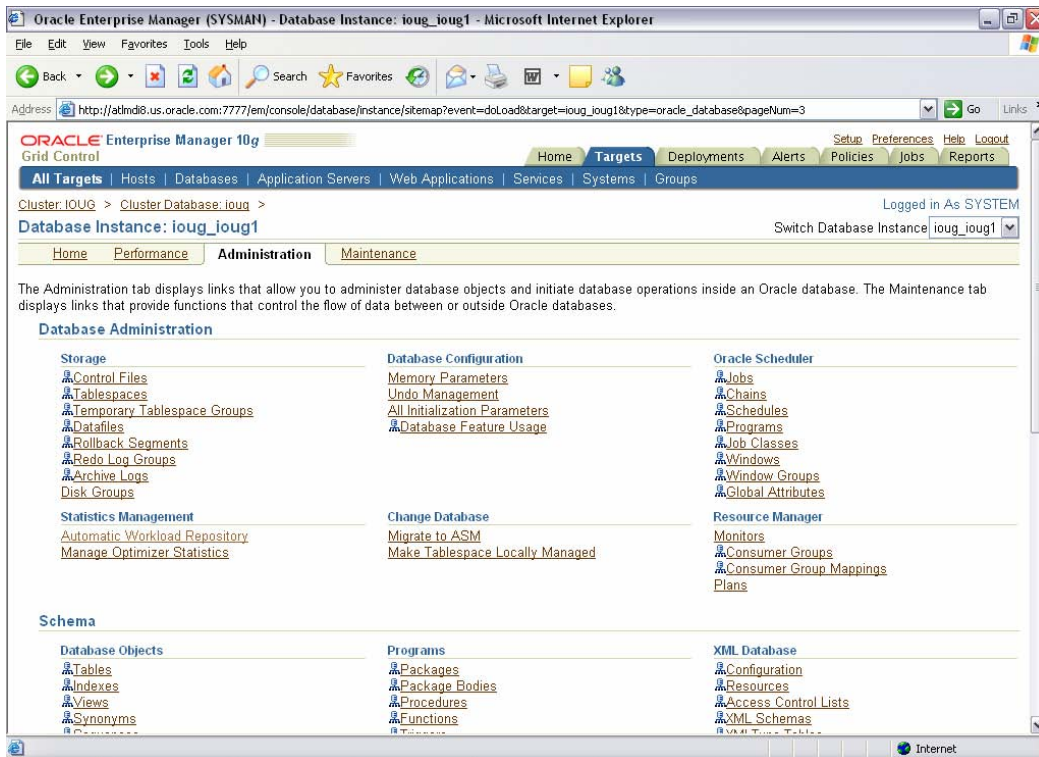


**Figure 4: Looking at CPU for 4 of the selected nodes in the IOUG Cluster**

### RUNNING THE AWR REPORT FROM ENTERPRISE MANAGER

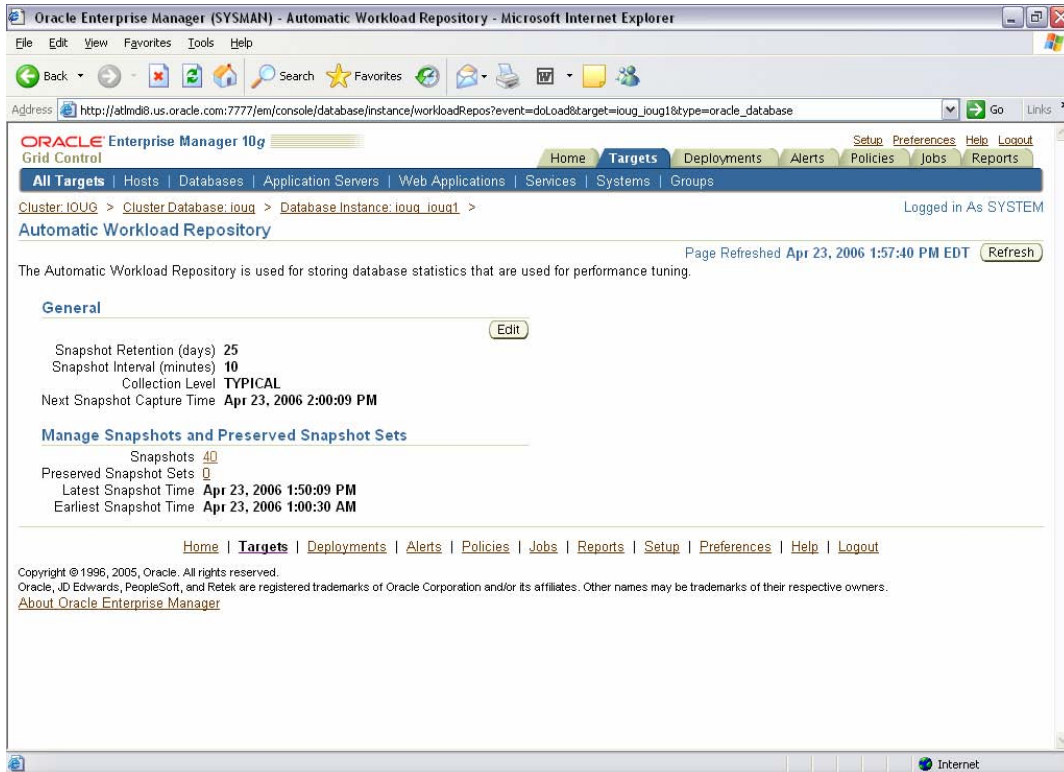
The Database Administration tab of Enterprise Manager can also be used at the Instance level to run the Automatic Workload Repository (AWR) Report. An option from