

15

Managing Performance by SQL Tuning

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Objectives

After completing this lesson, you should be able to:

- Use the SQL Tuning Advisor to:
 - Identify SQL statements that are using the most resources
 - Tune SQL statements that are using the most resources
- Use the SQL Access Advisor to tune a workload

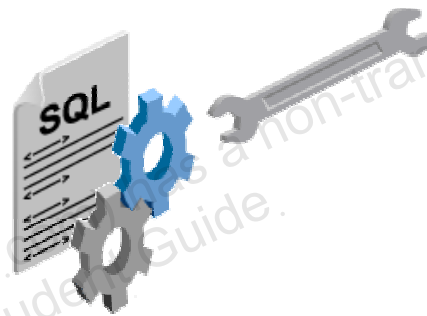
ORACLE

Copyright © 2009, Oracle. All rights reserved.

SQL Tuning

SQL tuning process

- Identify poorly tuned SQL statements.
- Tune the individual statements.
- Tune the application as a whole.



ORACLE

Copyright © 2009, Oracle. All rights reserved.

SQL Tuning

Generally, the tuning effort that yields the most benefit is SQL tuning. Poorly tuned SQL uses more resources than required. This inefficiency prevents scalability, uses more OS and database resources, and increases response time. To tune poorly tuned SQL statements, they must be identified, and then tuned. SQL statements can be tuned individually, but often the solution that optimizes one statement can hurt the performance of several others.

The SQL statements that use the most resources are by definition the statements in need of tuning. These are statements that have the longest elapsed time, use the most CPU, or do the most physical or logical reads.

Tune the individual statements by checking the optimizer statistics, check the explain plan for the most efficient access path, test alternate SQL constructions, and test possible new indexes, materialized views, and partitioning.

Test the application as a whole, using the tuned SQL statements. Is the overall performance better?

The methodology is sound, but tedious. Tuning an individual statement is not difficult. Testing the overall impact of the individual statement tuning on an application can be very difficult.

In Oracle Database 11g, a set of SQL advisors are available to identify and tune statements, individually or as a set.

SQL Advisors

Advisor Central

Advisors **Checkers**

View Data Real Time: 15 Second Refresh

Advisors

[ADDM](#) [Automatic Undo Management](#) [Data Recovery Advisor](#)
[Memory Advisors](#) [MTTR Advisor](#) [Segment Advisor](#)
[SQL Advisors](#) [SQL Performance Analyzer](#) [Streams Performance Advisor](#)

Advisor Tasks [Change Default Parameters](#)

Search

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

Advisory Type Task Name Advisor Runs Status

SQL Performance Analyzer Last 31 Days All Go

case matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the string and the symbol (%) in a double quoted string.

Re-schedule Go Previous 1-25 of 55 Next 25

Description	User	Status	Start Time	Duration (seconds)	Expires (days)
1161_1_49	ADDM auto run:	SYS COMPLETED	Aug 25, 2009 2:00:49 AM	1	3

ORACLE

Copyright © 2009, Oracle. All rights reserved.

SQL Advisors

Oracle Database 11g provides a set of advisors for SQL: the SQL Access Advisor, the SQL Tuning Advisor, the SQL Performance Analyzer, and the SQL Repair Advisor. The AWR identifies and records statistics about the recent high-load SQL statements.

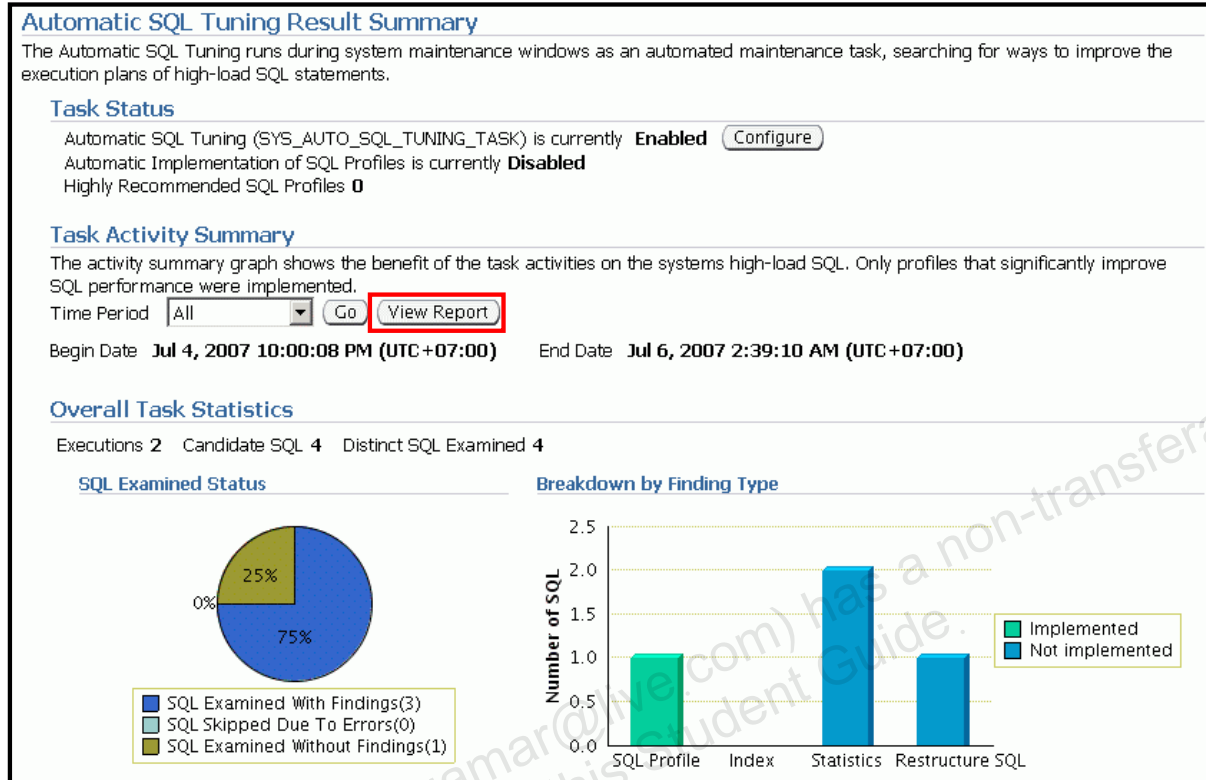
The SQL Tuning Advisor analyzes one or more SQL statements one at a time. It examines statistics, SQL profiles, indexes, materialized views, and restructured SQL. The SQL Tuning Advisor can be run manually at any time, but it is run during every maintenance window against the recent high-load SQL statements. Click Automatic SQL Tuning Results to view and implement the recommendations. This automatic job can be configured to automatically implement recommended SQL profiles for the high-load statements.

