Precise for Oracle™

User Guide

Version 10.0.x

IDERA

Precise™ for Oracle User Guide

Global Headquarters
Brookhollow Central III
2950 North Loop Freeway West, Suite 700
Houston, Texas 77092
Phone: +1-877-693-1886
Fax: +1-650-898-1666

Third-party legal notices

Third-party software may be recommended, distributed, embedded, or bundled with this Precise product. Such third-party software is licensed separately by its copyright holder. All third-party copyrights associated with this product are listed in the accompanying release notes.

Customer Support

For technical assistance, visit our customer portal at http://www.idera.com/support/productsupport where you can find an extensive knowledge base, product updates and our online community forums. You can also contact our Customer Support Team using our customer portal, or go to http://www.idera.com/support/productsupport for a list of our support access numbers in your country.

Chapter 3   Getting an overview of your application

Chapter 4   Getting current application information

Chapter 5   Examining performance over time

Chapter 6   Examining the relationship between database objects

Chapter 7   Storage Device entities

Chapter 8   Evaluating explain plans

Chapter 9   Simulating the effect of changes

Chapter 10 Examining Oracle instance statistics

Chapter 11   Examining concurrent managers and requests

Chapter 12   Oracle Findings

Glossary

Introducing Precise for Oracle

This section includes the following topics:

About Precise for Oracle

Precise for Oracle lets you manage the performance of your applications. It provides you with the tools necessary for proactive monitoring, diagnosing, detecting, and correcting of performance problems before they become visible in your Oracle databases and related applications.

Precise for Oracle addresses the needs of Database Administrators (DBAs), system administrators, managers, quality assurance specialists, and planners. It thereby provides a complete view of application performance by capturing, measuring, and correlating performance metrics from all critical system components.

Precise for Oracle offers a wide range of data presentations to enable quick and accurate monitoring and performance analysis. Current operational data is displayed to enable easy identification of problematic conditions. Recent operational data, for designated time periods, supplements operational data to enable operations staff to detect problematic conditions that occurred in the past and that are likely to occur again and again, if not corrected.

Performance data is archived and a fully detailed display of the historical data is provided for use in performance, QA, and capacity planning analysis.

Additionally if the DBA sees that a business transaction is performing badly, he can focus the tuning efforts on it. Precise for Oracle will give specific findings and recommendations to help him improve the transaction statement performance and the business as a whole. To that end, Precise for Oracle now has a possibility to show application and transaction context as part of the tuning efforts.

Improved performance management

IT staff frequently search for answers to questions like "Why did the application run so slowly last night?" or "Where is the application spending its time?"

Precise for Oracle can help IT staff find the answer to these questions. The first step is to establish a baseline level of database resource consumption by understanding the behavior of the average application. The behavior of the average application represents the average percentage of the total time that all applications spend consuming or waiting for each resource. These aggregate figures provide a broad picture of the operation of your database. They enable you to identify major bottlenecks in your system, for example CPU and I/O limitations, Oracle database waits due to locks, remote queries, and so on.

Precise for Oracle also lets you focus on and drill down to individual applications, SQL statements, and database objects in your system that are responsible for heavy consumption of system and database resources and to quickly find the definitive root cause of performance degradation.

In a multi-Tiered application, you can extend the reach of Precise for Oracle by installing Precise for Database & Storage, which drills down into the storage layer behind the database Tier and the optional Interpoint extensions to see into the Tier in front of the database.

Integration with other Precise products

Precise for Oracle is fully integrated with other Precise products. You can receive alerts using Alerts, examine Oracle reports within Report Manager, and view Oracle data in Insight and the Precise Custom Portal. You can launch Precise for Oracle from Insight, Alerts, Report Manager, Precise for SAP, and Precise for J2EE. For more information on the individual components, see the user's guide of the relevant product.

Alerts provides a notification system that alerts you to application response time problems and other fault conditions, and lets you launch Precise for Oracle in context to view more detailed data.

Report Manager gathers and organizes historical information that enables the IT team to focus on infrastructure hot-spots. You can use Report Manager to compare period-to-period performance against a baseline and identify response time problems "at-a-glance", before they affect your users. It queries the PMDB and displays the results in reports that are generated on a scheduled basis or on demand.

If you are following the Precise tuning methodology, both Alerts and Report Manager enhance your ability to detect performance problems by answering the question "Do I have a performance problem?"

The Precise Custom Portal, is a lightweight, configurable portal application that provides access to various types of data from different applications. The Precise Custom Portal features a Web-based view. It is highly customizable and extensible, allowing you to build a dashboard for each user or each function within your organization. For example, you can build a dashboard for all the information that a system administrator or director of IT operations would need to constantly follow up on. This component is automatically installed as part of the Precise framework installation.

Insight gives you an end-to-end view of your applications performance, across all your Tiers. It helps you find the source of performance problems in multi-Tier applications by allowing you to compare the resource consumption of each Tier. In this way, you can answer the question "Where is my performance problem?". If Insight shows that the problem lies within Oracle, you can use Precise for Oracle to drill down further.

Components of Precise for Oracle

To understand the Precise for Oracle architecture, it is important to know the overall architecture of the Precise product suite. For more information, see the *Precise Installation Guide.*

Precise for Oracle employs client/server architecture to provide comprehensive 24 x 7 performance monitoring, vital for both short and long-term monitoring of your Oracle database.

The following figure illustrates the typical architecture of Precise for Oracle.

Figure 1-1 Precise for Oracle architecture
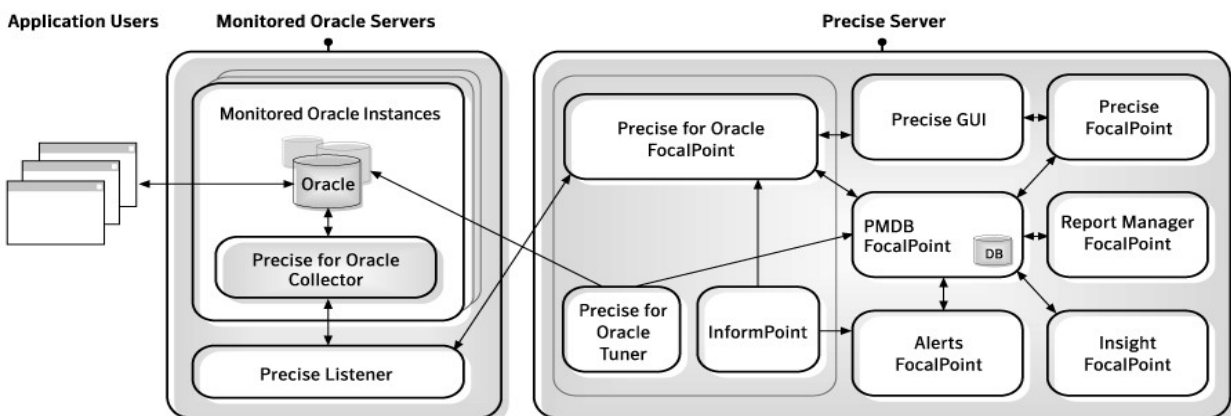
On the monitored server, Precise for Oracle consists of the following components:

- Precise Listener

- Precise for Oracle Collector

- Optional Collector extensions for Storage and ERP applications (not shown) On the Precise server, Precise for Oracle consists of the following components:

- Precise FocalPoint

- Precise for Oracle FocalPoint

- Precise for Oracle Tuner

- PMDB FocalPoint

- Precise user interface including Precise for Oracle

Other Precise components can integrate with Precise for Oracle. These components can also be present on the Precise server:

- Report Manager FocalPoint

- Alerts FocalPoint

- Alerts InformPoint

- Insight FocalPoint

See the Precise Installation Guide.

See the Precise Administration Guide.

Precise Listener

There is one Precise Listener per physical server or domain for each installation of Precise. Its purpose is to handle communication with other Precise components, using TCP/IP. It listens on port 20702 by default. The Precise Listener handles all requests from the user interface and Precise for Oracle FocalPoint; no requests go directly to the Precise for Oracle Collector. If necessary, the Precise Listener runs a Data Retriever process to carry out the work.

Precise for Oracle Collector

There is one Precise for Oracle Collector per monitored Oracle instance. The Collector runs in the background, monitoring and recording all Oracle database activity on a 24 x 7 basis. It works using a sampling technique and thereby consumes fewer resources than a trace agent. The collected information is stored locally in historical files, which are subsequently transferred and loaded into the PMDB.

Optional Collector extensions

Optional Collector extensions are available. They provide additional information in certain applications, namely for storage and for Enterprise Resource Planning (ERP). Precise for Database & Storage correlates Oracle database activity with the I/O on the back-end storage. ERP extensions correlate Oracle database activity with front-end ERP users and transactions.

Precise for Database & Storage is available for the following technologies:

- EMC Symmetrix and CLARiiON

- HP XP Array

If storage system support is installed, the Collector provides detailed information regarding the I/O resource consumption across the storage devices. By investigating this information, you can relate I/O activity to specific components of the storage device and pinpoint the cause of the I/O performance degradation.

Within Precise for Database & Storage, it is possible to browse device-specific mappings and statistics information. For more information, see the *Precise Installation Guide*.

ERP extensions (also known as Interpoints) are available for the following technologies:

- Oracle Applications

- SAP

- PeopleSoft

- Siebel

If ERP support is installed, the Collector provides detailed information on the activities and resource consumption of packaged application entities. Heavy users of Oracle resources, such as: ERP users, transactions, and reports can be identified. You can also drill down to the statements executed by them and the objects they accessed.

Precise FocalPoint

There is a single instance of the Precise FocalPoint for each installation of Precise. Precise FocalPoint is the management software for Precise. It manages and communicates with other FocalPoints, such as Precise for Oracle FocalPoint, PMDB FocalPoint, and the Precise Listener on monitored servers. Precise FocalPoint also contains the HTTP server used for the user interface.

Precise for Oracle FocalPoint

There is a single instance of the Precise for Oracle FocalPoint for each installation of Precise. It performs the following major tasks:

- It is responsible for replying to user interface requests. Depending on the nature of the request, it will retrieve data from the PMDB or from the Collector using the Precise Listener or run a Tuner process to query the Oracle database itself.

- It periodically loads Collector data into the PMDB. According to the schedule defined in the PMDB, Precise for Oracle FocalPoint executes the Precise Listener on each monitored server to copy across the prepared historical activity files. They are then loaded into the PMDB.

- It periodically loads Oracle statistics, explain-plan data, schema changes, and database definition changes. According to the schedule defined in the PMDB, Precise for Oracle FocalPoint runs a Tuner process to communicate with each monitored Oracle instance.

- When Alerts is installed, it provides Alerts with Oracle-related metrics.

Precise for Oracle Tuner

The Tuner process is only created when certain information from the monitored database is requested and lasts for the duration of the request. For example, if you want to view the explain plan of a SQL statement and see the definitions and statistics for the database objects involved, the Tuner is executed on the server to retrieve the relevant information from the Oracle Data Dictionary.

PMDB FocalPoint

There is a single instance of the PMDB for each installation of Precise. It maintains the Oracle PMDB, in which most of the performance information is held.

Precise user interface including Precise for Oracle

Precise provides a Web-based graphical user interface (GUI). Client requests are sent to Precise FocalPoint using HTTP. The Precise GUI is accessed through a browser. After you log in as an authorized user, the Precise StartPoint page opens. From the StartPoint, you can access the Precise for Oracle user interface and all other Precise products.

How Precise for Oracle works

Precise for Oracle stores a complete history of your database activity in the PMDB. This provides you -with a performance tracking tool that gives you the ability to assess database performance trends and analysis changes in database activities over long periods of time. This data is vital for identifying patterns in resource consumption and predicting future resource consumption in your Oracle database.

The Precise for Oracle Collector runs on your server using a unique, accurate and comprehensive sampling technology that requires low overhead. It collects performance data from an instance and then correlates and loads the data into the PMDB tables. This data is then analyzed and summarized by the FocalPoint. You can later access this data using the Precise for Oracle Web browser interface.

This interface lets you submit queries in a flexible intuitive manner, and then displays responses to your queries in a format that easily permits you to detect performance problems, drill down to the source of these problems, and subsequently tune your applications and database. The user interface is divided into a number of tabs to enable quick, accurate, and focused monitoring and performance analysis.

See About Precise for Oracle tabs.

The interface and tabs support the Precise tuning model of Detect, Find, Focus, Improve, and Verify. The following example shows how you might follow the tuning model in some of the tabs:

- The Dashboard tab, with its at-a-glance view of all Oracle instances side-by-side, lets you detect which is the most problematic instance. You can drill down, in context, to find and focus on the problem in the Current, Activity, or Statistics tabs. Automatic findings will guide you to the problems which, in turn, will take you to the next step of the problem solving action.
  The Dashboard tab also lets you compare performance and statistics among your Oracle instances.

- The Current tab gives a near real-time picture of activity in your instance and lets you easily find problematic conditions that are occurring now.

- The Activity tab lets you find and focus on the top resource consumers within your application over time. It also lets you identify patterns in resource consumption and predict future resource consumption in your Oracle database, such as:

  o When comparing business cycles

  o When comparing software releases

  o In capacity planning or trend analysis

- The Statistics tab lets you focus on instance-wide statistics that affect all applications.

- Having identified a performance problem, you can either improve the performance of your application, or use one of the automated tuning tools in Precise for Oracle, such as:

  o The SQL tab, where 'Alternative SQL' suggests more efficient SQL, and Advice recommends index changes.

  o The What-If tab, where you can simulate the effect of any proposed index change without physically creating the index.

- After implementing a performance improvement, return to the Activity tab and check if the change created the desired effect on your application or database. For example, return to the Activity tab to see if resource consumption was reduced.

What you can do with Precise for Oracle

Use Precise for Oracle to detect, diagnose, and resolve performance problems in your Oracle database. Precise for Oracle lets you identify real-time performance problems (for example locking problems) and monitor a complete history of your Oracle database.

Where to get more information

More information on Precise, its products, technical notes, and so on, can be found in the Release Notes document for this version.

Precise for Oracle basics

This section includes the following topics:

About Precise for Oracle tabs

The Precise for Oracle user interface is made up of seven or eight tabs depending on whether Oracle Applications is installed or not. Each tab has a different focus and provides a different view into your application and database performance. You can move between these tabs to examine the information necessary to successfully track your database performance and to identify patterns in resource consumption.

The following table describes which tasks can be performed in each tab.

Table 2-1 Tabs in Precise for Oracle

Tab	Description
Dashboard	Lets you quickly identify which Oracle instance has a performance problem, providing the first glimpse into where the problem lies. The Dashboard tab also lets you compare performance and statistics among your Oracle instances.
Current	Lets you identify and solve the performance problems that are happening now.
Activity	Lets you examine the application's performance over time and helps locate performance bottlenecks in its behavior.
Objects	Lets you browse Oracle database objects. It can help you understand the relationships and associations between database schema objects and the SQL statements that access them. It is primarily used for object tuning.
SQL	Lets you tune problematic SQL statements.
What-If	Lets you simulate the effect of any index change on SQL statements running in your application.
Statistics	Lets you examine Oracle's own statistics and see graphs of their variation over time. It is primarily used for instance tuning.

Table 2-1 Tabs in Precise for Oracle

Tab	Description
Oracle Applications	Lets you identify problems in Oracle Applications concurrent managers and requests. Note that this tab is only available if you have Interpoint for Oracle Applications installed.

How most tabs are structured

Though each tab is structured differently, most tabs consist of two different areas. Each area can include different control elements, such as tabs and view controls, and displays information in various formats, such as tables, graphs, or charts. The various areas are related to each other in that performing an action in one area affects the information displayed in other areas on the page. For example, the Current tab contains an upper and a lower area. The lower area (Association area) shows information in a table format. Each row in the table represents an application. The upper area (Main area) displays general information on the selected entity as well as related applications.

The entities displayed in the Association area are associated with the selected entity or related application displayed in the Main area. At times, the relationship between the entity displayed in the Main area and those displayed in the Association area is that of parent to child, and sometimes it merely represents that there is a relationship between the selected entity and the entities displayed in the Association area (such as when a statement is the selected entity displayed in the Main area and the list of programs associated with the selected statement is displayed in the Association area).

It is possible to click a row and drill down to an additional level below the selected level or use the filter feature to determine which information is displayed in the table. When you perform an action on an element in the lower area (in this case the Association area), the information displayed in the upper area (in this case the Main area) changes to reflect the action you requested.

You can control the tab by doing any of the following:

- Keeping the selected entity but changing the associated entities—by selecting a different entity from the Association controls in the Association area.

- Changing the selected entity—by selecting a row in the Association area.

- Changing the view for the selected entity—by selecting a different view in the Main area.

For example, when you want to view information on a specific application in the Current tab, choose Active Applications from the Association controls. The Association area changes to display applications-related information. The Tab heading and the Main area remain unchanged.

In the Association area, click the row of the application you want to view detailed information for. The Tab heading indicates the newly selected entity; the Main area displays information on the application you drilled down to, and the Association area shows information on statements executed by this application.

Figure 2-1 How a typical tab is structured
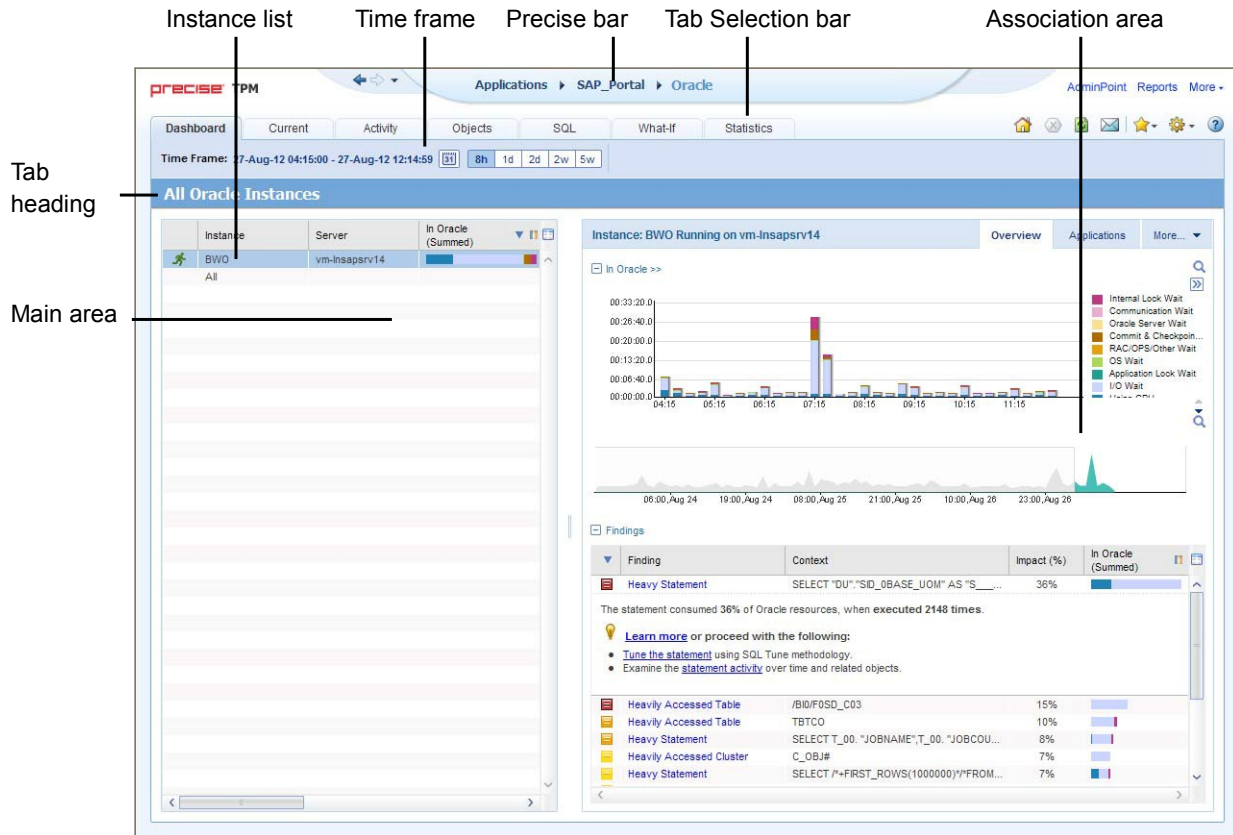
About the Precise Menu Area

The Precise Menu area is the area at the top part of each Precise screen. It contains menus, icons, and breadcrumbs. The Precise Menu area enables you to keep track of where you have been and provides various controls.

The following table describes the function of each of the toolbar buttons.

Table 2-2 Precise bar functions

Icon	Name	Description
	Back	During a work session, keeps track of where you have navigated to. The Back button enables you to navigate between previously visited views. The Back control displays your previous view.
	Forward	Enables you to navigate to the next view. This button is only enabled if you clicked Back or if you chose a history option.
AdminPoint	AdminPoint	Launches Precise AdminPoint.
	Home	Navigates to the highest level entity, usually the instance or Tier (all instances). The time frame settings remain the same.

Table 2-2 Precise bar functions

Icon	Name	Description
	Stop	Stops a request for information from the server.
	Refresh	Updates the data currently displayed.
	Favorites	Enables you to add or remove favorites in your Favorites list.
	Send	Opens a new email message in your email program with the link to the current application in context.
	Settings	Opens the Findings Settings, General Settings, and Time Frame Settings dialog boxes.
	Help	Opens the online help in context.

About the Main area

The Main area displays general information on the selected entity. The structure of this area depends on the selected entity and tab.

Information can be displayed as follows:

- As two over-time graphs that are displayed side-by-side and describe two data series.
- As a pie chart and a table that are displayed side-by-side. The table displays more details regarding a specific slice in the pie chart.

The View Control drop-down menu allows you to choose different view options. You can access this menu by clicking the arrow icon.

In some tabs, you can filter the contents of the view by clicking the Filter button. See Filtering data.

About the Association area

The Association area lists entities that have similar attributes and that are related to the selected entity named in the Tab heading. Besides the Tab bar, the Association area is the main navigation tool. It enables you to drill down from one entity to related entities (usually of a different type).

The entities displayed in this area depend on the selected tab and the selected entity. The same entity may be displayed in several tabs, with different data displayed for each one (for example, an Oracle table will be displayed in the Objects tab with data from the Oracle data dictionary, and in the Activity tab with data sampled by the Collector).

The Association control, accessed by the arrow icon, allows you to associate with different entities. The last option, More..., opens a dialog box that lets you carry out additional operations, such as to view additional associations, change the sort order, or control the number of returned rows.

The information displayed in the Association area is arranged in a tabular format. Each row in the table represents an entity, and each column describes an attribute of the entity, such as text or a graph. It is possible to change the sort order of the entities in the table by clicking a column title.

The Association tabs located above the table let you display different sets of attributes for the entities displayed in the table. The Association tabs are entity-dependent. For some entities, no tabs are displayed.

A numeric column can optionally display a graph, and at times, it can display a stacked bar graph as well. Switching to a graph view in these columns offers some level of breakdown to the information displayed in the column.

The following icons are used to indicate which format information is displayed.

Displays the information in numerical format.

Displays the information in a bar graph.

Displays the information as a stacked bar graph.

Displays the information in percentage format

The Association area lets you drill down to another entity by clicking a row. The information displayed in both the Main area and the Association area will change to reflect your selection.

See Filtering data and Associating entities with data that meets specific criteria.

About the Findings area

For selected tabs in Precise products, the association area includes the Findings area, displaying problematic findings for the application. The findings feature is a high level tool, designed to provide the user with an overview of performance issues within the monitored application and enable quick and efficient navigation to the relevant tab for further analysis and handling. The displayed performance findings may indicate performance deteriorations as well as incorrect methods usage.

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. When expanded, the finding details area provides important guidelines as to what may be the root cause for the reported problem, and what the recommended steps are to resolve this problem.

The table below describes the information that is displayed in the Findings area.

Table 2-3 Information displayed in the Findings area

Column	Description
Severity	The severity of the finding is calculated using a formula. The position of the finding in the list is determined by an internal scoring system that is based on the knowledge of Precise product experts. The severity is indicated by the following colors: ■ Red: High severity ■ Orange: Medium severity ■ Yellow: Low severity ■ Blue: No severity - the finding is strictly informative By default, findings are displayed according to severity.
Finding	A short name of the Finding.
Context	Entity/Method name (unless specified for the whole instance). The entity/method name is a short name but the long name is displayed in the ToolTip. Some of the findings are identified in specific Methods while others are relevant for the entire instance. In the latter case, a finding is specified as an instance-related finding. **Note:** Rows are aggregated by finding and the URI/method/SQL name and not by ID.
Finding overview	Displays specific details regarding the finding in context. **Note:** This is displayed in the expanded view only.
Learn more (advice)	Provides recommendations for solving the selected finding. For each finding, it lists all relevant pieces of advice and all applicable solutions. You should carefully review all data for the finding and then choose the advice that best suits your needs. **Note:** This is displayed in the expanded view only.
Proceed with the following (bullets)	Provides expert knowledge about the selected finding. The information displayed will direct you if you have difficulties deciding which advice to take or which solution to implement. **Note:** This is displayed in the expanded view only.

Investigating a finding

Perform the following steps to investigate a finding. To investigate a finding

1. In the Time Frame list, select the period of time you want to analyze.

2. In the All Oracle Instances table, select the instance you want to investigate.

3. In the Finding area, review the top Findings for the selected instance displayed in the Findings table. The findings displayed in this table are sorted by severity.

4. In the Findings table, select the finding you want to analyze further.

5. In the selected finding (the expanded view), read the data displayed for the finding and follow any links provided to view additional information (advice) or next steps (bullets) to resolve the problem.

For more information, see [Oracle Findings](#).

About drilling down in context

The term 'in-context' means that you can display additional information on a selected item by drilling down to another tab or view. The filter settings you defined (for example, the selected time frame you chose) and the entity you selected are carried over to the other view or tab, to allow you to continue analyzing your subject from a different perspective. This concept takes on slightly different meanings depending upon where you are attempting to drill down in context from.

For example, when viewing information on a database partition in the Dashboard tab, you can click a link in the Details area (right pane) to view additional information on the related tab, in context of your original selection.

Alternatively, when viewing a list of alerts for your product in Alerts, you can launch your product in context and continue investigating the factors that led the system to issue that alert.

In short, the information displayed when drilling down in context is always related to your original selection's settings.

How a drilldown affects the Tab heading

The Tab heading displays the name of the currently selected entity on this screen. When you drill down to a new entity in the Association area, the Tab heading changes to reflect the name of the newly selected entity.

About configuring your settings

The following settings allow you to control the appearance and behavior of the user interface. They can be configured from the Settings menu in the Precise Menu area:

*   General settings - Includes the General, SQL, Display, Tree view, and What-If/Recommend tabs.