Administration Screen only at the instance level is the link to the Automatic Workload Repository (AWR). Once the AWR option from the Administration screen is clicked, the AWR General information is displayed. This screen includes information on all Snapshots and Collection Levels.





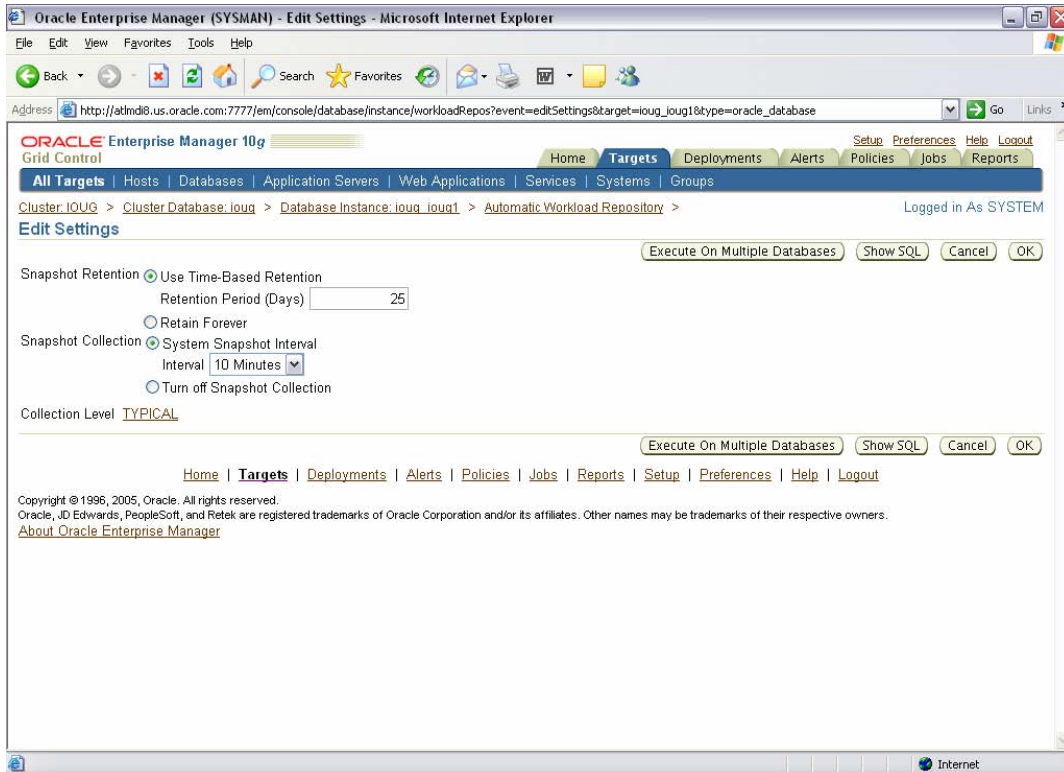
**Figure 5: Database Administration links Instance level**

