TURBOCHARGE YOUR DATABASE: USE THE ORACLE DATABASE 10G SQLACCESS ADVISOR

Dr. Lilian Hobbs, Oracle Corporation Ltd

IMPROVING DATABASE PERFORMANCE

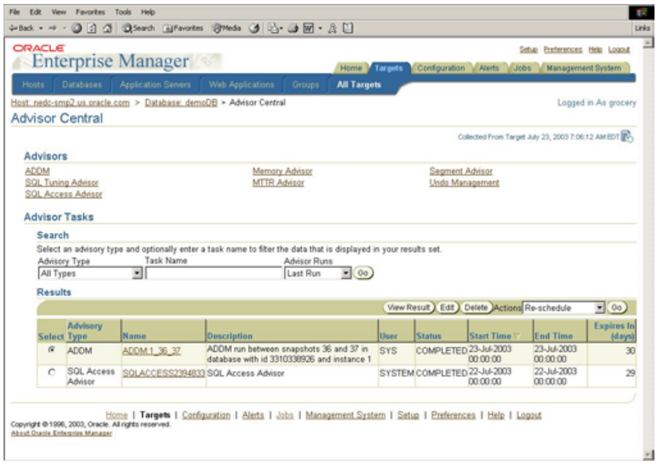
Probably every DBA or person responsible for managing a computer system, would like to improve the query response times and throughput of their system. However, as anyone who has ever tried to do this will tell you, it is not easy. There are many techniques that can be used to tune a computer system, some involve changing the hardware and many involve software changes; at the operating system, application and database level.

The problem when tuning a database is that when it is done manually it can be quite a time consuming operation, typically taking days and if it proves to be a challenge, weeks or even months to achieve optimum performance. A major concern is that as you tuning, are you using a representative example of how the database is actually being used. If not, then you may be creating objects like indexes and materialized views, that will not improve your system response time.

SQLACCESS ADVISOR

Once all the traditional tuning options have been explored, and performance could still be improved, our database tuner is now looking for new ways to tune the database and this is where the *SQLAccess Advisor* in Oracle Database 10g can help. Alternatively, if you are just starting out building a system and you are not sure which indexes or materialized views you may need, then the SQLAccess Advisor can help her also.

Figure 1 Advisor Central



File Edit View Favorites Tools Help
Back Search Favorites Media Links
ORACLE Enterprise Manager
Setup Preferences Help Logout
Home Targets Configuration Alerts Jobs Management System
Hosts Databases Application Servers Web Applications Groups All Targets

Host: nedc-smp2.us.oracle.com > Database: demoDB > Advisor Central

Logged in As grocery

Advisor Central

Collected From Target July 23, 2003 7:06:12 AM EDT

Advisors

ADDM
SQL Tuning Advisor
SQL Access Advisor
Memory Advisor
MTTR Advisor
Segment Advisor
Undo Management

Advisor Tasks

Search

Select an advisory type and optionally enter a task name to filter the data that is displayed in your results set.

Advisory Type: All Types
Task Name:
Advisor Runs: Last Run

Results

Select	Advisory Type	Name	Description	User	Status	Start Time	End Time	Expires In (days)
●	ADDM	ADDM:1_36_37	ADDM run between snapshots 36 and 37 in database with id 3310338926 and instance 1	SYS	COMPLETED	23-Jul-2003 00:00:00	23-Jul-2003 00:00:00	30
○	SQL Access Advisor	SQLACCESS2394833	SQL Access Advisor	SYSTEM	COMPLETED	22-Jul-2003 00:00:00	22-Jul-2003 00:00:00	29

Home | Targets | Configuration | Alerts | Jobs | Management System | Setup | Preferences | Help | Logout
Copyright © 1996, 2003, Oracle. All rights reserved.
About Oracle Enterprise Manager

The *SQLAccess Advisor* is a new feature which will recommend indexes and materialized views to create, retain or drop based on the workload supplied.