The SQL Access Advisor considers changes applied to a set of SQL statements and looks for a net gain in performance. This set can be a hypothetical set of SQL, a historical set, or a manually created set.

The SQL Performance Analyzer can be used to predict and prevent potential performance problems for any database environment change that affects the structure of the SQL execution plans.

The SQL Repair Advisor is run from the Support Workbench when a SQL statement fails with a critical error. A critical error also produces an incident. The repair advisor attempts to find and recommend a SQL patch. If no patch is found, you can continue in the Support Workbench to package the incident and submit the package to Oracle Support as a Service Request (SR).

Automatic SQL Tuning Results



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Automatic SQL Tuning Results

The Automatic SQL Tuning Task runs by default every night. The Automatic SQL Tuning Results link shows the result summary page. If you click **View Report**, each of the distinct SQL statements that were examined can be viewed.

Clicking the Configure button displays a page where you can change the defaults of the Automatic Tuning Task and enable automatic implementation of SQL profiles.

Implement Automatic Tuning Recommendations

Automatic SQL Tuning Result Details

Begin Date Jul 4, 2007 10:00:08 PM (UTC +07:00) End Date Jul 6, 2007 2:39:10 AM (UTC +07:00)

Recommendations

Only profiles that significantly improve SQL performance were implemented.

[View Recommendations](#)
[Implement All](#)

Select	SQL Text	Parsing Schema	SQL ID	Statistics	SQL Profile	Index	Restructure SQL	Miscellaneous	Error	Date
<input checked="" type="radio"/>	SELECT NULL AS table_cat, t.owner...	SYSMAN	0prhvnva3f97z	✓	(82.9%) ✓					7/5/07

Recommendations for SQL ID:0prhvnva3f97z

Only one recommendation should be implemented.

[Return](#)

SQL Text

SELECT NULL AS table_cat, t.owner AS table_schem, t.table_name AS table_name, t.column_name AS column_name, DECODE (t.data_type, 'CHAR', 1, 'VARCHAR2', 12, 'NUMBER', 3, ...)

Select Recommendation

[Original Explain Plan \(Annotated\)](#)

[Implement](#)

Select	Type	Findings	Recommendations	Rationale	Benefit (%)	New Explain Plan	Compare Explain Plans
<input checked="" type="radio"/>	SQL Profile	A potentially better execution plan was found for this statement.	Consider accepting the recommended SQL profile.		82.87		

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Implement Automatic Tuning Recommendations

If you click View Report on the Automatic Tuning Results Summary page, you will see the Automatic SQL Tuning Result Details. You can implement all the recommendations or drill down to view or implement individual recommendations. On the Recommendations page, you can click the eyeglass icon on the right to see the differences that implementing a SQL profile will make in the explain plan.

SQL Tuning Advisor: Overview



SQL Tuning
Advisor

Comprehensive SQL tuning

Detect stale or missing
statistics

Tune SQL plan
(SQL profile)

Add missing index

Restructure SQL

ORACLE

Copyright © 2009, Oracle. All rights reserved.

