

Quest[®] SQL Optimizer for Oracle 8.8

User Guide

© 2013 Quest Software, Inc.
ALL RIGHTS RESERVED.

This guide contains proprietary information protected by copyright. The software described in this guide is furnished under a software license or nondisclosure agreement. This software may be used or copied only in accordance with the terms of the applicable agreement. No part of this guide may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's personal use without the written permission of Quest Software, Inc.

The information in this document is provided in connection with Quest products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Quest products. EXCEPT AS SET FORTH IN QUEST'S TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, QUEST ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL QUEST BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF QUEST HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Quest makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Quest does not make any commitment to update the information contained in this document.

If you have any questions regarding your potential use of this material, contact:

Quest Software World Headquarters
LEGAL Dept
5 Polaris Way
Aliso Viejo, CA 92656
email: legal@quest.com

Refer to our Web site (www.quest.com) for regional and international office information.

Patents

SQL Optimizer for Oracle contains patent pending technology.

Trademarks

Quest, Quest Software, the Quest Software logo, Simplicity at Work, Benchmark Factory, Foglight, LeccoTech, Quest vWorkspace, SQLab, Toad and T.O.A.D. are trademarks and registered trademarks of Quest Software, Inc. For a complete list of Quest Software's trademarks, please see <http://www.quest.com/legal/trademark-information.aspx>. Other trademarks are property of their respective owners.

Table of Contents

Introduction to SQL Optimizer	5
About Quest SQL Optimizer for Oracle	5
Optimize SQL	5
Optimize Indexes	6
Batch Optimize	6
Scan SQL	6
Inspect SGA	6
Analyze Impact	6
Manage Plans	7
SQL Optimization Workflow	7
Database Privileges	9
Use SQL Optimizer	15
Tutorial: Optimize SQL (SQL Rewrite)	15
Step 1: Optimize the SQL Statement	15
Step 2: Test Alternative SQL Statements	15
Tutorial: Optimize SQL (Plan Control)	17
Step 1: Generate and Execute Execution Plan Alternatives	17
Step 2: Deploy Execution Plan as a Baseline	17
Tutorial: Generate Indexes in Optimize SQL	19
Tutorial: Best Practices	20
Tutorial: Deploy Outlines	21
Tutorial: Batch Optimize SQL	22
Tutorial: Scan SQL	26
Tutorial: Inspect SGA	28
Tutorial: Analyze Impact	30
Tutorial: Send Indexes to Analyze Impact	32

Tutorial: Manage Outlines	33
Optimize Indexes for SGA Workload	34
Review Optimize Indexes Results	37
Appendix: Contact Quest	40
Contact Quest Support	40
Contact Quest Software	40
About Quest Software	41
Index	42

Introduction to SQL Optimizer

About Quest SQL Optimizer for Oracle

Quest SQL Optimizer for Oracle automates the SQL optimization process and maximizes the performance of your SQL statements. SQL Optimizer analyzes, rewrites, and evaluates SQL statements located within database objects, files, or collections of SQL statements from Oracle's System Global Area (SGA). Once SQL Optimizer identifies problematic SQL statements, it optimizes the SQL and provides replacement code that includes the optimized statement.

SQL Optimizer also provides a complete index optimization and plan change analysis solution. It provides index recommendations for multiple SQL statements or a SQL workload, simulates index impact analysis, and generates SQL execution plan alternatives.

SQL Optimizer consists of the following modules:

Optimize SQL

Optimize SQL includes a SQL Rewrite mode and a Plan Control mode.

SQL Rewrite Mode	Description
Optimize SQL Statements	Uses SQL Optimizer's Artificial Intelligence engine to execute SQL syntax rules and apply Oracle optimization hints to create semantically equivalent SQL statement alternatives. In addition, you can create user-defined alternatives to test with your database environment. See "About Optimizing SQL" in the online help for more information.
Execute SQL Alternatives	Executes SQL statement alternatives to view execution statistics. This provides execution times that allow you to identify the best SQL statement for your database environment. See "Execute Scenarios" in the online help for more information.
Generate Index Alternatives	Analyzes SQL statement syntax and database structure to provide index alternatives that improve performance. SQL Optimizer uses virtual indexes to generate alternatives without physically creating the indexes on your database. See "About Index Generation" in the online help for more information.
Test for Scalability	Uses Quest Benchmark Factory to simulate potential workload conditions to test SQL statement performance. See "Test for Scalability" in the online help for more information.

Incorporate Best Practices	Incorporates common best practices techniques to improve database performance. See "Best Practices" in the online help for more information.
Plan Control Mode	Description
Generate Execution Plan Alternatives	Generates execution plan alternatives for your SQL statements without changing the original source code. See "Generate Execution Plan Alternatives" in the online help for more information.
Deploy Baselines	Creates baselines from the execution plan alternatives and deploys these baselines to ensure optimal database performance. See "Deploy Baselines" in the online help for more information.

Optimize Indexes

Optimize Indexes analyzes a SQL workload or any group of SQL statements and determines the best index set for the workload or group of statements. See "About Optimize Indexes" in the online help for more information.

Batch Optimize

Batch Optimize submits files, database objects, SQL text, or statements stored in a Foglight Performance Analysis repository for batch processing. Batch Optimize scans and extracts the SQL statements, optimizes the statements, and tests the alternatives to find the best performing SQL statements for your database environment. See "About Batch Optimize" in the online help for more information.

Scan SQL

Scan SQL identifies problematic SQL statements in your source code and database objects without execution. Scan SQL then analyzes the problematic SQL statements and categorizes them according to performance levels. See "About Scanning SQL" in the online help for more information.

Inspect SGA

Inspect SGA analyzes SQL statements from Oracle's SGA. You can specify the criteria used to retrieve SQL statements and execution statistics to review SQL performance. See "About Inspect SGA" in the online help for more information.