*   Time Frame settings

*   Findings settings - Includes the Dashboard, Activity, Objects, and SQL tabs.

About configuring general settings

On the General tab of the General Settings dialog box, you can adjust the following:

*   Number of items that will be displayed for each association
    From the list, select the maximum number of items that will be displayed in a table. The default is 50 items. You can control the number of items returned for a specific association by using the More... option in the Association controls. This way, you can display up to 500 items.

*   Maximum number of characters to display in a text ToolTip
    When you move your cursor over text columns in a table listing Statements or PL/SQLs, a ToolTip appears, displaying more text than can normally fit into a table cell. This field controls the maximum number of characters that can be displayed. The default is 500 characters.

*   How often the information in the Current tab is refreshed
    If you select the Auto-refresh Current tab every check box, the Current tab automatically refreshes at the specified interval. By default, the tab is not refreshed automatically. If you select this option, the default refresh interval is every five minutes.

*   Display concurrent managers history for the last x days

*   Display concurrent managers request history for the last x hours

About SQL settings

On the SQL tab, you can adjust the following:

- When Precise for Oracle carries out an explain plan operation, it needs to connect to the database. By default, it uses the Oracle account specified during the installation. This option lets you override the database user and **password** that will be used.

- Precise for Oracle's own parsing functions can be disabled at the expense of some of the formatting features of the explain plan. This option is useful when Precise for Oracle cannot parse a statement. See About Precise for Oracle tabs.

You can fine-tune the way that Precise for Oracle generates alternatives to SQL in a number of ways:

- You can control the maximum number of alternative SQL statements.

- You can control the percentage of the items that are highlighted in the Oracle explain tree based on "In Oracle Time".

- You can have Precise for Oracle ignore alternatives that generate the same execution plan.

- When Precise for Oracle generates alternatives (referred to as Related SQL), any views are expanded to show the full view text, rather than just the name.

- If a Precise for Oracle view is expanded, Precise for Oracle may consider rewriting the text of the view itself, not just the statement that uses the view, when it generates alternatives.

- You can have Precise for Oracle use different Oracle Optimizer modes.

About configuring display settings

When you drill down to or choose a different entity, selecting the option on the Display tab of the General Settings dialog box allows you to maintain the selected view or tab when changing from one tab to another. This is sometimes referred to as sticky tabs.

About configuring Tree view settings

(Only applies to Precise products that include a Tree view)

On the Tree View tab of the General Settings dialog box, you can adjust the following:

- Display the Tree View as the default tab of the Activity tab.

- Sorting of items.

- Number of categories to display when expanding an item.

- Number of items to display when expanding a category.

About configuring What-If/Recommend settings

On the What-If/Recommend settings tab of the General Settings dialog box you can adjust the maximum number of SQL Statements shown in the What-If tab and the maximum duration of the What-If evaluation or Recommend process.

About configuring Findings settings

You can set the severity thresholds for the findings displayed in the Dashboard tab, Activity tab, SQL tab, and Objects tab, in the Findings dialog box. You determine whether a finding is to be displayed as a high, medium or low severity.

About configuring Time Frame settings

You can determine the resolution of the data that is displayed in the overtime graphs using the Time Frame Settings dialog box. By using this dialog box you can define the default time frame to display.

Tasks common to most tabs

The following tasks are commonly performed in most tabs:

- Switching to a different tab

- Selecting a time frame

- Selecting an entity

- Filtering data

- Associating entities with data that meets specific criteria

- Focusing on information in overtime graphs

- Sending an email message

- Adding, viewing, and deleting Favorites

- Determining which table columns to display

- Copying data to the clipboard

- Exporting to the Precise Custom Portal

Switching to a different tab

You can easily switch between the different tabs in the Precise Menu area. When you start your Precise product, the Dashboard tab opens by default. For other Precise products, another tab will open by default. The button of the selected tab is displayed in a darker blue.

To select a tab, click the required Tab to display information on the selected entity.

Selecting a time frame

You can configure Precise for Oracle to display transaction performance data for a specific time frame using the predefined time frame options or calendar icons.

Selecting a predefined time frame from the toolbar displays transaction performance data for the selected time period up to the current time. See Selecting a predefined time frame from the Precise for Oracle toolbar.

Selecting the time frame using the calendar icon, you can choose to define a time range independent of the current time, or to define a time range up to the current time. See Selecting a time frame using the calendar icon.

The predefined time frame options are:

- Last 8 hours (8h) (default)

- Last 1 day (1d)

- Last 2 days (2d)

- Last 2 weeks (2w)

- Last 5 weeks (5w)

The time frame selected affects all information displayed in Precise for Oracle. Only data that falls within the selected time frame is shown in these areas.

Selecting a predefined time frame from the Precise for Oracle toolbar

To select a predefined time frame, in the Precise Menu are, select one of the predefined time frames.

Selecting a time frame using the calendar icon

To select a time frame

1. Click the calendar icon. In the dialog box that is displayed perform one of the following:

   a. To define a time frame independent from the current time, select the 'Time Frame' option and select the Start and End dates and times.

   b. To define a time frame up to the current time, select the 'Last' option and enter the desired time frame.

   c. To use one of the three previously used time frames, select the 'Recently used' option and from the drop-down menu select the desired time frame.

   d. To use a previously saved time frame, select **Use a previously saved time frame** and from the drop-down menu select the desired time frame.

2. To save your settings for future access, select **Save these definitions for future use as:** and enter a name in the corresponding field.

3. Click **OK**.

Selecting an entity

You can select the entity that you want to examine in more detail. To select an entity, on the Main area of the tab, select the entity you want to examine.

Filtering data

You can focus on specific contents of the Dashboard and Activity workspaces by showing a subset of the information. This shows the contribution of entities such as programs and users.

You can define flexible selection criteria in the Filter dialog box. To open the dialog box, you need to click the Filter Off icon next to the Time Frame selector. Filter On or Filter Off indicates the current state of the filtering mechanism. Filter Off means all information is shown; Filter On means a subset of information is shown.

When you apply your selections in the Filter dialog box, the information displayed in both the Main area and the Association area is modified to reflect your selections. Also, the filtering continues to apply when you drill down to associated entities.

NOTE	You can enter multiple criteria, in which case each criterion is applied using the logical AND operator.

To filter data

1. Click Filter is Off/Filter is On.

2. In the Filter dialog box, do the following for each entity you want to filter:

From the left drop-down list	Select an entity.
From the middle drop-down list	Select an operator, such as, Like, <>, Not Like, In, Not In.
In the text box	Type the criteria (case-sensitive) for the selected entity.
	If you select the operator Like or Not Like, you can use the % wildcard character to represent 0 or more characters, and the _ wildcard character to represent exactly one character. If you select the operator In or Not In, type a comma to separate values.

3. Click **OK**.

Associating entities with data that meets specific criteria

You can associate the displayed entity with specific data to focus your analysis.

NOTE	The criteria no longer apply when you drill down to another entity.

To associate entities with data that meets specific criteria

1. Click the arrow located to the left of the Association controls and select **More...**

2. In the Associate With dialog box, on the Entries tab, select the entity you want to associate data with from the Populate table with list.

3. In the Sort entries by list, determine according to which criteria you want the information to be sorted and in which order.

4. From the Display top list, select the number of rows to display.

5. On the Criteria tab, do the following for each entity you want to associate data with:

From the left drop-down list	Select an entity.
From the middle drop-down list	Select an operator, such as, Like, <>, Not Like, In, Not In.
In the text box	Type the criteria (case-sensitive) for the selected entity.
	If you select the operator Like or Not Like, you can use the % wildcard character to represent 0 or more characters, and the _ wildcard character to represent exactly one character. If you select the operator In or Not In, type a comma to separate values.

6. Click **OK**.

Editing the properties of a statement

NOTE	This function is only available in Table View.

Any statement can be saved in the PMDB. You can change the name of a statement, place it in a different cabinet and folder, change the parsing user, and add a comment. But you cannot save a statement with the same combination of cabinet, folder, and name as an existing statement.

Saving a statement gives you the chance to keep important statements for future reference. You will probably want to rename it to something more meaningful, such as the name of the function it is trying to perform. This name is then used throughout the product instead of its numeric identifier, making it easier, for example, to pick it out of the list of statements in the Activity tab.

It is a good idea to keep a statement and its alternative in their own separate folder rather than leaving them in the default folder. The comment that you add is particularly useful for describing your attempts to tune a statement and the differences between an alternative and the original.

For example, consider a slow-running statement called 56549.16561.40035.65472, which performs a customer search in a sales application. You might choose to save it in a cabinet called Sales Application, a folder called Customer Search, and with a name such as Customer Search original version. You might add a comment such as This needs tuning. Later, when you generate an alternative, you might save it as Customer Search with new index and a comment such as Simulated index, cost reduced by 10.

Saving a statement is possible in the following tabs:

- Activity tab: only when the selected entity is a statement.

- SQL tab: in all views except Compare (because it is not obvious which of the two statements being compared should be saved).

To edit the properties of a statement

1. Click the Actions icon>Edit Properties.

2. In the Edit Statement Properties dialog box, choose the relevant properties for the statement from the drop-down lists and enter a statement name. The following properties are available:

Instance	Name of the instance that the statement belongs to. The instance cannot be changed.
Parsing User	Choose an Oracle user name that will parse the statement from the drop-down list
Cabinet	Choose a cabinet to save the statement in, from the drop-down list or type in a name.
Folder	Choose a folder to save the statement in, from the drop-down list or type in a name.
Statement	Type a user-defined name for the statement.
Comment	Type an optional tuning comment that is to be associated with the statement.

The maximum number of items displayed in the Cabinet, Folder, and Statement lists is limited. You may also type the names.

3. Click **OK.**

Focusing on information in overtime graphs

Some entities display an overtime graph. The overtime graph displays entity statistics over a specified time period. Depending on the number of points displayed in the graph, you may need to zoom in or out. The text displayed on the x-axis varies according to the time frame. If there is a year or day change, x-axis labels will display accordingly.

To focus on information in overtime graphs, select the desired time frame on the overtime graph, click and drag the left or right handle to increase or decrease the time range. The small zoom (spyglass) icon will display on the upper right of the selected time range, and above the overtime graph legend. Click the zoom icon to zoom in according to the selected time frame.

Sending an email message

You can send an email message to one or more recipients from the Precise toolbar. The default subject for the message will be "Link to a Precise application". The email will include a link to the Precise product in the current context (time frame and selected entries). To send an email message

1. Click the email icon in the Precise Menu area. The default email program opens.

2. Fill in the required fields and click **Send**.

Adding, viewing, and deleting Favorites

The Favorites feature allows you to save a specific location in your application and to retrieve the same location later without having to navigate to it.

About the Favorites feature

The new Favorites feature includes the following options:

- Relative Time Frame – Saving relative time frame instead of static date. For example, saving the last 7 days will always display the last 7 days, depending on the day entered.

- One click to specific location – Once you open Precise by launching a saved Favorite item, you will not have to enter a login credential nor click the login button.

- IE Favorites support – Adding a new Favorite item in Precise will also add it to the IE Favorites menu.

- Auto Complete – The Favorites dialog includes a new combo box which supports AutoComplete.

- Auto Naming – The Favorites dialog generates item names based on the current location.

UI description

A Favorites icon with a menu has been added to the Precise Menu area in each product.

An Add/Delete Favorites option under the Favorites icon with menu allows you to save the current location or delete an existing one.

To add a new Favorite location

1. In the Add/Delete Favorites dialog box, enter the name of the new Favorites entry.

2. Click **Add**. The dialog box is closed and the new Favorite is added to the list.

To view a Favorites location

1. In the Precise Menu area, click the **Favorites icon**.

2. Select the Favorites location you want to view.

To delete an existing Favorite location

1. On the Add/Delete Favorites dialog box, select the Favorite location to be deleted.

2. Click **Delete**. The dialog box closes and the selected Favorite is deleted from the list.

| NOTE | The favorite address is displayed in the Address field and cannot be edited. |

Determining which table columns to display

Tables are used to display information about a set of related entities in the Main and Association areas. It is possible to determine which columns to display in the Association area tables.

To determine which columns to display in the Association area

1. Click the Table icon on the upper right-hand side of a table and select **Column Chooser.**

2. In the Table columns dialog box, click the arrows to move the names of the columns that you want to display to the Visible box and the ones that you do not want to display to the Invisible box.

3. Click **OK**.

Copying data to the clipboard

At times you may want to save data displayed in the table area in a Microsoft Excel spreadsheet for further analysis or save an image of a graph to the clipboard.

To copy data displayed in the Association area to the clipboard, click the Table icon on the upper right-hand side of a table and select **Copy to clipboard**. The table can be pasted into Microsoft Excel or as an HTML file.

To copy a graph to the clipboard, right-click a graph and choose **Copy to clipboard**. You can now paste the image into any application that works with the clipboard.

Exporting to the Precise Custom Portal

The Export to the Precise Custom Portal Portlet feature enables you to export the view of the chosen table or graph and generate a portlet with that view in the Precise Custom Portal, so that it will provide you with another way of monitoring your application.

Prerequisites

To be able to use this feature, you need to have the following rights in Precise:

- View permissions to all Tiers in the application

If you do not have sufficient rights, you will get an error message when trying to execute this feature.

Exporting the information

You can either export a table view or a graph view.

NOTE	The name field has the following restrictions: maximum 100 characters.

To export a table view

1. Click the Column Chooser icon.

2. Select Export to the Precise Custom Portal Portlet.

3. Insert a name in the name field that clearly describes the table view.

4. Click **OK**.

To export a graph view

1. Right-click the graph.

2. Select Export to the Precise Custom Portal Portlet.

3. Insert a name in the name field that clearly describes the graph view.

4. Click **OK**.

About role management in Precise for Oracle

The Roles management feature in Precise for Oracle lets you assign roles to users and control whether they will be able to do the following operations:

- Bind—view bind sets.

- Run alternative—run an alternative statement.

- Gen_Estimated_Plan—determine whether viewing existing or real plans collected from Oracle will always be allowed.

- Gen_Massive_Plan—explain unexplained statements in the Object tuner (depends on what is configured for Gen_Estimated_Plan).

- Whatif—lets you simulate the effect of creating, modifying, or dropping a B*Tree or bitmap index, on the performance of all statements that Precise for Oracle has collected, without having to physically make the change (depends on what is configured for Gen_Estimated_Plan).

It is possible to grant permissions to all roles or to revoke permission from individual roles. Permission is a combination of permission type (such as: INSTALL, ADMIN, MONITOR) and operation (such as: VIEW, UPDATE).

Certain types of permission enforce permission dependencies. For example, if you grant a TUNE.RECOMMEND permission it will automatically grant TUNE.EXPLAIN. In the same manner, if you revoke TUNE.EXPLAIN then the TUNE.RECOMMEND permission is also revoked.

You can configure permissions and roles in AdminPoint. For more information on configuring permissions and roles see the *Precise Administration Guide*.

How information is collected

Precise for Oracle collects and displays performance data that enables quick and accurate performance monitoring and analysis. To understand the information displayed in the various graphs, tables, and views, it is important to understand how information is collected.

Precise for Oracle Collector sampling methods

The Precise for Oracle Collector is a non-Oracle process that can collect data from both the server's operating system and the Oracle instance. This is important because both the server's operating system and the Oracle instance contain data necessary for the solution of a performance problem, but neither is aware of the other's internal state.

For example, the Oracle instance may have data indicating that a specific process is waiting in the operating system, but it does not have data indicating why. Only the servers operating system has data indicating that the process is waiting for CPU, memory, and so on.

A product that can support all phases of tuning from the entire system down to individual applications and objects must meet the following criteria:

- It must be able to collect data from both the server's operating system and the Oracle database system.

- It must be fully independent of the Oracle system that it monitors to provide a true picture of Oracle activity.

- It must be able to achieve high sampling rates.

Precise for Oracle meets all of these criteria.

Non-intrusive monitoring of the Oracle instance

Most monitors collect data on user activity and system status intrusively by issuing queries to the Oracle X$ and V$ dynamic performance views. These queries consume Oracle resources and tend to be neglected by Oracle when system load is heavy and collecting data on Oracle is most important.

Precise for Oracle's Collector gathers data non-intrusively by sampling data in the System Global Area (SGA) of the Oracle instance. This makes monitoring the Oracle instance independent of the Oracle system.

How Precise for Oracle monitors availability

The Precise for Oracle Collector monitors the Oracle instance. If the instance status (Available/Unavailable) has changed, a message code is sent by the Collector and causes the current availability of the instance saved in the PMDB to be changed. In addition, the availability of the instance is saved, over time, in the PMDB. During a planned downtime period, the instance availability is shown as it was available during this period.

NOTE	If the Precise for Oracle Collector is down, the instance availability status reported is as Unknown.

High sampling rates

The Collector can be configured to sample data using a high sampling rate. This enables the Collector to collect data on entities that are executed for only very short periods of time.

High resource consumption can be the result of an entity (such as a program or statement) with heavy resource demands running for an extended period of time (long duration, low frequency). It can also be the result of an entity with modest resource demands running often for a short period of time (short duration, high frequency). Identifying the latter entity as the cause of a performance problem can be difficult if the Oracle instance is sampled using a low sampling rate. If execution duration is short enough, complete executions of the problematic entity can take place between samples, and data on these executions will not be included in the overall picture of system performance.

Wait States

Fundamental to the operation of Precise for Oracle is the capturing of wait state information to identify resource consumption. The Precise for Oracle Collector, using its unique collection technology, samples every Oracle session and SQL statement, and at each sample point determines the Oracle wait event. It also correlates the session information with the corresponding operating system process information to capture additional wait states. In fact, it employs a two-Tier categorization: wait state and wait sub-state. It defines the status of sessions currently connected to Oracle by these states. By aggregating this wait state information over time, Precise for Oracle can give an accurate time-based, resource consumption profile for every statement, table, session, program, and so on, up to the instance level.

This wait state information forms the basis of the resource consumption graphs, particularly in the Dashboard, Current, and Activity tabs.

Precise for Oracle displays the following resource consumption graphs:

- Overall activity
- In Oracle
- I/O wait

Overall activity

The Overall Activity resource consumption graphs give the total time that a session or a set of sessions is connected to Oracle. During that time, the session may be actively executing statements within Oracle, or it may be idle. Overall Activity can help answer the question: Is the application spending its time in Oracle or elsewhere?

Overall Activity is divided into the following states:

- Request Wait
  The session is idle. The underlying Oracle wait state is typically an SQL*Net message from the client or pipe_get, and the client is performance-related and not Oracle-related activity.

- In Oracle (Sum.)
  The total time sessions spent actively executing statements and PL/SQL within Oracle. This is further broken down into In

- Oracle states in the In Oracle resource consumption graphs.

- Parallel Query Sync
  The time that a parallel query coordinator and parallel query slave sessions spent waiting for other slaves to complete an operation. It is not shown as time spent actively executing In Oracle because the parallel query sessions are effectively idle during this time.

- Miscellaneous
  The time not accounted for by one of the other states. It is usually due to session initialization or termination. For example, a dead session not yet cleaned up by Oracle PMON may have some time in this state.

In Oracle

The In Oracle resource consumption graphs give a breakdown of how Oracle is spending its time executing statements and PL/SQL. The wait states with the longest duration indicate the bottlenecks within Oracle. Reducing the duration of these wait states will improve the response time or alternatively increase the throughput.

Precise for Oracle employs a two-level hierarchy of In Oracle wait states. Each In Oracle state has one or more In Oracle sub-states. In turn, sub-states have one or more Oracle or operating system wait states. States and sub-states relate to each other, as described in the following table. As a consequence, Precise for Oracle displays two types of In Oracle resource consumption graphs: either In Oracle state or In Oracle sub-state.

The following table lists the existing In Oracle states.

Table 2-4 In Oracle states

In Oracle State	Icon	In Oracle Sub-State	Description	Typical Oracle Events
Using CPU	CPU	Using CPU	The session is using CPU executing SQL.	N/A
I/O Wait	I/O	Direct I/O Wait	The session is waiting on direct I/O.	direct path read; direct path write
I/O Wait	I/O	Scattered I/O Wait	The session is waiting on scattered I/O.	db file scattered read
I/O Wait	I/O	Sequential I/O Wait	The session is waiting on sequential I/O.	db file sequential read; control file sequential read
I/O Wait	I/O	Other I/O Wait	The session is waiting on another type of I/O.	db file parallel write; log file parallel write; control file parallel write; asynch disk I/O
I/O Wait	I/O		**Note:** The I/O sub-states Direct I/O Wait, Scattered I/O Wait, Sequential I/O Wait, and Other I/O Wait only apply to Oracle files, objects, and storage entities. For other entities, the sub-state is simply shown as I/O Wait.	
Application Lock Wait		Row Lock Wait	The session is waiting on row locks.	Enq: TX - row lock contention (Oracle 10g)
Application Lock Wait		Table Lock Wait	The session is waiting on table locks.	Enq: TM - contention (Oracle 10g)
Internal Lock Wait		Other Lock Wait	The session is waiting on other Oracle locks, such as streams, latches, or internal locks.	STREAMS events; enqueue; other enq: events; latch activity; latch free; latch: events

Table 2-4   In Oracle states

In Oracle State	Icon	In Oracle Sub-State	Description	Typical Oracle Events
Application Lock Wait			**Note:** Enqueue may appear in Row Lock Wait and Table Lock Wait, depending on the values of the P1, P2, and P3 parameters. The Collector captures these parameters and assigns the wait state accordingly. This can be seen in the Current and Activity tabs. However, the Collect Instance Statistics PMDB process does not capture the parameters so that in the Statistics tab, the wait is always shown as Other Lock Wait.	
Operating System Waits		CPU Wait	The session is on the operating system run queue waiting for CPU.	N/A
Operating System Waits		Memory Wait	The session is waiting on a memory operation, such as page-in or page-out. This sub-state may also include memory-mapped I/O.	N/A
Operating System Waits		Other Host Wait	The session is waiting on another type of OS operation, such as sorting.	N/A
RAC/OPS/Other Waits		RAC/OPS Wait	The session is waiting on RAC or OPS synchronization.	DFS events; DLM events; gc events; gcs events; ges events; global events;
RAC/OPS/Other Waits		Other Wait	The session is waiting on another, less common lock event.	N/A
Commit & Checkpoint Waits		Rollback Segment Wait	The session is waiting on a rollback segment operation.	alter rbs offline; undo segment events
Commit & Checkpoint Waits		Redo Log Buffer Wait	The session is waiting on a redo log buffer operation.	log buffer space; log file syn
Commit & Checkpoint Waits		Log Switch and Clear Wait	The session is waiting for a log switch and clear operation to occur.	checkpoint events; log file switch events
Oracle Server Wait		Parallel Query Server Wait	The session is waiting for a parallel query server to become available.	PX create server; PX server shutdown; parallel query; create server; parallel query; dequeue wait
Oracle Server Wait		Parallel Query Sync. Wait	The session is waiting for a parallel query server to get synchronized with another parallel query server.	Other PX events; other parallel query events

Table 2-4 In Oracle states

In Oracle State	Icon	In Oracle Sub-State	Description	Typical Oracle Events
Oracle Server Wait	RES MGR	Resource Mgr Wait	The session is waiting for a resource to become available (only if the database resource manager is enabled).	resmgr events
Oracle Server Wait	MTS [	Multi-Threaded Server Wait	The session is waiting for a multi-threaded server to become available.	alter system set dispatcher; alter system set mts_dispatcher
Oracle Server Wait		Background Process Wait	The session is waiting for an Oracle background process, such as DBWR or LGWR.	LGWR events
Client Request Wait		Request wait	Request wait is generally considered as an idle event as the Oracle server process is waiting for the client process.	Client messages
Communication Wait		Oracle/Oracle Communication Wait	The session is waiting for another Oracle instance, usually as a result of a DBLINK event or a standby database.	SQL*Net break/reset to DBLINK; SQL*Net message from DBLINK; SQL*Net message to DBLINK; SQL*Net more data from DBLINK; SQL*Net more data to DBLINK
Communication Wait		Oracle/Client Communication Wait	The session is waiting for data sent to the client, usually as a result of a selected statement, or for data sent from a client, usually as a result of bind variables.	SQL*Net break/reset to client; SQL*Net message to client; SQL*Net more data from client; SQL*Net more data to client
Internal Lock Wait		Shared Pool Wait	The session is waiting for a shared pool operation.	library cache load lock; library cache lock; library cache pin
Internal Lock Wait		Buffer Wait	The session is waiting for a buffer cache operation.	buffer busy; buffer busy wait; buffer latch

I/O wait

I/O Wait resource consumption graphs are shown for Oracle files, objects, and storage entities. They give a breakdown of the following I/O wait into:

- Direct I/O Wait
- Scattered I/O Wait
- Sequential I/O Wait
- Other I/O Wait

Modified In Oracle graphs are shown for Oracle objects such as tables and indexes. In addition to the listed I/O sub-states, Precise for Oracle records the following sub-states for an object:

- Buffer Wait
- Table Lock Wait
- Row Lock Wait

The other In Oracle wait states, such as Using CPU, do not apply to an object. See In Oracle.

Statement identifier vs. Oracle hash value

SQL statements in Precise for Oracle are assigned a numeric identifier (a 20-digit number that is divided into 4 sections, with dot separators) or a user-defined name. The numeric identifier is similar to, but not the same as, the hash value that Oracle assigns to a SQL statement. When Precise for Oracle calculates the numeric identifier, it ignores white space and case, unlike Oracle. It also takes into account whether Collapse Statements is switched on, in which case string literals are replaced with place holders or pseudo-bind variables.

Executions vs. opens and fetches

Precise for Oracle employs the following methods to determine how many times a statement has been executed.

- At each sample point, the Precise for Oracle Collector examines the equivalent area of the SGA that V$SQL maps on to and captures the number of executions of the statement that Oracle has recorded. (It also captures buffer gets and rows processed.

NOTE	Precise for Oracle only increments the counters when the statement has finished executing similar to V$SQL.

- At each sample point, the Precise for Oracle Collector examines whether the session is opening or fetching from a cursor, that is executing a statement. Oracle does not provide enough information to allow the Collector to determine whether the session is opening (executing) the statement or fetching results, and therefore it counts both. This potential over-reading of the actual execution count may be offset by the fact that any sampling technology may miss some executions, particularly for fast-running statements.

Where possible, Precise for Oracle displays the number of executions that Oracle has recorded because it is more accurate. However, V$SQL only contains statistics about the statement; it does not tell you which session ran it. Therefore, it is not possible to accurately divide the number of executions between the sessions. This means that in the user interface, wherever you drill down in the context of a session (and this includes program and user), you will see opens and fetches. Where you drill down directly from the instance level to a statement, you will see executions. The Current tab, for example, displays opens and fetches because you always view a statement in the context of a session, whereas the Activity tab may display either executions or opens and fetches, depending on your navigation path. Other derived values, such as average response time, are only shown if the number of executions is available.