SQL Tuning Advisor: Overview

The SQL Tuning Advisor is the primary driver of the tuning process. It performs several types of analyses:

- **Statistics Analysis:** Checks each query object for missing or stale statistics, and makes recommendations to gather relevant statistics.
- **SQL Profiling:** The optimizer verifies its own estimates and collects auxiliary information to remove estimation errors. It builds a SQL profile using the auxiliary information and makes a recommendation to create it. When a SQL profile is created, it enables the query optimizer to generate a well-tuned plan.
- **Access Path Analysis:** New indexes are considered if they significantly improve access to each table in the query. When appropriate, recommendations to create such objects are made.
- **SQL Structure Analysis:** SQL statements that use bad plans are identified and relevant suggestions are made to restructure them. The suggested changes can be syntactic as well as semantic.

The SQL Tuning Advisor considers each SQL statement included in the advisor task independently. Creating a new index may help a query, but may hurt the response time of DML. So a recommended index or other object should be checked with the SQL Access Advisor over a workload (a set of SQL statements) to determine whether there is a net gain in performance.

Using the SQL Tuning Advisor

- Use the SQL Tuning Advisor to analyze SQL statements and obtain performance recommendations.
- Sources for SQL Tuning Advisor to analyze:
 - Top Activity: Analyzes the top SQL statements currently active
 - SQL Tuning Sets: Analyzes a set of SQL statements you provide
 - Historical SQL (AWR): Analyzes SQL statements from statements collected by AWR snapshots

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Using the SQL Tuning Advisor

The SQL Tuning Advisor runs automatically every night as the Automatic SQL Tuning Task. There may be times when a SQL statement needs immediate tuning action. You can use the SQL Tuning Advisor to analyze SQL statements and obtain performance recommendations at any time. Typically, you run this advisor as an ADDM performance-finding action.

Additionally, you can run the SQL Tuning Advisor when you want to analyze the top SQL statements consuming the most CPU time, I/O, and memory.

Even though you can submit multiple statements to be analyzed in a single task, each statement is analyzed independently. To obtain tuning recommendations that consider overall performance of a set of SQL, use the SQL Access Advisor.

SQL Tuning Advisor Options

Schedule SQL Tuning Advisor

Specify the following parameters to schedule a job to run the SQL Tuning Advisor.

* Name

Description

* SQL Tuning Set

SQL Tuning Set Description

SQL Statements Counts

Overview

The SQL Tuning Advisor analyzes individual SQL statements, and suggests indexes, SQL profiles, restructured SQL, and statistics that improve the performance of the SQL statements.

The SQL Tuning Advisor operates on a collection of SQL. You can choose a SQL Tuning Set to run the advisor. If you do not have a SQL Tuning Set with the desired SQL for running the advisor, you can create a new one.

You can click on one of the following sources, which will lead you to a data source where you can tune SQL statements using the SQL Tuning Advisor.

[Top Activity](#)
[Historical SQL \(AWR\)](#)
[SQL Tuning Sets](#)

Scope

Total Time Limit (minutes)

Scope of Analysis ☐ Limited ☒ Comprehensive

The analysis is done without SQL Profile recommendation and takes about 1 second per statement.

This analysis includes SQL Profile recommendation, but may take a long time.

Time Limit per Statement (minutes)

ORACLE

Copyright © 2009, Oracle. All rights reserved.

SQL Tuning Advisor Options

On the Schedule SQL Tuning Advisor page, you can choose the SQL statements to include and change the automatic defaults for a tuning task. You can set the source of the SQL statements, and if you have been granted the ADVISOR system privilege, you can submit the task. Enterprise Manager then creates a tuning task for the SQL Tuning Advisor.

The SQL statement options allow you to choose one or more SQL statements from recent Top Activity, choose Historical SQL stored in the AWR, or choose from a SQL Tuning Set that you have already created.

It is important to choose the appropriate scope for the tuning task. If you choose the Limited option, then the SQL Tuning Advisor produces recommendations based on statistics check, access path analysis, and SQL structure analysis. The Limited option will not make a SQL profile recommendation. If you choose the Comprehensive option, the SQL Tuning Advisor produces all the recommendations that the Limited option produces, but it also invokes the optimizer under the SQL profiling mode to build a SQL profile. With the Comprehensive option, you can also specify a time limit for the tuning task, which by default is 30 minutes. After you select Run SQL Tuning Advisor, configure your tuning task using the SQL Tuning Options page.

SQL Tuning Advisor Recommendations

SQL Tuning Results:SQL_TUNING_1183667475959

Status **COMPLETED**
Started **Jul 6, 2007 3:31:23 AM**
Completed **Jul 6, 2007 3:31:28 AM**

Page Refreshed **Jul 6, 2007 3:31:31 AM GMT+07:00** [Refresh](#)
Tuning Set Owner **SYS**
Tuning Set Name **TOP_SQL_1183667475554**
Time Limit (seconds) **1800**
Running Time (seconds) **5**

Recommendations

View

Select	SQL Text	Parsing Schema	SQL ID	Statistics	SQL Profile	Index	Restructure SQL	Miscellaneous Error
<input checked="" type="radio"/>	DECLARE table_nonexistent EXCEPTION; PRAGMA EXCEPTION_INIT (table_nonexistent, -942); BEGIN BEG...	HR	6t4uxuxdaxpff					✓
<input type="radio"/>	delete from sh.sales_copy	SYSTEM	0ggwcxx1quwuv	✓				
<input type="radio"/>	insert into sh.sales_copy select * from sh.sales	SYSTEM	axn4pkyybt51a	✓				