Analyze Impact

Use Analyze Impact to evaluate the impact on a SQL workload from database changes by tracking execution plan and Oracle cost changes for SQL statements. You can run an impact analysis to estimate the performance impacts from parameter changes and new indexing. You can

also run a comparison of two different databases that run the same application. See "About Analyze Impact" in the online help for more information.

Manage Plans

Manage Plans organizes stored baselines and outlines used to improve SQL statement performance. See "About Managing Plans" in the online help for more information.

SQL Optimization Workflow

The SQL Optimization workflow ensures that your SQL statements perform optimally in your database environment.

Procedure	Description
Identify Problematic SQL	<p>Batch Optimize extracts embedded SQL statements from your database objects. After extracting the statements, it analyzes execution plan operations and identifies potential performance bottlenecks.</p> <p>Notes:</p> <ul style="list-style-type: none">You can use Inspect SGA to capture dynamic SQL statements. Save the captured dynamic SQL statements into an inspector file and use Batch Optimize to extract the statements.You can also use Scan SQL to extract embedded SQL statements.
Optimize SQL Statements	<p>Once Batch Optimize identifies problematic SQL statements, it automatically optimizes these statements and generates alternatives with unique execution plans. Batch Optimize generates the alternatives by analyzing SQL statement syntax and database structure. You can also use hints during the optimization process.</p> <p>Note: You can also use SQL Rewrite mode in Optimize SQL to optimize SQL statements extracted with Scan SQL.</p>
Execute SQL Alternatives	<p>After Batch Optimize generates alternatives, it automatically executes the alternatives and provides you with the best statement for your database environment.</p> <p>Note: Since Batch Optimize automates the SQL optimization process, you are only provided with the best alternative statement it generates. You can send your statement to Optimize SQL to view all statement alternatives available.</p>
Compare SQL Alternatives	<p>Batch Optimize compares the SQL text and execution plans of your original SQL statement and the best alternative.</p> <p>Note: If you send your statement to Optimize SQL, you can compare your original SQL statement with any of the statement alternatives available.</p>

Procedure	Description
Replace Problematic SQL Statements	Batch Optimize creates a script that you can use to replace your original source code.
Generate Index Alternatives	In addition to optimizing SQL statements, you can generate index alternatives for your statement in Optimize SQL. You can generate new indexes for a group of SQL statements or for a SQL workload using Optimize Indexes.
Generate Execution Plan Alternatives	You can use Plan Control mode in Optimize SQL to generate execution plan alternatives for your SQL statements without changing the original source code. Plan Control mode finds the best execution plan for your SQL statement and deploys it as a baseline.
Analyze Performance Changes	Analyze Impact evaluates what impact certain changes, such as indexing and parameter changes, will have on SQL statement performance.

Database Privileges

Oracle database privileges limit access for individual users. The following list summarizes the functions in SQL Optimizer that require specific Oracle database privileges.

Module	Functionality	Privilege
Optimize SQL (SQL Rewrite)	Alter session parameters for executing SQL	Requires access to SYS.V_\$PARAMETER view.
	Generate virtual indexes	Requires Oracle 8i or later.
	Deploy outlines	Requires Oracle 8i or later. Requires CREATE ANY OUTLINE and DROP ANY OUTLINE privileges. Requires access and UPDATE privileges to the following views: OUTLN.OL\$HINTS OUTLN.OLS OUTLN.OL\$NODES
Optimize SQL (Plan Control)	Open database connection	Requires Oracle 11g or later.
	Retrieve execution plans, generate alternative plans	Requires ADMINISTER SQL MANAGEMENT OBJECT privileges. Requires access to the following packages: SYS.DBMS_SQL SYS.DBMS_SPM SYS.DBMS_XPLAN Requires access to the following views: SYS.DBA_SQL_PLAN_BASELINES SYS.V_\$SQLAREA SYS.V_\$SQLTEXT_WITH_NEWLINES
Optimize SQL and Batch Optimize	Execution method option: Run on server setting	Requires access to the SYS.DBMS_SQL package.
	Retrieve run-time statistics	Requires access to the following views:

Module	Functionality	Privilege
		<p>SYS.V_\$MYSTAT SYS.V_\$STATNAME SYS.V_\$PARAMETER</p>
Optimize Indexes	Recommend indexes	<p>Requires Oracle 8i or later. Requires access to the SYS.V_\$SESSION view.</p>
	Access the AWR	<p>Requires Oracle 10g or later. Requires access to the following system views: SYS.DBA_HIST_SNAPSHOT SYS.DBA_HIST_SQLTEXT SYS.DBA_HIST_SQLSTAT</p>
	Display the control information for the Workload Repository	<p>Requires access to the following system view: SYS.DBA_HIST_WR_CONTROL</p>
	Display the SQL summary for the Workload Repository	<p>Requires access to the following system view: SYS.DBA_HIST_SQL_SUMMARY</p>
	Access Foglight PA Repository	<p>Requires access to the following tables: QUEST_SC_ACTION_DIM QUEST_SC_CLIENT_INFO_DIM QUEST_SC_MODULE_DIM QUEST_SC_SQL_STAT_FACT QUEST_SC_SQL_SYNTAX_DIM QUEST_CTRL_PYRAMID_LEVELS QUEST_DB_USER_DIM QUEST_INSTANCE_DIM QUEST_PROGRAM_DIM QUEST_TIME_DIM</p>
	Access the SGA	<p>Requires access to the SYS.V_\$SQLAREA view.</p>
Inspect SGA	<p>SQL to collect: Executed SQL from SQL area</p>	<p>Requires access to the following views: SYS.V_\$SQLAREA SYS.V_\$SQLTEXT_WITH_NEWLINES (or SYS.V_\$SQLTEXT depending on your version</p>