In Oracle time vs. Duration time

In Oracle time indicates the period that the entity consumed within the monitored box, not including IDLE time (sleep, waiting for user requests, etc.).

Duration time is the "wall clock time", including idle and non-idle consumption periods. The following calculations are used:

Table 2-5 Calculations

Calculations	Description
In Oracle time (Avg.)	In Oracle time divided by the amount of executions.
Duration time (Avg.)	Duration time divided by the amount of executions or sessions (the parameter depends on the entity that is used).

The following is an explanation based on an example statement using a parallel query.

When a session runs a statement for a table with parallel degree of *n* (4 degrees for example), the Oracle engine creates *n* new sessions to serve it (these sessions are called: "Parallel Server Agents"). Each of the parallel server agents runs as an individual entity and analyzes part of the requested data, while the original session (the one that you logged in with) is called the coordinator session. The last goal is to receive all the information from the parallel server agents and summarize (sort/filter/calculate/join) it. Usually the coordinator session will not perform an "In Oracle" operation. In fact it is waiting "All the time" when the parallel server agents are working. These parallel server agents can perform I/O / Locks / RAC operations - like any other regular session. These sessions (4 in our example) have the same session identifiers as the coordinator session. Therefore in the collector's aggregation phase all the sessions are aggregate into one row. For example, if a specific statement runs for 1 hour (wall clock time) you can see in the Activity tab that the statement consumed 4 hours "In Oracle" time. You can also find the parallel degree of the statement within the Activity tab.

Oracle statistics usage within Precise for Oracle

Precise for Oracle captures standard Oracle statistics that you can see in the V$SESSTAT table. However, the Collector reads the values in a low-overhead fashion directly from the SGA memory that the V$ tables map onto. Where necessary, the statistics are divided between the sessions executing each SQL statement and then aggregated to the user, program, and instance levels. You will see these statistics in the Current and Activity tabs, displayed as name-value pairs. Each statistic represents a different counter and has its own units. See Oracle's documentation for more information on these statistics.

Precise Interpoint - how resource consumption of packaged applications is examined

Precise Interpoint is the only tool that helps the ERP administrator identify the exact cause of problems in ERP applications.

To tune any application, you need to identify specific users and programs and be able to review their activities and resource consumption in relation to the statements that they execute. In this respect, packaged application tuning poses several problems.

First, ERP applications identify themselves to Oracle through only one user ID or a few user IDs, and by only a few program names. When faced with a problematic statement, you have no way of identifying the user who ran the statement, or the transaction from which the statement was executed.

Second, the statements executed by ERP transactions can rarely be changed. This means that your tuning can only focus on the objects referenced by these statements.

Precise Interpoint provides solutions to ERP application tuning restrictions by allowing you to display explicit information on the activities and resource consumption of your packaged application components. This lets you identify the ERP users, transactions, reports, and so on, that consume the most Oracle resources.

You can look at the information about a packaged application work type (a type of working process executed by an internal packaged application) collected during a historical time frame. The collected information tells you about the application work type, its resource consumption, and the activities it performed in Oracle.

After using Precise Interpoint to detect the biggest resource consumers, you can drill down to the statements that they executed and the objects they accessed. After ascertaining the objects referenced by these statements, you can use the What-If tab to tune them, thus consequently reducing the response times of your statement and improving the performance of your packaged applications.

Currently, Precise Interpoint supports the technologies Oracle Applications, SAP, PeopleSoft, and Siebel.

Precise Interpoint replaces some of the entities that appear in the base Precise for Oracle product with entities that are relevant to the ERP technology.

The following table shows how Precise Interpoint maps the entities in each ERP application.

You can find information about how to install Oracle Applications, SAP, PeopleSoft, and Siebel Tier Collectors in the *Precise Installation Guide*.

Table 2-6 Mapping of Precise Interpoint entities per technology

Technology	Entity without Interpoint Installed	Entity with Interpoint Installed
Oracle Applications	User	User
Oracle Applications	Program	Form
Oracle Applications	Module	Application
Oracle Applications	Action	Request
Oracle Applications		Work Type: OA Batch, OA Interactive, OA Self-Service, OA CRM, or Other
SAP	User	User
SAP	Program	SAP Transaction
SAP	Module	Function
SAP	Action	Screen
SAP		Work Type: SAP Dialog, SAP Batch, SAP Spool, or SAP Lock/Enqueue

Table 2-6 Mapping of Precise Interpoint entities per technology

Technology	Entity without Interpoint Installed	Entity with Interpoint Installed
PeopleSoft	User	User
PeopleSoft	Machine	Machine
PeopleSoft		Work Type: PS Interactive, PS Batch, or Other
Siebel	Program	Siebel Views
Siebel	Module	Siebel Business Objects
Siebel	Action	Siebel Applications
Siebel		Work Type: Interactive, Task Batch, Task Background, or Task Interactive

Precise for Storage, Oracle version - how resource consumption of storage devices is examined

Precise for Storage, Oracle version supports EMC Symmetrix and Clariion, and Hitachi HDS. It offers a unique method to identify I/O problems in your attached storage and reports I/O wait time instead of just I/O counters. This lets you easily identify the resource that is actually being waited for and tune it instead of guessing by rules of thumb. Furthermore, you can associate the storage device with the applications and statements that are waiting on I/O.

You may look at a device, for example, with a relatively high throughput and I/O rate. However, does this mean that the device causes performance problems in the Oracle applications? Even though the device seems to be very active, it does not necessarily follow that the applications are waiting for I/O on this device.

It becomes even harder to decide whether I/O problems are occurring according to the activity parameters for the device in the Oracle application. Oracle uses mechanisms like sequential read to read in advance blocks required by applications before they actually demand them and background write to write data to files using background processes. These mechanisms may increase the volume of activity on a device without causing I/O problems.

Because Precise for Storage, Oracle version, reports I/O wait time and not just I/O counters, you are able to find out if the applications are actually waiting for the device.

Performance problems caused by high I/O wait time may be tuned on the following levels:

- Oracle files
  After finding Oracle files for which I/O wait time is high, you can improve I/O wait time using different techniques. For example, you can partition a table residing in the file, separate tables residing in the file into different files, or decrease the number of extents making up the file.

- Devices
  After finding devices for which I/O wait time is high, you can improve I/O wait time using different techniques. For example, you can move a heavily used Oracle file to a less active device, change the configuration, and relocate devices with high waiting time to different physical devices or balance the volume of the disk directors.

It is important to understand how Precise for Oracle can aid you with this latter activity. Precise for Oracle provides two different views of device performance. The Activity tab shows the time waited on devices used by the currently selected instance: you use this tab to find the performance bottlenecks within your application. The Objects tab shows performance counters for all devices, not just those used by the current instance. This gives you a view of the overall performance of the storage arrays and lets you choose less busy devices to move your Oracle files to.

Launching Precise for Oracle from Precise StartPoint

Precise is a Web-based application. You can access the Precise user interface using the Internet Explorer browser, version 6.0 or later. The syntax of the Precise URL address is: `http://<server>:<port>`, where `<server>` refers to the Precise FocalPoint server and `<port>` refers to the port number used by the GUI Web server. By default, the port number is 20790 (20760 for J2EE). For example: `http://beanotown:20790`.

This URL provides secure access to the StartPoint using authorized roles. From here, you can launch all Precise products. It gives you a quick overview of the status of your applications and access to the AdminPoint, where you can perform various management tasks (see the *Precise Administration Guide* for details).

| NOTE | You must have local administrator privileges on the server where the StartPoint is running. |

To launch your product using Precise StartPoint

1. Type the address of the StartPoint user interface into the Address bar of your browser and click **Enter**. The Precise login page opens. The login page provides secure access to Precise and to your specific product.

2. Specify your authorized role name and password. By default, both role name and password are `admin`. For more information about role names, see the *Precise Administration Guide*.

3. Click **Login**. The StartPoint page opens. This is the Precise home page.

4. On the Product Selection bar, from the drop-down list, select the product you want to launch.

Getting an overview of your application

This section includes the following topics:

- About the Dashboard tab
- How the Dashboard tab is structured
- How the Dashboard tab can help you identify performance problems

About the Dashboard tab

The Dashboard is the starting point for your Precise analysis since it displays an overview of all the instances available in the selected application and information that is also available in some of the other tabs.

The Dashboard tab displays availability, statistics over time, and resource consumption of all Oracle instances for the selected time frame. This tab can be used to monitor your system's current state by viewing all the available instances, a list of all active sessions, and the resource consumption for the last minute (Current tab). In addition, you can view the highest resource consumer in terms of work type, statistics over time, resource consumption over time, and top resource consumers within the selected time frame (on the Overview, Statistics, Applications, Current, Availability, and Activities tabs). If you are deploying the Oracle Real Application Clusters (RAC) option, the RAC Database tabs displays load balancing information, lists the top RAC events and top objects experiencing RAC wait.

The Dashboard tab provides answers to the following types of questions:

- Are all my Oracle instances up?
- Which instance is consuming the most resources?
- Which program is consuming the most resources for the specified instance and across all instances?
- From what types of performance problems is an instance suffering?

The All row in the All Oracle Instances table located in the Main area (left pane) displays the average availability of all instances. When you select the All row, a summary of the activities of all the instances and a comparison of application identifiers is displayed in the various tab pages viewed in the Instance Details area (right pane).

In general, launching to a tab from the All row focuses in on the Tier entity, which summarizes the data collected for all instances. Tabs that do not have a Tier entity, open on the default instance.

NOTE   If you are working with Oracle Applications, SAP, PeopleSoft, or Siebel and ERP support is installed, the Dashboard tab displays information on Oracle activity in relation to the ERP application and users, and specific information on executed views and transactions.

When an Interpoint for ERP is installed, the Precise for Oracle Collector divides Oracle activities into several groups, called work types.

The following table lists the work types available that are available per technology.

Table 3-1 Work types per technology

Technology	Work type
Oracle Applications	Can be any of the following: ■ OA Interactive—Activities generated from the Oracle Applications interactive work process. ■ OA Batch—Activities generated from Oracle Applications batch processes. ■ OA Self Service—Activities generated from Oracle Applications self-service operations. ■ OA CRM—Activities generated from Oracle CRM activities.
SAP	Can be any of the following: ■ SAP Dialog—Activities generated from the SAP Dialog work process. ■ SAP Update—Activities generated from the SAP Update work process. ■ SAP Spool—Activities generated from the SAP Spool work process. ■ SAP Background—Activities generated from the SAP Background work process.
PeopleSoft	Can be any of the following: ■ PS Interactive—Activities generated from the PeopleSoft Interactive panel. ■ PS Batch—Activities generated from the PeopleSoft batch.
Siebel	Can be any of the following: ■ Siebel Interactive- Activities generated from a client request. ■ Siebel Batch-Activities that must be started manually, but require no user intervention. ■ Siebel Background- Activities generated in the background and require no user intervention.

See About Precise for Oracle tabs, "About the Current tab" on page 51, "About the Activity tab" on page 61, "About the Objects tab" on page 87, "About the SQL tab" on page 131, "About the What-If tab" on page 159, "About the Statistics tab" on page 165, and "About the Oracle Applications tab" on page 178.

How the Dashboard tab is structured

The Dashboard tab is divided into two areas the Main area and the Instance Details area. The Main area lists all the instances that are monitored by Precise for Oracle. The Instance Details area provides comparative information regarding the selected instance. You can filter the information shown in the Instance Details area so that only the contribution of certain entities is included. See Filtering data.

About the Main area in the Dashboard tab

The instance table displayed in the Main area lists all the instances that are monitored by Precise for Oracle. Each row corresponds to an instance. The All row displays the average availability (%) of all instances.

The following table shows the information displayed in the Instance table.

Table 3-2 Instance table

Column	Description
Icon	Indicates whether or not the instance is available.
Instance	Name of the instance.
Server	Name of the server.
In Oracle (Sum.)	Total In Oracle time of the instance within the selected time frame.
Executions	Total statements executions within the selected time frame.
Availability	Average percentage of time the instance was available within the selected time frame.
Last Update	Time the instance availability status was last updated.

Table 3-2   Instance table

Column	Description
Clustered	Indicates whether or not the instance belongs to a cluster.
Database	Displays the Oracle database name for the instance.
DB Name	Displays the Oracle database ID for the instance.
Instances	Indicates how many instances share the same database with the instance.

Note: When you open the Dashboard tab, only the first few columns are displayed. To view additional columns (such as: Executions and Last Update columns), use the scroll bar at the bottom of the screen.

See "Determining which table columns to display" on page 33.

About the Instance Details area in the Dashboard tab

This area is divided into tabs and displays overall activity, active sessions, top performance findings, and statistical information for the selected instance. Clicking on the hyperlinked title of specific data items launches, in context, to the related tab from which the data was derived. For example, clicking In Oracle on the Overview tab launches the Activity tab, in context of the current instance (or all instances if you have selected the All row in the instance table), displaying the In Oracle overtime graph. Additionally there is also a Findings section at the bottom part of this area.

The following table describes the information displayed in the Instance Details area.

Table 3-3   Information displayed in the Instance Details area

Tab	Information displayed
Overview	The Overview tab displays In Oracle data and performance findings for the selected instance, over the selected time period.
Applications	The Applications tab displays the top resource-consuming programs, users and machines, for the selected instance, over the selected time period.
RAC Database	Displays information related to the RAC database, such as top RAC events, top objects experiencing RAC wait, and a graph comparing the time spent by the selected instance In Oracle vs. the average time spent by all instances in the same RAC database. (if applicable)
More	The following options are available in the More tab: ■ The Activities tab displays the top resource-consuming statements, objects, and files for the selected instance, over the selected time frame. ■ The Current tab displays active sessions that occurred within the last minute for the selected instance. ■ The Availability tab displays the average percentage of time the instance was available or unavailable, as compared to the Availability SLA.

Note: If you select the All row, the information displayed reflects the average availability of all instances, and the Current tab is not available.

About the overview of your system's performance

The Overview tab displays overall activity and performance findings for the selected instance, within the selected time frame. Items on this tab can launch the Activity, SQL, or Object tab in context for further investigation.

The following table describes the entities displayed on the Overview tab.

Table 3-4   General instance data

General instance data	Description
In Oracle	Displays the total time spent In Oracle over time, broken down into the different In Oracle wait states, such as Using CPU, I/O Wait, Application Lock Wait, and so on. See "In Oracle" on page 36.
Findings	Displays top performance findings. All problems are ranked by severity and lets you launch the finding, in context, to the appropriate tab. The information is displayed in the findings table.
Availability	Displays the Availability SLA (as defined in AdminPoint) vs. the actual availability over time. See the *Precise Administration Guide.*

About top resource-consuming entities

The Applications tab displays the top resource-consuming programs, users, and machines for the selected instance, over the selected time frame.

All entities in this display launch the Activity tab, in context of the Oracle instance.

See "Precise Interpoint - how resource consumption of packaged applications is examined" on page 40.

About RAC database activity

Oracle RAC allows two or more instances to simultaneously access a single database. This allows an application to connect to either computer and have access to the same data. Because all instances access the same database, it is essential that data changes on different computers are coordinated. The RAC Database tab displays activities related to the RAC database, such as top RAC events, top objects experiencing RAC wait, and a graph comparing the time spent by the selected instance In Oracle vs. the average time spent by all instances in the same RAC database. This lets you monitor the RAC activity in your system and check how well Oracle is synchronizing its buffer pool caches.

Table 3-5   RAC Database tab display

Display	Description
In Oracle: Instance vs. All Instances in Same RAC database (Avg.)	Displays a graph comparing the time spent by the selected instance In Oracle vs. the average time spent by all instances In Oracle, in the same RAC database. This indicates whether or not the database is balanced.
Top RAC Events	Displays the top RAC events experienced by the selected instance. Lets you compare the average amount of time spent by the top RAC wait events of all the instances in the RAC database to the amount of RAC wait time spent by the selected instance.
Top Objects Experiencing RAC Wait	Displays the top objects experiencing RAC wait, sorted by how much they suffered from RAC wait. Lets you compare the average amount of time spent by the top objects of all the instances in the RAC database to the amount of RAC wait time spent by the objects of the selected instance. It also displays the object name and schema name of the object and indicates whether it is a table, index or cluster.

About top resource-consuming statements, objects, and files

The Activities tab, accessed from the More tab, displays the top resource-consuming statements, objects, and files for the selected instance, over the selected time frame.

All entities in this display launch the Activity tab, in context of an Oracle instance or Tier.

The following table describes the information displayed in the Statement activities table.

Table 3-6   Statement activities

Column	Description
Statement	Statement name. This is either a numeric hash value or a user-defined name.
In Oracle (Sum.)	Total In Oracle time of the statement, broken down into the different In Oracle wait states.
	See "In Oracle" on page 36.
	Click the respective icon to display the data in this column in a stacked graph, bar graph, or numerical format.
Executions	Number of executions of the statement.

The following table describes the information displayed in the Object activities table.

Table 3-7   Object activities

Column	Description
Object	Name of the object.
Type	Type of the object, typically a table or index.
In Oracle (Sum.)	Total In Oracle time of the object, broken down into the In Oracle wait states relevant to Objects.
	See "In Oracle" on page 36.
	Click the respective icon to display the data in this column in a stacked graph, bar graph, or numerical format.

The following table describes the information displayed in the File activities table.

Table 3-8   File activities

Column	Description
Oracle File	Path and name of the Oracle file.
I/O Wait	Total I/O wait time of the file, broken down into I/O wait states relevant to files.
	See "I/O wait" on page 38.
	Click on the respective icons to display the data in this column in a stacked graph, bar graph, or numerical format.

About the session activity within the last minute

The Current tab, accessed from the More tab, displays active sessions that occurred within the last minute for the selected instance.

All items in this display launch the Current tab, in context of an Oracle instance or Tier.

The following table describes the information displayed in the Session statistics table.

Table 3-9   Session statistics

Counter	Description
Connected	Total number of sessions connected to the selected instance.

Table 3-9   Session statistics

Counter	Description
Active	Total number of sessions currently active.
Locked	Total number of locked sessions.

The following table describes the information displayed in the Session activity table.

Table 3-10   Session activity

Column	Description
Status icon	Indicates the current In Oracle sub-state.
Session ID	The current session ID.
Program	The name of the program that opened the session
Sub-State	A snap-shot of the current In Oracle sub-state of the session, as recorded by the Collector at the last sample point.
In Oracle (Last Minute)	Total In Oracle time for the session occurring within the last minute, broken down into the different In Oracle wait states.

In addition, two pie charts summarize the overall activity and the In Oracle activity that occurred within the last minute, broken down into Overall Activity states and In Oracle wait states, respectively.

See "Overall activity" on page 35.

See "In Oracle" on page 36.

About the availability of an instance

The Availability tab, accessed from the More tab, displays the average percentage of time the instance was available or unavailable, as compared to the Availability SLA. The second graph shows the number of times for unavailability.

About the Findings area in the Dashboard tab

The Findings area in the Dashboard tab displays performance problems detected in the select instance(s). The findings list is displayed according to severity (highest to lowest).

To view the details of a specific finding, hover over the finding name. The area for the selected finding will expand, displaying information and recommended actions. Click on the relevant link to launch the appropriate tab in context, or **Learn more** to open the help for that finding.

For more information about identifying performance problems in the Dashboard tab, see "How the Dashboard tab can help you identify performance problems" on page 48.

How the Dashboard tab can help you identify performance problems

You can identify a performance problem by doing one or more of the following:

■   Examining Oracle instance availability

■   Identifying the most resource-consuming instance

■   Identifying the top resource consumers in various entities

Examining Oracle instance availability

It is possible to examine the availability of Oracle instances and observe their current status as well as their level of availability over time. In addition, you can determine whether their level of availability meets the SLA standard defined in the SLA availability setting in AdminPoint.

Note: The Availability SLA is defined in AdminPoint.

For more information, see the *Precise Administration Guide*

See "How Precise for Oracle monitors availability" on page 35.

The following example shows how to examine the availability of an Oracle instance, observe its current status, and its availability over time.

A blank area in the graph denotes periods when the Collector was down.

 To examine Oracle instance availability

1   In the Time Frame list, choose the period of time you want to analyze.

2   In the Main area of the Dashboard tab, check the icon status. The icon displayed in the first column indicates the current availability state of the instance. A currently running instance displays a running figure icon.

3   Move the mouse pointer over the icon to display a ToolTip. The ToolTip shows the current status of the instance.

4   Under the Availability column in the All Oracle Instances table, check the average availability (percentage) of the instance. The Availability column represents the average availability (percentage) of the instance within the selected time frame.

5   In the Instance Details area, click the **More>Availability** tab. This displays the instances' availability over time with respect to the configured SLA. If Precise for Oracle is integrated with Alerts, an alert is issued at the time an SLA is breached.

Identifying the most resource-consuming instance

Precise for Oracle enables you to determine which are the most resource-consuming instances, indicating a potential performance problem.

The following example shows how to determine which instance consumes the most resources. This example can show the top resource-consuming instances, as measured by the total time spent executing In Oracle.

To identify the most resource-consuming instance

1   In the Instance list, choose the instance you want to analyze.

2   In the Time Frame list, choose the period of time you want to analyze.

3   In the Instance Details area, click **More>Statistics.**

4   Check the physical reads.

5   Check the redo graph.

6   Use any of the following options to investigate further:

■   Launch the Statistics tab for the selected instance.

■   In the Activity tab, check what the application was doing at specific times.

Identifying the top resource consumers in various entities

A look at the Applications tab quickly helps you identify the top resource-consuming entities.

The following example shows how to identify the top resource-consuming entities.

To identify the top resource consumers in various entities

1   In the Instance list, choose the instance you want to analyze.

2   In the Time Frame list, choose the period of time you want to analyze.

3   In the Instance Details area, click **Applications.** For example, a SAP database, PS8 on hp8, is monitored by Interpoint for SAP.

The Application tab shows the following SAP-related top entities:

- Background work type

- SAP user SUPER

- Transaction SAP_CCMS_MONI_BATCH_DP

Most likely, these entities are related.

4   To investigate further, click an entity to launch the Activity tab. In the Activity tab you can confirm the connection between the problematic entities. For this example, you can identify a background SAP transaction run by the SAP CCMS administration tool.

Getting current application information

This section includes the following topics:

- About the Current tab

- How the Current tab is structured

- How the Current tab can help you identify performance problems

About the Current tab

The Current tab displays near real-time information on all sessions that were connected to your Oracle database during the last minute. You can use this tab to monitor your system's current activities and resource consumption and to identify performance problems, such as sessions involved in locks, as they occur.

The Current tab provides answers to the following types of questions: How many sessions are currently running? or Why doesn't my application work when I click OK?

Using the Current tab, you can monitor current connections to your Oracle instance, check the state of each connection, and identify the resources currently utilized by each session. Further investigation of each session provides a thorough understanding of the SQL statement being executed, the login name executing the statement, the program, and the resource utilization for the session.

Note: If you are working with Oracle Applications, SAP, PeopleSoft, or Siebel and ERP support is installed, the Current tab displays information on the current Oracle activity in relation to the ERP application and users, and specific information on executed views and transactions.

See "Precise Interpoint - how resource consumption of packaged applications is examined" on page 40.

In the Current tab, data on a current session is continuously updated to provide information on what happened during the last minute. Because the displayed session information is derived from a dynamic application, it is recommended to regularly refresh the screen, either manually or automatically, to ensure that the information displayed is up-to-date.

Note: The Current tab displays information on the last minute of activity. For example, if you have worked at your screen for ten minutes and then refresh the information displayed on your screen, you will be able to view new information that occurred during the last minute of activity. Similarly, any items that were previously displayed, but have had no activity in the very last minute (rather than the 10 minutes since the last refresh) will not be displayed again, and so will disappear from the screen.

You can set the display to automatically refresh every few minutes; refresh can either be enabled or disabled from the Settings dialog box. If it is enabled, the display will be refreshed every <n> minutes, where <n> is the number specified in the Settings dialog box.

See "About configuring general settings" on page 27.

See "How most tabs are structured" on page 22.

See "About the Dashboard tab" on page 43.

How the Current tab is structured

The Current tab displays information on a selected entity and its associated entities. For example, if the selected entity is a Session, its associated entities are Statements. The selected entity is always reflected in the Tab heading, which serves as a point of orientation. The highest-level entity you can view information for in the Current tab is an Instance. You can view another instance by selecting it from the Instance list. When you open the Current tab from any tab, it always displays information on the instance level, even if the currently selected entity is not an instance.

About the Main area in the Current tab

The Main area shows comprehensive information on the selected entity. You can choose from several views to examine the entity from different angles. For example, if you are analyzing a session entity, you can focus exclusively on the data displayed in the overview, such as, In Oracle data (default), or choose to view the text for the current statement being run by the session.

About the Association area in the Current tab

The Association area provides corresponding information on the entities associated with the selected entity (displayed in the Main area). For example, you can associate with the Active Sessions related to the selected instance. The selection you make is reflected in the Association area only; the Main area remains unchanged. For some entities, tabs above the Association area enable you to view additional information. For example, the tabs displayed for a session entity are: General, In Oracle, and Connection. Clicking a tab displays different table columns with the same type of associated entities.

From the Association area, you can also drill down to another entity by clicking a table row. A drill down affects the entire tab. When you drill down to another entity, the Tab heading displays the new selection, the Main area displays information on the newly selected entity, and the Association area displays the entities associated with the selected entity.

For example, when you want to view information on a specific active session, from the Association controls, click Active Sessions. The Association area changes to display session-related information. Note that the Tab heading and the Main area remain unchanged.

If you want detailed information on an active session, in the Association area, click the row of the session that you want to view detailed information for. The Tab heading indicates the newly selected entity; the Main area displays In Oracle information on the session you drilled down to, and the Association area shows information on the statements associated with this session.

See "About Precise for Oracle tabs" on page 21.

See "About the Dashboard tab" on page 43.

About examining entities in the Current tab

The Current tab displays information on different entities. This section provides an overview of all entities, their meaning, and their views.

The following entities can be examined in the Current tab:

- Instance
- Session
- Statement

About the Instance entity

The Instance entity shows a summary of all sessions connected to Oracle during the last minute.

Getting an overview of current Instance activities

The Instance Overview displays information on the activity of the instance in the Main area.

The following table describes the information displayed in the Instance Overview.