The SQLAccess Advisor is part of the Oracle Database 10g Server Manageability feature and is completely integrated into that environment. Therefore, the SQLAccess Advisor can be run from *Advisor Central*, shown in figure 1, thus providing an easy to use interface although there is a command line option if you prefer that approach.

FOUR SIMPLE STEPS TO GENERATE RECOMMENDATIONS & IMPLEMENT

Only a few steps have to be completed in order to generate some recommendations. All the information about this tuning process, such as what is to be tuned, the recommendations and actions are held inside a *task*. If you are using the SQLAccess Advisor from Enterprise Manager, then the task is transparent and you only need to give it a name.

STEP 1 – CREATE THE TASK

There is no limit to the number of tasks that you can create. Therefore you can create tasks to model a number of different scenarios. For example, you could configure one task to represent your daily environment, another task for your overnight batch work and then compare the recommendations from each task to see if they have anything in common. Since tasks do not have to be deleted, this would enable you to build up a view of your system over a period of time. Therefore, you could compare a task that represents your typical daily activity now, to one that you created say six months ago and see if anything has changed.

If you are using the command line interface then a task can be used as a template for another task, which can be very useful when you may want to modify the tuning parameters for a task slightly to see if that has any effect on the recommendations.

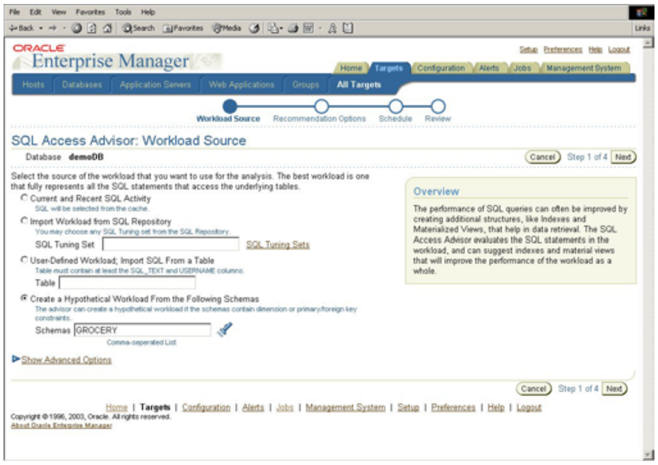
STEP 2 – DEFINE A WORKLOAD

Obtaining a good set of recommendations is very dependent on having a representative workload for your system. Database tuners will tell you that this is one of the most challenging aspects of database tuning, as gathering this data is not easy. The SQLAccess Advisor allows the following workload sources:

- Current contents of the SQL cache
- User-defined via a table containing SQL statements
- SQL Tuning Set from the Workload Repository
- Hypothetical workload, by referencing schemas(s) in the database
- Oracle 9i Advisor workload (command line only)
- Manually add SQL statements to a workload (command line only)

With all these workload sources, the database tuner now has plenty of opportunity to gather good data for the tuning process. In Figure 2 we see the first screen in the SQLAccess Advisor where the workload source is specified.

Figure 2 SQLAccess Advisor Workload Source


File Edit View Favorites Tools Help
Back Search Favorites Media
Links
ORACLE
Enterprise Manager
Setup Preferences Help Logout
Home Targets Configuration Alerts Jobs Management System
Hosts Databases Application Servers Web Applications Groups All Targets
Workload Source Recommendation Options Schedule Review
Cancel Step 1 of 4 Next
Show Advanced Options
Home | Targets | Configuration | Alerts | Jobs | Management System | Setup | Preferences | Help | Logout
About Oracle Enterprise Manager

SQL Access Advisor: Workload Source

Database **demoDB**

Select the source of the workload that you want to use for the analysis. The best workload is one that fully represents all the SQL statements that access the underlying tables.

○ Current and Recent SQL Activity
SQL will be selected from the cache.

