NINE STRENGTHS OF PRECISE FOR DATABASES

Precise for Databases provides a complete view of the performance of databases by capturing, measuring, and correlating performance metrics from all critical components of the system (Figure 1). It proactively monitors, diagnoses, detects and corrects performance problems before they become noticeable for databases and related applications. Precise provides a wealth of expert knowledge regarding possible areas of deterioration and links that guide to the resolution of specific problems to optimize the performance of databases.
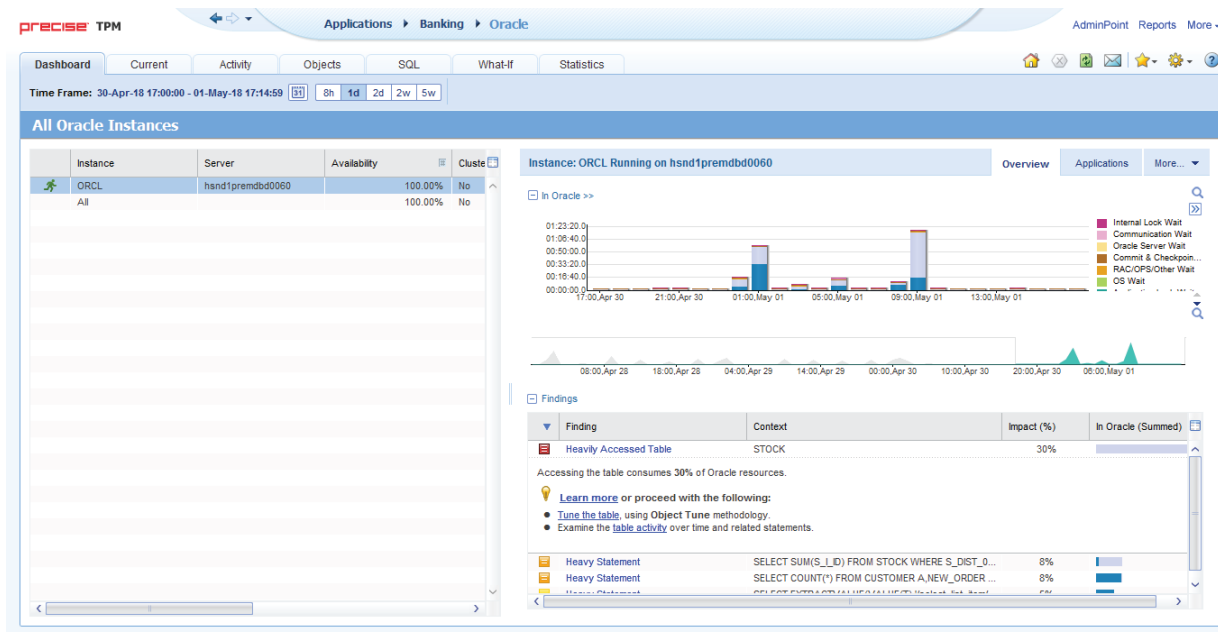

Figure 1: The *'Dashboard'* workspace.

The solution brief presents a selection of ten strengths of Precise for Databases that includes supporting multiple database platforms, sampling frequently and continuously with low overhead, viewing findings, tuning of objects, displaying actual execution plans, viewing recommendations, performing what-if analysis, tuning performance smartly, and setting up flexible licensing.

1 SUPPORT MULTIPLE DATABASE PLATFORMS (ALSO IN THE CLOUD)

Precise for Databases supports multiple relational database management systems. It supports Microsoft SQL Server, Oracle Database, IBM DB2, and SAP Sybase ASE (Figure 2). For each of these database platforms, the user interface of Precise looks and behaves the same to minimize the learning curve and enable the same process to solve problems rapidly.

Figure 2: Precise for Databases supports Microsoft SQL Server, Oracle Database, IBM DB2, and SAP Sybase ASE.

In addition to monitoring of on-premises instances of these databases, Precise can monitor instances of SQL Server, Oracle Database, IBM DB2, and SAP Sybase ASE that run in virtual machines hosted by Amazon Elastic Compute Cloud (EC2) and Microsoft Azure Virtual Machine (VM).