Table 4-1   Instance Overview

Item	Description
Sessions	Displays information on the following items: ■ Connected—Number of sessions that were connected to the Oracle instance during the last minute. ■ Active—Number of sessions that were connected to the Oracle instance during the last minute and spent time executing In Oracle. ■ Locked—Number of locked sessions. ■ Sessions via Dedicated Server—Number of sessions that were connected to the Oracle instance using an Oracle dedicated server process (or shadow process) during the last minute. ■ Sessions via Shared Server—Number of sessions that were connected to the Oracle instance using an Oracle multi-threaded server process (MTS) during the last minute. ■ Active Sessions In Oracle—Average In Oracle time of active sessions during the last minute. Equals the total In Oracle time divided by the number of active sessions. The format is: hh:mm:ss.s.
Parallelism	Displays information on the following items: ■ Idle Parallel Servers—Number of PQ servers that have not been used in the last minute. ■ Active Parallel Servers—Number of PQ servers that have been used in the last minute.
In Oracle (Last Minute)	Displays information on the following items: ■ Sub-State—In Oracle sub-state. ■ Bar graph—Graphical representation of the Time column. ■ Time—Amount of time that sessions were in this state. ■ %—Percentage of time that sessions were in this state. ■ Sessions—Number of sessions in each state. A session may be in more than one state during one minute. See "In Oracle" on page 36.
Overall Activity (Last Minute	A pie chart illustrating the overall activity of all sessions during the last minute. See "Overall activity" on page 35.
Works Types (Last Minute)	If the chosen instance has ERP support installed, this tab also displays a pie chart of work types.

About associating an Instance with other entities

You can associate an Instance, in the Main area of the Current tab, with the following entities in the Association area:

■ Current Sessions

   A list of all sessions that were connected during the last minute.

■ Active Sessions

   A list of all sessions that were connected and spent some time executing In Oracle during the minute.

These lists of sessions are each divided into different sets of information, showing different aspects of the sessions. Each set is accessed by a tab.

The following table describes the information shown on Instance Association tabs.

Table 4-2 Instance Association tabs

Tab	Columns
General	Displays the following table columns: ■ Icon—Indicates the current sub-state. ■ Icon—Enables the Kill Session process. ■ Program—Name of the program connected to Oracle. ■ Session ID—Session ID or number. ■ Oracle Process ID—OS-assigned process ID of the Oracle server process. ■ Serial #—Oracle serial number, which, along with the session ID, uniquely identifies a session. ■ In Oracle—Bar chart showing the In Oracle breakdown for the session. ■ Sub-State—Snapshot of the current In Oracle wait sub-state. ■ Wait Event—Snapshot of the current Oracle wait event. ■ Tune icon—Allows launching the SQL tab in the context of the current statement. ■ Text—Portion of any SQL statement that is actually being executed at the time of the refresh; otherwise, it is blank. See "In Oracle" on page 36.
In Oracle	Displays the following table columns: ■ Icon—Indicates the current sub-state. ■ Session ID—Session ID or number. ■ Oracle Process ID—OS-assigned process ID of the Oracle server process. ■ In Oracle—Bar chart showing the Oracle breakdown for the session. ■ Using CPU—In Oracle wait state. ■ I/O wait—In Oracle wait state. ■ Application Lock wait—In Oracle wait state. ■ Commit & Checkpoint wait—In Oracle wait state. ■ Communication wait—In Oracle wait state. ■ Internal Lock wait—In Oracle wait state. ■ Oracle Server wait—In Oracle wait state. ■ Operating System wait—In Oracle wait state. ■ RAC/OPS/Other wait—In Oracle wait state. See "In Oracle" on page 36.

Table 4-2   Instance Association tabs

Tab	Columns
Connection	Displays the following table columns: ■ Icon—Indicates the current sub-state. ■ Session ID—Session ID or number. ■ Oracle Process ID—The ID of the Oracle server process. This ID is assigned by the operating system. ■ Oracle Process Type—Any of the following: Dedicated: Indicates a multi-threaded server Shared: Indicates a server that is not multi-threaded Parallel: Indicates that the session is a sub-session of another session ■ Connect Time—The date and time that the session got connected to Oracle. ■ Work Type—Group of Oracle activities. Only relevant when an Interpoint is installed. See "About the Dashboard tab" on page 43. ■ ERP—Indicates the kind of Interpoint that is used. Can be OA (Oracle Applications), SAP, PS (PeopleSoft), or SBL (Siebel). If no Interpoint is installed, N/A is displayed. ■ Host User—The name of the host user. The information in this column is derived from Oracle's v\$session table. ■ User—The name of the user. The information in this column is derived from Oracle's v\$session table. ■ Program—The name of the program. The information in this column is derived from Oracle's v\$session table. ■ Machine—The name of the machine. The information in this column is derived from Oracle's v\$session table. ■ Module—The name of the module. The information in this column is derived from Oracle's v\$session table. ■ Action—The name of the action. The information in this column is derived from Oracle's v\$session table. ■ Terminal—The terminal name of the operating system. The information in this column is derived from Oracle's v\$session table. ■ Status—Status of the session. See "In Oracle" on page 36.

Note: Precise for Oracle retrieves the displayed information from Oracle's V\$SESSION table. If this table does not contain relevant data for a specific cell in the Association table, this cell is not populated with data.

By clicking a row in the Association table, you can drill down to the specific session.

See "About the Session entity" on page 56.

About the Kill Session process

The Kill Session process is executed by the Kill Session button. This button enables you to kill the selected session. You can instruct Oracle to execute this command immediately (using the Oracle Alter System Kill Session Immediate command) or during the time of less activity (using the Oracle Alter System Kill Session command).

To perform the Kill Session during the time of less activity

1   On the general tab of the Instance Association, select the session that needs to be killed.

2   Click **Kill Session**.

3   In the dialog, insert the Oracle DBA user name and password.

4   Click **OK**.

To perform the Kill Session immediately

1   On the general tab of the Instance Association, select the session that needs to be killed.

2   Click **Kill Session**.

3   In the dialog, insert the Oracle DBA user name and password.

4   Mark **Kill immediately**.

5   Click **OK**.

About the Session entity

The Session entity displays information on a single connection with Oracle.

About the overview of current and active sessions

The Session Overview displays general information on the session.

The following table describes the information displayed in a Session Overview.

Table 4-3   Session Overview

View area	Description
Session	Displays information for the following items: ■ Serial #—Oracle serial number, which, along with the session ID, uniquely identifies a session. ■ Connected Time—Time that the session connected to Oracle. ■ Duration—Duration of the session since it connected. ■ Sampling Start Time—Time that the Collector started sampling this session. ■ Sampling Duration—Duration that the Collector has sampled this session. ■ Oracle Process ID—Process ID of the Oracle server or shadow process. ■ Client Process ID—Process ID of the client or application process, if it is running on the same server as the instance; otherwise, it is zero. ■ Oracle Process Type—Type of Oracle server process: Dedicated or MTS. See "In Oracle" on page 36.
Connection	Displays information for the following items: ■ Program—Name of the program connected to Oracle. ■ Module—Current Oracle module, if Oracle Modules and Actions are used. ■ User—Name of the Oracle user. ■ Host User—Name of the OS user connected to Oracle. ■ Machine—Name of the machine connected to Oracle. ■ Terminal—Name of the terminal connected to Oracle. ■ Sub-State—Snapshot of the In Oracle wait state. ■ Wait Event—Snapshot of the Oracle wait event. This is equivalent to the EVENT column in the V$SESSION_WAIT table. ■ Open + Fetches—The number of times that the Collector encountered the session either opening or fetching from a cursor. See "Executions vs. opens and fetches" on page 39.
In Oracle (Last Minute)	Displays information for the following items: ■ Sub-StateIn—Oracle sub-state. ■ Bar graph—Graphical representation of the Time column. ■ Time—Amount of time the session was in this state. ■ %—Percentage of time the session was in this state.

About the text of a current or active session

The Text view displays the SQL text for the statement that the session is currently executing, if any.

About associating a session with other entities

You can associate a Session, in the Main area of the Current tab, to different entities in the Association area. The information displayed is divided into different sets that show different aspects of the entity. Each set is accessed by a tab.

The following table describes the entities that are available in the Association area.

Table 4-4 Association area entities

Entity	Description
Statements	Displays a list of statements run by the session during the last minute.
Sessions Involved in Lock	Displays a list of other sessions locking or locked by this session during the last minute. Three of the tabs displayed here are the same as those displayed for a regular session. In addition, the Lock tab displays locking information that is equivalent to the information displayed in Oracle's V$SESSION_WAIT table (For more information, see the Oracle documentation).
Parallel Query Server Sessions	Displays Parallel Query Coordinator or Parallel Query Slave sessions associated with this session during the last minute. The tabs displayed here are the same as those displayed for a regular session. See "About associating an Instance with other entities" on page 53. **Note:** If you drill down to a Parallel Query Server process, the Main area displays information on this session. You can then identify the Parallel Query Coordinator Session using the Association controls.

The following table explains the information that is displayed on the tabs in the Association area, when the Statements entity is selected.

Table 4-5 Statements Association area table

Tab	Columns
Statement	Displays the following table columns: ■ Statement—Unique identifier assigned by the Collector to the SQL statement. ■ Hash Value—Value assigned to the statement by Oracle. ■ In Oracle—Bar graph showing the In Oracle breakdown. ■ Tune icon—Allows launching the SQL tab in the context of the current statement. Clicking elsewhere in the row drills down to the statement in Current tab. ■ Text—Portion of the SQL text for the statement. Note that if the cursor is moved over the text, then a tool-tip appears containing the full text of the statement. See "Statement identifier vs. Oracle hash value" on page 39. See "In Oracle" on page 36.
In Oracle	Displays the following table columns: ■ Statement—Unique identifier assigned by the Collector to the SQL statement. ■ In Oracle—Bar chart showing the In Oracle breakdown for the session. ■ Using CPU—In Oracle wait state. ■ I/O wait—In Oracle wait state. ■ Application Lock wait—In Oracle wait state. ■ Commit & Checkpoint wait—In Oracle wait state. ■ Communication wait—In Oracle wait state. ■ Internal Lock wait—In Oracle wait state. ■ Oracle Server wait—In Oracle wait state. ■ Operating System wait—In Oracle wait state. ■ RAC/OPS/Other wait—In Oracle wait state. See "Statement identifier vs. Oracle hash value" on page 39. See "In Oracle" on page 36.
Statistics	Displays the change in Oracle statistics for the statements in the last minute. See "Oracle statistics usage within Precise for Oracle" on page 40.

The following table explains the information that is displayed on the Lock tab in the Association area, when the Sessions Involved in Lock entity is selected.

Table 4-6 Lock tab information

Column	Description
Program	Name of the program connected to Oracle.
Session ID	ID of the session.
In Oracle (Last Minute)	Bar chart showing the In Oracle breakdown for the session.
Status	Indicates whether the session is waiting for or holding the lock.
Type	Transaction type: TX, TM, and so on. This information is derived from the P1 parameter in the V$SESSION_WAIT table.
Mode	Transaction mode: Exclusive, Shared, and so on. This information is derived from the P1 parameter in the V$SESSION_WAIT table.
ID1	This information is equivalent to P2 parameter in the V$SESSION_WAIT table.
ID2	This information is equivalent to P3 parameter in the V$SESSION_WAIT table.
Object	Locked object, if appropriate.
Text	Portion of SQL text for the session being locked.

About the Statement entity

The Statement entity, in the Current tab, displays information on a single SQL statement that has been run by a session in the last minute or is still running. You cannot associate a statement with any other entity. This implies that if the Main area displays information on a statement, the Association area remains blank.

See "About Precise for Oracle tabs" on page 21.

Getting an overview of current Statement entities

The Statement Overview displays general information on a single SQL statement.

The following table describes the information displayed in a Statement Overview.

Table 4-7 Statement Overview

View area	Description
Statement	Displays information on the following items: ■ Hash Value—Oracle hash value. ■ In Oracle (Sum.)—Total time spent In Oracle. ■ Opens + Fetches—Number of times sessions opened and fetched from a cursor for this statement.
In Oracle (Last Minute)	■ Sub-State—In Oracle sub-state. ■ Bar graph—Graphical representation of the Time column. ■ Time—Amount of time sessions were in this state. ■ %—Percentage of time that sessions were in this state. ■ Sessions—Number of sessions in each state. Note a session may be in more than one state during a minute.
Overall Activity (Last Minute)	A pie chart illustrating the overall activity of the statement during the last minute. See "Overall activity" on page 35.
Text	The SQL text of the statement.

About the text of a statement

The Text view displays the SQL text of the statement.

About statement-related statistics changes

The Statistics view displays a list of the changes that occurred in the Oracle statistics for the statement during the last minute.

See "Oracle statistics usage within Precise for Oracle" on page 40.

The following table describes the information displayed in the Statement Statistics view.

Table 4-8   Statement Statistics view

Column	Description
Statistic	Oracle statistic name
Value	Statistic value

How the Current tab can help you identify performance problems

You can identify a performance problem by doing one or more of the following:

- Examining resource consumption of an entire instance

- Examining a single connection

- Resolving a locking situation

Examining resource consumption of an entire instance

The Current tab displays near real-time information on all sessions that were connected to your Oracle database during the last minute. Usually you will use this tab to monitor the system's current activities and resource consumption, and to identify performance problems, such as sessions involved in locks as they occur.

The following example shows how to examine resource consumption of an instance. By examining the current activity at an instance level of display, you can see how it is currently behaving and determine if there is an overall problem that affects all sessions or if there is a problem with specific sessions. For this example, the Main area of the Current tab shows the top two sub-states (in the Sub-State column) for the instance's In Oracle time as Using CPU and Redo Log Buffer Wait.

To examine resource consumption of an entire instance

1   In the Instance list, choose the instance you want to analyze.
    Do one of the following:

    - On the View controls in the Main area, click **Overview.** This will show you a breakdown of resource consumption for the last minute.

    - On the View controls, click **Statistics** to analyze the current statistics of the selected instance.

2   In the Main area, check for any unproductive wait states. In our example, the sessions on average (in the % column) are spending roughly a third of their time (34.91%) waiting on the redo log buffer.

3   To further investigate, launch the Statistics tab. View the Statistics tab to look for system events that relate to the redo log buffer. Launch to the Statistics tab, in context, to view additional statistical information for any selected instance.

4   In the Association area, move the cursor over the In Oracle bar in the list of Active Sessions to display a ToolTip that shows which sessions are spending the majority of their time waiting on Redo Log Buffer Wait.

5   Check how much time is spent using CPU. If all other wait states have been eliminated, obviously this will be 100%; therefore, it is important to look at the actual time (in hh:mm:ss), in the Main area, as well as the percentage value. High CPU usage can indicate excessive logical I/O operations.

6   To further investigate, open the Activity tab. Try tuning any SQL with a high Using CPU time and high values for the statistics consistent gets and db block gets.

Examining a single connection

You can also monitor current connections to your Oracle instance, check the state of each connection, and identify the resources currently utilized by each session.

The following example shows how to examine a single connection. For this example, the Association area of the Current tab displays the sub-state Using CPU in the Active Sessions list.

To examine a single connection

1   In the Instance list, choose the instance you want to analyze.

2   In the Association area, drill down to the session whose connection you want to analyze. For an example, click the top session ID 10 (in the Session ID column), in the Active Sessions list, for the sub-state Using CPU. In this example, we see all the statements that the selected Session entity has executed in the last minute, or is possibly still executing. The top statement is mainly using CPU and waiting for I/O.

3   Continue your analysis by examining its resource breakdown. To further investigate, in the Association area, click the Tune icon in the row of the top statement. Clicking the Tune icon opens the SQL tab for tuning this statement.

Resolving a locking situation

You can view any sessions that are locked in the Current tab. Then you can fully investigate the situation to determine the type of lock and which session is holding the lock.

In the Main area, the following example shows three sessions (shown in the Sessions column) that have waited for a row lock over the last minute. The time waited (shown in the Time column) is 1 minute 57 seconds or 100% of the total In Oracle time for the instance.

In the Association area, three active sessions are in the state of a row lock wait. For our example, Session 117 (the top locked session shown in the Association area) is attempting to execute and UPDATE on the CUSTOMER table (shown in the Program column).

To resolve a locking situation

1   In the Instance list, choose the instance you want to analyze.

2   On the Association controls, in the Association area, you can associate Session 117 with the other Sessions involved in the lock. For an example, Session ID124 is holding the lock in Exclusive mode.

3   Using the Association controls, you can now drill down to see the statements being executed by the session that is holding the lock (Session ID 124).
     For our example, in the Main area, you can see that the holding session (for Duration) has not run any statements in the last 15-minute time slice, meaning that the UPDATE to the CUSTOMER table (for Program) was executed earlier but that the session has not yet issued a COMMIT to release the row lock.

Examining performance over time

This section includes the following topics:

About the Activity tab

The Activity tab allows you to analyze performance behavior over time. The information displayed in the Activity tab is derived from historical data and enables you to access information for different time intervals at varying levels of detail. You can analyze what happened at a specific time, pinpoint problematic time periods, and drill down to analyze a bottleneck that caused a performance problem.

The Activity tab was designed to answer questions such as, What happened yesterday, as well as a month ago and a year ago? and What has been the performance level of my applications over time?

Data in the Activity tab is available up to the last 15 minutes (information on the last minute is available in the Current tab). The Activity tab is usually the place to identify and investigate tuning problems.

When working in the Activity tab, you can choose to work on a specific instance or to view information on all your instances.

Note: You usually switch to the Activity tab after focusing on an instance in the Dashboard tab, or by launching Precise for Oracle from another product (Insight or Alerts). In these cases, the product automatically focuses on the relevant instance.

When you do not select a specific instance to view in-context, a summary of the performance information of All Instances in the application is displayed. You can analyze the average behavior of the entire Oracle Tier. You can then identify the instance which consumes the most resources in your system.

Note: If you are working with Oracle Applications, SAP, PeopleSoft, or Siebel and ERP support is installed, the Activity tab displays information on the historical Oracle activity in relation to the ERP application and users, and specific information on executed forms (Oracle Applications) and transactions (SAP).

How the Activity tab is structured

The Activity tab displays information on a selected entity and its associated entities. For example, if the selected entity were a Program, its associated entities would be Statements. The selected entity is always reflected in the Tab heading, which serves as a point of orientation.

The Activity tab can be displayed in two different views—Table view and Tree view. Both views display the same information, unless otherwise specified. The functionality of these views differ.

Note: The initial view show when opening the Activity tab is the Tree view.

When you open the Activity tab from the Dashboard tab, information on the instance selected in the Dashboard is displayed, or on the entire Tier if the All row was selected. When you open the Activity tab from another tab, the settings from your last visit to the Activity tab are retained, such as a drill down to a Session entity. This is similar to clicking the History button and returning to a previous tab.

The highest-level entity you can view information for in the Activity tab is Tier. You can view a particular instance by selecting it from the Instance selector.

You can use the Filter dialog box to filter the Association area so that only the contribution of certain entities is included.

See "Filtering data" on page 29.

About the Table view

The Table view is divided into two areas, the Main area and the Association area. Information on the entity you are analyzing is displayed in the Main area. The entities that are associated with the entity displayed in the Main area are displayed in Association area.

About the Main area in the Activity tab

The Main area shows comprehensive information on the selected entity. You can choose from several views to examine the entity from different angles. For example, if you are analyzing an Instance entity, you can focus on the data displayed in the overview, or choose to view scalability data, or availability information in more detail.

About the Association area in the Activity tab

The Association area provides corresponding information on the entities associated with the selected entity (displayed in the Main area). For example, you can associate with the Programs related to the selected instance. The selection you make is reflected in the Association area only; the Main area remains unchanged. For some entities, tabs above the Association area enable you to view additional information. For example, the tabs displayed for a statement entity are: General, In Oracle, Statistics, and Workshop. Clicking a tab displays different table columns with the same type of associated entities.

From the Association area, you can also drill down to another entity by clicking a table row. A drill down affects the entire tab. When you drill down to another entity, the Tab heading displays the new selection, the Main area displays information on the newly selected entity, and the Association area displays the entities associated with the selected entity.

For example, when you want to view information on a specific program, click `Programs` from the Association control. The Association area changes to display session-related information. The Tab heading and the Main area remain unchanged.

In the Association area, click the row of the program you want to view detailed information for. The Tab heading indicates the newly selected entity; the Main area displays In Oracle information on the program you drilled down to, and the Association area shows information on the statements associated with this program.

You can also launch a different tab altogether. For example, clicking the Tune icon next to a statement will take you to the SQL tab, in context, to allow detailed investigation and tuning of the SQL statement.

About the Tree view

Figure 5-1 Oracle Activity tab Tree view
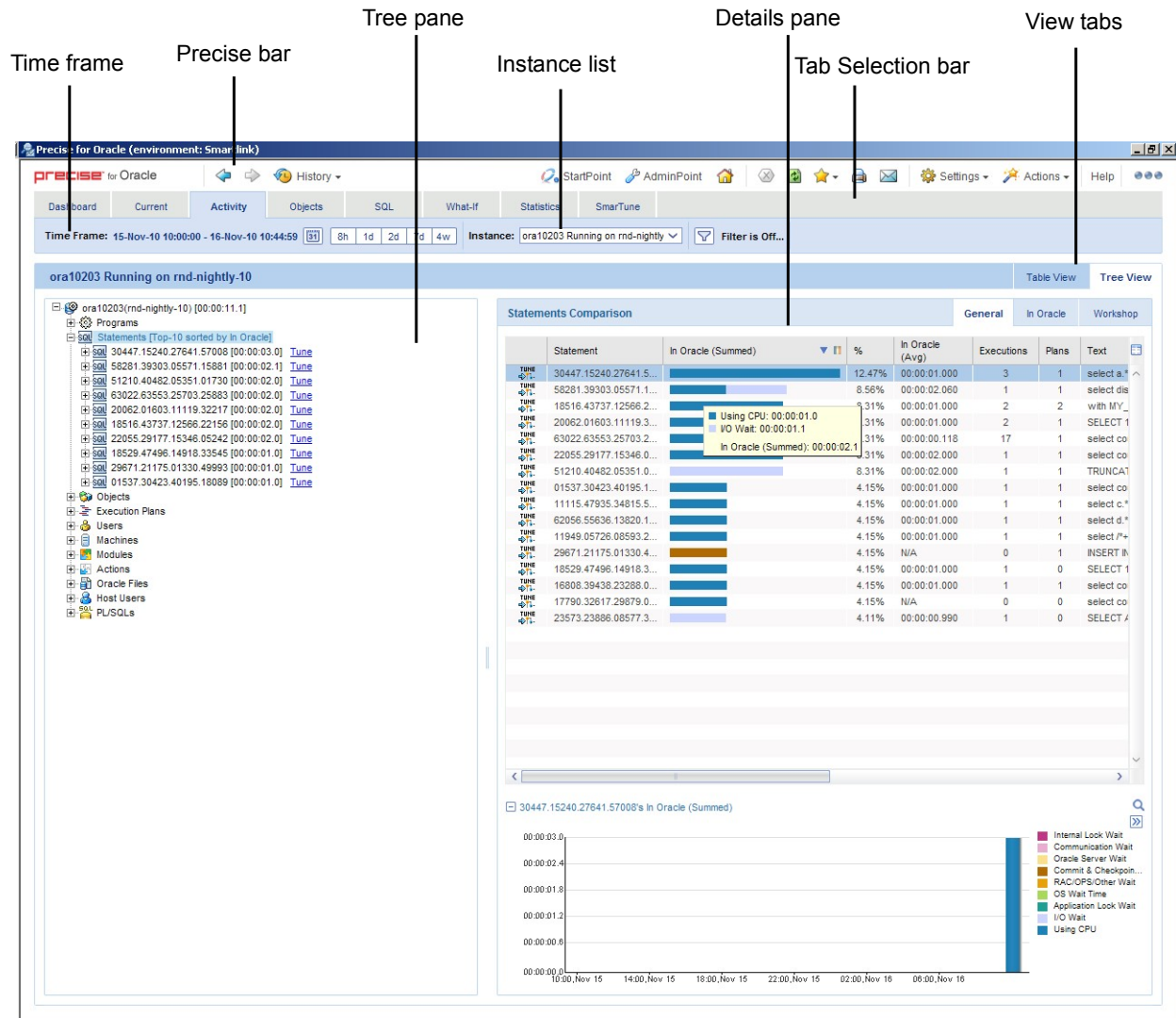
The Tree view displays the same information shown in the Table view, but in a different format that offers different functionality. The Tree view is divided into two areas, the Tree pane and the Details pane. A breakdown of the entities that are associated with the selected entity is displayed in the Tree pane. Information on the entity you selected is displayed in the Details pane.

The order in which the entries in the Tree pane are displayed and which entries are displayed can be changed in the Settings menu.

The duration value displayed next to each entry reflects the duration of each instance in the selected entity, according to the sort parameter you selected. The sort parameter is displayed next to each entity. The default entity for the top entity in the tree is In Oracle time. You change this setting in the Settings dialog box.

See "About configuring Tree view settings" on page 27.

Note: The following entities are not displayed in the Tree view: Recent Sessions, Extended Collections and Imported Statements. The Database entity is only displayed when All Instances is selected.

Determining what information is displayed in the Tree

You can control the number of entries displayed and the order in which they are displayed in the tree using the sort and filter option.

To determine what information to display in the tree

1 Right-click on an entity in the tree, and then click **Sort & Filter.**

2 In the Associate With dialog box, on the Rows tab, select the entity you want to associate data with from the Populate table with list.

3 In the Sort entries by list, determine according to which criteria you want the information to be sorted and in which order.

4 From the Display top list, select the number of rows to display.

5 On the Criteria tab, do the following for each entity you want to associate data with:

From the left drop-down list	Select an entity.
From the middle drop-down list	Select an operator, such as, Like, <>, Not Like, In, Not In.
In the text box	Type the criteria (case-sensitive) for the selected entity.
	If you select the operator Like or Not Like, you can use the % wildcard character. If you select the operator In or Not In, type a comma to separate values.

6 Click **OK.**

About viewing information for entities associated with a selected entity

You can drill down to the entities that are associated with the selected entity by clicking on the plus sign (+) located next to each entity. When you drill down on an entity, the information displayed in the Main area of the Table view is displayed in the Details pane. When you drill down on an instance that is associated with the selected entity, you display the instances that are associated with the selected entity and you can view additional information on the selected instance in the Details pane.

Viewing information from a different time frame

You can switch to a different time frame and view additional information on the selected entity that was collected for that time frame. Only the information in the subnodes of the entity you have selected is refreshed.

To select a different time frame

1 Click on the entity in the tree.

2 In the Time Frame list, select a different time frame than the one already being displayed.

Note: Only the information for the selected entity is refreshed, and it is moved up to the top of the tree.