In the example in Figure 6, there are 40 snapshots with a Retention of 25 days and an interval of 10 minutes (way too often - an hour may be a better interval).



**Figure 6: Automatic Workload Repository (AWR)**

By clicking on the “Edit” button (see Figure 7), the interval or retention of the information may be changed. The collection level can also be edited here.



**Figure 7: Automatic Workload Repository (AWR) Edit Settings**

By clicking on the number of snapshots displayed in the AWR General information screen (the number 40 as shown in Figure 6), the 40 snapshots will then be displayed one at a time as shown in Figure 8. The time that the snapshot was generated is listed along with the collection level.

Oracle Enterprise Manager 10g

Cluster: IOUG > Cluster Database: ioug > Database Instance: ioug\_ioug1 > Automatic Workload Repository

Logged in As SYSTEM

Switch Database Instance: ioug\_ioug1 Go

Page Refreshed Apr 23, 2006 1:58:33 PM EDT Refresh

**Snapshots**

A snapshot is a collection of database statistics at a single point in time. You can use the information in snapshots to diagnose database problems.

**Select Beginning Snapshot**

Go To Time: 4/23/06 2:00 PM Go

(Example: 12/15/03) Create

Select	ID	Capture Time	Collection Level	Within A Preserved Snapshot Set
<input type="radio"/>	26	Apr 23, 2006 11:40:34 AM	TYPICAL	
<input type="radio"/>	27	Apr 23, 2006 11:50:37 AM	TYPICAL	
<input type="radio"/>	28	Apr 23, 2006 12:00:37 PM	TYPICAL	
<input type="radio"/>	29	Apr 23, 2006 12:10:38 PM	TYPICAL	
<input type="radio"/>	30	Apr 23, 2006 12:11:38 PM	TYPICAL	
<input type="radio"/>	31	Apr 23, 2006 12:16:25 PM	TYPICAL	
<input type="radio"/>	32	Apr 23, 2006 12:30:04 PM	TYPICAL	
<input type="radio"/>	33	Apr 23, 2006 12:40:04 PM	TYPICAL	
<input type="radio"/>	34	Apr 23, 2006 12:50:04 PM	TYPICAL	
<input type="radio"/>	35	Apr 23, 2006 1:00:05 PM	TYPICAL	
<input type="radio"/>	36	Apr 23, 2006 1:10:05 PM	TYPICAL	
<input type="radio"/>	37	Apr 23, 2006 1:20:07 PM	TYPICAL	
<input type="radio"/>	38	Apr 23, 2006 1:27:41 PM	TYPICAL	
<input checked="" type="radio"/>	39	Apr 23, 2006 1:40:06 PM	TYPICAL	

**Figure 8: Automatic Workload Repository (AWR) Snapshot Listing**

Clicking on any specific snapshot to begin and end with will generate some basic snapshot details listed in Figure 9 (like a very mini-statspack), or we can run a report by clicking on Report. This will run and display the AWR Report (Figure 10).