2 SAMPLE FREQUENTLY AND CONTINUOUSLY WITH LOW OVERHEAD

Precise for Databases collects performance data from databases, and then correlates and loads the data into its data repository (Figure 3). The data collector of Precise runs on database instances using a unique, accurate and comprehensive sampling technology that requires low overhead. The data is then analyzed and summarized by the focal point of Precise. The web browser interface for Precise provides access to the data at a later time without affecting the performance of the monitored database instances. The repository provides performance tracking for assessing performance trends for databases and analyze changes in activities of databases over long periods of time. This setup samples the monitored database once per second, has low overhead, is non-intrusive and is safe for production environments. Precise captures all of the information for the monitoring of SQL statements continuously (that is, 24 by 7). The ranking of top SQL statements is crucial to performance monitoring. Correlating the SQL statements to servers, programs, and users places the SQL statements in the context of the application. Instead of having to activate additional high-overhead database monitoring and to recreate the problem after the problem occurred, Precise always captures everything so that there is no delay in solving problems.
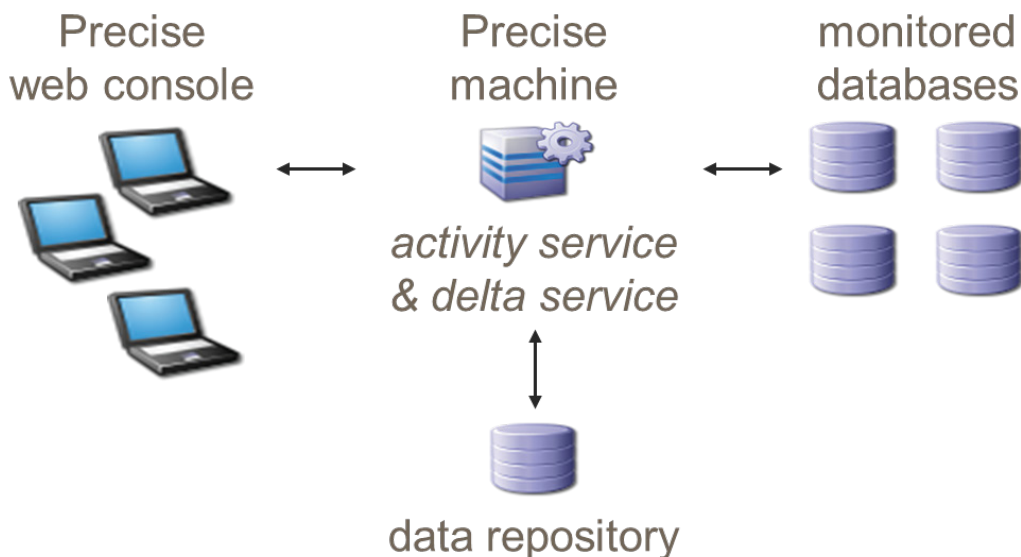

Figure 3: The architecture of Precise for Databases.

3 VIEW FINDINGS

Precise for Databases displays a 'Findings' area in relevant workspaces (Figure 4) that shows problematic findings for databases. The 'Findings' area provides an overview of performance issues within the monitored database and enables efficient navigation to the relevant area in the graphical user interface for further analysis and handling. The displayed performance findings may indicate performance deteriorations and incorrect use of methods. Each finding appears as a row in the displayed table of findings. Hover the mouse indicator over the single-line displayed finding to expand the finding to view essential guidelines as