Module	Functionality	Privilege
		<p>of Oracle) Requires access to SYS.V_\$SQL_PLAN view in Oracle 9 or later.</p>
	<p>SQL to collect: Currently executing SQL</p>	<p>Requires access to the following views: SYS.V_\$OPEN_CURSOR SYS.V_\$SESSION SYS.V_\$SQLAREA SYS.V_\$SQLTEXT_WITH_NEWLINES (or SYS.V_\$SQLTEXT depending on your version of Oracle) Requires access to SYS.V_\$SQL_PLAN view in Oracle 9 or later.</p>
	<p>Flush Oracle shared pool</p>	<p>Requires ALTER SYSTEM privileges.</p>
	<p>Execution plan information</p>	<p>Requires access to SYS.V_\$SQL_PLAN view in Oracle 9 or later.</p>
<p>Manage Plans (Baselines Management)</p>	<p>Open a Baselines Management session or database connection</p>	<p>Requires Oracle 11g or later.</p>
	<p>View baselines</p>	<p>Requires access to the SYS.DBA_SQL_PLAN_BASELINES view.</p>
	<p>View baseline plans</p>	<p>Requires access to the SYS.DBA_SQL_PLAN_BASELINES view. Requires access to the SYS.DBMS_XPLAN package.</p>
	<p>Import, export, and migrate baselines</p>	<p>Requires ADMINISTER SQL MANAGEMENT OBJECT and CREATE TABLE privileges. Requires access to the SYS.DBA_SQL_PLAN_BASELINES view. Requires access to the SYS.DBMS_SPM package.</p>
	<p>View and modify baseline configuration values</p>	<p>Requires ADMINISTER SQL MANAGEMENT OBJECT and ALTER SYSTEM privileges. Requires access to the SYS.DBMS_SPM package. Requires access to the following views:</p>

Module	Functionality	Privilege
	Obtaining the Oracle parameter values OPTIMIZER_CAPTURE_SQL_PLAN_BASELINES and OPTIMIZER_USE_SQL_PLAN_BASELINES	SYS.V_\$SYSTEM_PARAMETER SYS.DBA_SQL_MANAGEMENT_CONFIG SYS.DBA_DATA_FILES
Manage Plans (Outlines Management)	Open an Outlines Management session or database connection	Requires Oracle 8i or later.
	View outlines	Requires OUTLN.OL\$HINTS (SELECT only) and OUTLN.OL\$ (SELECT only) privileges.
	Modify outline configuration values	Requires ALTER SYSTEM and OUTLN.OL\$ (SELECT only) privileges.
	Modify outlines	Requires DROP ANY OUTLINE and ALTER ANY OUTLINE privileges. Requires access to the SYS.OUTLN_PKG package.
Analyze Impact	Virtual indexes for Index Impact	Requires Oracle 8i or later. Requires access to the following system view: SYS.V_\$SESSION
	Altering session parameters for Parameter Impact	Requires access to the following system view: SYS.V_\$PARAMETER
	Access AWR	Requires Oracle 10g or later. Requires access to the following system views: SYS.DBA_HIST_SNAPSHOT SYS.DBA_HIST_SQLTEXT

Module	Functionality	Privilege
		SYS.DBA_HIST_SQLSTAT
	Display the control information for the Workload Repository	Requires access to the following system view: SYS.DBA_HIST_WR_CONTROL
	Display the SQL summary for the Workload Repository	Requires access to the following system view: SYS.DBA_HIST_SQL_SUMMARY
Analyze Impact (cont.)	Access Foglight PA Repository	Requires access to the following tables: QUEST_SC_ACTION_DIM QUEST_SC_CLIENT_INFO_DIM QUEST_SC_MODULE_DIM QUEST_SC_SQL_STAT_FACT QUEST_SC_SQL_SYNTAX_DIM QUEST_CTRL_PYRAMID_LEVELS QUEST_DB_USER_DIM QUEST_INSTANCE_DIM QUEST_PROGRAM_DIM QUEST_TIME_DIM
	Access SGA	Requires access to the following system view: SYS.V_\$SQLAREA
	Display database information for Compare Database Impact	Requires access to the following system views: SYS.PRODUCT_COMPONENTS_VERSION SYS.V_\$PARAMETER SYS.NLS_SESSION_PARAMETERS SYS.NLS_INSTANCE_PARAMETERS SYS.NLS_DATABASE_PARAMETERS SYS.V_\$SGAINFO SYS.V_\$DATABASE SYS.V_\$INSTANCE SYS.V_\$STATISTICS_LEVEL SYS.V_\$OPTION
All Modules	Trace setup	Requires ALTER SESSION privileges.

Module	Functionality	Privilege
	options: Enable collection of Oracle trace statistics	Requires access to the following views: SYS.V_\$SESSION SYS.V_\$PROCESS
	General	If the Oracle init parameter O7_DICTIONARY_ACCESSIBILITY for Oracle 8 or later is set to false, you cannot access objects under SYS even if you have SELECT ANY TABLE privileges. In this case, you need SELECT ANY DICTIONARY privileges or SELECT_CATALOG_ROLE to access the objects under SYS.

Use SQL Optimizer


Tutorial: Optimize SQL (SQL Rewrite)




Using SQL Rewrite mode in Optimize SQL consists of two steps. In the first step, SQL Optimizer generates semantically equivalent alternatives with unique execution plans for your original SQL statement. An Oracle cost estimate displays for each alternative generated. In the second step, SQL Optimizer executes the alternatives to test each statement's performance. This provides execution times and run time statistics that allow you to find the best SQL statement for your database environment.

Tip: The Oracle cost only provides an estimate of resource usage to execute a SQL statement. Since statements with higher cost may perform better, you should test alternatives generated to determine the best statements for your database environment.

Step 1: Optimize the SQL Statement

1. Select the **Optimize SQL** tab in the main window.
2. Select **SQL Rewrite** from the Optimize SQL start page.



Note: If the start page does not display, click the arrow beside  and select **New SQL Rewrite Session**.


3. Enter a SQL statement in the Alternative Details pane.
4. Click  to retrieve the execution plan for your SQL statement. The Select Connection and Schema window displays.
5. Select a connection and schema to use.
6. Click  to optimize the SQL statement.
7. Click  **Compare** after SQL Optimizer completes the SQL rewrite process to compare your original SQL statement with the alternatives generated.