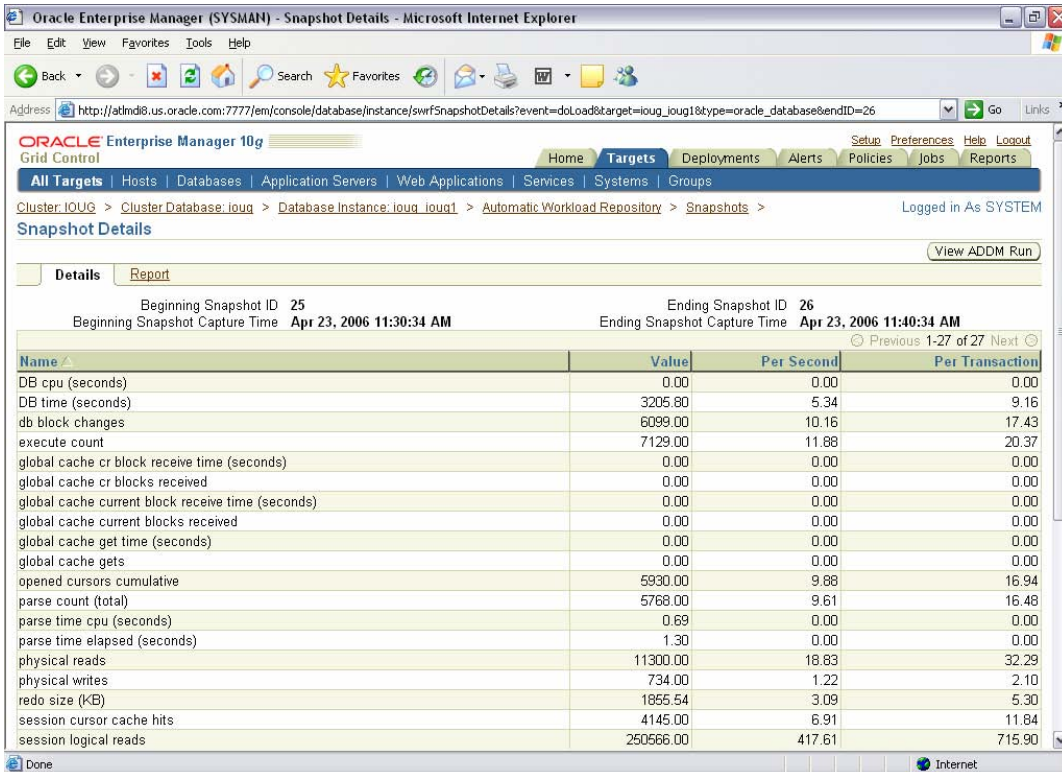


Figure 9: Automatic Workload Repository (AWR) Snapshot Listing

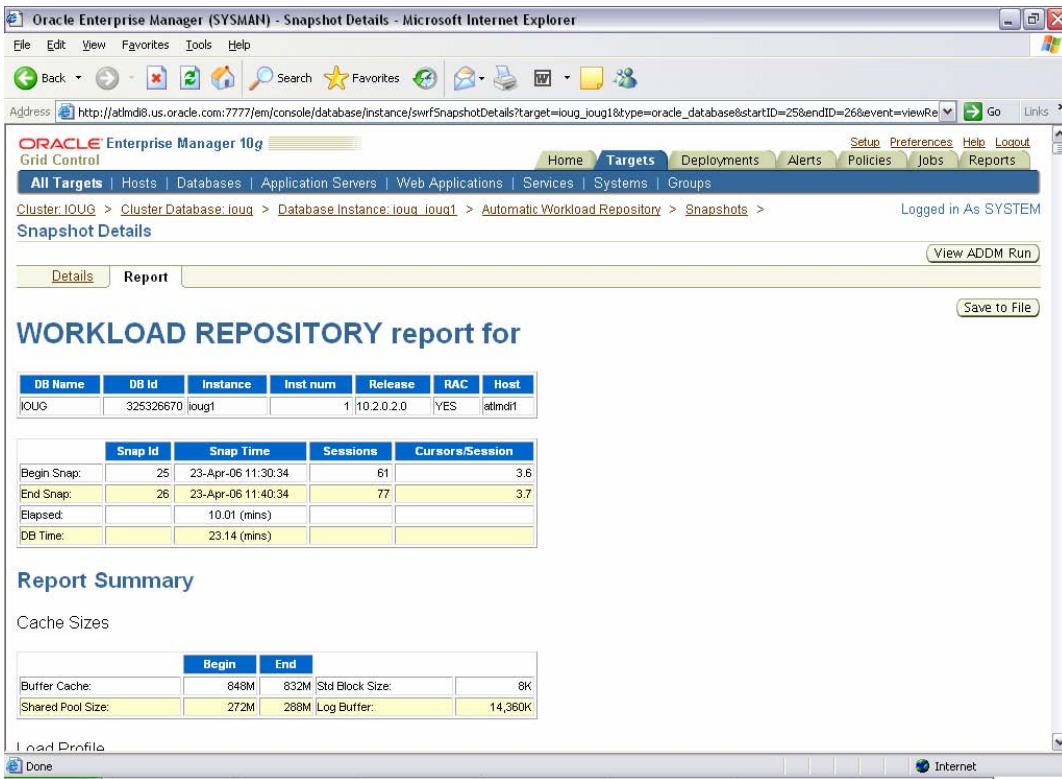


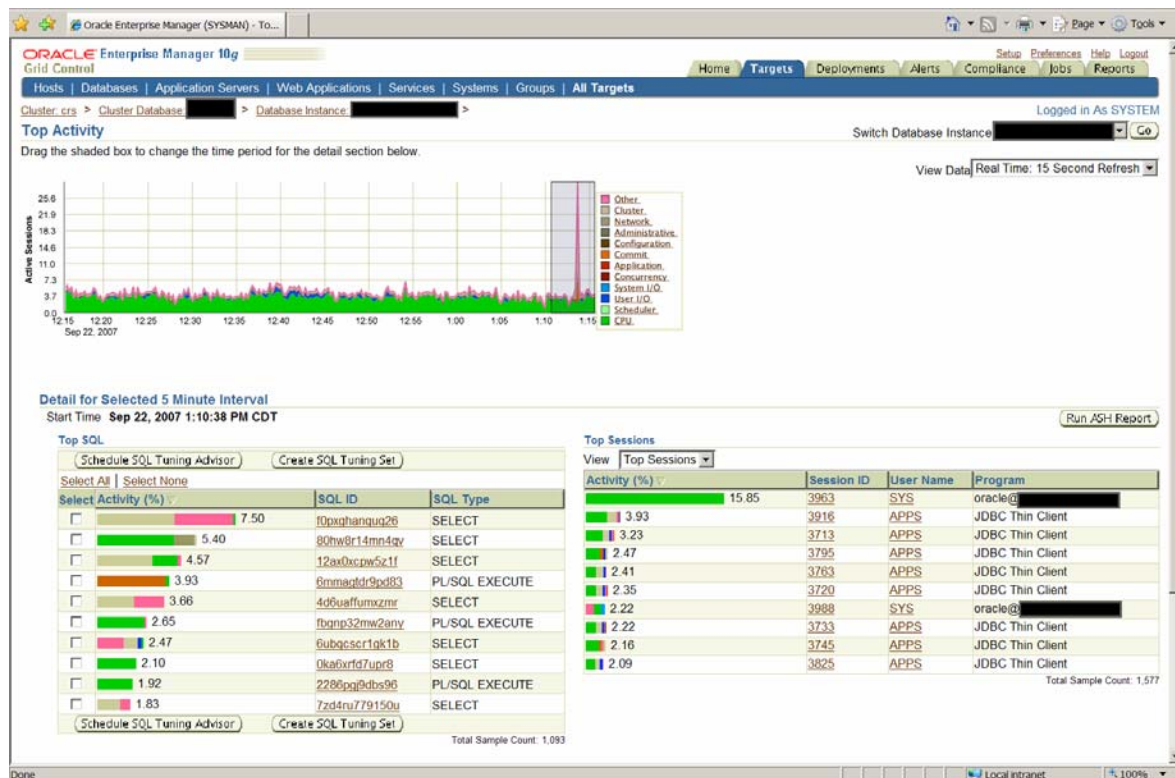
Figure 10: AWR Report Output



And there's much more to this tool including interconnect information and global block transfer information. It is a great tool for monitoring the grid. See the Collaborate '08 talk by Rich Niemiec on Grid Control for a lot more information on this great tool.