to what may be the cause for the reported problem, and what the recommended steps are to resolve the problem. The 'Learn more' hyperlink provides recommendations for solving the selected finding. For each finding, it lists all relevant pieces of advice and all applicable solutions. The 'Proceed with the following' bulleted hyperlinks provide expert knowledge about the selected finding to provide direction to decide which advice to take or which solution to implement.
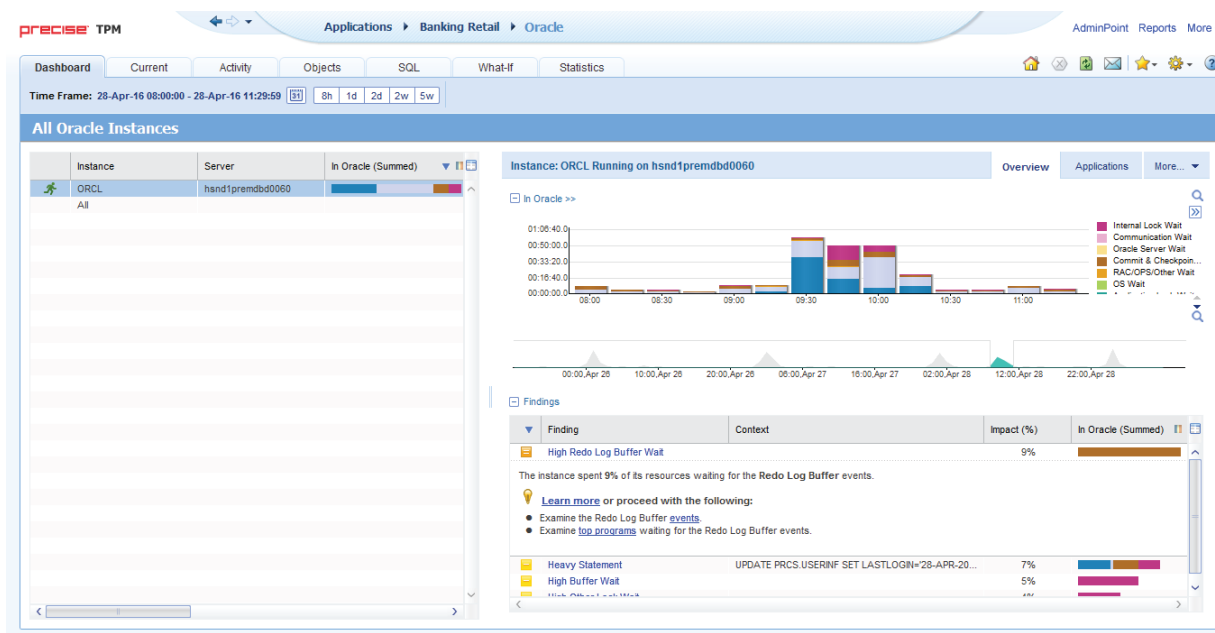

Figure 4: The *'Findings'* area at the bottom of the *'Overview'* tab of the *'Dashboard'* workspace.

4 TUNE OBJECTS

Precise for Databases can tune database objects when it is not possible to modify the text of the SQL statements that access the objects.

To identify suitable candidates for object tuning, examine the performance of active objects in the 'Explore Objects' tab of the 'Objects' workspace (Figure 5). Observe whether an object is an index or table, the name of the database that contains the object, a summary of the time the object spent in the database instance, and the size of the object measured in pages. Precise can also display additional information regarding input/output operations and waits, contentions, index usability, and logical, user and system operations. Select the launch icon next to an object to launch the 'Recommend' tab (Figure 6) to view recommendations on the selected table.

Examples of using the 'Objects' workspace to identify performance problems include:

Analyze how configuration changes affect resource consumption: Changing instance configuration and database parameters may affect the performance of the system. Track the configuration changes and compare them with the resource consumption of the database over time.

Analyze how schema changes affect the resource consumption of various entities: Changing the database schema may affect the performance of the system. These changes are implemented to improve the performance of the application. Monitor the changes to ensure that the goals have been achieved. Track the schema changes and compare them with the resource consumption of the entire database or a table to detect changes that affect the resource consumption of the database.

Examine the space usage of the object over time: Space usage information, collected by the 'Collect Space Utilization' process, allows the examination of the space allocated versus the space used, in a table and index. The information allows keeping track of object growth over time easily and decide how to spread database files over various disks and how to connect objects to file groups in the specified database.

Rank the busy tables in the system: Identifying the most critical tables to processing can ensure that they are on the appropriate maintenance schedule to minimize fragmentation.

Examine the access methods to tables and indexes: Examining the access methods used on tables and indexes can help make decisions regarding the types of indexes to employ.