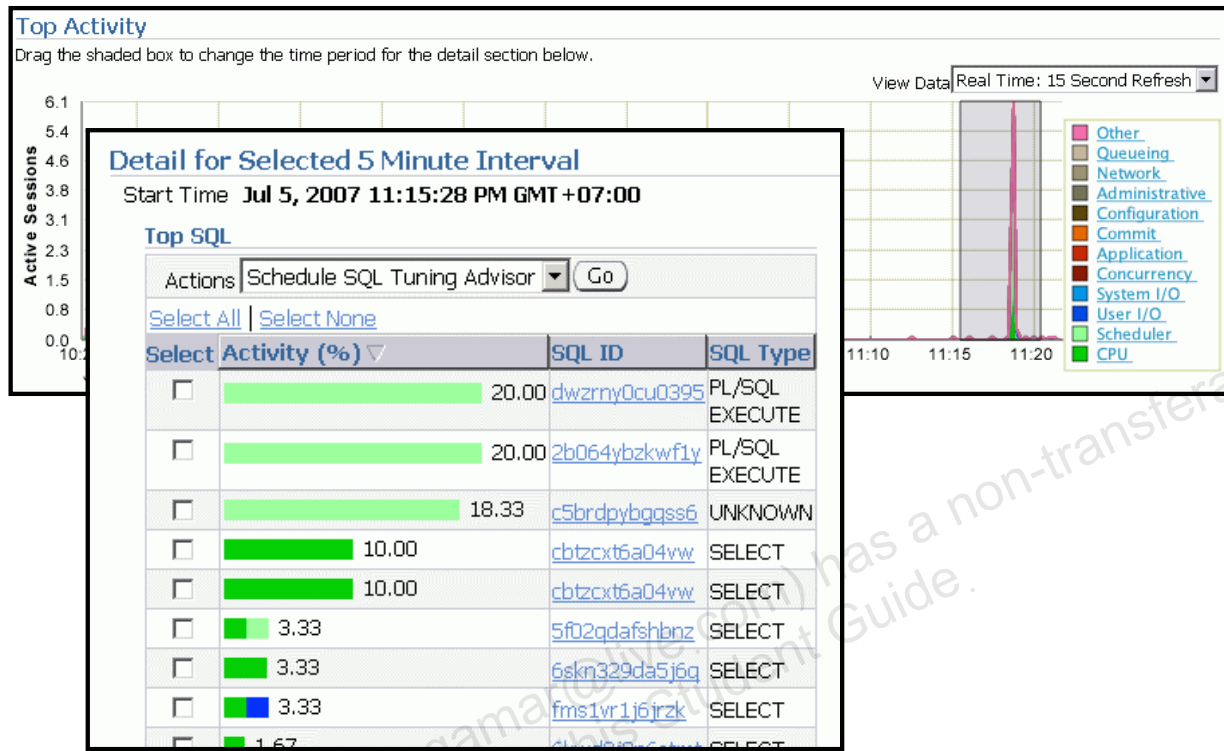
ORACLE

Copyright © 2009, Oracle. All rights reserved.

SQL Tuning Advisor Recommendations

The SQL Tuning Results for the task are displayed as soon as the task completes and can also be accessed later from the Advisor Central page. A summary of the recommendations are displayed. You can review and implement individual recommendations. Select the statement and click view.

Using the SQL Tuning Advisor: Example



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Using the SQL Tuning Advisor: Example

You can invoke the SQL Tuning Advisor by performing the following steps:

1. Click Advisor Central in the Related Links region on the Database home page.
2. Click SQL Advisors. The SQL Tuning Advisor Links page appears.

The advisor can be run on one of the following sources:

- **Active SQL:** Analyzes the Top SQL statements currently active
- **SQL Tuning Sets:** Analyzes a set of SQL statements you provide
- **Historical SQL (AWR):** Analyzes SQL captured by snapshots in the AWR

3. Select Active SQL. Select a five-minute interval to analyze by dragging the shaded box over the target time period. Select one or more statements to analyze during the selected period.
4. Click Run SQL Tuning Advisor. The SQL Tuning Options page appears showing the SQL statements in the interval. Give your task a name and description, select Comprehensive as the scope, and select Immediately for start time. Click OK.
5. Navigate back to the Advisor Central page. The status of Advisor Tasks is listed under this heading in the Results region. Wait until your task status is completed. Check the status by clicking Refresh in your browser. Select your task and click View Result. The SQL Tuning Result page appears.
6. Select the SQL statement and click View Recommendations.

Duplicate SQL

Duplicate SQL

Page Refreshed Jul 6, 2007 4:49:59 AM GMT +07:00
Refresh

Applications can cause database to consume excessive CPU by parsing SQL statements that are similar and that can share the same SQL text. Such applications can also cause slow performance by creating contention in the database for Library Cache or Shared Pool.