Step 2: Test Alternative SQL Statements

The Batch Run function provides an efficient way to test alternatives SQL Optimizer generates. You can execute selected alternatives to obtain actual execution statistics. This function does not affect network traffic since SQL Optimizer can provide these statistics without having to retrieve result sets from the database server. Additionally, data consistency is maintained when using SELECT, SELECT INTO, INSERT, DELETE, and UPDATE statements because these statements run in a transaction that is rolled back after execution.

To test a SQL statement alternative

1. Click  after you finish comparing your original SQL statement with the alternatives generated.
2. Click  to execute all SQL alternatives.

Tip: You can review batch run settings before executing the alternatives by clicking  and selecting **Optimize SQL | Batch Run**.

3. Review the execution statistics in the Alternatives pane.


Tutorial: Optimize SQL (Plan Control)

Using Plan Control mode in Optimize SQL consists of two steps. In the first step, SQL Optimizer generates execution plan alternatives for your SQL statement without changing the source code. You can then execute the alternatives to retrieve run time statistics and identify the best alternative for your database environment. In the second step, you can use Plan Control mode to deploy the execution plan to the Manage Plans module as an Oracle plan baseline.

Note: This topic focuses on information that may be unfamiliar to you. It does not include all step and field descriptions.



Step 1: Generate and Execute Execution Plan Alternatives

1. Select the Optimize SQL tab in the main window.
2. Select Plan Control from the Optimize SQL start page.


Note: If the start page does not display, click the arrow beside  and select **New Plan Control Session**.


3. Enter a SQL statement in the Original SQL pane.

Tip: Select the **This SQL is contained inside a PL/SQL block** checkbox if your SQL statement originated from a PL/SQL block. Selecting this checkbox ensures that the SQL text for the baseline you create matches the SQL text in your database.

4. Click  to generate alternative execution plans for your SQL statement. The Select Connection and Schema window displays.
5. Select a connection and schema to use.
6. Click  to execute all alternative execution plans to retrieve run time statistics.
7. Review the run time statistics in the Plans pane to identify the best alternative.

Step 2: Deploy Execution Plan as a Baseline

1. Click .
2. Review the following for additional information:

Deploy	Description
Select a plan to deploy	Click  and select an execution plan alternative to deploy as a baseline plan.



Deploy	Description
Performance comparison	Use this to review the performance of selected plans against the original.
Mark the plan as	Review the following for additional information: <ul style="list-style-type: none">• Enabled—Select whether to enable or disable this plan.• Fixed—Select whether to deploy this plan as fixed or non-fixed.• Not Auto-Purged—Select whether to auto-purge when it is not used.
Plan name	Enter a name for the plan.
Description	Enter a description for this plan.

3. Click  to deploy the plan to Manage Plans.

Tutorial: Generate Indexes in Optimize SQL

SQL Optimizer identifies columns to use as index alternatives for a SQL statement after it analyzes SQL syntax, relationships between tables, and selectivity of data. SQL Optimizer then combines identified alternatives into index sets.

To generate an index alternative

1. Select the **Optimize SQL** tab in the main menu.
2. Enter a SQL statement in the Alternative Details pane.
3. Click . The Select Connection and Schema window displays.
4. Select a connection and schema to use.
5. Select Index Details in the SQL Information pane to view index generation information.
6. Select an index set to view index name, details and generation script in the Alternative Details pane.
7. To test an index, select it in the Alternatives pane and click .

Note: The Execute function allows you to test an index set SQL Optimizer generated. It physically creates the indexes on the database, runs the SQL statement, retrieves execution statistics, and drops the indexes. Since this process physically creates indexes on your database, it may impact performance of other SQL statements.



Tutorial: Best Practices

You can use Best Practices to analyze your SQL statement and database to recommend common techniques for improving database performance. Since the recommendations can also affect performance of other statements in your database, you should review and test the recommendations before implementing them. When evaluating the recommendations, take into account that database performance is affected by the following:

- System resources (CPU, I/O, memory, database architecture, and more)
- Data distribution
- System architecture
- SQL execution plans
- User's usage behavior

Note: The Best Practices function is only available in SQL Rewrite mode in Optimize SQL.



To view best practices


1. Select the Optimize SQL tab in the main window.
2. Click  Best Practices .
Tip: To display the best practices tab, click **Options**, select **Optimize SQL | Best Practices | General**, and select the **Display Best Practices tab in SQL Rewrite mode** checkbox.
3. Enter a SQL statement into the Alternative Details pane.
4. Click . The Select Connection and Schema window displays.
5. Select a connection and schema to use.
6. Review the recommendations provided.

Tutorial: Deploy Outlines

The Deploy Outline function in Optimize SQL improves SQL statement performance without changing your original source code. Using Optimize SQL, you can generate SQL statements that are semantically equivalent to your original SQL statement with alternative execution plans. Once you identify the best alternative for your database environment, you can deploy it as a stored outline to use with your original statement.

To deploy an outline

1. Select the Optimize SQL tab in the main window.
2. Click the arrow beside  and select **New SQL Rewrite Session**.
3. Enter your original SQL statement in the Alternative Details pane and click . The Select Connection and Schema window displays.
4. Select a connection and schema to use.
5. Right-click the alternative you want to deploy as an outline in the Alternatives pane and select **Deploy Outline**. The Deploy Outline window displays.
6. Review the following for additional information:

Outline name	Enter a name for the stored outline
Category	Click  and select a previously created category or enter a new category name. Notes: <ul style="list-style-type: none">• The default category name is SQL_OPTIMIZER.• You can add the outline to a disabled category until you test the SQL statement with and without using the outline.

7. Click .






Note: You can use the Outline Management feature in Manage Plans to enable and disable categories or to move outlines to different categories.











Tutorial: Batch Optimize SQL



This topic focuses on information that may be unfamiliar to you. It does not include all step and field descriptions.