## APPLICATIONS ISSUES ON RAC:

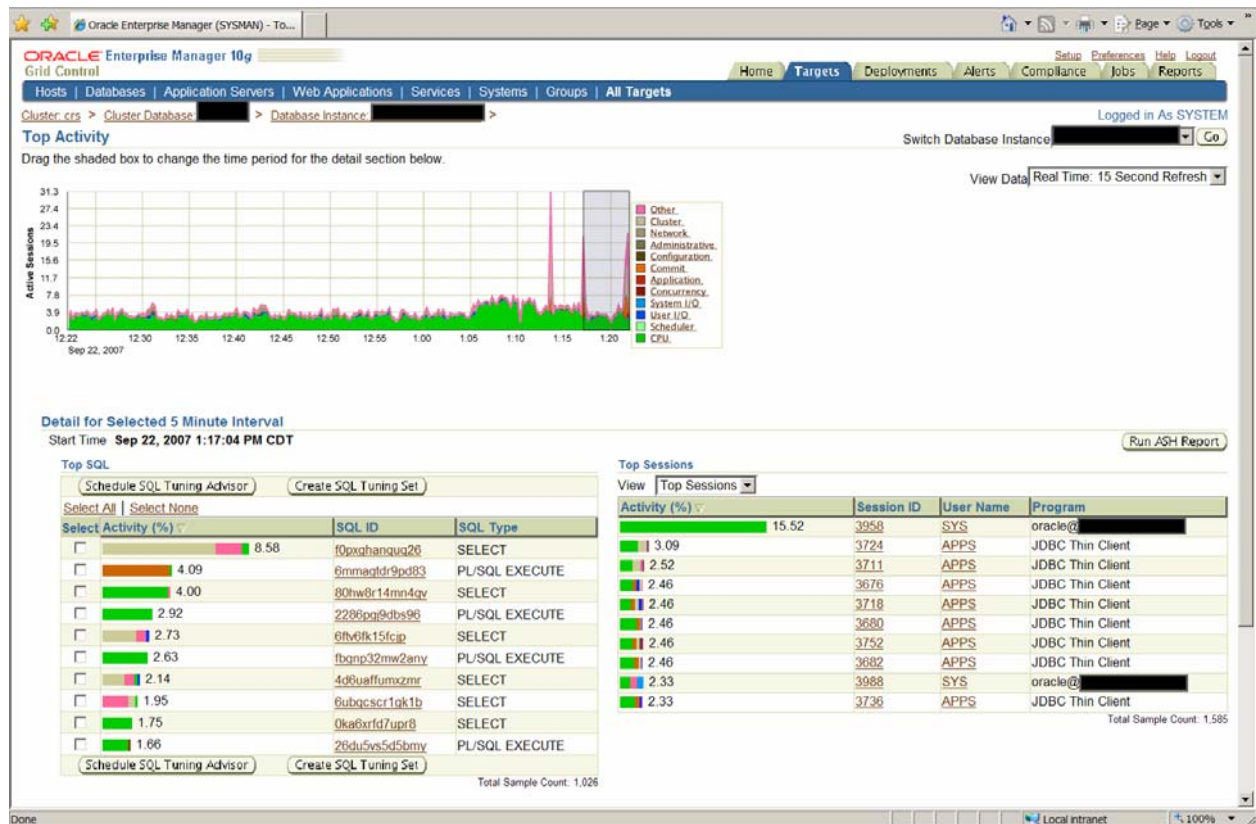
One of the biggest issues you have to face is when the system is overwhelmed to the degree that one of the nodes crashes. If Oracle can't get a heartbeat (due to being overloaded) for a node, it may shut it down (and auto restart it – given you've set it up to do that). The problems with that issue is that when a node gets overloaded, if it gets shut down, the load is transferred to another node (one of the reasons you never only want 2 nodes). That transfer of the load can cause a spike on the other node(s). Consider Figure 11, one of the nodes in a cluster has crashed on the load spikes on this node.