CPU Consumption Since Instance Started

CPU Used as percentage of Total CPU (%) **2.19**
CPU Used for Parsing as percentage of CPU Used(%) **17.20**

Duplicate SQL Statements

This report identifies similar SQL statements that could be shared by a single SQL statement if the database application used bind variables to replace literals and SQL coding conventions to remove differences based only on character case or whitespace. You can re-write the SQL statements to gain the efficiency of using a single, shared statement.

Note: Only the first 2000 SQL statements that are executed only once are examined. The actual number of SQL statements that are duplicates can be more than 2000.

[Expand All](#) | [Collapse All](#)

Duplicates	Plan Hash Value	SQL Text
▼ Duplicates		
▼ 5	1445457117	select * from hr.employees where employee_id = 148
	1445457117	select * from hr.employees where employee_id = 148
	1445457117	select * from hr.employees where employee_id = 145
	1445457117	select * from hr.employees where employee_id = 133
	1445457117	select * from hr.employees where employee_id = 100
	1445457117	select * from hr.employees where employee_id = 132

Bind variable candidates

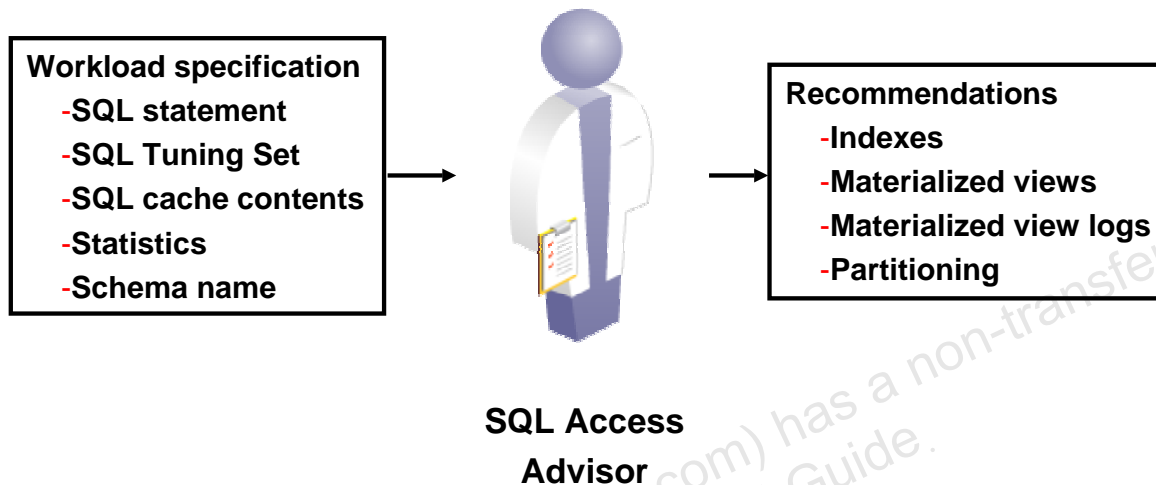
Copyright © 2009, Oracle. All rights reserved.

Duplicate SQL

Duplicate SQL statements are statements that are different only in the literal values they use or in their formatting. Statements that are different each get a separate cursor in the Library Cache. Statements that are duplicates can use the same cursor if the literals are replaced with bind variables, and the formatting is made to be the same.

Duplicate SQL statements can be identified by clicking Duplicate SQL on the Performance tabbed page in the Additional Monitoring Links section. SQL that is determined to be duplicate, except for formatting or literal differences, is listed together. This helps you determine which SQL in your application can be consolidated, thus lowering the requirements on the Library Cache and speeding up the execution of the statement.

SQL Access Advisor: Overview



Copyright © 2009, Oracle. All rights reserved.

ORACLE

SQL Access Advisor: Overview

The SQL Access Advisor can recommend the proper set of materialized views, materialized view logs, partitioning, and indexes for a given workload. Understanding and using these structures is essential when optimizing SQL because they can result in significant performance improvements in data retrieval.

The SQL Access Advisor recommends bitmap, function-based, and B-tree indexes. A bitmap index offers a reduced response time for many types of ad hoc queries and reduced storage requirements compared to other indexing techniques. B-tree indexes are most commonly used in a data warehouse to index unique or near-unique keys.

Another component of the SQL Access Advisor also recommends how to optimize materialized views so that they can be fast refreshable and take advantage of general query rewrite.

Note: For more information about materialized views and query rewrite, see the *Oracle Database Performance Tuning Guide*.

Typical SQL Access Advisor Session

ORACLE Enterprise Manager 11g Database Control

Setup Preferences Help Logout

Database

Advisor Central > Logged in As SYS

SQL Access Advisor: Initial Options

Select a set of initial options.

☐ Verify use of access structures (indexes, materialized views, partitioning, etc) only

☒ Recommend new access structures

☐ Inherit Options from a previously saved Task or Template

Cancel Continue

Overview

The SQL Access Advisor evaluates SQL statements in a workload Source, and can suggest indexes, partitioning, materialized views and materialized view logs that will improve performance of the workload as a whole.