To batch optimize SQL

1. Select the Batch Optimize tab in the main window.
2. Click **Add Code to Optimize** in the Batch Job List pane and select **All Types**. The Add Batch Optimize Jobs window displays.
3. Review the following for additional information:

Connection Page	Description
Connection	Click  to select a previously created database connection. Tips: <ul style="list-style-type: none"> • Click  to open the Connection Manager to create a new connection. • You can select an alternative connection for executing the SQL statement alternatives Batch Optimize generates.
Database Objects Page	Description
Database objects	Select a schema, database object type, or individual database object, and then click  to add the object. Tip: <ul style="list-style-type: none"> • Click  to browse for database objects. • Your database privileges determine if you can scan all selected database objects.
Execute using schema	Click  and select an alternative schema for executing the SQL statement alternatives.
Source Code Page	Description
Source code type	Select Text/Binary files , Oracle SQL *Plus Script , or COBOL programming source code to indicate the source code type for the file or directory you want to scan.

Add by file	Click  and browse to the files you want to add.
Add by directory	Click  and browse to the directories you want to add. Note: Select the Include Sub-directory checkbox to scan sub-directories.
Scan using schema	Click  and select a schema to scan.
Execute using schema	Click  and select an alternative schema for executing the SQL statement alternatives.
SQL Text Page	Description
SQL text	Enter SQL statement text.
Scan using schema	Click  and select a schema to scan.
Execute using schema	Click  and select an alternative schema for executing the SQL statement alternatives.
Scan SQL Page	Description
Group	Select the Scanner group that contains the SQL statements you want to scan.
Scan using schema	Click  and select a schema to scan.
Execute using schema	Click  and select an alternative schema for executing the SQL statement alternatives.
Inspect SGA Page	Description
Group	Select the Inspector group that contains the SQL statements you want to scan.
Scan using schema	Click  and select a schema to scan.
Execute	Click  and select an alternative schema for executing the

using schema	SQL statement alternatives.
Foglight Performance Analysis for Oracle Page	Description
Select a database to search for the repository used to store captured SQL	Click  to select a previously created database connection, and then click Check for PA Repository to locate the repository. Tip: Click  to open the Connection Manager to create a new connection.
Note: Batch Optimize helps you manage jobs by organizing them into batches. Use the Batch Info page to create a new batch or to add the current job to an existing batch.	

- Click **Finish** to start batch optimization.

Batch Optimize scans the job you created, classifies and optimizes the statements, and executes the SQL statement alternatives it generates.

Notes:

- Scanning starts automatically if you select the **Automatically start extracting SQL when job is added** checkbox in the Batch Optimize options page. Batch Optimize selects this checkbox by default.
- Batch Optimize selects the SQL statement to optimize based on the classification types selected in the Batch Optimize options page. Batch Optimize selects Problematic and Complex SQL classification types by default.
- Batch Optimize executes SQL alternatives it generates based on the statement types selected in the Batch Optimize options page. Batch Optimize selects SELECT statements by default.

- Select Batch List in the Batch Job List pane to view information about the jobs you created.

The Batch List pane sorts information about your jobs by batches. Additional information displays in the Jobs Improved pane.

- Select a batch from the batch list node to see details for the batch in the Job List pane.

The Job List pane displays the type of job, job status, and time of improvement for each job in the batch. Additional information displays in the SQL Classification and Cost and Elapsed Time Comparison panes for a selected job.

Tip: Select a job in the Job List pane and click  to generate a replacement script with the optimized SQL statement.

7. Select a job from the batch node to see details of the job.

The SQL List pane displays SQL classification information for the SQL statements in the job you select. The Original SQL Text and Best Alternative SQL Text panes allow you to compare your original SQL statement with the best alternative Batch Optimize generated.



Tip: Select a SQL statement in the SQL List pane and click  to send the statement to Optimize SQL and view all SQL alternatives.

Tutorial: Scan SQL


Scan SQL helps you identify problematic SQL statements in your database environment by automatically extracting statements embedded in database objects, stored in application source code and binary files, captured from Oracle's System Global Area, or saved in Foglight Performance Analysis repositories. Scan SQL retrieves and analyzes execution plans for the extracted statements and classifies them according to complexity. You can then send statements that Scan SQL classifies as problematic or complex to Optimize SQL.




Note: This topic focuses on information that may be unfamiliar to you. It does not include all step and field descriptions.





To scan SQL

1. Select the Scan SQL tab in the main window.
2. Click  to select a previously created group or click  to create a new group for your scan job.

Note: Scan SQL helps you manage scan jobs by organizing them into groups.

3. Click . The Add Scanner Jobs window displays.
4. Review the following for additional information:

Page	Description
Database Objects Page	
Database objects	Select a schema, database object type, or individual database object, and then click  to add the object. Tip: Click  to browse for database objects.
Source Code Page	
Source code type	Select Text/Binary files , Oracle SQL *Plus Script , or COBOL programming source code to indicate the source code type for the file or directory you want to scan.
Add by file	Click  and browse to the files you want to add.
Add by directory	Click  and browse to the directories you want to add. Note: Select the Include Sub-directory checkbox to scan sub-directories.
Scan using	Click  and select a schema to scan.

Page	Description
schema	
Inspect SGA Page	
Group	Select the Inspector group that contains the SQL statements you want to scan.
Scan using schema	Click  and select a schema to scan.
Foglight Performance Analysis for Oracle Page	
Select a database to search for the repository used to store captured SQL	Click  to select a previously created database connection, and then click Check for PA Repository to locate the repository. Tip: Click  to open the Connection Manager to create a new connection.
Scan using schema	Click  and select a schema to scan.


5. Click **Finish** to start scanning.

6. Select a scan job from the Job List pane to view additional information.

Details displayed in the Job List pane include the number of SQL statements found and the classification for each statement.

Tip: Click  and select a different group to display scan jobs from a different group.

7. Select a SQL statement in the SQL List pane to view additional information for the selected statement in the SQL Text and Execution Plan panes.

Tip: Click  **Send to Optimize SQL by SQL Rewrite** in the SQL Text pane to send the selected statement to Optimize SQL.