○ Import Workload from SQL Repository
You may choose any SQL Tuning set from the SQL Repository.
SQL Tuning Set [ ] SQL Tuning Sets

○ User-Defined Workload; Import SQL From a Table
Table must contain at least the SQL_TEXT and USERNAME columns.
Table [ ]

● Create a Hypothetical Workload From the Following Schemas
The advisor can create a hypothetical workload if the schemas contain dimension or primary/foreign key constraints.
Schemas GROCERY
Comma-seperated List

Overview

The performance of SQL queries can often be improved by creating additional structures, like Indexes and Materialized Views, that help in data retrieval. The SQL Access Advisor evaluates the SQL statements in the workload, and can suggest indexes and material views that will improve the performance of the workload as a whole.

Copyright © 1996, 2003, Oracle. All rights reserved.

Within Server Manageability there is a repository of workloads, which are known as SQL Tuning sets, therefore, any one of these can be selected by the SQLAccess Advisor. Since they can be created from problems identified by the other Advisors, they can be a good workload source to use for tuning.

Alternatively you may prefer to specify your workload manually. Perhaps you extracted the SQL generated by your tool and then stored that in a table, which can be used to create a workload.

If your system is being initially designed, then you may not have any workloads available. In this case, you can use the *hypothetical* workload, where you specify one or more schemas, and from their structure, the SQLAccess Advisor will generate what it thinks a workload may represent.

WORKLOAD FILTERING

Depending on the source of the workload, it is very easy to create one that contains far more data than you require, especially if it comes from say the current contents of the SQL cache. In this scenario, if you have many applications running, then you would extract SQL statements for all of those applications, but you may be only interested in tuning some of them. Therefore all workloads can be filtered by any of the following:

- Application or module name
- Number of SQL statements
- Only queries during a specified time window

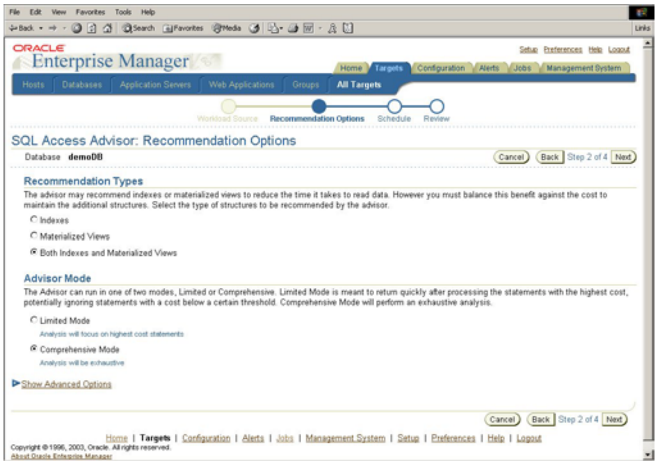
- Username
- Tables that either must or must-not be the list specified

STEP 3 – GENERATE THE RECOMMENDATIONS

Once the workload exists a set of recommendations can then be generated. In Figure 3 we see how this is done from within Oracle Enterprise Manager. Here we have the ability to say what type of recommendations are required, only indexes, only materialized views, or both. There is also the ability to say in which mode the advisor should run, *limited* or *comprehensive.*

Depending on the complexity of the workload, the number of possible solutions with respect to the indexes and materialized views that could be created can be quite significant. There are times, when you may not be interested in looking at all of these solutions and only considering the most important ones. This can be achieved by using the *limited* mode. Results will be shown quicker and then if they are not thought to be adequate, then the SQLAccess Advisor can be run again in *comprehensive* mode.

Figure 3 SQLAccess Advisor Recommendation Options


File Edit View Favorites Tools Help
Back Search Favorites Media
Links
ORACLE
Enterprise Manager
Setup Preferences Help Logout
Home Targets Configuration Alerts Jobs Management System
Hosts Databases Application Servers Web Applications Groups All Targets
Workload Source
Recommendation Options
Schedule
Review
Cancel Back Step 2 of 4 Next

SQL Access Advisor: Recommendation Options

Database demoDB

Recommendation Types

The advisor may recommend indexes or materialized views to reduce the time it takes to read data. However you must balance this benefit against the cost to maintain the additional structures. Select the type of structures to be recommended by the advisor.