Examine the distribution statistics of indexes: Distribution statistics of an index can determine whether or not the index is helpful to the SQL optimizer. The more selective an index is, the more useful it is since the means more rows can be eliminated from consideration.

Determine whether to recommend an index for a table: The decision to add a specific index to a table is complex. The index can improve one SQL statement and may negatively affect another. Therefore, before adding an index, examine every SQL statement accessing the table to see how the new index will impact it.
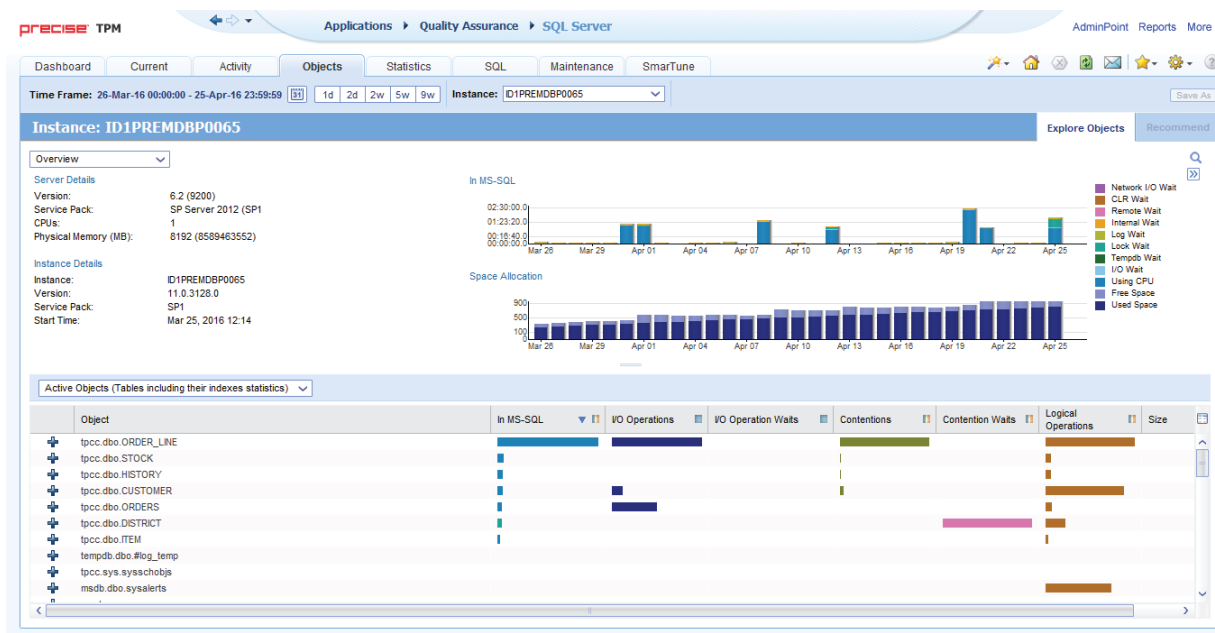