Tutorial: Inspect SGA




Inspect SGA retrieves executed SQL statements from Oracle's System Global Area or currently running SQL statements from Oracle's open cursor. Once you retrieve the statements, Inspect SGA displays the statements and their run time statistics so you can identify resource intensive statements in your database environment.


Note: This topic focuses on information that may be unfamiliar to you. It does not include all step and field descriptions.


To retrieve a previously executed SQL statement



1. Select the Inspect SGA tab in the main window.

Note: To retrieve previously executed SQL statements, you must have privileges to view SYS.V_\$SQLAREA and either SYS.V_\$SQLTEXT_WITH_NEWLINES or SYS.V_\$SQLTEXT.

2. Click  to select a group or click  to create a new group in the **Group** list.
3. Click . The Add Inspect SGA Job wizard displays.
4. Complete the following fields in the wizard:

General Information Page	Description
Job type	Select the Executed SQL from SQL Area option.
Collecting Criteria Page	Description
Collecting Criteria	Select the Top n records option and enter the number of records to display.
First by	Click  and select the statistic to use to extract SQL statements if you are not displaying all records. Note: A large SGA increases processing time.
Collection Time Page	Description
Collection Time	Select the Start collecting when you click the Inspect button option.

5. Click  to retrieve the SQL statements and run time statistics.

6. Select a statement that requires optimization in the SQL Statistics pane and click  to send to Batch Optimize SQL or click the arrow beside  **Send to Optimize SQL by SQL Rewrite** and select an option.




Tip: You can add an Inspect SGA job in Batch Optimize to optimize all the SQL statements in the collection.

Tutorial: Analyze Impact

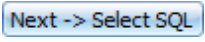
Use Analyze Impact to evaluate the impact that a change can have on a group of SQL statements or a SQL workload. The types of changes Analyze Impact will allow you to evaluate include: the addition of indexes and changes to database parameters. You can also run a comparison of two different databases that run the same application.

You can collect the SQL workload from one of several sources: an Oracle Automatic Workload Repository (AWR), Foglight PA Repository, Oracle System Global Area (SGA), or from source code.

To perform an impact analysis

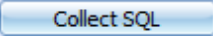
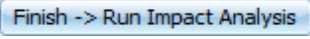
1. Select Analyze Impact.
2. Select the type of change you want to analyze. For this tutorial, select **Additional Indexes**.
3. Select the database connection and schema to use.
4. Click  to add a new index.
5. Enter a new index name or use the default.
6. In the Select Index Columns and Options pane, select the table containing the columns you want to index. The table's columns display in the lower pane. Select a column and click  to move it to the Indexed Columns list.
7. Click  to create a functional index.
8. Specify the remaining index criteria for the selected index. Review the following for additional information:



Index Type	Select an index type.
Advanced Options	Click to select from additional options for the selected index.
Compress Key	Select to add key compression. If you select <i>Compress Key</i> , you must also include the number of columns to compress in the Columns field.

9. Click  to start collecting the SQL to evaluate.
10. Select the source from which you want to collect SQL. For this tutorial, select **SGA**.
11. On the SQL collection page, click the **edit** link beside the parsing schema name. All the available parsing schemas are displayed in the graph. Select one of the parsing schemas by clicking its bar in the graph. Click the bar a second time to deselect it.
12. Click the **edit** link beside **Module**. Notice that all available modules (for the selected

schema) display in the graph. You can filter the SQL to collect by selecting a particular module or you can select **all** to collect SQL from all modules.

Note: Use the mouse pointer to hover over a bar in the graph to display a descriptive tooltip containing SQL workload details to help you select a workload.

13. Click the **edit** link beside **Action**. Notice that all available actions (for the selected schema) display in the graph. You can filter the SQL to collect by selecting a particular action or you can select **all** to collect SQL from all actions.
14. When you are finished selecting a SQL workload, click  to start the collection process.
15. SQL Optimizer collects the specified SQL statements. The SQL page opens displaying the list of SQL statements collected.
16. In the SQL Workload pane, select a SQL statement to display the statement text and the parsed execution plan.
17. You can remove a SQL statement from the workload by deselecting (clearing) the checkbox in the Include column.
18. When you are satisfied with the SQL workload, click  to start the analysis process.
19. When the analysis process is finished, one of the following will display, depending on the analysis results:
 - If an impact was found, the Impacts page displays. See "Review Impact Analysis Results" in the online help for more information.
 - If the analysis determines that non of the specified SQL were impacted, the log page displays and reports "No impacted SQL found."


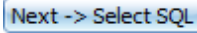
Tip: Use  and  to navigate back and forth between the pages of a tab.

Tutorial: Send Indexes to Analyze Impact

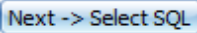
You can analyze the impact of new indexes on your SQL statement's execution plans before you physically create the indexes on your database. You can create indexes in Optimize SQL or Optimize Indexes and then send the index or indexes to Analyze Impact to evaluate the impact on your SQL workload.

Use the Optimize SQL module to generate index alternatives for a *single* SQL statement. Or use the Optimize Indexes module to generate index alternatives for a *SQL workload* or *group* of SQL statements.

To send indexes to Analyze Impact from Optimize SQL

1. Generate index alternatives using the Index Generation feature in Optimize SQL. See "Tutorial: Generate Indexes in Optimize SQL" (page 19) for more information.
2. Select the virtual index alternative you want to use for the analysis in the Alternatives pane in Optimize SQL.
3. Click  in Optimize SQL. A new Analyze Impact session opens.
4. The Indexing Change Details page is populated with the new index that you sent from the Optimize SQL session. Use this page to modify the index, if necessary. See "Analyze the Impact of Additional Indexes" in the online help for more information.
5. When you have finished modifying the index, click  to start collecting the SQL to evaluate. See "Collect SQL and Run an Impact Analysis" in the online help for more information.