About switching from one view to another

When you switch from one view to another, the information on the selected entity is displayed, in context. For example, if you switch from Table View to Tree View, the entity you selected is highlighted in the tree, and is displayed at the top of the tree, even if according to the sort criteria it should be displayed in another position.

Refreshing a view

The Refresh option refreshes the display from the PMDB.

To refresh the information displayed in a view

1 Right-click on an entity in the tree.

2   Click **Refresh.**

Printing the Tree View display

You can print an image of what is displayed in the Tree view.

To print the Tree View display

1   Right-click on an entity in the tree.

2   Click **Print.**

3   Use the print features of your browser to print out an image of the information displayed in the Tree view.

About viewing additional information on an entry in a table

When information on a selected instance is displayed in table and graph format, in the Details pane, there is a direct correlation between the selected entry in the table and the information displayed in the graph below the table. Selecting a different row changes the information displayed in the graph displayed below the table.

About the entities you can examine

The Activity tab displays information on different entities. This section provides an overview of all entities, their meaning, and their views.

The following entities can be examined in the Activity tab:

■   Statements

■   PL/SQLs

■   Execution Plans

■   Objects

■   Oracle Files

■   Extended Collection

■   Sessions (in Table view only)

■   Recent sessions (in Table view only)

■   Imported Statements

■   EMC Storage Devices

■   HP Storage Devices

■   Entities mentioned in the following section which display the same fields (further called aggregated entities)

About aggregated entities in the Activity tab

In the Activity tab, there are the following available aggregated entities:

■   All Instances

■   Actions

■   Siebel Business Objects

■   Databases

■   Instances

■   Programs

■   OA Forms

■   SAP Programs

- Users

- Host Users

- Machines

- Modules

- OA Applications

- Siebel Applications

- SAP Functions

- OA Requests

- SAP Screens

- Siebel Views

- J2EE Callers

- J2EE JVM Servers

- Web Transactions

These entities are aggregated entities. They display the aggregated resource consumption of all sessions with the same criteria. For example, User is the aggregation of all sessions executed by the same user; Program is the aggregation of all sessions executed by the same program; Instance is the aggregation of all sessions within an instance. The highest level entity is All Instances. It displays the resource consumption of all the instances in the Tier.

Note: A user may run many programs; a program may be run by many users. To find the resource consumption of a particular user running a particular program, you can either use the Association area in the Table view or navigate to the required context in the Tree view.

All of these entities can be associated with each other and with Statements, PL/SQL, Oracle Files, Objects as well as File Systems, Logical Volumes, and UNIX Devices (only if Precise for Storage, VERITAS version is installed). In addition, all but All Oracle Instances can be associated with Recent Sessions, and Extended Collections.

Database entities can be associated with Instances, Programs, Statements, Execution Plans, Databases, Users, Modules, Actions, Machines, Host Users, PL/SQLs, Oracle Files, Objects, Recent Sessions and Extended Collections.

For these entities, there are the following available views in the Table view:

- Overview

- In Oracle

- Overall Activity

- Scalability

- Availability

- Statistics

- RAC Databases (if a RAC database instance is selected)

In the Tree view, the following entity details are available:

- Highlights (currently only for the J2EE Caller Service entity)

- Over Time

- Overview

When selecting a category under the entity in the Tree view, the following details are available per category:

Table 5-1   Details per category

Category	Details
Aggregated entities	General, In Oracle, Statistics
Statements	General, In Oracle, Workshop
Objects	General, In Oracle, I/O Wait
Execution Plans	General, In Oracle
Oracle Files	General, In Oracle, I/O Wait
PL/SQLs	General, In Oracle, Workshop

About aggregated entities in the Activity tab

Overviews are available for the following entities:

■   All Instances

■   Actions

■   Siebel Business Objects

■   Databases

■   Instances

■   Programs

■   OA Forms

■   SAP Programs

■   Users

■   Host Users

■   Machines

■   Modules

■   OA Applications

■   Siebel Applications

■   SAP Functions

■   OA Requests

■   SAP Screens

■   Siebel Views

■   J2EE Callers

■   J2EE JVM Servers

■   Web Transactions

The Overview displays general information about the selected entity and a table listing the In Oracle sub-states for this entity.

See "In Oracle" on page 36.

The following table describes what is displayed in the overview.

Table 5-2   General entity information

Item	Description
Instance Number	Displays the number of the instance in the RAC database.
In Oracle (Sum.)	Total time the entity spent In Oracle.
Sessions	Total number of active sessions associated with the entity. An active session is one that executed or was executing a statement during the selected time frame and spent some time In Oracle.
In Oracle (Avg.)	Total In Oracle time for the entity or number of active sessions associated with the entity.
Duration (Avg.)	Average time active sessions associated with the entity were connected to Oracle.

The following table describes the In Oracle (Sum.) table.

Table 5-3   In Oracle (Sum.) table

Column	Description
Sub-State	In Oracle sub-state.
Bar graph	Graphical representation of the Time column.
Time	Amount of time the entity was in this state.
%	Percentage of time the entity was in this sub-state.

Additional information on In Oracle sub-states is available in the stacked bar graph of the In Oracle view and on the In Oracle tab of the table in the Association area.

About In Oracle data view

The In Oracle view displays a stacked bar graph on the In Oracle resource consumption over time, for the selected time frame, broken down into the In Oracle wait states.

See "In Oracle" on page 36.

About the overall activity view

The Overall Activity view displays a stacked bar graph on the Overall Activity resource consumption over time, for the selected time frame, broken down into the Overall Activity wait states.

*See "Overall activity" on page 35.

About the scalability view

The Scalability view displays stacked bar graphs that show how the average In Oracle response time varies with the number of sessions or the number of executions. These graphs indicate whether your application scales as the number of sessions or statements executed increases.

The following table explains which graphs are shown for a specific entity.

Table 5-4   Graphs displayed in the Scalability view

Entity	Graphs
Instance	In Oracle (Avg.) vs. Sessions
	In Oracle (Avg.) vs. Statement Executions
Program, User, Module, Action, Machine, Host User	In Oracle (Avg.) vs. Statement Executions

About the availability view

The Availability view is only available for the All Instances and Instance entities.

The following table describes the stacked overtime graphs that are displayed in this view.

Table 5-5 Graphs displayed in the Availability view

Graph	Description
Availability	Compares the overall availability with the Availability SLA, in percent. The settings for the Availability SLA are defined in AdminPoint. For more information, see the *Precise Administration Guide*.
	Availability information is also displayed on the Overview tab in the Dashboard tab.
Unavailability Times	Indicates the amount of time that the instance or instances were unavailable.

About the statistics view

The Statistics view displays a table that indicates a change in Oracle statistics for the selected entity during the specified time frame. The Statistics column displays statistics that represent different counters with different units. For more information, see the Oracle documentation.

See "Oracle statistics usage within Precise for Oracle" on page 40.

About the RAC database view

The RAC Database view compares activity of the selected instance to the activity of the all the instances in the entire RAC database. The RAC Database view is only available for RAC Database instances.

The following table describes the information displayed in the RAC Database view.

Table 5-6 RAC Database view

Item	Description
Database	Displays the logical name of the RAC database.
DB Name	Displays the Oracle database ID for the instance.
DB ID	Displays the Oracle database ID.
In Oracle (Sum.)	Total time the instance spent In Oracle.
Sessions	Total number of active sessions associated with the instance. An active session is one that executed or was executing a statement during the selected time frame and spent some time In Oracle.
Execution Counts	Number of executions.
In Oracle (Avg.)	Average In Oracle time for all instances in the RAC database.
Instances	Displays the number of instances that share the same RAC database with the selected instance.
In Oracle (Avg. per instance)	Average In Oracle time for the selected instance.
RAC Database table	Shows additional details on the instances in the RAC database, such as the number of sessions run by each instance, the amount of time each spent actively executing statements and PL/SQLs within Oracle, and the number of statements executed by each instance.

About Statement and PL/SQL entities

Statements and PL/SQL are the fundamental building blocks of an application. Application tuning will nearly always involve looking at the performance of the SQL. These entities share a common set of views in the Main area of the Activity tab.

Note: To tune a statement, you can launch the SQL tab by clicking the SQL tab button for a statement in the Main area or the Tune icon for a statement in the Association area.

You can drill down from a PL/SQL procedure to the SQL statements that it contains or from a statement to the PL/SQL procedure that it contains. You can also associate statements and PL/SQL with the following entities: Instances, Programs, Users, Modules, Actions, Machines (or their ERP equivalents), Host Users, Oracle Files, Objects, as well as File Systems, Logical Volumes, and UNIX devices (only if Precise for Storage, VERITAS version is installed).

You can also save a statement.

See "Editing the properties of a statement" on page 31.

The following views are available for a statement and PL/SQL entity:

■   Overview

■   In Oracle

■   Overall Activity

■   Scalability

■   Statistics

■   Text

■   Workshop (only for Statement)

About an overview of a statement or PL/SQL

The Overview displays general information about the selected entity in the Activity tab, a table that lists the In Oracle sub-states for this entity, and the complete SQL text of the SQL statement.

If bind variables were captured for a statement, a link is displayed that launches to the Bind Variables tab in the SQL tab, with the statement in context, to run a statement with the actual bind variables that are being used.

If you drill down to a statement from an execution plan entity, and execution plans were captured for the statement, a link is displayed that launches to the All Plans tab in the SQL tab, with the statement in context.

See "In Oracle" on page 36.

See "About the Bind Variables view" on page 148.

The following table explains how the general information is structured.

Table 5-7   Structure of general information for a statement or PL/SQL

Item	Description
In Oracle (Sum.)	Total time the statement or PL/SQL spent In Oracle.
Executions/Execution Counts	Number of times the statement or PL/SQL was executed. See "Executions vs. opens and fetches" on page 39.
In Oracle (Avg.)	For a statement only. Total In Oracle time for the statement or number of executions. If Opens + Fetches are displayed, this field displays N/A.
Duration (Avg.)	For a statement only. Total overall activity time for the statement or the number of executions.
Buffer Gets (Avg.)	For a statement only. Total number of buffer gets for the statement or the number of executions. If Opens + Fetches are displayed, this field displays N/A.
Rows Processed (Avg.)	For a statement only. Total number of rows processed by the statement or the number of executions. If Opens + Fetches are displayed, this field displays N/A.
Parallel Servers (Min.)	Minimum number of parallel query slaves that executed the statement.
Parallel Servers (Max.)	Maximum number of parallel query slaves that executed the statement.

Table 5-7 Structure of general information for a statement or PL/SQL

Item	Description
End of Fetch Count	Number of times the statement returned all rows to the user rather than returning only the first rows and dropping the cursor. "End of fetch count" will always be less than or equal to executions.
Version Count (Max.)	Shows the maximum number of cursors created for the particular statement in parallel

The following table describes the In Oracle (Sum.) table for a statement or PL/SQL.

Table 5-8 In Oracle (Sum.) table for a statement or PL/SQL

Column	Description
Sub-State	In Oracle sub-state.
Bar graph	Graphical representation of the Time column.
Time	Amount of time the entity was in this state.
%	Percentage of time the entity was in this sub-state.

Additional information on In Oracle sub-states is available in the stacked bar graph of the In Oracle view and on the In Oracle tab of the table in the Association area.

About In Oracle data for a statement or PL/SQL

The In Oracle view displays a stacked bar graph in the Activity tab on the In Oracle resource consumption over time, for the selected time frame, broken down into the In Oracle wait states.

See "In Oracle" on page 36.

About the overall activity of a statement or PL/SQL

The Overall Activity view, in the Activity tab, displays a stacked bar graph on the overall activity resource consumption over time, for the selected time frame, broken down into overall activity wait states.

See "Overall activity" on page 35.

About the scalability of a statement or PL/SQL

The Scalability view, in the Activity tab, displays a stacked bar graph showing how the average In Oracle response time varies with the number of executions. This graph indicates whether your application scales as the number of sessions or statements executed increases.

When you drill down from an instance or Tier, the In Oracle vs. Executions graph is shown. Otherwise, no graph is displayed.

About the statistics of a statement or PL/SQL

The Statistics view, in the Activity tab, displays a table indicating a change in Oracle statistics for the selected statement or SQL during the specified time frame. The Statistics column displays statistics representing different counters with different units. For more information, see the Oracle documentation.

See "Oracle statistics usage within Precise for Oracle" on page 40.

About the text view of a statement or PL/SQL

The Text view, in the Activity tab, displays the full SQL text.

About information on the cabinet or folder related to a statement

The Workshop view displays details on the cabinet or folder where the statement is saved, and on how it will be explained.

The following table describes how the information in the Workshop view is structured.

Table 5-9   Structure of information in the Workshop view

Item	Description
Parsing User	Oracle user name used to explain and parse the statement.
Cabinet	Name of the cabinet where the statement is stored.
Folder	Name of the folder where the statement is stored.
Origin	Source of the statement. The Collector captures most statements automatically when they execute. The following values are possible: ■ Unknown ■ Automatically collected ■ Generated as related SQL ■ Predicted plan (What-If) ■ Entered by user ■ Imported from source file ■ Generated as related SQL by user
Comment	User-defined comment associated with the statement.

See "Editing the properties of a statement" on page 31.

See "About the SQL tab" on page 131.

About the Execution Plan entity

The Oracle server generates execution plans at run-time. Precise for Oracle displays information on these plans so that you can analyze them and determine if any enhancements can be made to improve their execution.

You can drill down from an execution plan to the SQL statements that are using it. You can also associate execution plans with the following entities: Instances, Databases, Programs, Statements, PL/SQLs, other Execution Plans, Users, Modules, Actions, Machines (or their ERP equivalents), Host Users, Oracle Files, Objects, Recent Sessions and Extended Collections.

The following views are available for an execution plan entity:

■ Overview

■ In Oracle

■ Overall Activity

■ Scalability

About an overview of an execution plan

The Overview, in the Activity tab, displays general information about the selected entity, a table that lists the In Oracle sub-states for this entity, and the complete SQL text of the SQL statement.

See "In Oracle" on page 36.

The following table describes the information displayed for an execution plan.

Table 5-10   General over of an execution plan

Item	Description
In Oracle (Sum.)	Total time spent in Oracle by statements using the selected plan.

Table 5-10 General over of an execution plan

Item	Description
Execution Counts	Number of executions performed by the statements.
In Oracle (Avg.)	Displays the average time spent by the execution plan in the Oracle database.
Duration (Avg.)	Displays the average activity time for the execution plan.
Buffer Gets (Avg.)	Average number of buffer gets for the execution plan.
Rows Processed (Avg.)	Average number of rows processed by the execution plan.

The following table describes the In Oracle (Sum.) table for an execution plan.

Table 5-11 In Oracle (Sum.) for an execution plan

Column	Description
Sub-State	In Oracle sub-state.
Bar graph	Graphical representation of the Time column.
Time	Amount of time the entity was in this state.
%	Percentage of time the entity was in this sub-state.

Additional information on In Oracle sub-states is available in the stacked bar graph of the In Oracle view and on the In Oracle tab of the table in the Association area.

About In Oracle data for an execution plan

The In Oracle view, in the Activity tab, displays a stacked bar graph on the In Oracle resource consumption for the execution plan, over time, broken down into the In Oracle wait states.

See "In Oracle" on page 36.

About the overall activity view of an execution plan

The Overall Activity view, in the Activity tab, displays a stacked bar graph describing the overall activity resource consumption of the execution plan over time, broken down into overall activity wait states.

See "Overall activity" on page 35.

About the scalability view of an execution plan

The Scalability view, in the Activity tab, displays a stacked bar graph showing how the average In Oracle response time varies with the number of statement executions. This graph indicates whether your application scales as the number of statement executions increase.

When you drill down from an instance or Tier, the In Oracle (Avg.) vs. Statement Executions graph is shown. Otherwise, no graph is displayed.

About the Object entity

Whereas statements and PL/SQL are the fundamental building blocks of an application, the objects are the fundamental building blocks of a database. This view gives a DBA an opportunity to tune the objects in the database and is a far more powerful and less intrusive approach than looking at, for example, the V$SEGMENT_STATISTICS table. It is particularly useful if the application code cannot be changed, which is often the case when application packages have been purchased.

The Activity tab shows resource consumption information for objects that are accessed by SQL statements. These are the objects that can be seen in explain plans, namely tables, indexes, and clusters, plus some special objects

representing Control file I/O, Log File I/O, Temporary I/O, and Undo I/O. Wait time on other types of objects is not displayed. For example, the Oracle Optimizer will expand a synonym to its underlying table name during the parse phase before the explain plan is created.

You can also view how many partitions an object has in the Association area.

Note: To look at the selected object in detail, or to understand its performance over time, you can launch the Objects tab by clicking the Object tab button for an object in the Main area or the launch icon for an object in the Association area.

An Object can associate with Instances, other Objects, Oracle Files, Statements, PL/SQLs, Programs, Users, Modules, Actions, Machines (or their ERP equivalents), and Host Users.

For an Object, the following views are available:

- Overview
- In Oracle
- Dictionary

About the overview of an object

The Overview, in the Activity tab, displays a table that lists the I/O Wait sub-states for this object.

Precise for Oracle records the following sub-states:

- Sequential I/O Wait
- Scattered I/O Wait
- Direct I/O Wait
- Other I/O Wait
- Buffer Wait
- Table Lock Wait
- Row Lock Wait

Additional information on In Oracle sub-states is available in the stacked bar graph of the In Oracle view and on the In Oracle tab of the table in the Association area.

See "I/O wait" on page 38.

About In Oracle data for an object

The In Oracle view, in the Activity tab, displays a stacked bar graph on the In Oracle resource consumption over time, for the selected time frame, broken down into the In Oracle wait states. Particularly important for an object is the amount of time spent in I/O wait, Internal Lock wait, and Application Lock wait, that is how well cached the object is.

See "In Oracle" on page 36.

About the dictionary view of an object

The Dictionary view, in the Activity tab, describes the properties of the object. The information displayed is equivalent to information that can be found in the DBA_OBJECTS table.

About the Oracle File entity

Precise for Oracle shows resource consumption information for those Oracle files where the application spends time waiting directly on I/O. These are most likely to be data and temporary files. I/O wait time on other types of files such as redo log files are less likely because Oracle writes to these asynchronously using a background job such as LGWR. This information can be found in the Statistics tab.

Note: To look at the selected Oracle file in detail, or understand its performance over time, you can launch the Objects tab by clicking the Objects tab button for an Oracle file in the Main area or the launch icon for an Oracle file in the Association area.

An Oracle File can associate with Objects, Oracle Files, Statements, PL/SQLs, and Storage Devices (only if Precise for Storage is installed).

For an Oracle File, the following views are available in the Main Area:

- Overview

- In Oracle

- Dictionary

About an overview of an Oracle file

The Overview, in the Activity tab, displays a table listing the I/O Wait sub-states for this Oracle file. Additional information on In Oracle sub-states is available in the stacked bar graph of the In Oracle view and on the In Oracle tab of the table in the Association area.

See "I/O wait" on page 38.

About the In Oracle view of an Oracle file

The In Oracle view, in the Activity tab, displays a stacked overtime bar graph on the I/O waits of the Oracle file.

About the dictionary view of an Oracle file

The Dictionary view, in the Activity tab, describes the properties of the Oracle file. The information displayed is equivalent to information that can be found in the DBA_DATA_FILES table.

About the Extended Collection entity

You can define an Extended Collection when you need to carry out a more intensive investigation into a particular performance problem. It lets you capture trace-level information from Oracle, allowing you to see Oracle wait event information for sessions and statements. You can, for example, find a unique index contention or ITL contentions.

The wait event information is similar to that which you can get from Oracle's SQL_TRACE table or event 10046 (for example, alter session set events 10046 trace name context forever, level 8) but with much less overhead. The Precise for Oracle Collector automatically captures all the wait events when it is sampling, but it does not normally store the full details. Rather it stores a summary to keep down the storage requirements. When an Extended Collection is enabled, the full wait event data, including wait event parameters, is stored in a trace file on the monitored server. It is not loaded into the PMDB. Wait event parameters allow you to precisely identify the wait cause, such as the exact block in the Oracle file that has caused a db file sequential read wait event.

You can create an Extended Collection by using the Define Extended Collection dialog box.

When an Extended Collection is no longer required, it should be deleted using either the icon in the Association area, or the action control button when in the Main area. Similarly, pending and running Extended Collections can be stopped, and subsequently deleted.

Note: To see Extended Selections listed in the Association controls, you need to have an Instance selected. When you drill down to an Extended Collection, the time frame in the Time Frame list and all graphs change to the start and end time of the collection.

The information displayed is the same as you would see for the Instance over the same time period filtered by the same criteria. However, in addition to this, an Extended Collection is the container for the Sessions, Statements, and Oracle Events that are captured.

For an Extended Collection, the following views are available in the Main Area:

- Overview

- In Oracle

- Overall Activity

See "Defining extended collections" on page 80.

See "Filtering data" on page 29.

About the overview of an extended collection

The Overview, in the Activity tab, displays general information about the Extended Collection for the specified time frame and a table that lists the In Oracle sub-states for this entity.

See "In Oracle" on page 36.

The following table describes how the general information for an extended collection is structured.

Table 5-12 Structure of general information for an extended collection

Item	Description
Status	Status of the collection, which can be one of the following: Pending, Completed, Running, or Stopped.
Criteria	List of programs and other criteria that are included or excluded in the collection.
Current Size	Size of the collection. The size is limited when the collection is created. See "Defining extended collections" on page 80.
Start Collecting at	Start date and time of the collection.
Stop Collecting at	End date and time of the collection.

The following table describes the In Oracle (Sum.) table for an extended collection.

Table 5-13 In Oracle (Sum.) table for an extended collection

Column	Description
Sub-State	In Oracle sub-state.
Bar graph	Graphical representation of the Time column.
Time	Amount of time the entity was in this state.
%	Percentage of time the entity was in this sub-state.

Additional information on In Oracle sub-states is available in the stacked bar graph of the In Oracle view and on the In Oracle tab of the table in the Association area.

About In Oracle data for an extended collection

The In Oracle view, in the Activity tab, displays a stacked bar graph on the In Oracle resource consumption over time, for the selected time frame, broken down into the In Oracle wait states.

See "In Oracle" on page 36.

About the overall activity view of an extended collection

The Overall Activity view, in the Activity tab, displays a stacked bar graph on the Overall Activity resource consumption over time, for the selected time frame, broken down into the Overall Activity wait states.

See "Overall activity" on page 35.

About Session and Recent Session entities (in Table view only)

Precise for Oracle captures a detailed history of every session but does not load this information into the PMDB because it quickly becomes out of date and because of the storage requirements. Instead, this information is saved on the monitored server for only as long as is required. This is accomplished using two mechanisms: Extended Collections and Recent Sessions.

When an Extended Collection is defined, the session information is captured and saved in the Extended Collection files. You can then drill down from an Extended Collection to a Session.

In addition, Precise for Oracle keeps the last n hours of session information on the server and makes it available using Recent Sessions. By default, information for the last 4 hours is saved. You can specify and change the length of time using Precise Agent Installer (see the *Precise Installation Guide* for details), but increasing it will obviously require more storage.

Note: Because session information is only saved for a limited amount of time (by default, 4 hours), if you drill down to recent sessions, you will only see session information for that period, regardless of the time frame specified in the Time Frame list.

A Recent Session can associate with Statements.

For a Session (in an Extended Collection) or Recent Session, the following views are available in the Main Area:

- Overview

- In Oracle

- Overall Activity

See "About the Extended Collection entity" on page 75.

About the overview of a session or recent session

The Overview, in the Activity tab, displays general information about the Session (in an Extended Collection) or Recent Session for the specified time frame and a table listing the In Oracle sub-states for this entity (for details on these sub-states, see See "In Oracle" on page 36.).

The following table describes how the general information for a session or recent session is structured.

Table 5-14   Structure of general information for a session or recent session

Item	Description
In Oracle	Total time the session spent in Oracle
User	Oracle user name
Host User	Host user name.
Machine	Name of the machine connected to Oracle
Oracle Process ID	OS-assigned process ID of the Oracle server process
Client Process ID	Process ID of the client or application process, if it is running on the same server as the instance; otherwise it is zero.
Oracle Process Type	Type of Oracle server process: Dedicated, MTS, or Parallel.

The following table describes the In Oracle (Sum.) table for an extended collection.

Table 5-15   In Oracle (Sum.) table for a session or recent session

Column	Description
Sub-State	In Oracle sub-state.
Bar graph	Graphical representation of the Time column.

Table 5-15 In Oracle (Sum.) table for a session or recent session

Column	Description
Time	Amount of time the entity was in this state.
%	Percentage of time the entity was in this sub-state.

Additional information on In Oracle sub-states is available in the stacked bar graph of the In Oracle view and on the In Oracle tab of the table in the Association area.

About In Oracle data for a session or recent session

The In Oracle view, in the Activity tab, displays a stacked bar graph on the In Oracle resource consumption over time, for the selected time frame, broken down into the In Oracle wait states.

See "In Oracle" on page 36.

About the overall activity of a session or recent session

The Overall Activity view, in the Activity tab, displays a stacked bar graph on the Overall Activity resource consumption over time, for the selected time frame, broken down into the Overall Activity wait states.

See "Overall activity" on page 35.

About the Imported Statement entity

Statements can be imported from external source files into Precise for Oracle and stored in the cabinet and folder hierarchy. To display them in the Activity tab, use the More... option in the Association area and click Imported Statements.

Note: To tune a statement, you can launch the SQL tab by clicking the SQL tab button for a statement in the Main area or the Tune icon for a statement in the Association area.

You cannot drill down from an imported statement to other entities.

For an Imported Statement, the following views are available in the Main Area:

■ Text

■ Workshop

About the text view of an imported statement

The Text view, in the Activity tab, displays the full SQL text of the statement.

About information on the cabinet or folder related to an imported statement

The Workshop view, in the Activity tab, displays details on the cabinet or folder where the statement or PL/SQL is saved, and on how it will be explained.

The following table describes how the information in the Workshop view is structured.

Table 5-16 Structure of information

Item	Description
Explained on	The date and time of the last explain.
Parsing User	Oracle user name used to explain and parse the statement.
Cabinet	Name of the cabinet where the statement is stored.
Folder	Name of the folder where the statement is stored.

Table 5-16 Structure of information

Item	Description
Origin	Source of the statement. In this case, it is always Imported from source file.
Cost	The cost of the imported statement.
Saved on	Information on when the imported statement was saved.
Statement	The name of the imported statement.
Oracle Hash Value	The Oracle Hash value of the imported statement.
Comment	User-defined comment associated with the statement. This is blank until the statement has been saved.

See "Editing the properties of a statement" on page 31.

See "About the SQL tab" on page 131.