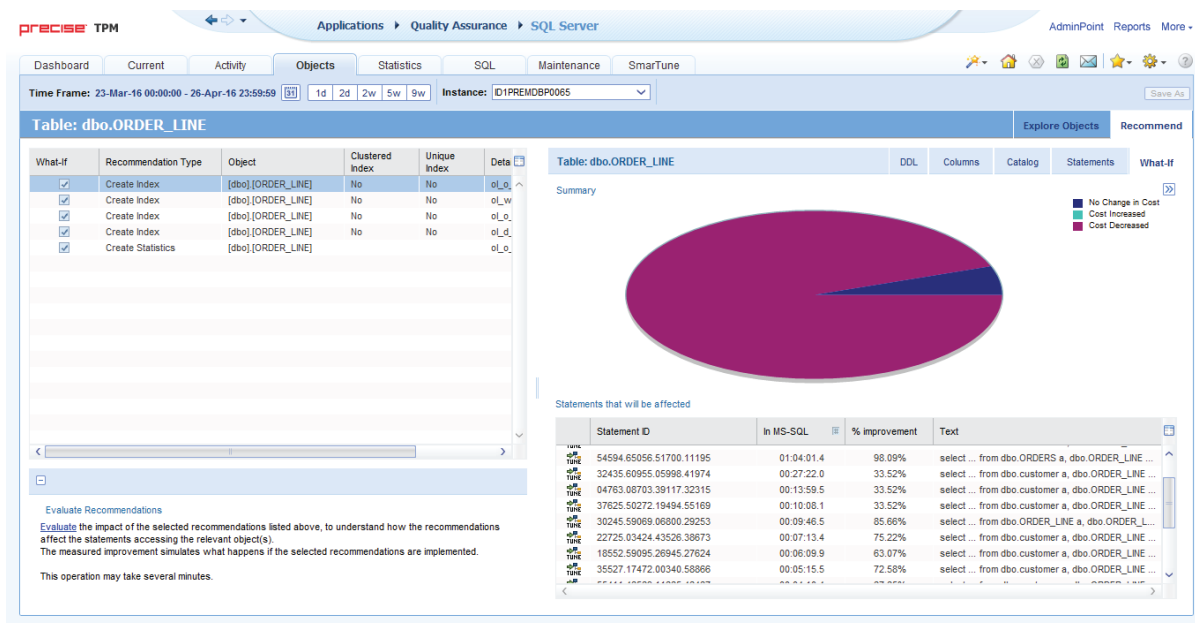**Figure 5:** The *'Explore Objects'* tab of the *'Objects'* workspace.

Figure 6: The *'Recommend'* tab of the *'Objects'* workspace.

5 DISPLAY ACTUAL EXECUTION PLANS

Precise for Databases readily identifies the most resource-consuming SQL statements to display a detailed explanation of their execution plan. The explain process can use actual plans when available, instead of estimated plans, for SQL statements that executed against monitored databases instances.

To view the actual execution plan of a SQL statement and understand how the SQL optimizer chooses to perform the SQL statement, navigate to the 'SQL' workspace. The 'Plan' tab (Figure 7) displays the actual execution plan of a SQL statement and various related information (such as statistics, referenced objects, and operations performed in the actual execution plan). It also displays a list of findings for the selected SQL statement. The 'Plan' tab displays two areas. The left area displays the 'Execution Plans' tree. The right area displays the 'Details' area with 'Highlights', 'Objects', 'Statistics', 'Operations', and 'Properties Information'.

In the left pane with the 'Execution Plans' tree, each branch in the actual execution plan represents a single operation. The actual execution plan displays the operations in order of their execution from top to bottom and left to right. Each operation is displayed using the format '[<execution order>] <operation type> (<options>) <accessed object> [<cost in percentage>]'. Precise marks operations whose estimated cost percentage is greater than a predefined threshold as critical with a red icon. The red icons enable quickly locating the most resource-consuming operation in the actual execution plan.