TIP You are selecting the starting point for the wizard. All options can be changed from within the wizard.

Typical SQL Access Advisor Session

When starting a SQL Access Advisor session, you can select Use Default Options and start with a predefined set of advisor options that are recommended. Additionally, you can start a task and have it inherit a set of option values as defined by a template or task by selecting “Inherit Options from a Task or Template.” These include several generic templates designed for general-purpose environments, OLTP, and data warehousing databases. You can save custom templates from a previous task and reuse them when needed.

Click Continue to launch the SQL Access Advisor Wizard.

Note: You can access SQL Access Advisor from the Advisor Central page of Database Control.

Workload Source


SQL Access Advisor: Workload Source

Database **orcl** Cancel Step 1 of 4 **Next**

Logged In As **sys**

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.

☐ **Use an existing SQL Tuning Set**
SQL Tuning Set 

☐ **Create a Hypothetical Workload from the Following Schemas and Tables**
The advisor can create a hypothetical workload if the tables contain dimension or primary/foreign key constraints.
Schemas and Tables Add

Comma-separated list

☒ **TIP** Enter a schema name to specify all the tables belonging to that schema.

[Filter Options](#)

Workload Source

Use the SQL Access Advisor Wizard's Workload Source page to provide a defined workload that allows the Access Advisor to make recommendations. Supported workload sources are:

- **Current and Recent SQL Activity:** Uses the current SQL from the cache as the workload
- **Use an existing SQL Tuning Set:** Enables you to specify a previously created SQL Tuning Set as the workload source
- **Create a Hypothetical Workload from the Following Schemas and Tables:** Provides a schema that allows the advisor to search for dimension tables and produce a workload

The scope of the workload can be further reduced by applying filters that you can access in the Filter Options section. With these options, you can reduce the scope of the SQL statements that are present in the workload. The filters are applied to the workload by the advisor to focus the tuning effort.

Possible filter options are:

- Top resource consuming SQL statements
- Users, module identifier, or actions
- Tables

Recommendation Options

SQL Access Advisor: Recommendation Options

Database: **orcl** Cancel Back Step 2 of 4 **Next**

Logged In As: **sys**

Access Structures to Recommend

- ☒ Indexes
- ☒ Materialized Views
- ☐ Partitioning

Scope

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
Analysis will focus on highest cost statements
- ☒ Comprehensive
Analysis will be exhaustive

Advanced Options

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Recommendation Options

Use the Recommendations Options page to choose whether to limit the advisor to recommendations based on a single access method. Choose Indexes, Materialized Views, Partitioning, or a combination of those from the “Access Structures to Recommend” section. You can choose Evaluation Only to evaluate only existing access structures. In this mode, the advisor does not generate new recommendations but comments on the use of existing structures. This is useful to track the effectiveness of the current index, materialized view, and MV log usage over time.

You can use the Advisor Mode section to run the advisor in one of two modes. These modes affect the quality of recommendations as well as the length of time required for processing. In Comprehensive mode, the advisor searches a large pool of candidates resulting in recommendations of the highest quality. In Limited mode, the advisor performs quickly, limiting the candidate recommendations.

Recommendation Options

Advanced Options

Workload Categorization

Workload Volatility

☐ Consider only queries
Choose this option if this is a Data Warehouse workload

Workload Scope

☐ Recommend dropping unused access structures
Select when workload represents all access structure use cases

Space Restrictions

Do you want to limit additional space used by recommended indexes and materialized views?

☒ No, show me all recommendations (unlimited space)

☐ Yes, limit additional space to MB

Set to zero or negative to recommend dropping existing access structures to make room for better ones.

Tuning Prioritization

Prioritize tuning of SQL statements by

SQL statements will be analyzed in descending order of the value of the prioritized statistic.

☒ Consider access structures creation costs recommendations
If checked, the SQL Access Advisor will weigh the cost of creation of access structures against the frequency and potential improvement of SQL statement execution time. If unchecked, the cost of creation will be ignored. Check this box if you do not want specific recommendations generated for statements that are not executed frequently.

Default Storage Locations

By default, indexes will be placed in the schema and tablespace of the table they reference, materialized views will be placed in the schema and tablespace of the first table referenced in the query, and materialized view logs will be placed in the default tablespace of the schema of the table they reference. These fields allow you to change these default locations.

Index Tablespace	<input type="text"/>
Index Schema	<input type="text"/>
Materialized View Tablespace	<input type="text"/>
Materialized View Schema	<input type="text"/>
Materialized View Log Tablespace	<input type="text"/>
Partitioning Tablespaces	<input type="text"/>

Comma-separated list

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Recommendation Options (continued)

You can choose Advanced Options to show or hide options that enable you to set space restrictions, tuning options, and default storage locations. Use the Workload Categorization section to set options for Workload Volatility and Workload Scope. You can choose to favor read-only operations or you can consider the volatility of referenced objects when forming recommendations. You can also select Partial Workload, which does not include recommendations to drop unused access structures, or Complete Workload, which does include recommendations to drop unused access structures.