About storage device entities

In the Activity tab, there are the following Storage device entities:

■ EMC storage device

■ HP storage device

Note: These entities are only available if you have installed Precise for Storage. For more information on the supported platforms, see the *Precise Installation Guide.*

The Activity tab offers a unique method for identifying I/O problems in EMC and HP storage devices. It reports I/O wait time instead of just I/O counters. This lets you easily identify the resource that is actually being waited for and tune it instead of guessing by rules of thumb. Furthermore, you can associate the storage device with the applications and statements that are waiting on I/O.

See "Precise for Storage, Oracle version - how resource consumption of storage devices is examined" on page 41.

Note: To look at a storage device in detail or to understand its performance over time, you can launch the Object tab by clicking the Object tab button for a storage device in the Main area or the launch icon for a storage device in the Association area.

When an HP Storage Device is selected, the Launch HP PA button appears to the right of the Define Extended Collection button. This option launches the Web-based user interface of the HP Performance Advisor in another Internet Explorer window.

Storage devices can associate with Objects, Oracle Files, Statements, and PL/SQLs.

For a storage device, the following views are available in the Main Area:

■ Overview

■ I/O Wait

■ Dictionary

About the overview of a storage device

The Overview, in the Activity tab, displays a table listing the In Oracle sub-states for this storage device.

See "In Oracle" on page 36.

The following table describes the In Oracle (Sum.) table for an extended collection.

Table 5-17   In Oracle (Sum.) table for a storage device

Column	Description
Sub-State	In Oracle sub-state.
Bar graph	Graphical representation of the Time column.
Time	Amount of time the entity was in this state.
%	Percentage of time the entity was in this sub-state.

Additional information on In Oracle sub-states is available in the stacked bar graph of the In Oracle view and on the In Oracle tab of the table in the Association area.

About the I/O wait view of a storage device

The I/O Wait view, in the Activity tab, displays a stacked overtime bar graph on the I/O waits of the storage device.

About the dictionary of a storage device

This view, in the Activity tab, describes the properties of the storage device. Note that despite the name of the view, these are properties not derived from the Oracle data dictionary.

About the Findings area in the Activity tab

Note: This is only relevant for the Tree view.

The Findings area in the Activity tab displays performance problems detected in the selected J2EE Caller Service or Web Transactions. The findings list is displayed according to severity (highest to lowest).

To view the details of a specific finding, hover over the finding name. The area for the selected finding will expand, displaying information and recommended actions. Click on the relevant link to launch the appropriate tab in context, or **Learn more** to open the help for that finding.

How to perform actions in the Activity tab

The Tuning Action buttons in the Activity tab enable you to perform the following actions:

■   Defining extended collections

■   Importing statements

■   Stopping extended collections

See "Editing the properties of a statement" on page 31.

Defining extended collections

The Define Extended Collection button opens the Define Extended Collection dialog box, allowing you to define a period where more detailed Oracle Performance information is collected.

Using the Define Extended Collection dialog box, you can specify the instance that you want to monitor, the name of your collection, the start and stop time, and the maximum size of the trace. You then use the entity table to define the entities that you want to include or exclude. In the Include and Exclude fields, you can use the Oracle SQL-style wildcards '-' and '%' to match zero or more characters and '_' to match a single character. The AND operator is used to produce the final selection.

Note: Try to only include the entities that you need and do not run too many extended collections simultaneously. Although the overhead is still much less than for SQL_TRACE, there can be a slight performance penalty in having to write a lot of information to the traces file on the monitored server.

See "About the Extended Collection entity" on page 75.

To define extended collections

1   Click the **Define Extended Collection** button to open the Define Extended Collection dialog box.

2   In the Instance list, select the appropriate instance name.

3   If appropriate, type in the Collection name.

4   For Start At date and time, select the following data:

   ■   day (number)

   ■   month (name)

   ■   year (number)

   ■   hour (number)

   ■   minute (number)

5   Type the desired maximum number of megabytes for collection information.

6   Define the criteria for activities to be gathered in the collection. For each entity listed in the Entity column, define whether to include or exclude the entity to the collection.

7   Click **OK** to accept or **Cancel** to reject the defined entities.

Importing statements

The Import Statement button opens the Import Statements from File dialog box, allowing you to import SQL statements from a file. This dialog box includes three tabs: Source, Catalog, and Options.

See "Editing the properties of a statement" on page 31.

See "About the Imported Statement entity" on page 78.

Defining information on the Source tab

On the Source tab, you can specify the name of the instance and user that will own the statement and the full path to the file containing the SQL. Wild cards in the path are not supported.

If duplicates are detected during the import process, the newer version of a statement will not over-ride the older version.

To define Source tab details

1   In the Instance list, select the appropriate instance.

2   In the User list, select the appropriate user name.

3   In the Path field, you can type in directly the path or click the . . . (browser) to locate and select the path.

4   Check the box, if appropriate, to view Explain imported statements.

5   Click **OK** to accept or **Cancel** to reject the Source tab information.

Defining information on the Catalog tab

On the Catalog tab, you can specify the destination cabinet and folder. Either specify the name of the cabinet or folder explicitly by selecting Use the cabinet/folder, or specify a naming convention based on either the name or full path of the file by selecting Use an automatically created cabinet/folder. By default, the statement is saved into the predefined cabinet or folder called Default.

To define Catalog tab details

1 Use the upper two lists for a cabinet to select an available parameter.

2 Use the lower two lists for a folder to select an available parameter.

3 Click **OK** to accept or **Cancel** to reject the Source tab information.

Defining information on the Options tab

On the Options tab, you can control the import process and the way that statements are found. You can filter statements in the following ways:

■ Ignore strings in either single or double quotes during the import and replace them with bind variables.

■ Ignore comments as defined in various programming languages.

■ Convert names in respect of hyphens and periods.

■ Control the sort of SQL statements imported.

■ Ignore names that are likely to refer to system objects.

■ Perform a semantic check on the statement by looking up referenced objects in the data dictionary to ensure that the named objects exist.

To define Options tab details

1 Check the appropriate box(es) if you want to ignore strings enclosed by single or double quotes.

2 Check the appropriate box(es) if you want to ignore specific types of comments.

3 Check the appropriate box(es) if you want to convert hyphens or periods to underscores.

4 Check the appropriate box(es) regarding statement parameters (select, insert, update, or delete).

5 Check the appropriate box if you want to ignore specific statements.

6 Check the appropriate box if you want to scan the data dictionary for a column name.

7 Click **OK** to accept or **Cancel** to reject the Source tab information.

Stopping extended collections

There are two possible actions you can take to stop extended collections.

To stop extended collections

1 When you have selected an Extended Collection, click **Stop** to stop this collection if it has already started but has not completed yet.

2 As an alternative, when a list of Extended Collections is displayed, you can click the appropriate action icon in the Association area.

How the Activity tab can help you identify performance problems using the Table view

You can identify a performance problem by doing one or more of the following:

■ Examining resource consumption of an entire instance

■ Identifying heavy resource consumers

■ Examining resource consumption over time

■ Identifying offending SQL statements

■ Correlating Oracle data with ERP data

■ Examining the scalability of your application

■ Examining EMC Storage devices

Examining resource consumption of an entire instance

When performing a tuning audit, it is very important to analyze and understand the performance behavior of your instances. You may have already drilled down to analyze the performance of a single query or program. However, examining the entire instance's behavior can alert you to the health of your system. The instance represents the performance of the average application. This view will help you answer questions such as: This query suffers from lock wait, but does the entire system suffer from lock wait? If, for example, you discover that your system suffers from I/O wait and you tune a query that performs memory operations, you most likely will not improve the entire system's I/O wait problem. So examining the entire instance can provide a quick overview of the dominant resources consumed.

The following example shows how to examine resource consumption of an instance.

To examine resource consumption of an entire instance

1   In the Instance list, choose the item you want to analyze.

2   In the Time Frame list, choose the period of time you want to analyze.

3   On the View controls in the Main area, click **Overview.** Examine the entire instance and determine which are the dominant resources that are consumed by your system. For our example, we have an Activity tab that shows an instance that is fairly busy. In the Main area, the instance shows that it has spent 1 day out of 7 executing. But it has spent 40% of that time on Other Lock Wait. This is an unproductive wait.

4   Launch to the Statistics tab to find out more about the underlying Oracle wait events comprising the Other Lock Wait. It is most likely to be an enqueue. In fact, the instance has spent over 50% of its time in unproductive waits (not using CPU or I/O).

Note: As a rule of thumb, a healthy system should have a high Using CPU value; a 10-15% I/O Wait, and the remaining non-productive wait states should show values that are as low as possible. For databases that process large data volumes, such as a data warehouse, the I/O Wait percentage may be higher. It is true, however, that applications/statements that experience high Using CPU or I/O Wait times may be performing excessive logical I/Os and should be investigated. To determine this, it is important to look at the actual time spent waiting (in hh:mm:ss) as well as just the percentage.

Identifying heavy resource consumers

Precise for Oracle lets you drill down to application components (such as Programs, Users, Host Users, Machines, Statements, and so on) to determine which application component consumes the most resources. To do so, click `Programs` from the Association controls to show a breakdown of the Oracle activity by program. You can now click any of the programs to see just the activity for that program and the entities that contributed to that activity. This process is iterative and you may continue to drill down until you discover the application component that you want to tune.

There are several reasons to drill down and focus on different application components:

■   During the tuning process, you drill down to determine which are the top resource-consuming components. Tuning a statement (User, Program, Machine, and so on) that consumes most of the resources will free them for other entities. The tuning process is an iterative process. Continue to list all components, examine their resource consumption, and focus on several of them until you have completed the tuning process.

■   When you try to analyze user experience, you need to isolate one user's activity from the other users. To achieve this, you can focus on a single User or Machine.

Note: When trying to determine which application component to drill down to, take into consideration both the total In Oracle time of the application and its number of Sessions or Executions and Duration (Avg.). Otherwise, you can try to tune a program that runs once a year, consuming 20 hours, instead of tuning a program that consumes 2 seconds but runs every time a user logs on to the system.

You can display the average duration for statements with the format shown as hh:mm:ss.

The following example shows how to drill down to application components and identify resource consumers. For our example, the Active tab shows a list of programs that you can drill down to in the Tree pane. We want to drill down to the program that spent the most time.

To identify heavy resource consumers

1   In the Instance list, choose the item you want to analyze.

2   In the Time Frame list, choose the period of time you want to analyze.

3   On the Association controls in the Association area, click **Programs**. This shows a breakdown of the Oracle activity by program.

4   You can now click any of the programs to see just the activity for that program and the entities that contributed to that activity.

5   Continue to drill down until you discover the application component that you want to tune.

Examining resource consumption over time

When you analyze and tune the performance of an application component (such as Program or Statement), it is important to consider its performance over time. Precise for Oracle lets you view the component's performance over a selected time frame.

■   There are several reasons to study a component's performance over time:

■   When analyzing a performance problem, you will want to determine if it is a random problem or has happened before.

■   To look for a pattern. For example, does the lock problem always happen on Tuesday at around 10 AM?

■   You will want to determine how your system is behaving throughout an entire day. For example you may observe that during the night, your system experiences much more I/O wait than during the day.

The following example shows how to examine resource consumption over time.

To examine resource consumption over time

1   In the Instance list, choose the item you want to analyze.

2   In the Time Frame list, choose the period of time you want to analyze.

3   In the table, click a program to drill down to. The tab now shows the result of clicking a program. The Main area shows the In Oracle (Sum.) overtime graph of the activity for the selected program. For an example, the overtime graph may show the selected program to run only during the working day, but is performing bulk inserts.

Although this particular example is performing a simple load test, in a production application you may want to consider running such statements outside of working hours.

Identifying offending SQL statements

Having identified a major resource consuming entity, you can further drill down to the statements and PL/SQL executed by that entity. These may include long running SQL statements and SQL statements that use few resources but were executed frequently.

See "Identifying heavy resource consumers" on page 83.

In many cases, analyzing the queries executed by the entity and tuning them will result in the most performance gain.

 For each statement, you can do the following:

■   View which entities executed the statement.

■   View resources consumed by the statement, which lets you determine if the statement is CPU or I/O bound, waiting for a lock, and so on.

■   View resource consumption patterns, which indicate if the statement execution is affected by excessive use of the same resource by another statement.

- View access plan information and display summary information on the operations used in the access plan. This is available only for Statements.

- Tune the statement by selecting the SQL tab or the Tune icon.

The following example shows how to identify offending SQL statements. For our example, we have identified a major resource consuming entity and have drilled down to the statements and PL/SQL executed by that entity.

To identify offending SQL statements

1   In the Instance list, choose the item you want to analyze.

2   In the Time Frame list, choose the period of time you want to analyze.

3   In the Association area, click the **General** tab so as to view the SQL text, number of executions, average In Oracle time, and the percentage contribution to the overall In Oracle time. In the Main area, the bar graph shows a breakdown of the In Oracle time so that you can easily see if a statement is CPU or I/O bound, or waiting on other things.

4   In the Association area, click the **Statistics** tab to view Oracle statistics for each statement. For example, you can see the number of Logical Reads, which is the total of the Consistent Gets + DB Block Gets. Similarly, you can see the number of Physical Reads, which is the number of reads that involved a physical I/O from disk because the data was not available in the Oracle Buffer Cache.

5   The tab shows that the number of logical and physical reads is high and contributing to the I/O wait time. In addition, it indicates the number of Sorts and how many were performed in memory compared to disk. Disk sorts take longer than sorts in memory, so attempts should be made to avoid them where possible by sizing the Oracle sort areas settings appropriately and tuning statements that do unnecessary sorting. For our example, the number of sorts is low and there is little sorting to disk.

Correlating Oracle data with ERP data

Installing the Precise for Oracle Interpoint extension lets you correlate Oracle information with ERP information. For example, if you install Interpoint for SAP, you will be able to display the breakdown of your Oracle activity into SAP Work Types.

The following example shows how to correlate Oracle data with ERP data.

To correlate Oracle data with ERP data

1   In the Instance list, choose the item you want to analyze.

2   In the Time Frame list, choose the period of time you want to analyze.

3   In the Activity tab, the overtime graph indicates that the major bottleneck is the I/O wait. However, in this 24 hour period, nearly all the I/O wait time arises from batch jobs (SAP Background).

4   Check the interactive sessions (SAP Dialog). Interactive sessions (SAP Dialog) are not waiting much on I/O. If the emphasis in this application is on tuning interactive sessions, then despite first appearances, tuning the I/O is not going to make an appreciable difference.

5   Drill down to the entity whose data you want to correlate. For our example, drill down to SAP Background to view the transactions and statements, to investigate why batch jobs are taking so long.

Examining the scalability of your application

Applications are called upon to support the addition of more and more users, and data, over the years. One of today's tasks is to ensure that our applications and servers are scalable for tomorrow's tasks.

The following example shows how to examine scalability of your application. In the Activity tab, an application runs a particular statement over 10 million times during a selected time frame.

To examine scalability of your application

1   In the Instance list, choose the item you want to analyze.

2   In the Time Frame list, choose the period of time you want to analyze.

3 On the View controls in the Main area, click **Scalability**. The Scalability view shows that the number of Executions has remained fairly constant.

4 Check the In Oracle time. For our example, the In Oracle time for the statement is now much lower (towards the right-hand side of the graph), after some tuning has taken place. In other words, it should be possible to increase the number of executions and hence the throughput of the application by a factor of 3 for the same resource consumption.

Examining EMC Storage devices

Precise for Storage, Oracle version, supports EMC devices and lets you see the resource consumption of the EMC Storage devices from the application's point of view.

The following example shows how to examine EMC Storage devices resource consumption.

To examine EMC Storage devices

1 In the Instance list, choose the item you want to analyze.

2 In the Time Frame list, choose the period of time you want to analyze.

3 For our example, the Association area in the Activity tab shows the total I/O wait on each of the devices used by instance ora920 in just over one hour, broken down into scattered and sequential I/O types.

4 Check the I/O wait on the devices. The I/O wait on devices, for our example, 0E1 and 0E0 is significantly higher than for 0DE and 0DF. These are the busy devices as far as this instance is concerned.

5 Drill down into device 0E1 to see its resource consumption over time. Find out which statements have been waiting on it.

Examining the relationship between database objects

This section includes the following topics:

- About the Objects tab
- How the Objects tab is structured
- About Objects tab entities
- How the Objects tab can help you identify performance problems

About the Objects tab

It is important to understand the relationship between database objects and statements to make sound and well-founded tuning decisions.

The Objects tab is a comprehensive browser that lets you understand the relationships and associations between database schema objects. It also provides further understanding into the relationships between data dictionary components and the statements that access them. This information lets you identify the statements that will be affected by changes to a database object, and the schema relationship impacts that are associated with an object change. The Objects tab also lets you observe storage utilization, over time and at varying levels, such as double-space, table, index, and Oracle file.

While the Activity tab focuses on SQL tuning, the Object tab focuses on database object tuning. In many cases, you may want to tune a database object rather than a specific query or program. This is especially true if you did not personally write the application in question, as for example, in ERP and CRM applications, and cannot therefore change the text of any statement.

Object tuning is a powerful tool. Taking into consideration important aspects regarding objects can ensure maximum application performance. Precise for Oracle lets you examine your index structure by displaying the usage of each index. Storage layout can be optimized by analyzing the object storage configuration and metric information.

Because Precise for Oracle tracks changes in the database objects, the effect of schema changes on performance can be easily observed.

Note: This section provides a general description of how to use the Objects tab in your analysis of Oracle objects. For further information on a specific Oracle object, see Oracle's relevant documentation.

How the Objects tab is structured

The Objects tab consists of two different views: Explore Objects and Tune Objects.

The Explore Objects view displays information on a selected entity and its associated entities. The selected parent entity is always reflected in the Tab heading, which serves as a point of orientation. The highest-level entity you can view information for in the Objects tab is Instance. You can choose another instance from the Instance selector.

The Tune Objects view displays performance and usage information per table or index, based on new and existing data sources.

About the Explore Objects view

When you open the Objects tab from the Activity tab (and the selected entity is Statement, Table, Index, Oracle File or Storage Device) or the SQL tab (Table, Index or Column) you launch to the tab in context, and information on the entity you selected is displayed. In all other circumstances the settings from your last visit to the Objects tab are retained. This is similar to clicking the History button and returning to a previous tab.

Although a large number of entities can be displayed in the Objects tab, they nearly all share the same set of Main Area views and Associations.

About the Main area in the Explore Objects view

The Main area, in the Objects tab, shows comprehensive information on the selected entity. You can choose from several views to examine the entity from different angles. For example, you can focus exclusively on Instance Parameters, or Instance Parameters Changes.

The following table explains the standard views that are available for most entities in the Objects tab.

Table 6-1 Standard views in the Objects tab

View	Description
Overview	Usually the first view that is displayed. The Overview shows general properties of the entity that can be found in the Oracle data dictionary, such as DBA_OBJECTS, and DBA_TABLES, and also provides graphs that can assist you in analyzing changes in objects.
RAC Database	RAC allows multiple instances on different nodes to access a shared database on a cluster system. This allows multiple servers to share the load. The RAC Database view is only displayed for RAC instances.
Storage	This view displays current storage information for objects such as information on extents.
Changes	The changes displayed by this view are represented differently depending on whether they occurred on the instance level or on the table or index level, as follows:
	■ Precise for Oracle FocalPoint periodically scans for changes in the instance that have been made to components, initialization parameters, redo log parameters, rollback segments, tablespaces, datafiles, and schemas. This can be valuable information when you are trying to determine why performance degradation has occurred. These changes are captured by the Collect Instance Definitions process. The scan typically occurs once a week (this interval can be configured in AdminPoint; see the *Precise Administration Guide* for details). At least two snapshots must have been captured before Precise for Oracle can detect whether a change has occurred—the first to establish the baseline and the second to calculate the differences.
	■ Changes that occur on the table or instance level are captured in the same way as other changes but are represented in accordance with their equivalent DDL (for example, DROP INDEX or ALTER TABLE). These changes are captured by the Collect Schema Changes process. The scan typically occurs once a week (this interval can be configured in AdminPoint; see the *Precise Administration Guide* for details). This process is disabled by default.
	Note: The information displayed in the Changes view is not the same as an audit trail. Precise for Oracle only tells you that the change occurred somewhere between the two snapshots and does not tell you who or what was responsible for the change. If a change is made and then restored to its original value before the next sample, it will not be detected. Similarly, if a number of consecutive changes of the same type are made to an object between samples, only the overall change will be recorded.
Statistics	This view displays statistics generated by the Oracle statistics collection mechanism (for example, distinct values for a column, or number of rows for a table). If you are using the COST or CHOOSE database optimizer modes, it is important that the statistics are accurate and up-to-date.

About the Association area in the Explore Objects view

The Association area, in the Objects tab, provides corresponding information on the entities that are related to or can be associated with the selected entity (the entity displayed in the Tab heading). You can view information on one type of associated entity at a time, such as Statements or Tablespaces, by selecting an entity from the Association Controls list. The selection you make is reflected in the Association area only; the Main area remains unchanged. For some entities, tabs above the Association area enable you to view additional information. Clicking a tab changes the columns of the table displayed for the associated entities.

From the Association area, you can also drill down to another entity by clicking on a table row. A drill down affects the entire tab. When you drill down to another entity, the Tab heading reflects the new selection, the Main area displays information on the newly selected entity, and the Association area shows data on the entities that are associated with the selected entity.

For example, when you want to view information on a specific Tablespace, from the Association controls, click `Tablespaces`. The Association area changes to display tablespace-related information. Note that the Tab heading and the Main area remain unchanged. Click the row of the tablespace you want to view detailed information for. The Tab heading indicates the newly selected entity; the Main area displays an overview of the tablespace you drilled down to, and the Association area shows a list of Oracle files that belong to the tablespace and their properties.

The following table explains the standard tabs that are available for most entities in the Association area.

Table 6-2   Standard tabs in the Association area of the Objects tab

Tab	Description
Dictionary	Displays Oracle data dictionary information that can be found in DBA_OBJECTS, DBA_TABLES, etc. This information is then generally displayed in the Overview in the Main area when the entity is selected.
Storage	Displays extent information similar to that found in DBA_SEGMENTS. This information is then generally displayed in the Storage view in the Main area when the entity is selected.
Statistics	Displays some statistics generated by Oracle statistics collection mechanism (for example, distinct values for a column, or number of rows for a table). If you are using the COST or CHOOSE database optimizer modes, it is important that the statistics are accurate and up-to-date. You may want to sort by the Last Analyzed On date or use the Filter dialog to search for entities whose Last Analyzed On date is before a certain date.
Usage	Displays a breakdown of the different ways in which an object such as a table, index or column has been accessed within explain plans—for example the number of full table scans or index range scans. This information is only available for statements that have been explained. It can help you determine if your indexing strategy is working effectively. You may want to sort tables by the number of Full Scans or use the More feature to find tables that have been Accessed via Full Scan and have a large number of rows or blocks.
	The Access columns are specific to a type of object but they all share a References column. This represents the total of all individual usages, and provides an overall view of the usage of a particular object.

About the Tune Objects view

You can launch directly to the Tune Objects view of the Objects tab from the following tabs:

Table 6-3   Direct launch

Tab	Description
Dashboard	From the launch icon for object findings.

Table 6-3   Direct launch

Tab	Description
Activity	From the Association area, by clicking on the launch entity, or when clicking on the Objects workstation button when an Object entity is displayed in the Main area.
Objects	From the launch icon for an active object.
SQL	From the launch icon in the Objects tab, when tables and indexes are selected.

The Main area of the Tune Objects view displays information on the table you drilled down to and its indexes.

You can also directly open a table by using the Open option.

Selecting a table to tune

You can select a table to tune by using the Open option.

To open a table for tuning

1   On the Tune Objects tab in the Objects tab, click **Open**.

2   In the Open Table dialog box, in the Instance drop-down list, select the instance related to the table you want to tune.

3   In the Owner drop-down list, select the owner of the table you want to tune.

4   In the Table drop-down list, select the name of the table you want to tune.

5   Click **Ok**.

About the Main area in the Tune Objects view

The segment table, located in the upper left corner of the Main area of the Objects tab, displays information on the selected table and its corresponding indexes. Additional tabs (Details, Partitions, and Columns), located in the lower left corner, provide additional information on the selected table. You can view this information by clicking the + sign to expand the view.

The following table provides an explanation of the information displayed in the Segments table.

Table 6-4   Segments table

Column	Description
Segment icon	Icon indicating whether a segment is a table or an index.
Name	Displays the name of the segment.
In Oracle (Sum.)	Displays a breakdown of the total waits that contributed to the In Oracle time for the object.
Logical Reads	Summarizes the logical reads made on the object.
Physical I/Os	Summarizes the direct physical reads, direct physical writes, physical reads, and physical writes made on the object.
Statements	Displays the number of statements that accessed the table within the selected time frame.
Hit Ratio	Displays the cache hit ration of the selected object.
Index Columns	When the selected object is an index, displays the names of the index columns.

The Details area displays relevant data dictionary information for the selected index, table or segments.

The Columns area shows information on the columns of the selected tables or index.

The following table describes the information displayed in the Dictionary and Columns areas.

Table 6-5   Information displayed in the Dictionary and Columns area

Selection	Dictionary	Columns
All Segments	Rows, Degree, Last analyzed on, Chained rows	Column, Type, Distinct values, Nullable, Position in partition
Tables	Rows, Degree, Pct free, Last analyzed on, Chained rows, Partitioned, Number of partitions, Partition type, Partition columns, Number of subpartitions, Subpartition type, Subpartition columns	Column, Type, Distinct values, Nullable, Position in partition
Indexes	Number of columns, Distinct keys, Leaf blocks, B-levels, Clustering factor, Columns Last analyzed on, Pct free, Partitioned, Locality, Partition type, Partition columns, Number of subpartitions, Subpartition type, Subpartition columns	Column, Type, Distinct values, Nullable, Position in index, Position in partition

Note: You can launch with the selected row of the Segments table in context to the Active Objects association in the Explore Objects view by clicking the Dictionary links.

About the Details area in the Tune Objects view

The Details area, in the Objects tab, is divided into the following tabs:

- Highlights

- Overtime

- Statements

- Joins

- Partitions

About the Highlights tab

The Highlights tab shows the statement access type distribution for the segment, indicating how the segment was accessed. The Findings table displays information on problematic objects and provides links to the appropriate tabs where you can continue your analysis or tune the object. It displays the percentage of time the object spends in the Oracle database, with respect to the amount of time spent by all segments in the Oracle database.

When you click on a row in the Findings table, additional details on the selected finding are displayed in the Details area below the table.

If you want to continue your examination of a particular finding, click on the launch icon to launch to the appropriate tab with the selected finding in context.