Precise displays the formatted text of the actual execution plan below the 'Execution Plans' tree. It highlights the text that relates to the selected step in the 'Execution Plans' tree to view the text of the SQL statement, the actual execution plan and additional information (such as the objects referenced by the SQL statement) all at once.
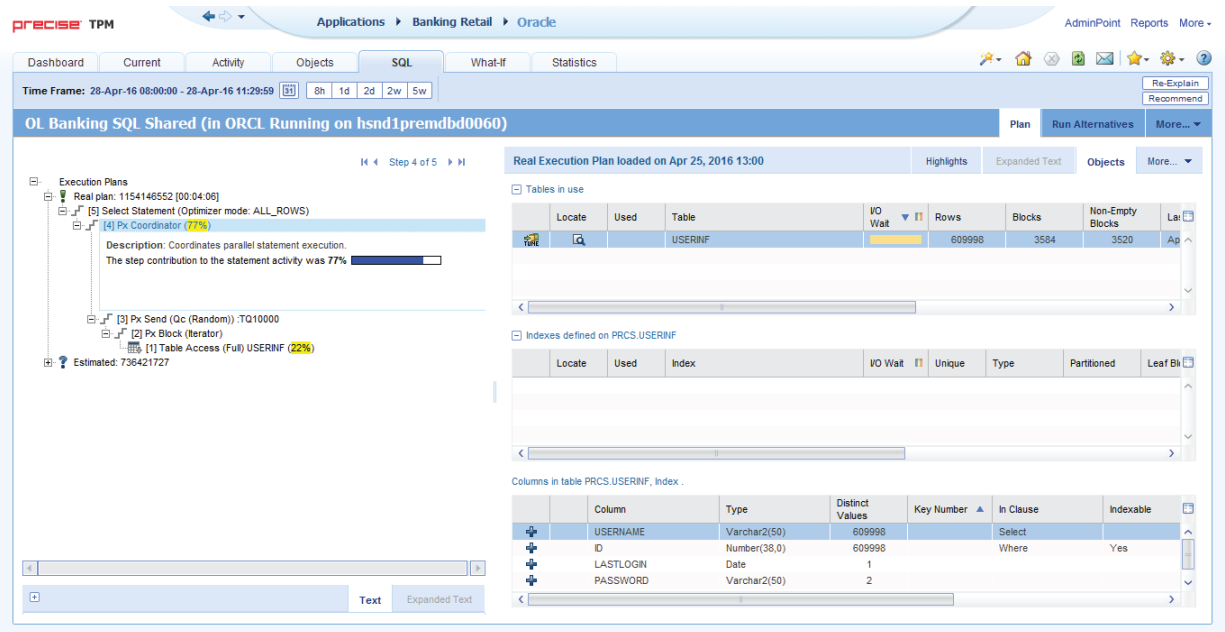
Figure 7: The *'Plan'* tab of the *'SQL'* workspace.

6 VIEW RECOMMENDATIONS

Precise for Databases displays recommended indexes or statistics for the selected table in the 'Recommend' tab (Figure 8). The indexes and statistics help the SQL optimizer choose a better access plan and enhances the performance of the SQL statements or batches that access the selected table. Precise bases the recommendations on a workload. The workload contains all the SQL statements stored in the repository that the database executed during the selected period with an average duration time exceeding the value defined in the registry.
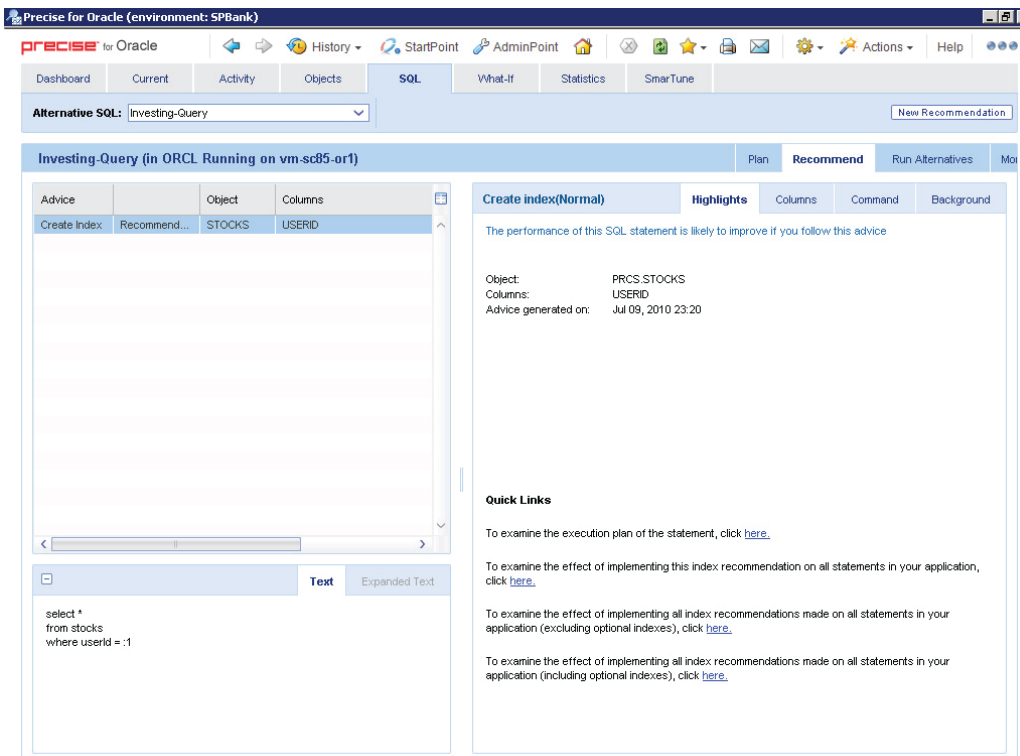
7 PERFORM WHAT-IF ANALYSIS