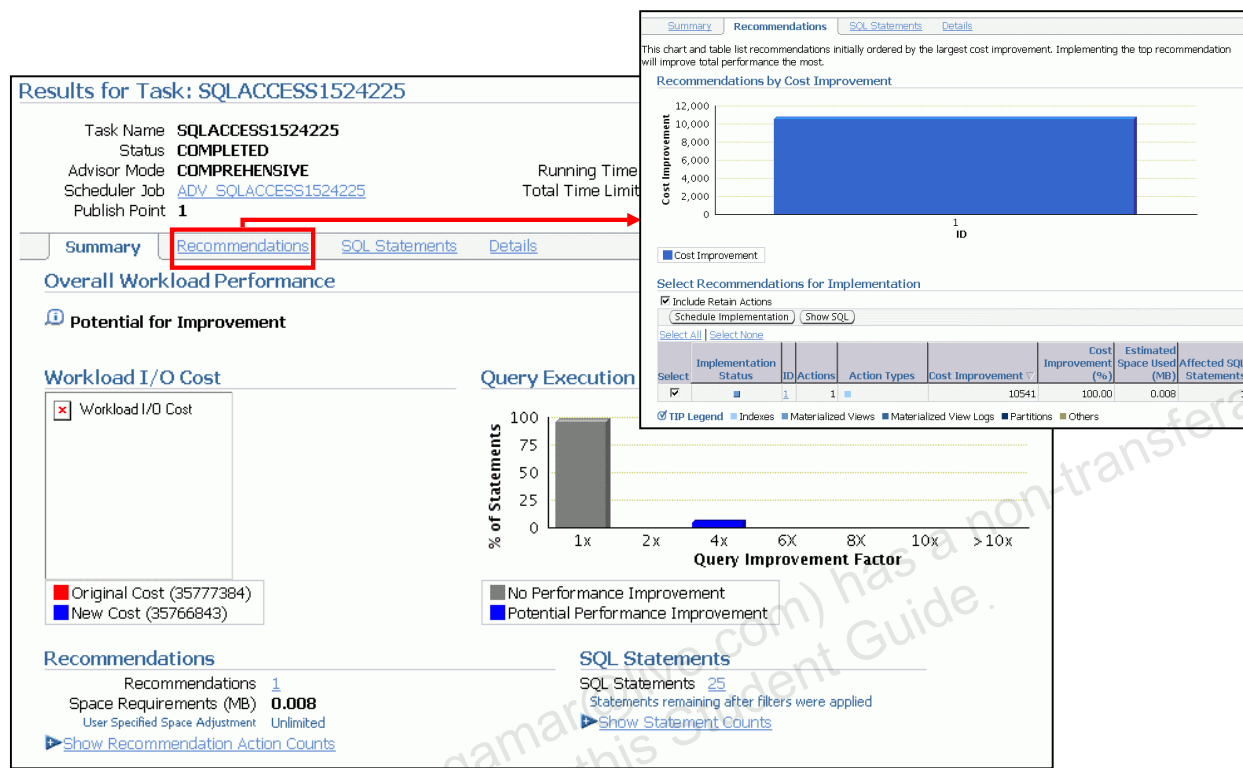
Use the Space Restrictions section to specify a hard space limit, which forces the advisor to produce recommendations only with total space requirements that do not exceed the specified limit.

Use the Tuning Options section to specify options that customize the recommendations made by the advisor. Use the “Prioritize Tuning of SQL Statements by” drop-down list to prioritize by Optimizer Cost, Buffer Gets, CPU Time, Disk Reads, Elapsed Time, and Execution Count.

Use the Default Storage Locations section to override the defaults defined for schema and tablespace locations. By default, indexes are placed in the schema and tablespace of the table they reference. Materialized views are placed in the schema and tablespace of the user who executed one of the queries that contributed to the materialized view recommendation.

After you define these parameters, you can schedule and review your tuning task.

Reviewing Recommendations



Reviewing Recommendations

Using the Advisor Central page, you can list all the completed SQL Access Advisor tasks. Select the one for which you want to see the recommendations, and then click the View Result button. Use the Results for Task Summary page to get an overview of the advisor findings. The page presents charts and statistics that provide overall workload performance and query execution time potential improvement for the recommendations. You can use the page to show statement counts and recommendation action counts.

To see other aspects of the results for the advisor task, click one of the three other tabs on the page: Recommendations, SQL Statements, or Details.

The Recommendations page displays a chart and a table that show the top recommendations ordered by their percentage improvement to the total cost of the entire workload. The top recommendations have the biggest total performance improvement.

By clicking the Show SQL button, you can see the generated SQL script for the selected recommendations. You can click the corresponding recommendation identifier in the table to see the list of actions that need to be performed in order to implement the recommendation. On the Actions page, you can actually see all the corresponding SQL statements to execute in order to implement the action. For recommendations that you do not want to implement, keep those check boxes deselected. Then, click the Schedule Implementation button to implement the retained actions. This step is executed in the form of a Scheduler job.