○ Indexes
○ Materialized Views
● Both Indexes and Materialized Views

Advisor Mode

The Advisor can run in one of two modes, Limited or Comprehensive. Limited Mode is meant to return quickly after processing the statements with the highest cost, potentially ignoring statements with a cost below a certain threshold. Comprehensive Mode will perform an exhaustive analysis.

○ Limited Mode
Analysis will focus on highest cost statements
● Comprehensive Mode
Analysis will be exhaustive

▶Show Advanced Options
Cancel Back Step 2 of 4 Next
Home | Targets | Configuration | Alerts | Jobs | Management System | Setup | Preferences | Help | Logout
Copyright © 1996, 2003, Oracle. All rights reserved.
About Oracle Enterprise Manager

Clicking on the *Advanced Options* shown in Figure 3 will also allow you to specify any space restrictions that the recommendations must work within. A default tablespace and schema into which the indexes and materialized views will be placed and to specify the priority to be used for tuning the SQL statements such as average elapsed time or average disk reads.

Now that all of the information has been supplied the recommendations can be generated. In Enterprise Manager a job will be submitted and its progress can be monitored from *Advisor Central* which is shown in Figure 1.

STEP 4 – VIEWING & OPTIONALLY IMPLEMENTING THE RECOMMENDATIONS

Once the recommendations have been generated, they will remain until the task is deleted, thus giving the database tuner plenty of time to review the suggestions from the SQLAccess Advisor. This means that the recommendation process can be repeated many times, trying different workloads and options to see what effect this has on the suggested recommendations.

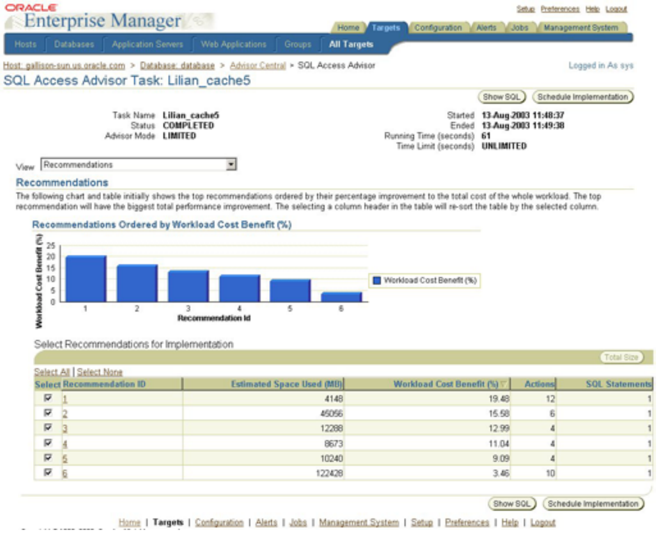
When the SQLAccess Advisor is used from within Enterprise Manager, the recommendations can be viewed by:

- Recommendations by workload cost benefit
- SQL Statements by workload cost benefit

This gives the database tuner the ability to see which recommendations will give the best improvement to the workload or individual SQL statements can be viewed to see how they can be specifically tuned, and the impact that will have on system performance.

In Figure 4 we see the *Recommendations by Workload Cost Benefit.* In this example we can see that there are 6 recommendations, where the improvement to the workload will be between 3% and 19%. In Figure 5 we see the SQL statements which comprise the workload and how each of these can be improved. Although this example shows a one to one mapping between SQL statement in the workload and recommendation, the SQLAccess Advisor recommendations can be used by multiple SQL statements.