Precise for Databases can evaluate the effects of implementing tuning recommendations (such as adding indexes) on the performance of the database via the workload of the affected SQL statements. The What If modeling for indexes minimizes the risk of applying detrimental changes to production databases. It is easy to add an index. Many long-running production systems have too many indexes or the wrong indexes. Every index must be maintained during an update, delete, or insert operation. This maintenance takes resources. Instead, Precise pinpoints unused indexes and uses modeling to analyze the impact of dropping these indexes. Precise shows the impact on the entire workflow.

The 'What-If' workspace (Figure 9) shows the number of SQL statements whose performance would improve and the number of SQL statements whose performance would deteriorate by implementing the selected recommendation. It also shows the cost of implementing the recommendation to weigh the benefits and disadvantages of implementing the change.
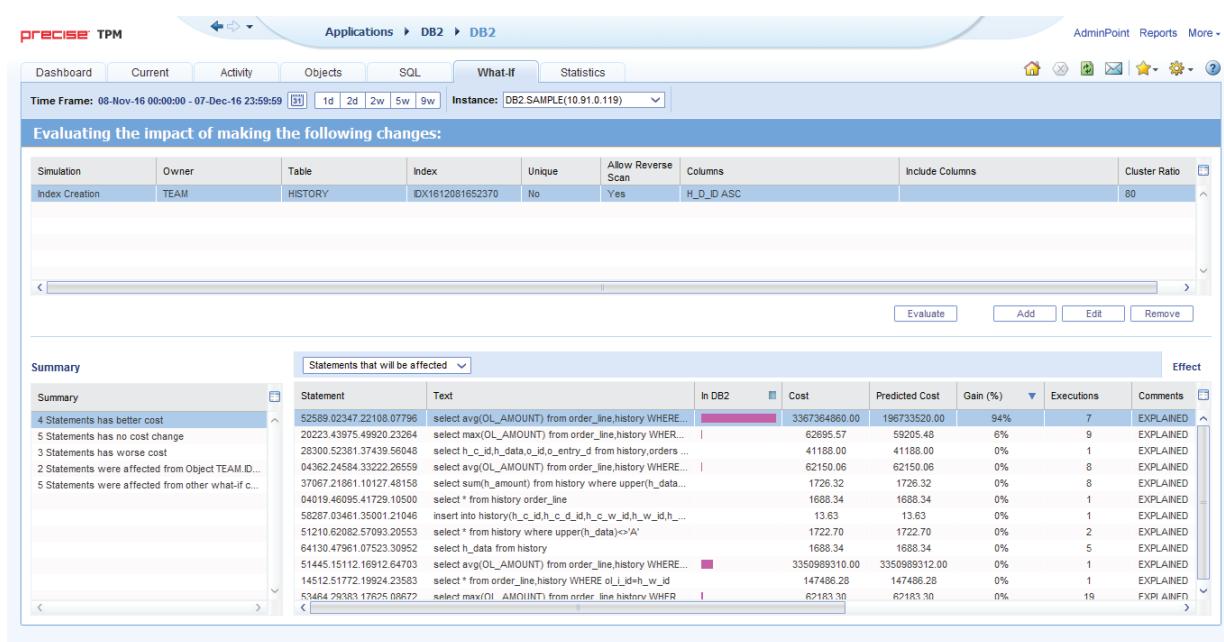

Figure 9: The *'What-If'* workspace.