SQL Performance Analyzer: Overview

- Targeted users: DBAs, QAs, application developers
- Helps predict the impact of system changes on SQL workload response time
- Builds different versions of SQL workload performance (that is, SQL execution plans and execution statistics)
- Executes SQL serially (concurrency not honored)
- Analyzes performance differences
- Offers fine-grained performance analysis on individual SQL
- Is integrated with SQL Tuning Advisor to tune regressions

ORACLE

Copyright © 2009, Oracle. All rights reserved.

SQL Performance Analyzer: Overview

Oracle Database 11g includes SQL Performance Analyzer, which gives you an exact and accurate assessment of the impact of change on the SQL statements that make up the workload. SQL Performance Analyzer helps you forecast the impact of a potential change on the performance of a SQL query workload. This capability provides DBAs with detailed information about the performance of SQL statements, such as before-and-after execution statistics, and statements with performance improvement or degradation. This enables you (for example) to make changes in a test environment to determine whether the workload performance will be improved through a database upgrade.

SQL Performance Analyzer: Use Cases

SQL Performance Analyzer is beneficial in the following use cases:

- Database upgrades
- Implementation of tuning recommendations
- Schema changes
- Statistics gathering
- Database parameter changes
- OS and hardware changes

Accessible through Enterprise Manager and the DBMS_SQLPA package



ORACLE

Copyright © 2009, Oracle. All rights reserved.

SQL Performance Analyzer: Use Cases

SQL Performance Analyzer can be used to predict and prevent potential performance problems for any database environment change that affects the structure of the SQL execution plans. The changes can include (but are not limited to) any of the following:

- Database upgrades
- Implementation of tuning recommendations
- Schema changes
- Statistics gathering
- Database parameter changes
- OS and hardware changes

You can use SQL Performance Analyzer to predict SQL performance changes that result from changes for even the most complex environments. As applications evolve through the development life cycle, database application developers can test changes to schemas, database objects, and rewritten applications to mitigate any potential performance impact.

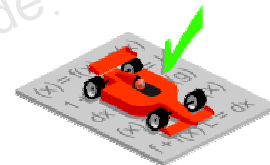
SQL Performance Analyzer also enables the comparison of SQL performance statistics.

You can access SQL Performance Analyzer through Enterprise Manager or by using the DBMS_SQLPA package.

For more information, see the *Oracle Database 11g: Performance Tuning* and the *Oracle Database 11g: New features for Administrators* course. For details about the DBMS_SQLPA package, see the *Oracle Database PL/SQL Packages and Types Reference Guide*.

Using SQL Performance Analyzer

1. Capture SQL workload on production.
2. Transport the SQL workload to a test system.
3. Build “before-change” performance data.
4. Make changes.
5. Build “after-change” performance data.
6. Compare results from steps 3 and 5.
7. Tune regressed SQL.



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Using SQL Performance Analyzer

1. **Gather SQL:** In this phase, you collect the set of SQL statements that represent your SQL workload on the production system.
2. **Transport:** You must transport the resultant workload to the test system. The STS is exported from the production system and the STS is imported into the test system.
3. **Compute “before-version” performance:** Before any changes take place, you execute the SQL statements, collecting baseline information that is needed to assess the impact that a future change might have on the performance of the workload.
4. **Make a change:** After you have the before-version data, you can implement your planned change and start viewing the impact on performance.
5. **Compute “after-version” performance:** This step takes place after the change is made in the database environment. Each statement of the SQL workload runs under a mock execution (collecting statistics only), collecting the same information as captured in step 3.
6. **Compare and analyze SQL Performance:** After you have both versions of the SQL workload performance data, you can carry out the performance analysis by comparing the after-version data with the before-version data.
7. **Tune regressed SQL:** At this stage, you have identified exactly which SQL statements may cause performance problems when the database change is made. Here, you can use any of the database tools to tune the system. After implementing any tuning action, you should repeat the process to create a new after-version and analyze the performance differences to ensure that the new performance is acceptable.

Quiz

Even when you enable Automatic Maintenance tasks, the SQL Tuning Advisor always has to be started separately.

1. True
2. False

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Answer: 2

Quiz

You can receive performance recommendations for historical SQL statements that are collected by AWR snapshots.

1. True
2. False

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Answer: 1

Quiz

The SQL Access Advisor can recommend the proper set of materialized views, materialized view logs, partitioning, and indexes for a given workload.

1. True
2. False

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Answer: 1

Quiz

The SQL Performance Analyzer provides you with detailed information about the performance of SQL statements, such as before-and-after execution statistics, and statements with performance improvement or degradation.

1. True
2. False

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Answer: 1

Summary

In this lesson, you should have learned how to:

- Use the SQL Tuning Advisor to:
 - Identify SQL statements that are using the most resources
 - Tune SQL statements that are using the most resources
- Use the SQL Access Advisor to tune a workload

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Practice 15 Overview: Managing Performance by SQL Tuning

This practice covers the following topics:

- Using SQL Tuning Advisor

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Usman Qamar (usman.qamar@live.com) has a non-transferable
license to use this Student Guide.