Figure 4 Recommendation by Workload Cost


ORACLE
Enterprise Manager
Setup Preferences Help Logout
Home Targets Configuration Alerts Jobs Management System
Hosts Databases Application Servers Web Applications Groups All Targets
Show SQL
Schedule Implementation
Recommendations Ordered by Workload Cost Benefit (%)
Workload Cost Benefit (%)
25
20
15
10
5
0
1
2
3
4
5
6
Recommendation Id
Workload Cost Benefit (%)
Total Size
Select All | Select None
Home | Targets | Configuration | Alerts | Jobs | Management System | Setup | Preferences | Help | Logout

Host: gallison-sun.us.oracle.com > Database: database > Advisor Central > SQL Access Advisor   Logged in As sys

SQL Access Advisor Task: Lilian_cache5

Task Name Lilian_cache5
Status COMPLETED
Advisor Mode LIMITED

Started 13-Aug-2003 11:48:37
Ended 13-Aug-2003 11:49:38
Running Time (seconds) 61
Time Limit (seconds) UNLIMITED

View: Recommendations

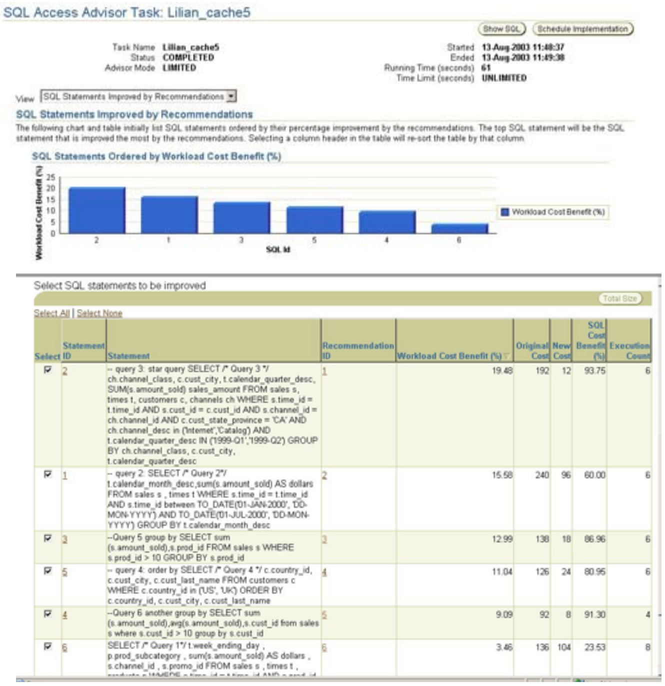
Recommendations

The following chart and table initially shows the top recommendations ordered by their percentage improvement to the total cost of the whole workload. The top recommendation will have the biggest total performance improvement. The selecting a column header in the table will re-sort the table by the selected column.

Select Recommendations for Implementation

Select	Recommendation ID	Estimated Space Used (MB)	Workload Cost Benefit (%)	Actions	SQL Statements
☑	1	4148	19.48	12	1
☑	2	45056	15.58	6	1
☑	3	12288	12.99	4	1
☑	4	8673	11.04	4	1
☑	5	10240	9.09	4	1
☑	6	122428	3.46	10	1

Figure 5 Recommendation by SQL statement


Show SQL
Schedule Implementation
SQL Statements Ordered by Workload Cost Benefit (%)
Workload Cost Benefit (%)
25
20
15
10
5
0
2
1
3
5
4
6
SQL Id
Workload Cost Benefit (%)
Total Size
Select All | Select None

SQL Access Advisor Task: Lilian_cache5

Task Name Lilian_cache5
Status COMPLETED
Advisor Mode LIMITED

Started 13-Aug-2003 11:48:37
Ended 13-Aug-2003 11:49:38
Running Time (seconds) 61
Time Limit (seconds) UNLIMITED