To send indexes to Analyze Impact from Optimize Indexes

1. Generate index alternatives in Optimize Indexes. See "About Optimize Indexes" in the online help for more information.
2. In the Optimize Indexes session, select the Results tab.
3. On the Results page, in the Results Summary pane, click **Send to Analyze Impact**. A new Analyze Impact session opens.
4. The Indexing Change Details page is populated with the new indexes that you sent from the Optimize Indexes session. Use this page to modify the index or indexes, if necessary. See "Analyze the Impact of Additional Indexes" in the online help for more information.
5. When you have finished modifying the indexes, click  to start collecting the SQL to evaluate. See "Collect SQL and Run an Impact Analysis" in the online help for more information.

Tutorial: Manage Outlines

Outline Management displays stored outlines deployed using SQL Rewrite mode in Optimize SQL.

To manage outlines

1. Select the Manage Plans tab in the main window.

Tip: Select the **Show Manage Plans** checkbox on the Manage Plans options page to display the Manage Plans tab in the main window.

2. Click **Manage Plans**. The Create a New Manage Plans Session window displays.
3. Select a connection to use.
4. Select the Outlines Management tab.
5. Select a category in the Category/Outline pane.

You can delete or rename the selected category.

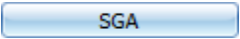
6. Select a stored outline from the category node.

You can move, rename, or reset the Used Flag on the selected stored outline.

Optimize Indexes for SGA Workload

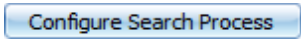
If you selected the SGA as a workload source, define the new workload and specify the search process criteria.

Define a new SGA Workload

1. If you have already created a new session, skip to 3. Otherwise, select **Optimize Indexes**.
2. Click .
3. Enter a SQL workload name and select a database connection from which to collect SQL.
4. Specify SQL filter criteria. To modify filter criteria, click the **Edit** link beside each filter. The available options display as bars in the graph. Click an option (bar) in the graph to select it. The selected option then displays in the filter criterion field in the top pane. Review the following for additional information:

SQL filter criteria pane	Description
Collect SQL for parsing schema	You must select a parsing schema before collecting SQL.
Module, Action	Use these fields to filter the available SQL statements. <i>Default = All</i>

Notes:

- Use the cursor to hover over an option (bar) in the graph to display option details.
 - Click **List statistics data** to display the options for the currently selected filter criterion in a grid format.
 - You can display additional data in the graph (or in the grid). To do this, select/deselect data options from the group of options at the bottom of the page.
5. To begin collecting SQL, click .
 6. Optimize Indexes immediately asks you to choose a second database connection. Select a database connection to use to retrieve execution plans and evaluate index alternatives.


Note: This step allows you to select an alternate database (other than the one used to collect SQL) to run the index optimization process. In other words, you can collect your SQL from one database, but run the optimize indexes process on a second database. The second database must have the same application environment and data volume statistics as the first.

7. The Search Process page opens. Before specifying search process criteria, you can review and modify your SQL workload.
 - Review collected SQL. This step is optional. You can modify the SQL workload if you want.
 - Specify search process criteria. See step four.

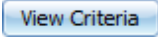
Note: To restart the SQL collection process, click **Collected from <source>** to return to the SQL Criteria page. Then click **restarted**.

Review collected SQL (optional).

1. To review and modify your SQL workload, click **Review/adjust workload SQL**. Review the following for additional information:

Pane / Button	Description
SQL Tuning Set / SQL Workload	This pane displays the list of SQL in the workload. Include—To include a SQL statement in the optimize indexes process, select the checkbox in the Include column.
SQL Text	This pane displays the SQL text of the selected SQL statement.
Parsed Execution Plan	This pane displays the execution plan of the selected SQL statement. SQL Statistics—Select this tab to display statistics of the selected statement.
 Send to Optimize SQL by :	Select to send your SQL to Optimize SQL or Batch Optimize. Click the arrow for more options.

Notes:

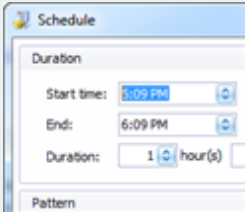
- Click  to return to the SQL Collection page. (On the SQL Collection page, click **Review SQL** to go back to the SQL Review page.)
- To collect a new SQL workload, return to the SQL Collection page and click **reset the workload**.

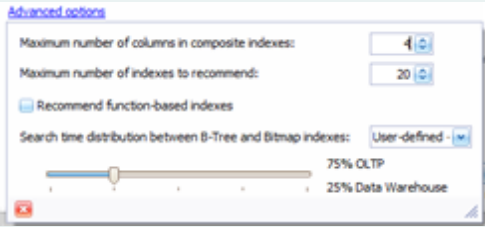
Caution: If you click **reset the workload**, the SQL workload will be lost, as well as any analysis data already gathered for the current session.

2. To return to the Search Process page, select the **Search Process** tab.

Specify search process criteria

1. Specify search process criteria in the Search Process Control pane. Review the following for additional information:

Search Process Control Pane	Description
Primary goal (minimize)	Select a primary goal. SQL Optimizer evaluates indexes to optimize workload based on this primary goal.
Enable advanced process control	<p>Click the plus sign to enable scheduling options.</p> <p>Scheduled to run on—Click the link to schedule a time to run the optimization process on this workload. Select a start time, end time, duration, and interval.</p>  <p>Stop condition—Select criteria to determine when to end the optimization process.</p> <p>Note: If you do not specify a Stop condition or an End date, the optimization process will run indefinitely.</p>
Advanced options	<p>Click this link to select the number of columns in a composite index, the maximum number of indexes to recommend, and an index-type time distribution.</p> <p>Recommend function-based indexes—Select this checkbox if you want SQL Optimizer to include function-based indexes.</p> <p>Search time distribution between B-Tree and Bitmap indexes—Select a pre-defined or user-defined time distribution. Determines the percentage of time spent searching for B-Tree vs Bitmap index types.</p>

Search Process Control Pane	Description
	

2. Click **Start** to begin searching for new indexes to improve the workload performance.

Review Optimize Indexes Results

You can review the results or the progress of your search for new indexes in Optimize Indexes.