About the Overtime tab

The Overtime tab shows the segment's I/O wait distribution, by read/writes, over time and the segment changes that took place during the selected time period.

The following table describes the information displayed in the Overtime tab.

Table 6-6   Overtime graphs

Graphs	Description
In Oracle	Displays In Oracle resource consumption over time, broken down into In Oracle wait states.

Table 6-6   Overtime graphs

Graphs	Description
Changes	Shows at which point in time database block changes occurred, and the space allocated to the segment by the data dictionary, on the DB Block Changes axis. Additional points on the graph indicate whether or not Statistics (analyze operations performed on the segment) or Schema Changes (structural changes made to the object) occurred at that point in time. This information helps you identify the impacts of a statistics or schema change on a database object.
Physical I/O vs. Logical Reads	Summarizes physical I/O (includes physical reads direct, physical writes and physical writes direct) vs. logical reads on the selected segment, over time.

About the Statements tab

The Statements tab shows the statement access type distribution for the segment, indicating how the segment was accessed. Each table is populated according to the row you select in the table above it. The Statements table will show information on the accessing statements using the access type you selected in the Access Types table. The Columns table provides information on how columns are being used in the statement.

The following table describes the information displayed in the Statement tab.

Table 6-7   Statement table

Table	Description
Access Types	The Access Types table shows the following information on the statement's access types: ■ Access Type Describes which access type was used to access the specific object, segment or segments. ■ In Oracle (Sum.) Displays the amount of time spent by the selected object, segment or segments, of the specified access type, in the Oracle database. ■ Statements In Oracle Displays the amount of time spent by the selected object, segment or segments, of the specified access type, in the Oracle database. ■ Statements Number of explained statements accessing the segment with the specified access type If you want to continue your examination of a particular access type, click on the launch icon to launch to the Activity tab with the selected access type in context.

Table 6-7 Statement table

Table	Description
Statements	The Statements table shows the following information on the statements using the access type you selected in the Access Types table: ■ Text Displays the beginning of the selected statement's text so that you can identify the statement. You can view more of the statement's text by placing the cursor over the text cell in the table and viewing the information displayed in the ToolTip. ■ In Oracle (Sum.) Displays the amount of time spent by the selected statement, of the specified access type, in the Oracle database. ■ Statements In Oracle Displays the amount of time spent by the specific statement, of the specified access type, in the Oracle database. ■ Access Type Displays how the statements accessed all segments, or the selected segment or object (access type). ■ Executions Displays the number of times the statement was executed. ■ Duration (Avg.) Displays the average time spent by the statement in the Oracle database. ■ Statement Displays the statement's calculated hash or user-defined name. ■ Programs Displays the program that executed the statement. ■ Host Users Displays the users that executed that statement. If you want to continue your examination of a particular statement, click on the launch icon to launch to the Activity tab with the selected statement in context.
Columns	The Columns table shows the following information on the statement you selected in the Statements table: ■ Column Displays the name of the column used by the statement. ■ In Clause Displays the number of times the column was accessed by SELECT, WHERE, ORDER, or UPDATE clauses. ■ Equal Operator Displays the number of equal signs (=) used by the column. ■ Distinct Values Displays the number of distinct values used by the column over the entire segment or object. ■ Position in Index Displays the position of the column in the index or partition.

About the Joins tab

The Joins tab displays information on the tables (and their corresponding statements), that participate in joins with the selected table.

Note: This tab is only displayed if a table is selected in the Segments table in the Main area.

The following table describes the information displayed in the Joins tab.

Table 6-8   Joins Tables

Table	Description
Tables	The Tables table shows the following information on the tables that are joined with the selected segment, segments, or object: ■ Name Displays the name of the joined table. ■ Owner Displays the owner of the joined table. ■ In Oracle Displays the dispersion of the In Oracle time of the selected object or segment with respect to the table it is joined with. ■ Statements In Oracle Displays the In Oracle time of all statements accessing both tables. ■ Explained Statements Displays the number of statements that both the selected table and the current table participate in. If you want to continue your examination of a particular join table, click on the launch icon to launch to the Tune Object view with the selected join table in context.
Statements	The Statements table shows the following information on the statements in which both the selected segment or object, and the selected table appear: ■ Text Displays the beginning of the selected statement's text so that you can identify the statement. You can view more of the statement's text by placing the cursor over the text cell in the table and viewing the information displayed in the ToolTip. ■ In Oracle Displays the dispersion of the In Oracle time of the selected object or segment with respect to the table it is joined with. ■ Statements In Oracle Displays a breakdown of the In Oracle time of all statements accessing both tables. ■ Columns Displays the name of the column appearing in both tables. ■ Statement Displays the statement's calculated hash or user-defined name. If you want to continue your examination of a particular statement, click on the launch icon to launch to the Activity tab with the selected statement in context.

About the Partitions tab

The Partitions tab displays partition details for the selected segment or object.

Note: This tab is only displayed if the object or segment you selected in the Segments table is partitioned.

The following table describes the information displayed in the Partitions tab.

Table 6-9   Partitions table

Column	Description
Partition	Displays the name of the segment partition.
Deleted	Indicates whether or not the partition was deleted.
In Oracle (Sum.)	Displays a breakdown of the total waits that contributed to the In Oracle time for the selected partition.
Logical Reads	Summarizes the logical reads made on the selected partition.

Table 6-9   Partitions table

Column	Description
Physical I/Os	Summarizes the direct physical reads, direct physical writes, physical reads, and physical writes made on the selected partition.
Segment	Displays the name of the segment owning the partition.
Blocks	Displays the segment size in blocks.
Tablespace	Displays the name of the tablespace in which the partition resides.
Buffer Pool	Displays the name of the buffer pool in which the partition resides.

If you want to continue your examination of a particular partition, click on the launch icon to launch to the Activity tab with the selected partition in context.

About Objects tab entities

The Objects tab displays information on many different entities. This section provides an overview of all these entities, their meaning, and their views.

In the Objects tab, you can examine the following entities:

Table 6-10   Entities

Entity	Entity
Instance entity	Index Partition and Index Sub-Partition entities
Oracle File entity	Index-type entity
Tablespace entity	LOB entity
Table entity	Materialized View entity
Index and unused index entities	Method entity
Column entity	Rollback entity
Statement entity	Sequence entity
PL/SQL entity	Synonym entity
Object entity	Table Partition and Table Sub-Partition entities
View entity	Trigger entity
Attribute entity	User entity
Cluster entity	User-defined type entity
Constraint entity	

Some entities are related to storage devices and are only available if you install the EMC Symmetrix and Clariion or HP SureStore versions of Precise for Storage. The Objects tab lets you see details for all the storage arrays connected to the server where the currently selected instance is running. You can then drill down inside an array and see details of its components. The performance counters are available for all devices and not just for those devices used by the selected instance. This gives you a view of the overall performance of the arrays and lets you identify less busy devices that you can move your Oracle files to.

The following entities are only available if you have installed Precise for Storage:

■   EMC Units

- Disk director entity

- Port entity

- Hyper entity

- HP Storage Device entity

For more information on the supported platforms, see the *Precise Installation Guide*.

About the Instance entity

The Instance entity, in the Objects tab, shows object-related information on the instance level.

The following views are available:

- Overview

- Init parameters

- Database options

- Database option changes

- Parameter changes

- Redo log changes

- Rollback segment changes

- Tablespace and datafile changes

- Schema changes

It is possible to associate from an instance to Active Objects and to the Users and Tablespaces defined in the instance, and to the Oracle Files that make up the instance. You can display additional entities by using the More... option.

About the overview of instance-related objects

The Overview displays general instance information and an overtime graph that illustrates the time (Sum.) that the instance spent In Oracle during the selected time period.

The following table describes the general information displayed in this view.

Table 6-11   Instance Overview

Column	Description
Version	Displays which version of Oracle software is being used.
Parallel	Indicates whether the instance is part of an OPS or RAC cluster.
Status	Displays the status of the instance.
Database Status	Displays the status of the entire database, in relation to the cluster.
Archiver Status	Displays the status of the database log archiver.
Up Time	Displays how long the database has been up.

Additionally Logical Reads and Space Allocation graphs are shown.

About init parameters for an instance

The Init Parameters view displays all Oracle initialization or INIT.ORA parameters.

The following table describes the table shown in this view.

Table 6-12   Init Parameters view

Column	Description
Parameter	Displays the Parameter Name.
Values	Displays the Current Value.
Is Default	Indicates whether the value displayed is the default value.
Is Session Modifiable	Indicates whether this parameter can be changed to take effect according to session.
Is System Modifiable	Indicates whether this parameter can be changed to take effect across the entire instance.
Is Modified	Indicates whether the parameter has been modified after instance startup.
Is Adjusted	Indicates whether Oracle adjusted the input value to a more suitable value.
Description	Provides a description of the parameter.

About database options

The Database Options view displays the available database options.

The following table describes the table shown in this view.

Table 6-13   Database Options view

Column	Description
Option	Displays the name of the database option.
Is Enabled	Displays a True or False value indicating whether or not the option is enabled.

About examining changes to database options

The Database Options Changes view displays the changes made to database options during the selected time period.

The following table describes the table shown in this view.

Table 6-14   Database Options Changes view

Column	Description
Timestamp	Displays the date the change took place.
Component	Displays the name of the item changed.
Old Value	Displays the old value of the component.
New Value	Displays the new value of the component.

About parameter changes

The Parameter Changes view displays the changes to the database initialization parameters during the selected time period.

The following table describes the table shown in this view.

Table 6-15   Parameter Changes view

Column	Description
Timestamp	Displays the date the change took place.
Parameter	Displays the name of the item changed.

Table 6-15   Parameter Changes view

Column	Description
Old Value	Displays the old value of the parameter.
New Value	Displays the new value of the parameter.

About changes to the redo logs

The Redo Log Changes view displays the changes to the redo logs during the selected time period.

The following table describes the table shown in this view.

Table 6-16   Redo Log Changes view

Column	Description
Timestamp	Displays the date the change took place.
Redo Log	Displays the name of the redo log that was changed.
Change Type	Displays the type of change that took place.
Field	Displays the type of change that took place—created, deleted or field changed.
Old Value	Displays the old value of the parameter.
New Value	Displays the new value of the parameter.

About changes to rollback segments

The Rollback Segment Changes view displays the changes to rollback segments during the selected time period.

The following table describes the table shown in this view.

Table 6-17   Rollback Segment Changes view

Column	Description
Timestamp	Displays the date the change took place.
Owner	Displays the Oracle user that owns the rollback segment.
Rollback Segment	Displays the name of the rollback segment that was changed.
Change Type	Displays the type of change that took place.
Field	Displays the type of change that took place—created, deleted or field changed.
Old Value	Displays the old value of the parameter.
New Value	Displays the new value of the parameter.

About changes to tablespaces and datafiles

The Tablespace and Datafile Changes view displays the changes to tablespaces and datafiles during the selected time period.

The following table describes the table shown in this view.

Table 6-18   Tablespace and Datafile Changes view

Column	Description
Timestamp	Displays the date the change took place.
<blank>	Datafile or Tablespace.

Table 6-18   Tablespace and Datafile Changes view

Column	Description
Name	The name of the datafile or tablespace that was changed.
Change Type	Indicates the type of change that took place—created, deleted or field changed.
Field	Displays which item was changed.
Old Value	Displays the old value of the parameter.
New Value	Displays the new value of the parameter.

About schema changes

The Schema Changes view displays the changes made to schema objects, such as tables and indexes, during the selected time period, but expressed as DDL. For example: DROP INDEX or ALTER TABLE

The following table describes the table shown in this view.

Table 6-19   Schema Changes view

Column	Description
Timestamp	Displays the date the change took place.
Change Type	Provides a description of the change, for example, Create TABLE CUSTOMER.

About the Oracle File entity

The Oracle File entity is comprised of information about datafiles and temporary files. This information lets you find objects that reside on the Oracle file and the statements that access these objects. You can view the I/O statistics of the Oracle file to understand the I/O read and write time and the number of physical I/Os. The Oracle File entity displays general details, I/O activity, and storage information for the selected file.

The following views are available:

■   Overview

■   RAC Database

■   Dictionary

■   Storage

About the overview of an Oracle file

The Overview, in the Objects tab, displays general Oracle file information and several overtime graphs.

The following table describes how the general information is structured.

Table 6-20   Structure of general information for an Oracle file

Item	Description
File Name	Displays the name of the Oracle file.
Folder	Displays the Operating System folder where the file is located.
Created on	Displays the date the file was created.
Status	Displays the status of the file—Online, Offline, Recover, System.
Tablespace	Displays the name of the Tablespace supported by the Oracle file.
File Type	Indicates whether the file is a datafile, or a temp file.

The following table describes the overtime graphs for an Oracle file.

Table 6-21 Overtime graphs for an Oracle file

Graph	Description
Physical I/O Operations	Illustrates the physical reads and physical writes over time.
Physical Read Time (Avg.)	Illustrates the average read times.
I/O Waits (Sum.)	Illustrates the numbers of I/O waits according to wait state.
Physical Write Time (Avg.)	Illustrates the average write times.

About the RAC Database of an Oracle file

The RAC Database view, in the Objects tab, compares the file activity of the entire RAC database to the file activity of the selected RAC instance. This view shows aggregated wait states, and load balancing and indicates whether only the specific instance is effected by an event, or whether the entire database is effected by the event.

The following table describes the information displayed on Oracle files in the RAC Database view.

Table 6-22 Information displayed on Oracle files in the RAC Database view

Item	Description
Database	Displays the logical name of the RAC database.
DB Name	Displays the Oracle database name.
DB ID	Displays the Oracle database ID.
Instance Number	Displays the number of the instance in the RAC database.
Instances	Displays the number of instances that share the same RAC database with the selected instance.
Overtime graphs	Displays the following overtime graphs that compare the Oracle file activity of the entire database to the Oracle file activity of the selected instance, over the selected time period: ■ Database physical I/O operations ■ Average database I/O waits ■ Average database physical read time ■ Average database physical write time

About the dictionary of an Oracle file

The Dictionary view, in the Objects tab, displays the general properties of the Oracle File.

The following table describes how the information in the Dictionary view is structured.

Table 6-23 Structure of information in the Dictionary view

Item	Description
File	Displays the full path of the Oracle file.
Name	Displays the name of the Oracle file.
Folder	Displays the Operating System folder where the file is located.
Tablespace	Displays the name of the Tablespace supported by the Oracle file.
File ID	Displays the Oracle ID for the file.
File Type	Indicates whether the file is a datafile or temp file.
Status	Displays the status of the file—Online, Offline, Recover, System.
Autoextension	Indicates whether the file can be autoextended.

Table 6-23 Structure of information in the Dictionary view

Item	Description
Created on	Displays the date the file was created.
Accessibility	Indicates whether or not the file can be written to: Read Write, or Read Only

About storage information related to an Oracle file

The Storage view, in the Objects tab, displays storage information related to the Oracle File.

The following table describes how the information in the Storage view is structured.

Table 6-24 Structure of information in the Storage view

Item	Description
File Size	Displays the current size of the file.
Blocks	Displays the number of blocks in the file.
Block Size	Displays the block size of the file.
Autoextension	Indicates whether or not the file can autoextended.
Max. File Size	Displays the maximum size the file can be extended to.
Max. Blocks	Displays the maximum number of blocks the file can be extended to.
Increment by	Displays the size allocated for further extension when auto-extending.

Associating an Oracle file with other entities

You can associate an Oracle file in the Main area of the Objects tab with the following entities in the Association area:

■ Tables and Indexes that are stored in the file.

■ Statements that are run against those tables and indexes in the file.

■ Tablespaces that the file belongs to.

■ Storage devices that the file is mounted on. This option is only available if Precise for Storage is installed.

You can display additional entities by using the More... option.

The information in the Association area table is displayed on the following tabs:

■ Dictionary tab—displays general properties of the Oracle File.

■ Storage tab—display storage information related to the Oracle File.

■ Physical I/O tab—displays a table of total I/O activity against the Oracle File, over the selected time period. On this tab, the table contains the following columns:

File	The name of the Oracle file.
Physical I/O Operations	A bar graph that shows the number of physical reads and writes.
Physical Reads	The total number of physical reads executed during the selected time period.
Physical Writes	The total number of physical writes executed during the selected time period.
Avg. Read REQ time (Msec.)	The average number of physical reads per minute, executed during the selected time period.
Avg. Write REQ time (Msec.)	The average number of physical writes per minute executed during the selected time period.
File ID	ID of the file.

About the Tablespace entity

The Tablespace entity displays general details, I/O activity and storage information for the selected Tablespace.

The following views are available:

- Overview
- RAC Database
- Dictionary
- Storage

About the overview of a tablespace

The Overview, in the Objects tab, displays general tablespace information and a stacked overtime graph that illustrates the physical reads and physical writes to the tablespace.

The following table describes how the general information is structured.

Table 6-25   Structure of general information for a tablespace

Item	Description
Status	Displays the status of the tablespace—online, or offline.
Datafiles	Displays the number of datafiles supporting the tablespace.
Contents	Indicates whether the tablespace is Temporary or Permanent.
Logging	Indicates whether the Tablespace has logging enabled.
Extent Management	Indicates whether the tablespace is locally managed or dictionary managed.
Allocation Type	Specifies how Oracle manages extents within the Tablespace. Values are System or Uniform.
Plugged in	Indicates whether a transportable tablespace is available.
Block Size	Displays the underlying block size of the tablespace.

Additionally a Physical I/O Operations graph is shown.

About the RAC Database of a tablespace

The RAC Database view, in the Objects tab, compares the tablespace activity of the entire RAC database to the tablespace activity of the selected RAC instance. This view shows aggregated wait states, and load balancing and indicates whether only the specific instance is effected by an event, or whether the entire database is effected by the event.

The following table describes the information displayed on tablespaces in the RAC Database view.

Table 6-26   Information displayed in the RAC Database view for tablespaces

Item	Description
Database	Displays the logical name of the RAC database.
DB Name	Displays the Oracle database name.
DB ID	Displays the Oracle database ID.
Instance Number	Displays the number of the instance in the RAC database.
Instances	Displays the number of instances that share the same RAC database with the selected instance.
Overtime Graphs	Displays Database physical I/O operations in the overtime graph. The graph compares the tablespace activity of the entire database to the tablespace activity of the selected instance, over the selected time period:

About the dictionary of a tablespace

The Dictionary view, in the Objects tab, displays the general properties of the tablespace.

The following table describes how the information in the Dictionary view is structured.

Table 6-27   Structure of information in the Dictionary view

Item	Description
Name	Displays the name of the tablespace.
Datafiles	Displays the number of datafiles supporting the tablespace.
Status	Displays the status of the tablespace—online, or offline.
Contents	Indicates whether the tablespace is Temporary or Permanent.
Logging	Indicates whether the tablespace has logging enabled.
Extent Management	Indicates whether the tablespace is managed locally or by dictionary.
Allocation Type	Specifies how Oracle manages extents within the Tablespace.Values are System, or Uniform.
Plugged in	Indicates whether a transportable tablespace is available.

About storage information related to a tablespace

The Storage view, in the Objects tab, displays storage information related to the tablespace, including used and free space, and extent information.

The following table describes how the information in the Storage view is structured.

Table 6-28   Structure of information in the Storage view

Item	Description
Datafiles	Displays the number of datafiles supporting the tablespace.
Total Space	Displays the total space available in the tablespace.
Used Space	Displays the amount of used space in the tablespace.
Free Space	Displays the amount of free space in the tablespace.
Free Extents	Displays the number of free extents in the tablespace.
Largest Free Extent	Displays the size of the largest free extent in the tablespace.
Initial Extent	Displays the default size allocated to the first extent when creating a table in the tablespace.
Next Extent	Displays the default size allocated to subsequent extents when creating a table in the tablespace.
Pct. Increase	Displays the default percentage increase in size on each subsequent extent allocation.
Min. Extent Size	Displays the smallest extent size Oracle may use.
Min. Extents	Displays the minimum default number of extents allowed for tables within the tablespace.
Max. Extents	Displays the maximum default number of extents allowed for tables within the tablespace.

Associating a tablespace with other entities

You can associate a tablespace in the Main area of the Objects tab with the following entities in the Association area:

- Oracle Files that are used for storage
- Statements that have accessed the Tablespace
- Tables and Indexes in the tablespace

You can display additional entities by using the More... option.

The information in the Association area table is displayed on the following tabs:

■ Dictionary tab—displays general properties of the Tablespace.

■ Storage tab—displays storage information related to the tablespace, including used and free space, and extent information.

■ Physical I/O tab—displays a table of total I/O activity against the tablespace, over the selected time period. On this tab, the table contains the following columns:

File	The name of the file.
Physical I/O Operations	A bar graph showing the number of physical reads and writes.
Avg. Read REQ Time (Msec.)	Average read request time in miliseconds.
Avg. Write REQ Time (Msec.)	Average write request time in miliseconds.

About the Table entity

The Table entity, in the Objects tab, displays general details, storage information, statistical information, and change-related information for the selected table.

The following views are available:

■ Overview

■ RAC Database

■ Dictionary

■ Storage

■ Statistics

■ Changes

About the overview of a table

The Overview, in the Objects tab, displays general table information and several overtime graphs.

The following table describes how the general information is structured.

Table 6-29   Structure of general information for a table

Item	Description
Created on	Displays the date the table was created.
Status	Displays the status of the table—whether it is valid or invalid.
Partitioned	Indicates whether the table is partitioned.
Nested	Indicates whether the table is nested.
Temporary	Indicates whether the table is temporary or permanent.
Index Organized	Indicates whether the table is Index Organized.
Cluster	Indicates whether the table is part of a cluster.
Tablespace	Displays the tablespace in which the table is created.
Logging	Indicates whether updates to the table are logged.
Cache	Indicates whether the table is cached in the buffer cache.
Rows	Displays the number of rows in the table.

Table 6-29   Structure of general information for a table

Item	Description
Last Analyzed on	Displays the date on which the table was last analyzed.

The following table describes the overtime graphs for a table.

Table 6-30   Overtime graphs for an table

Graph	Description
In Oracle (Sum.)	Displays the relevant In Oracle wait states.
Changes	Displays the changes made over time where each point on the overtime graph represents a separate statistics or schema change.
Blocks	Displays the number of blocks allocated, split, empty or used.
Rows	Displays the number of rows in the table.

The changes, blocks and rows data is dependent on the Collect Schema Changes process. Precise for Oracle reports statistical changes when this process is run.

About the RAC Database of a Table

The RAC Database view, in the Objects tab, compares the table activity of the entire RAC database to the table activity of the selected RAC instance. This view shows aggregated wait states, and load balancing and indicates whether only the specific instance is effected by an event, or whether the entire database is effected by the event.

The following table describes the information displayed on tables in the RAC Database view.

Table 6-31   Information displayed on tables in the RAC Database view

Item	Description
Database	Displays the logical name of the RAC database.
DB Name	Displays the Oracle database name.
DB ID	Displays the Oracle database ID.
Instance Number	Displays the number of the instance in the RAC database.
Instances	Displays the number of instances that share the same RAC database with the selected instance.
Overtime graphs	Displays the following overtime graph that compares the table activity of the entire database to the table activity of the selected instance, over the selected time period: ■ In Oracle (Sum.)

About the dictionary of a Table

The Dictionary view, in the Objects tab, displays the general properties of the table.

The following table describes how the information in the Dictionary view is structured.

Table 6-32   Structure of information in the Dictionary view

Item	Description
Created on	Displays the date the table was created.
Last Modified on	Displays the date on which the table was last modified.
Status	Displays the status of the table—whether it is valid, or invalid.

Table 6-32 Structure of information in the Dictionary view

Item	Description
Cache	Indicates whether the table is cached in the buffer cache.
Degree	Specifies the degree of parallelism to be used when performing full table scans using parallel query.
Partitioned	Indicates whether the table is partitioned.
Nested	Indicates whether the table is nested.
Temporary	Indicates whether the table is temporary or permanent.
Index Organized	Indicates whether the table is Index Organized.
Cluster	Indicates whether the table is clustered.
Logging	Contains "Yes" if logging is enabled for all operations or "NO" if minimal logging is enabled whenever applicable. For more information, see Oracle's documentation.

Note: If the selected table is partitioned, some of these fields will not be displayed in Precise for Oracle, but they will be displayed in the table partition itself.

About storage information related to a table

The Storage view, in the Objects tab, displays storage information related to the table, including used and free space, and extent information.

The following table describes how the information in the Storage view is structured.

Table 6-33 Structure of information in the Storage view

Item	Description
Table	Name of the table,
Owner	Owner of the table.
Tablespace	Displays the tablespace where the table is located.
Segment Blocks	Displays the sumber of segment blocks in the table,
Extents	Displays the number of extents in the table.
Initial Extent	Displays the size allocated to the first extent when the table was created.
Next Extent	Displays the size allocated when the table size is not sufficient.
Pct. Increase	Displays the default percentage increase in size on each subsequent extent allocation.
Pct. Used	Displays the percentage of used space in each data block.
Pct. Free	Displays the percentage of space reserved in each data block for updates.
Min. Extents	Displays the minimum number of extents allowed.
Max. Extents	Displays the maximum number of extents allowed.
Init. Trans.	Displays the initial number of transaction slots in each block.
Max. Trans.	Displays the maximum number of transaction slots in each block.
Free Lists	Displays the number of free lists.
Type	Displays Type of the object.

Table 6-33   Structure of information in the Storage view

Item	Description
Table #	Displays table id as shown in Oracle dictionary.

Note: If the selected table is partitioned, some of these fields are not displayed in Precise for Oracle, but they will be displayed in the table partition itself.

About the statistics of a table

The Statistics view, in the Objects tab, displays statistics on the table. This information is generated by the Oracle statistics collection mechanism.

The following table describes how the information in the Statistics view is structured.

Table 6-34   Structure of information in the Statistics view

Item	Description
Rows	Indicates the number of rows.
Chained Rows	Indicates the number of chained rows.
Row Length (Avg.)	Indicates the average row length.
Recent Growth	Indicates the growth of the table (in blocks) since the previous statistics collection and from the Collect Schema Changes process.
Last Analyzed on	Displays the date of the last analysis.
Empty Blocks	Displays the number of empty blocks.
Non-empty Blocks	Displays the number of non-empty blocks.
Free Space per Block (Avg.)	Displays the average amount of free space per block.

About table-related changes

The Changes view, in the Objects tab, displays details of the changes that were made to the table and the indexes defined on the table.

The following table describes how the information in the Changes view is structured.