View: SQL Statements Improved by Recommendations

SQL Statements Improved by Recommendations

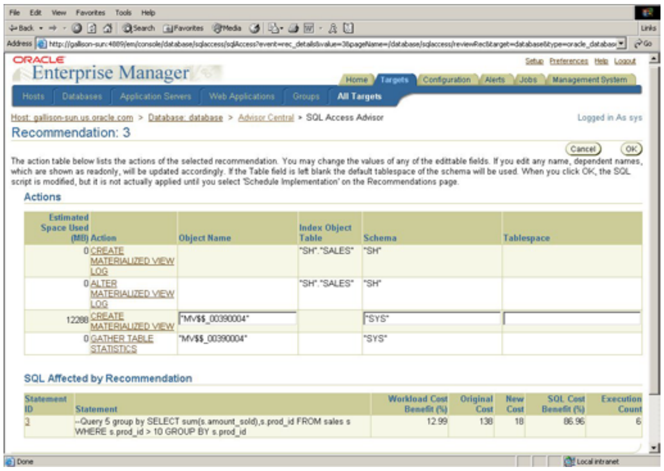
The following chart and table initially list SQL statements ordered by their percentage improvement by the recommendations. The top SQL statement will be the SQL statement that is improved the most by the recommendations. Selecting a column header in the table will re-sort the table by that column.

Select SQL statements to be improved

Select	Statement ID	Statement	Recommendation ID	Workload Cost Benefit (%)	Original Cost	New Cost	SQL Cost Benefit (%)	Execution Count
☑	2	-- query 3: star query SELECT /* Query 3 */ ch.channel_class, c.cust_city, t.calendar_quarter_desc, SUM(s.amount_sold) sales_amount FROM sales s, times t, customers c, channels ch WHERE s.time_id = t.time_id AND s.cust_id = c.cust_id AND s.channel_id = ch.channel_id AND c.cust_state_province = 'CA' AND ch.channel_desc in ('Internet','Catalog') AND t.calendar_quarter_desc IN ('1999-Q1','1999-Q2') GROUP BY ch.channel_class, c.cust_city, t.calendar_quarter_desc	1	19.48	192	12	93.75	6
☑	1	-- query 2: SELECT /* Query 2*/ t.calendar_month_desc,sum(s.amount_sold) AS dollars FROM sales s , times t WHERE s.time_id = t.time_id AND s.time_id between TO_DATE('01-JAN-2000', 'DD-MON-YYYY') AND TO_DATE('01-JUL-2000', 'DD-MON-YYYY') GROUP BY t.calendar_month_desc	2	15.58	240	96	60.00	6
☑	3	--Query 5 group by SELECT sum (s.amount_sold),s.prod_id FROM sales s WHERE s.prod_id > 10 GROUP BY s.prod_id	3	12.99	138	18	86.96	6
☑	5	-- query 4: order by SELECT /* Query 4 */ c.country_id, c.cust_city, c.cust_last_name FROM customers c WHERE c.country_id in ('US', 'UK') ORDER BY c.country_id, c.cust_city, c.cust_last_name	4	11.04	126	24	80.95	6
☑	4	--Query 6 another group by SELECT sum (s.amount_sold),avg(s.amount_sold),s.cust_id from sales s where s.cust_id > 10 group by s.cust_id	5	9.09	92	8	91.30	4
☑	6	SELECT /* Query 1*/ t.week_ending_day , p.prod_subcategory , sum(s.amount_sold) AS dollars , s.channel_id , s.promo_id FROM sales s , times t ,	6	3.46	136	104	23.53	8

Any of these links can be clicked on to view the appropriate information. For example, selecting recommendation 3, we can see in Figure 6, what the SQLAccess Advisor is proposing, which is both a materialized view and an index to improve the performance for this query and clicking on the CREATE MATERIALIZED VIEW will then display the actual statement.