To View Optimize Indexes Search Results

1. If you already have a session open, skip to 3. Otherwise, select **Optimize Indexes**.
2. Click **Open a Saved SQL Workload** and select a workload session. Click **Open** to open the session.


Note: You can view a summary of the results, or the progress of the search if it is ongoing, in the bottom pane of the Open SQL Workload window.

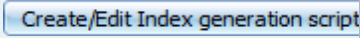
3. Select the **Search Process** tab to review details related to the search process. Each line on the graph indicates a new indexing recommendation.

Notes:

- Use your cursor to hover over a line on the graph to display a summary of the recommendation's improvements (in a tool tip). Then click **View details** (in the tool tip) to open the Results page for the *selected* recommendation.
 - Select **All** in the Display Range box, to display all index recommendations in the graph.
 - Click **List all results** to display all index recommendations and their cost improvement values in a grid format. Click a recommendation in the list to open the results page for the *selected* recommendation.
4. Select the **Results** tab to view the final recommendation (or the last recommendation if the search process is ongoing).

Review the following for additional information:

Pane / Option / Button	Description										
<p>n Recommended Indexes</p>	<p>Displays a list of the indexes for the recommendation.</p> <p>Index Name—Click an Index name to see the index details, including a list of existing indexes which you can use to check for duplication.</p> <p>Tip: Select (highlight) an index in the Recommended Indexes list to highlight the SQL statements in the Impacted SQL list that are impacted by that recommended index.</p>										
<p>Results Summary</p>	<p>(left side)—Displays a summary of workload improvement results.</p>  <table border="1"> <caption>Results Summary Data</caption> <thead> <tr> <th>Metric</th> <th>Estimated Improvement (%)</th> </tr> </thead> <tbody> <tr> <td>Cost</td> <td>89.71</td> </tr> <tr> <td>CPU Cost</td> <td>90.62</td> </tr> <tr> <td>IO Cost</td> <td>89.70</td> </tr> <tr> <td>elapsed Time</td> <td>87.43</td> </tr> </tbody> </table> <p>Note: If the links at the bottom of the Results Summary pane are not in view, click and drag the pane's bottom edge down to resize.</p> <p>(right side)—Displays the workload timeslots.</p> <p>Use the cursor to hover over a timeslot to display a summary of the improvement to that timeslot's SQL.</p> <p>Note: Timeslots display only for workloads collected from snapshots.</p>	Metric	Estimated Improvement (%)	Cost	89.71	CPU Cost	90.62	IO Cost	89.70	elapsed Time	87.43
Metric	Estimated Improvement (%)										
Cost	89.71										
CPU Cost	90.62										
IO Cost	89.70										
elapsed Time	87.43										
<p>View</p>	<p>Displays one of two different evaluations for the Workload:</p> <ul style="list-style-type: none"> • Plan Statistics view—Displays the estimated improvements based on changes in plan statistics and weighed by the number of executions for each SQL per timeslot. • Projected Actual Statistics view—Displays the estimated improvements calculated using actual statistics and changes in plan statistics. 										

Pane / Option / Button	Description
Publish Report	Click to view the Optimize Indexes Workload Report which you can customize and then export, print, or email.
Send to Analyze Impact	Click to send the results to the Analyze Impact module where you can evaluate the impact of the new indexes on other SQL.
Chart Options	Click this link to select which charts/statistics to display.
Impacted SQL	<p>SQL Name—Click a SQL name to review the SQL statement and to compare the recommended execution plan to the original execution plan.</p> <p>Tip: Select (highlight) a SQL statement in the Impacted SQL list to highlight the indexes in the Recommended Indexes list that contributed to the performance improvement of that statement.</p>
	Click to create a script to generate the new indexes recommended by Optimize Indexes.

Notes:

- To save the contents of a grid (list) as a file, right-click the grid and select **Save As**.
- An open database connection is not required when opening a completed session, reviewing results, or creating and publishing (printing) a report.

Tip: Use  and  to navigate back and forth between the pages of a tab.

Appendix: Contact Quest

Contact Quest Support

Quest Support is available to customers who have a trial version of a Quest product or who have purchased a Quest product and have a valid maintenance contract. Quest Support provides unlimited 24x7 access to our Support Portal at <http://support.quest.com>.

From our Support Portal, you can do the following:

- Retrieve thousands of solutions from our online Knowledgebase
- Download the latest releases and service packs
- Create, update and review Support cases

View the *Global Support Guide* for a detailed explanation of support programs, online services, contact information, policies and procedures. The guide is available at: <http://support.quest.com>.

You may also find help and additional information at the SQL Optimizer community at:

<http://sqloptimizeroracle.inside.quest.com>

Contact Quest Software

Email info@quest.com

Quest Software, Inc.
World Headquarters

Mail 5 Polaris Way
Aliso Viejo, CA 92656
USA

Web site www.quest.com

See our web site for regional and international office information.

About Quest Software

Quest Software simplifies and reduces the cost of managing IT for more than 100,000 customers worldwide. Our innovative solutions make solving the toughest IT management problems easier, enabling customers to save time and money across physical, virtual and cloud environments. For more information about Quest go to www.quest.com.

Index

A

Analyze Impact
tutorial 30

B

Batch Optimize
tutorial 22

best practices
tutorial 20

D

database privileges 9

deploy
outline 21

E

execute
SQL Optimizer tutorial 15

I

identify problematic SQL
process to find 7

Index Generation
tutorial 19

Index Impact Analysis
tutorial 32

Inspect SGA
tutorial 28

M

Manage Outlines
tutorial 33

O

optimize SQL
about 5
batch optimize 26
performance assurance 7
SQL Optimizer tutorial 15

P

Plan Control
tutorial 17

privileges, database 9

S

Scan SQL
tutorial 26

SGA
define workload (Optimize Indexes) 34

SQL Optimizer
about 5
tutorial 15

T

test alternative SQL 7

V

V_\$OPEN_CURSOR 11

V_\$SESSION 10

V_\$SQLAREA 9

V_\$SQLTEXT 9