The 'What-If' subtab at the right side of the 'Recommend' tab of the 'Objects' workspace (Figure 10) displays a pie chart with the distribution of the cost of all evaluated SQL statements and a data grid with the explained SQL statements that access the objects whose recommendations Precise is evaluating.
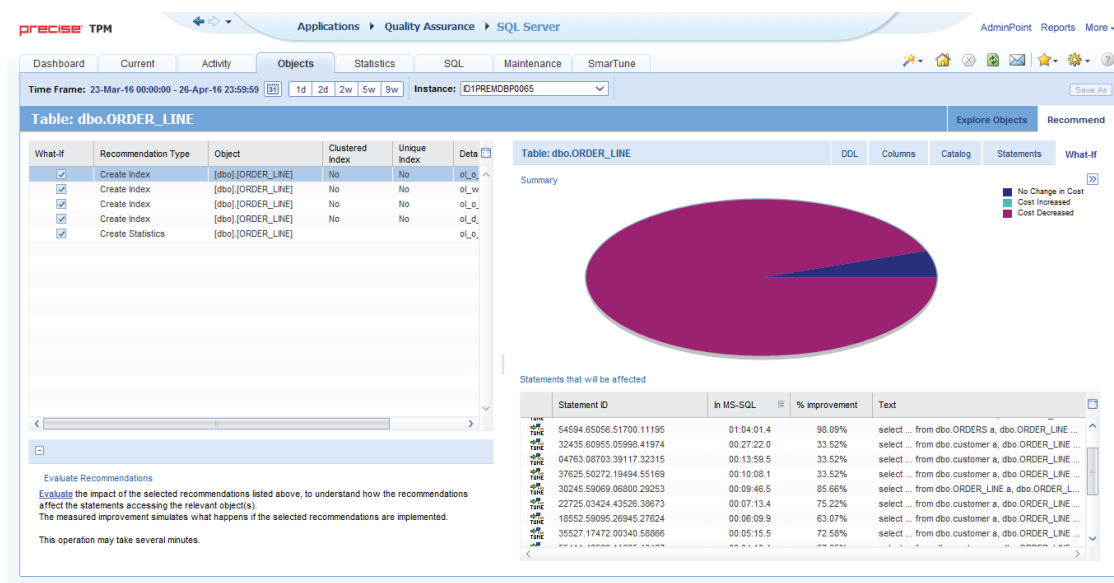
Figure 10: The *'What-If'* subtab of the *'Recommend'* tab of the *'Objects'* workspace.

8 TUNE PERFORMANCE SMARTLY

Precise for Databases provides the 'SmarTune' process that detects performance deterioration by regularly analyzing performance data for the database, and identifying less than optimal behavior. The 'SmarTune' workspace displays performance findings that may indicate potential problems in the database. The list of problems includes a list of SQL statements that require tuning, a list of objects displaying heavy activity or possessing an inefficient structure, and instance events. To start performance tuning on a database, start with the 'SmarTune' workspace to examine the 'SmarTune' findings, read expert knowledge regarding the area of deterioration, and obtain comprehensive advice on how to solve a specific problem and optimize performance. The 'Findings' area lists the problematic SQL statements, objects, and instance events that are causing poor performance. The 'Findings' area displays in the right pane the relevant information regarding each problem listed. The 'SmarTune' workspace includes the 'Quick Links' hyperlinks to launch to other tabs in context as part of the guidance provided to continue the investigation readily.

9 SETUP FLEXIBLE LICENSING

Precise for Databases provides flexible options for licensing. One option is to license Precise per database instance per year (that is, "eat as you go"). Another option is to license Precise for all instances in an organization for three years (that is, "all you can eat"). Both licensing options include professional services for installation, customization, and training assistance.

SUMMARY

Precise for Databases discovers why databases ran slowly during particular time intervals, and what resources databases used most of the time. Precise displays a list of SQL statements that may require tuning, a list of database objects experiencing a heavy activity or possessing an inefficient structure, and a list of events for databases. Precise archives performance data and provides detailed displays of historical data for performance analysis, quality assurance, and capacity planning.

PRECISE FOR DATABASES

ACCELERATE BUSINESS PERFORMANCE

- Database Performance Fuels Company Performance
- Multiple Platform Database Monitoring and Alerting
- Performance Management Database
- Root Cause Identification
- Tuning Recommendations
- What-if Analysis
- Capacity Planning

REQUEST TRIAL