Figure 6 Recommendation 3


File Edit View Favorites Tools Help
Back Search Favorites Media
Links
Go
ORACLE
Enterprise Manager
Setup Preferences Help Logout
Home Targets Configuration Alerts Jobs Management System
Hosts Databases Application Servers Web Applications Groups All Targets
Cancel
OK
Done
Local intranet

Address: http://gallison-sun:4889/em/console/database/sqlaccess/sqlAccess?event=rec_details&value=3&pageName=/database/sqlaccess/reviewRec&target=database&type=oracle_databas

Host: gallison-sun.us.oracle.com > Database: database > Advisor Central > SQL Access Advisor   Logged in As sys

Recommendation: 3

The action table below lists the actions of the selected recommendation. You may change the values of any of the edittable fields. If you edit any name, dependent names, which are shown as readonly, will be updated accordingly. If the Table field is left blank the default tablespace of the schema will be used. When you click OK, the SQL script is modified, but it is not actually applied until you select 'Schedule Implementation' on the Recommendations page.

Actions

Estimated Space Used (MB)	Action	Object Name	Index Object Table	Schema	Tablespace
0	CREATE MATERIALIZED VIEW LOG		"SH"."SALES"	"SH"	
0	ALTER MATERIALIZED VIEW LOG		"SH"."SALES"	"SH"	
12288	CREATE MATERIALIZED VIEW	"MV$$_00390004"		"SYS"	
0	GATHER TABLE STATISTICS	"MV$$_00390004"		"SYS"	

SQL Affected by Recommendation

Statement ID	Statement	Workload Cost Benefit (%)	Original Cost	New Cost	SQL Cost Benefit (%)	Execution Count
3	--Query 5 group by SELECT sum(s.amount_sold),s.prod_id FROM sales s WHERE s.prod_id > 10 GROUP BY s.prod_id	12.99	138	18	86.96	6

The SQLAccess Advisor first makes its recommendations, and then it lets you decide whether to implement all of its recommendations, or only some of them, or none. This is achieved by simply selecting the recommendations you require using either of the screens shown in Figure 4 or Figure 5.

At this stage, you have a choice on how to implement the recommendations. If you prefer to do it yourself, then the SQLAccess Advisor will generate a SQL script for you to use. You can then edit this if required, to match your company standards for data naming etc and then implement it at your leisure. Alternatively, you can schedule a job in Enterprise Manager and it will execute the script to implement these recommendations.

Command Line Interface

Although this paper has concentrated on showing how the SQLAccess Advisor is used from within Oracle Enterprise manager, it does have a command line interface, via the DBMS_ADVISOR package, which may be used if you prefer this approach. Everything described here is in fact done using the DBMS_ADVISOR package and within Enterprise Manager you can view these commands.

There are a couple of additional features that are available from the command line. This includes the ability to make a task or workload a template, that can be used to create another task or workload respectively. SQL statements can be added, changed or removed from a workload and it is possible to use a workload created for the Oracle 9i advisor.

There is also a option to tune a single statement using the DBMS_ADVISOR..QUICK-TUNE which takes as its input a SQL statement.

ORACLE 9I SUMMARY ADVISOR

If you are now interested in what the SQLAccess Advisor could do for your systems, but are not yet using Oracle Database 10g, but you are using Oracle 9i. Then you could try the Summary Advisor in Oracle 9i, which has similar functionality to the SQLAccess Advisor, but it only, recommends materialized views. Its interface is different, but it will give you an opportunity to see how the advisors work and any workloads that you create, can be used as input to the SQLAccess Advisor when you start using Oracle Database 10g.

CONCLUSION

The SQLAccess Advisor provides a very easy to use interface to help tune your systems. It requires very little system knowledge and can run without affecting production systems, since the data can be gathered from the production system and taken to another machine where the SQLAccess Advisor can be run. Since all data can be retained, it enables you to build up a library or workloads and recommendations, so you can see over a period of time, whether the patterns of usage of your systems are changing. Anyone who has to tune a database should consider using the SQLAccess Advisor because its so quick to use, that you may find that it decides that a part of your system needed tuning that you never even thought was a problem.