**Figure 11: Load Spike from another node that crashed**

The load may even be too much for the rest of the cluster to handle and the nodes might even start crashing one after another if all of them are heavily loaded. Consider Figure

12 where 3 other nodes have crashed. The surviving node has been able to “hold the fort down” so to speak, and the database continues to stay up. This is a great benefit of using RAC. As long as you’ve set it up to auto restart, you often will not feel the effects. Figure 12 shows a system that continues to be available despite 3 of the nodes in the cluster crashing and the load being moved and handled by this instance in the cluster.



**Figure 12: Load Spikes as 3 other nodes crash – this instance in the cluster continues to run and handles the load that is passed from the evicted nodes.**

Finding the problems that caused the crash or node overload is a key to stability of the cluster in the future. Using this screen (Figures 11 & 12) to find the Top Activity and corresponding SQL statements (listed below the graph) that may need to be fixed or submitted to Oracle for a patch are a key to success. Using grid control helps you show Oracle support or development that you have a problem and can also quickly show where the problem is exactly which will increase their speed in solving the issue. Grid Control is your key to finding, submitting and fixing potential problems. See the books listed in the references for more detailed information on tuning the grid. The first book listed has an entire chapter devoted to this tool and how to use it.

**REFERENCES:**

Oracle10g Performance Tuning Tips and Techniques; Niemiec, 2007

Oracle9i Performance Tuning Tips and Techniques, Niemiec, 2003

Oracle Performance Tuning Tips and Techniques, Niemiec, 1999

RAC SIG Event Update, Rich Niemiec & Murali Vallath

Oracle Clusterware, Oracle

Using Oracle Clusterware to Protect 3rd Party Applications, Oracle

IOUG Masters Tuning Class, Rich Niemiec, 2002

[www.oracleacsig.org](http://www.oracleacsig.org)

#### **AUTHOR BIOGRAPHY**

Rich Niemiec is the author of the Oracle Press book Oracle10g Performance Tuning Tips and Techniques. He is also the President of TUSC, the Chicago based Oracle Expert database integrator ([www.tusc.com](http://www.tusc.com)). TUSC is a wholly owned subsidiary of Rolta International, Inc. Rich is the President of the Midwest Oracle Users Group and the former president of the International Oracle Users Group - IOUG ([www.ioug.org](http://www.ioug.org)). Rich can be reached at [rich@tusc.com](mailto:rich@tusc.com).

TUSC is an expert level consultancy that helps companies optimize their investment in Oracle technology. We provide integrated functional and technical solutions since 1988 in the areas of Oracle's E-Business Suite, Business Intelligence/Data Warehousing, Custom Development, Managed Services/Remote DBA, Database Services, Training & Mentoring and Oracle Licensing. Please report errors in this article to [rich@tusc.com](mailto:rich@tusc.com). Neither TUSC nor the author warrants that this document is error-free. TUSC © 2008. This document cannot be reproduced without expressed written consent from an officer of TUSC except Collaborate 2008 may make copies and make this paper available as needed for the conference and proceedings.

---

Special Thanks to:



Angelo Pruscino, Murali Vallath, Sohan DeMel, Erik Peterson, Debbie Migliore, Sudip Datta, Philip Newlan, Jagan Athreya, Anil Khilani, Bharat Paliwal, Ron Weiss, Ara Shakian, Prabhaker Gongloor (GP), Barb Lundhild, Kirk McGowan, Carol Colrain, Stefan Pommerenk, Troy Anthony, Sue Tang, Joshua Solomin, Eunhei Jang, Supratim Chowdhury, Sourav Mukherjee and Khethavath Singh (KP).