Table 6-35   Structure of information in the Changes view

Column	Description
Timestamp	Displays the date the change took place.
Change Type	Displays a description of the change. For example: EMPLOYEE.DENSITY modified from 0.00201 to 0.00203

Associating a table with other entities

You can associate a table in the Main area of the Objects tab with the following entities in the Association area:

- Columns that make up the table
- Indexes that are related to the table
- Statements that access the table
- Table partitions
- Table subpartitions

You can display additional entities by using the More... option.

The information in the Association area table is displayed on the following tabs:

- Dictionary tab—displays data dictionary information for the selected table.

- Storage tab—displays storage information for the selected table.

- Statistics tab—displays the statistics for the table, generated by the Oracle statistics collection mechanism

About the Index and Unused Index entities

The Index entity, in the Objects tab, displays general details, storage information, Oracle statistics information, and change-related information for the selected index or unused index.

An index is unused if it has not been referenced in an explain plan. Until some statements have been explained using Precise for Oracle, all indexes are, by definition, unused. For this reason, it is important that all functions within an application are exercised before you search for unused indexes. Statements are automatically explained by Precise for Oracle FocalPoint once a day (by default), so there may be a lag before a statement is explained following its execution.

The following views are available:

- Overview

- RAC Database

- Dictionary

- Storage

- Statistics

- Changes

About the overview of an index

The Overview, in the Objects tab, displays general index information and several overtime graphs.

The following table describes how the general information is structured.

Table 6-36   Structure of general information for an index

Item	Description
Created on	Displays the creation date of the index.
Status	Displays the status of the index—indicates whether it is valid, or invalid.
Type	Displays the type of index—Normal or Bitmap.
Unique	Indicates whether the key values are unique.
Partitioned	Indicates whether the index is partitioned.
Table	Displays the name of the table that the selected index is referring to.
Owner	Displays the Oracle user that owns the index.
Distinct Keys	Displays the number of distinct values in the index.
Clustering Factor	Displays the clustering factor of the index. The clustering factor is an important factor in determining how efficiently an index range scan will retrieve data from the table. It measures the degree to which the data in the index and its table are in the same order or, put another way, the probability that the next row to be fetched from the table is in the same block as the current row. It can vary between the number of blocks in the table (the best case because they are in the same order) and the number of rows in the table (the worst case because they are completely out of sync). The clustering factor tends to become worse over time as data is inserted and deleted. Note that the clustering factor makes no difference for a unique index lookup.

Table 6-36 Structure of general information for an index

Item	Description
Leaf Blocks	Displays the number of blocks at the lowest index level.
B-Levels	Displays the number of levels above the leaf blocks in the index (indicates the depth of the index).
Last Analyzed on	Displays the date when the index was last analyzed.

The following table describes the overtime graphs for a table.

Table 6-37 Overtime graphs for a table

Graph	Description
In Oracle (Sum.)	Displays the relevant In Oracle wait states.
Changes	Displays the number of changes that took place.
Distinct Keys	Displays the number of distinct keys in the index.
Clustering Factor	Displays the clustering factor of the index.

The changes, distinct keys, and clustering factor data is dependent on the Collect Schema Changes process. Precise for Oracle reports statistical changes when this process is run.

About the RAC Database of an index

The RAC Database view, in the Objects tab, compares the index activity of the entire RAC database to the index activity of the selected RAC instance. This view shows aggregated wait states, and load balancing and indicates whether only the specific instance is effected by an event, or whether the entire database is effected by the event.

The following table describes the information displayed on indexes in the RAC Database view.

Table 6-38 Information displayed on indexes in the RAC Database view

Item	Description
Database	Displays the logical name of the RAC database.
DB Name	Displays the Oracle database name.
DB ID	Displays the Oracle database ID.
Instance Number	Displays the number of the instance in the RAC database.
Instances	Displays the number of instances that share the same RAC database with the selected instance.
Overtime Graphs	Displays the In Oracle (Sum.) overtime graph. The graph compares the index or unused index activity of the entire database, to the index or unused index activity of the selected instance, over the selected time period:

About the dictionary of an index

The Dictionary view, in the Objects tab, displays the general properties of the index.

The following table describes how the information in the Dictionary view is structured.

Table 6-39 Structure of information in the Dictionary view

Item	Description
Created on	Displays the creation date of the index.
Last Modified on	Displays the date on which the index was last modified.

Table 6-39 Structure of information in the Dictionary view

Item	Description
Status	Displays the status of the index (valid or invalid).
Table	Displays the name of the table that the selected index is referring to.
Owner	Displays the Oracle user that owns the index.
Type	Displays the type of index—Normal, Bitmap.
Unique	Indicates whether the key values are unique.
Partitioned	Indicates whether the index is partitioned.
Compression	Indicates whether the index has been defined with the COMPRESSES attribute.
Logging	Indicates whether updates to the index are logged.
Columns	Displays the number of columns in the index.

Note: If the selected index or unused index is partitioned, some of these fields will not be displayed in Precise for Oracle, but they will be displayed in the index partition itself.

About storage information related to an index

The Storage view, in the Objects tab, displays storage information related to the index.

The following table describes how the information in the Storage view is structured.

Table 6-40 Structure of information in the Storage view

Item	Description
Tablespace	Displays the Tablespace in which the index is located.
Blocks	Displays the number of blocks in the index.
Extents	Displays the number of extents in the index.
Initial Extent	Displays the size allocated to the first extent when the index was created.
Logging	Indicates whether updates to the index are logged.
Compression	Indicates whether the index has been defined with the COMPRESSES attribute.
Table	Displays the name of the table that the selected index is referring to.
Owner	Displays the Oracle user that owns the index.

Note: If the selected index or unused index is partitioned, some of these fields will not be displayed, but they will be displayed in the index partition itself.

About the statistics of an index

The Statistics view, in the Objects tab, displays statistics that are related to the index and generated by the Oracle statistics collection mechanism.

The following table describes how the information in the Statistics view is structured.

Table 6-41   Structure of information in the Statistics view

Item	Description
Distinct Keys	Displays the number of distinct values in the index.
Clustering Factor	Displays the clustering factor of the index.
	The clustering factor is an important factor in determining how efficiently an index range scan will retrieve data from the table. It measures the degree to which the data in the index and its table are in the same order or, put another way, the probability that the next row to be fetched from the table is in the same block as the current row. It can vary between the number of blocks in the table (the best case because they are in the same order) and the number of rows in the table (the worst case because they are completely out of sync). The clustering factor tends to become worse over time as data is inserted and deleted. Note that the clustering factor makes no difference for a unique index lookup.
Leaf Blocks	Displays the number of blocks at the lowest index level.
Leaf Blocks per Key	Displays the average number of leaf blocks in which each distinct key value appears. There must be at least one leaf block for a unique key.
Data Blocks per Key	Displays the average number of data blocks in which each distinct key value appears. There must be at least one leaf block for a unique key.
B-levels	Displays the number of levels above the leaf blocks in the index (indicates the depth of the index).
Last Analyzed on	Displays the date when the index was last analyzed.

About changes to indexes

The Changes view, in the Objects tab, displays details of the changes made to the index.

The following table describes how the information in the Storage view is structured.

Table 6-42   Structure of the information in the Changes view

Column	Description
Timestamp	Displays the date of the change.
Change Type	Displays a description of the change. For example, `EMP_DEPT_FK.DISTINCT_KEYS` modified from 25 to 30.

Associating an index with other entities

You can associate an index in the Main area of the Objects tab with the following entities in the Association area:

- Columns in the index

- Tables that are indexed

- Statements that make use of the index (only if the index is used. By definition, no statements are associated with an unused index.)

- Index partitions and subpartitions

You can display additional entities by using the More... option.

The information in the Association area is displayed on the following tabs:

- Columns tab—displays information about the columns that make up the index.

Name	Displays the name of the index.
Owner	Displays the Oracle user that owns the index.

Columns	Displays the number of columns that make up the index.
Column List	Displays the names of the columns that make up the index.

- Dictionary tab—displays data dictionary information for the selected index, similar to that described in the Main area.

- Storage tab—displays storage information for the selected index, similar to that described in the Main area.

- Statistics tab—displays statistics related to the index, as generated by the Oracle statistics collection mechanism.

Index	Displays the name of the index.
Owner	Displays the Oracle user that owns the index.
Distinct keys	Displays the number of distinct values in the index.
Clustering Factor	Displays the clustering factor of the index. The clustering factor is an important factor in determining how efficiently an index range scan will retrieve data from the table. It measures the degree to which the data in the index and its table are in the same order or, put another way, the probability that the next row to be fetched from the table is in the same block as the current row. It can vary between the number of blocks in the table (the best case because they are in the same order) and the number of rows in the table (the worst case because they are completely out of sync). The clustering factor tends to become worse over time as data is inserted and deleted. Note that the clustering factor makes no difference for a unique index lookup.
Leaf Blocks	Displays the number of blocks at the lowest index level.
Leaf Blocks per Key	Displays the average number of leaf blocks in which each distinct key value appears. There must be at least one leaf block for a unique key.
Data Blocks per Key	Displays the average number of data blocks in which each distinct key value appears. There must be at least one leaf block for a unique key.
B-Levels	Displays the number of levels above the leaf blocks in the index (indicates the depth of the index).
Last Analyzed on	Displays the date when the index was last analyzed.

About the Column entity

The Column entity, in the Objects tab, displays general details and Oracle analysis statistics for the selected column.

The following views are available:

- Overview
- Statistics
- Buckets

About the overview of a column

The Overview, in the Objects tab, displays general column information and three overtime graphs that show the variation in the number of distinct values, the number of nulls, and the density. Density is the proportion of selective values divided by the number of distinct selective values. For uniformly distributed data, this equates to one over the number of distinct values.

The data is dependent on the Collect Schema Changes process. Precise for Oracle reports statistics changes when this process is run.

The following table describes how the general information is structured.

Table 6-43   Structure of general information for a column

Item	Description
Table	Displays the name of the table to which the column belongs.
Owner	Displays the Oracle-owning user.
Position	Displays the numeric position of the column within the table.
Predefined Type	Indicates whether the column is a predefined data type.
Type	Displays the data type of the column.
Length	Displays the length of the column.
Nullable	Indicates whether the column is allowed to hold null values.
Default Value	Displays a default value for the column, if any.

Additionally the Distinct Values, Null Values, and Density graphs are shown.

About the statistics of a column

The Statistics view, in the Objects tab, displays statistics related to the column and generated by the Oracle statistics collection mechanism.

The following table describes how the information in the Statistics view is structured.

Table 6-44   Structure of information in the Statistics view

Item	Description
Value (Min.)	Displays the lowest value for the column.
Value (Max.)	Displays the highest value for the column.
Distinct values	Displays the number of different values.
Null values	Displays the number of nulls.
Density	Displays the proportion of selective values divided by the number of distinct selective values. For uniformly distributed data this equates to one over the number of distinct values.
Time	Displays the date of the analyses.
Sample Size	Displays the number of rows sampled.
Length (Avg.)	Displays the average length of the column.

About the buckets of a column

The Buckets view, in the Objects tab, displays the distribution of column data in the Oracle statistics that were gathered. This information is typically generated for indexed columns with skewed data by running a statistics collection.

The following table describes how the information in the Buckets view is structured.

Table 6-45   Structure of information in the Buckets view

Column	Description
Bucket	The divisions of the Oracle statistics histograms
Bucket Limit	The data value at the upper limit of each bucket.

Associating a column with other entities

You can associate a column in the Main area of the Objects tab with the following entities in the Association area:

- Statements that refer to the column

- Tables that the column belongs to

- Constraints that the column is involved in

- Indexes that the column belongs to

You can display additional entities by using the More... option.

The information in the Association area table is displayed on the following tabs:

- Dictionary tab—displays data dictionary properties of the column similar to those displayed in the Main area overview.

- Statistics tab—displays statistics on the column, as generated by the Oracle statistics collection mechanism.

- Usage tab—displays information on how the column has been accessed within explain plans.

Column	Displays the name of the column.
References	Displays the total of the counts displayed in the columns that followed, viewed as a numeric figure or bar chart.
As Indexable	Displays the number of times a column could have been used as part of an index.
In Select	Displays the number of times a column appeared in a Select clause.
In Where	Displays the number of times a column appeared in a Where clause.
In Update	Displays the number of times a column appeared in an Update clause.
In Join	Displays the number of times a column appeared in a Join predicate.
In Aggregate	Displays the number of times a column appeared in an Aggregate function.
In Distinct	Displays the number of times a column appeared in a Distinct clause.
In Order	Displays the number of times a column appeared in a Sort by clause.
In Group	Displays the number of times a column appeared in Group by clause.
Position	Displays the numeric position of the column within the entities.

About the Statement entity

The Statement entity, in the Objects tab, displays activity information, In Oracle performance, the text of the statement, and the storage location for the selected statement.

The following views are available:

- In Oracle

- Overall activity

- Text

- Workshop

About In Oracle information for a statement

The In Oracle view, in the Objects tab, displays an overtime graph of the In Oracle wait states.

See "Wait States" on page 35.

About the overall activity of a statement

The Overall Activity view, in the Objects tab, displays a graph that illustrates the statement's overall activity over the selected time period.

See "Wait States" on page 35.

About the text of a statement

The Text view, in the Objects tab, displays the full SQL text of the selected statement.

About information on the cabinet or folder related to a statement

The Workshop view, in the Objects tab, displays details on the cabinet or folder where the statement or PL/SQL is saved and on how it will be explained.

In general, if you prefer that a statement be referenced by a user-defined name in all tabs, rather than by a numeric identifier, you can save it to a particular cabinet or folder and assign a name to it. If required, you can also edit it later or add a tuning comment.

See "Editing the properties of a statement" on page 31.

See "About Precise for Oracle tabs" on page 21.

The following table describes how the information in the Workshop view is structured.

Table 6-46 Structure of information in the Workshop view

Item	Description
Explained on	Displays the date on the which the statement was last explained.
Optimizer Mode	Displays the Oracle optimizer mode.
Cost	Displays the optimizer cost.
Parsing User	Displays the Oracle user that the explain plan is generated for.
Cabinet	Displays the cabinet where the statement is stored.
Folder	Displays the folder within the cabinet where the statement is stored.
Origin	Indicates where the statement originated from, for example, automatically collected.
Saved on	Displays the date statement was saved.
Comment	Displays a textual comment associated with the statement.

About associating a statement with other entities

You can associate a statement in the Main area of the Objects tab with the following entities in the Association area:

■ Tables

■ Indexes

■ Views

You can display additional entities by using the More... option.

The information in the Association area table is displayed on the following tabs:

■ General tab—displays the statement name and a portion of the SQL. You can also click on the tuning icon to launch the SQL tab to further your analysis.

■ Workshop tab—displays similar information to the information displayed in the Workshop view in the Main area. It is possible to associate from a Statement to the Tables, Indexes, and Views that it accesses.

About the PL/SQL entity

The PL/SQL entity, in the Objects tab, displays general details and the text of the selected PL/SQL.

The following views are available:

- Overview
- Text

About the overview of a PL/SQL

The Overview, in the Objects tab, displays general information on the selected PL/SQL.

The following table describes how the general information is structured.

Table 6-47 Structure of general information for a PL/SQL

Item	Description
Name	Displays the name of the PL/SQL.
Owner	Displays the Oracle user to which the PL/SQL belongs.
Type	Displays the type of PL/SQL. Possible values are: ■ Procedure ■ Function ■ Package ■ Package Body
Created on	Displays the date on which the PL/SQL was created.
Last Modified on	Displays the date on which the PL/SQL was last updated.
Status	Displays the status of the PL/SQL (valid or invalid).

About the text of a PL/SQL

The Text view, in the Objects tab, displays the full text of the selected PL/SQL.

Associating a PL/SQL with other entities

You can associate a PL/SQL in the Main area of the Objects tab with the following entities in the Association area:

- Objects referenced in the text
- Views referenced in the text
- Tables referenced in the text

You can display additional entities by using the More... option.

About the View entity

The View entity, in the Objects tab, displays general details and the text for the selected view.

The following views are available:

- Overview
- Text

About the overview of a view

The Overview, in the Objects tab, displays general information on the selected view.

The following table describes how the general information is structured.

Table 6-48   Structure of general information for a view

Item	Description
Name	Displays the name of the view.
Owner	Displays the Oracle user to which the view belongs.
Created on	Displays the date on which the view was created.
Last Modified on	Displays the date on which the view was last updated.
Type	View
Status	Displays the status of the view—whether it is valid or invalid.
Object ID	Displays the unique Oracle object ID for the view.

About the text of a view

The Text view, in the Objects tab, displays the full text of the selected view.

Associating a view with other entities

You can associate a view in the Main area of the Objects tab with the following entities in the Association area:

- Statements that access the view
- Columns that the view includes

You can display additional entities by using the More... option.

About the Attribute entity

The Attribute entity, in the Objects tab, displays general information (Overview) that is retrieved from the data dictionary.

Associating an attribute with other entities

You can associate an attribute, in the Objects tab, with the following entities:

- User-defined type of attribute
- User-defined type containing attribute

You can display additional entities by using the More... option.

About the Cluster entity

The Cluster entity, in the Objects tab, displays information from the data dictionary (Overview and Storage view) and lists the schema changes that have been made to the cluster during the selected time period (Schema Changes view).

Associating a cluster with other entities

You an associate a cluster in the Objects tab with the following entities:

- Tables
- Columns
- Indexes

You can display additional entities by using the More... option.

The information in the Association area table is displayed on the following tabs:

- Dictionary tab—displays data dictionary information for the associated clusters, such as status, cache, creation date and when they were last modified.

- Storage tab—display storage information related to the associated clusters, such as tablespace, initial extent and next extent, and percent increased, used and free.

About the Constraint entity

The Constraint entity, in the Objects tab, displays general information that is retrieved from the data dictionary.

Associating a constraint with other entities

You can associate a constraint with the following entities:

- Columns

- Tables

- Views

- Constraints

You can display additional entities by using the More... option.

About the Index Partition and Index Sub-Partition entities

When an index partition (or index sub-partition) is displayed in the Main area of the Objects tab, the Overview and the Storage view are available. Both views display information from the data dictionary.

Associating an index partition or sub-partition with other entities

You can associate an index partition or sub-partition in the Main area of the Objects tab with the following entities in the Association area:

- Columns in the index (sub-)partition

- Index that owns the index (sub-)partition

- Columns in the partition key.

You can display additional entities by using the More... option.

The information in the Association area table is displayed on the following tabs:

- Dictionary tab—displays information from the data dictionary.

- Storage tab—displays information from the data dictionary.

- Statistics tab—displays Oracle analysis statistics.

About the Indextype entity

When an indextype is displayed in the Main area of the Objects tab, the Overview provides general information from the data dictionary.

About associating an indextype with other entities

You can associate an indextype in the Main area of the Objects tab with User Defined Types and PL/SQLs. You can display additional entities by using the More... option.

About the LOB entity

When an LOB is displayed in the Main area of the Objects tab, the Overview provides general information from the data dictionary.

About associating an LOB with other entities

You can associate an LOB in the Main area of the Objects tab with the table that it belongs to. You can display additional entities by using the More... option.

The information in the Association area is displayed on the Dictionary and Storage tabs. Both display information from the data dictionary.

About the Materialized View entity

When a materialized view is displayed in the Main area of the Objects tab, it shows general information from the data dictionary (Overview) and the definition of the materialized view (Text view).

Associating a materialized view with other entities

You can associate a materialized view in the Main area of the Objects tab with its related tables, views, and nested materialized views. You can display additional entities by using the More... option.

The information in the Association area is displayed on the Dictionary tab, which shows information from the data dictionary, and the Text tab, which shows the view's definition.

About the Method entity

When a method is displayed in the Main area of the Objects tab, it shows general information from the data dictionary (Overview) and the definition of the method (Text view).

Associating a method with other entities

You can associate a method in the Main area of the Objects tab with its return type, that is, a user-defined type. You can display additional entities by using the More... option.

About the Rollback entity

When a rollback is displayed in the Main area of the Objects tab, the following views are available:

- Overview—displaying information from the data dictionary.

- Statistics—displays information relating to the Oracle analysis statistics.

- Rollback Wait Graph—displays a graphical view of the rollback wait activity.

You cannot drill down from an imported statement to other entities.

When the rollback is selected in the Association area, information is displayed on the following tabs:

- Details—displays the same information as the Overview in the Main area.

- Statistics—displays Oracle analysis statistics; lists only rollback segments with an online status.

You can display additional entities by using the More... option.

About the Sequence entity

When a sequence is displayed in the Main area of the Objects tab, the Overview provides general information from the data dictionary.

Associating a sequence with other entities

You can associate a sequence in the Main area of the Objects tab with PL/SQLs or other objects that use it, for example triggers and synonyms. You can display additional entities by using the More... option.

About the Synonym entity

When a synonym is displayed in the Main area of the Objects tab, the Overview displays general information from the data dictionary.

When a synonym is displayed in the Association area, it displays data dictionary information.

Associating a synonym with other entities

You can associate a synonym in the Main area of the Objects tab with its referenced object. You can display additional entities by using the More... option.

About the Table Partition and Table Sub-Partition entities

When a table partition or sub-partition is displayed in the Main area of the Objects tab, the Overview and the Storage view are available. Both views display information from the data dictionary.

When a table partition or table sub-partition is displayed in the Association area, information is displayed on the following tabs:

- Dictionary—displays data dictionary information.

- Storage—displays data dictionary information.

- Statistics—displays Oracle analysis statistics.

You can associate an table partition or sub-partition in the Main area with the following entities in the Association area:

- Columns in the table (sub-)partition

- Columns in the partition key

- Table that spans the table (sub-)partition

You can display additional entities by using the More... option.

About the Trigger entity

When a trigger is displayed in the Main area of the Objects tab, the following views are available:

- Overview—displays information from the data dictionary.

- Text—displays the trigger definition.

- Schema Changes—displays changes that have been made to the trigger during the selected time period.

When a trigger is displayed in the Association area, it displays data dictionary information.

Associating a trigger with other entities

You can associate a trigger in the Main area of the Objects tab with the table or view that the trigger is defined for. You can display additional entities by using the More... option.

About the User entity

When a user is displayed in the Main area of the Objects tab, the Overview displays general information from the data dictionary.

When a user is displayed in the Association area, it displays data dictionary information.

Associating a user with other entities

You can associate a user in the Main area of the Objects tab with the tables, indexes, or views owned by the user, or with the statements that the user has executed. You can display additional entities by using the More... option.

About the User-Defined Type entity

When a user-defined type is displayed in the Main area of the Objects tab, the following views are available:

■ Overview—displays information from the data dictionary.

■ Text—displays a definition of the user defined type.

When a user-defined type is displayed in the Association area, it displays data dictionary information.

Associating a user-defined type with other entities

You can associate a user-defined type in the Main area of the Objects tab with attributes of that type, its own attributes and methods, and the objects where it is declared. You can display additional entities by using the More... option.

About the EMC Unit entity

This entity is only available, on the Objects tab, if you have installed Precise for Database & Storage. For more information on the supported platforms, see the *Precise Installation Guide*. For more information on the Storage Device entity, see

An EMC Unit is the highest level Symmetrix or Clariion entity and comprises the entire storage array. It contains devices, front-end directors, back-end directors and remote-directors. A unit may store data for many servers and applications. The EMC Unit entity displays configuration details and performance counters for units connected to the server that the currently selected instance is running on.

To determine if a unit is used by the selected instance, check the `Used by Instance` check box in the Association area. For more information on counters that are related to the current instance, drill down to the device level in either the Objects or Activity tabs.

The following views are available:

■ Overview

■ Performance

■ Load

■ Throughput

About the overview of an EMC unit

The Overview, in the Objects tab, displays general information on the Symmetrix unit, such as ID, model, and number of physical disks.

About the performance of an EMC unit

The Performance view, in the Objects tab, displays two overtime graphs that illustrate the following information:

■ The number of sequential reads and the total number of read and write requests during the selected time period.

■ The read, write, and overall cache hit ratios over the same time period.

About the load of an EMC unit

The Load view, in the Objects tab, displays an overtime graph that illustrates the megabytes of data read and written during the selected time period.

About the throughput of an EMC unit

The Throughput view, in the Objects tab, displays an overtime graph that illustrates the I/O rate (requests per second) and the throughput (megabytes per second) over the selected time period.

Associating an EMC unit with other entities

You can associate an EMC unit in the Main area of the Objects tab with the disk directors, front end directors, remote directors, and devices configured for the unit. You can display additional entities by using the More... option.

How the Objects tab can help you identify performance problems

The Objects tab helps you understand the relationship between database objects and statements.

You can identify a performance problem by doing one or more of the following:

■ Examining INIT.ORA parameters

■ About examining schema changes and tablespaces

Examining INIT.ORA parameters

INIT.ORA parameters can help you when tuning a statement, object or instance in the Objects tab.

To examine INIT.ORA parameters

1   In the Time Frame list, choose the period of time you want to analyze.

2   In the Instance list, choose the instance you want to view.

3   In the Main area, observe the Init Parameters view.

About examining schema changes and tablespaces

You can observe information on the schema changes in the Objects tab, by selecting an instance and the Schema Changes view.

To examine schema changes and tablespaces

1   In the Time Frame list, choose the period of time you want to analyze.

2   In the Instance list, choose the instance you want to analyze.

3   On the View controls in the Main area, click **Schema Changes** to open a view for that instance.

4   Now take a look at the Tablespaces entities, to understand how objects grow over time and what effect their growth has on performance. For example, the Objects tab shows storage information on tablespaces in Instance ora920, including the used and free space. Clicking a tablespace reveals details of an individual tablespace.

In another example, we analyze the number of reads and writes to a tablespace storing indexes. Here the number of physical write operations is much higher than the physical reads, probably as a result of heavy insert or update activity. This may indicate that the performance of insert or update statements is suffering because of the requirement to maintain the indexes.

For example, the number of physical reads from all the tables in tablespace XDB has increased over time. This is probably due to index range scans or full table scans that need to return more rows.

Storage Device entities

This section includes the following topics:

- About the Storage Device entities
- How the Objects tab can help you identify performance problems

About the Storage Device entities

Storage Device entities can be monitored in the Objects tab, if you have installed Precise for Database & Storage.

If you want to see the Storage Device entities that are monitored, select EMC Units in the Association area of the Objects tab. The information shown will give the DBA answers on whether there is a contention on a device due to re-tiering by performing the following actions:

Table 7-1   Actions

Action	Indicator	Possible Solution
Examine the device's response time overtime.	Marker on the Response Time graph.	Consider moving heavy files to other devices to split the I/O load or moving the data to a faster storage device.
Examine the device tier distribution and review other devices that share the same policy.	Marker on Tier Distribution graph.	Consider using a policy that secures more space on the high-end storage for the device. Verify if the policy is adequate for the type of activity.
Examine the device tier distribution and review other devices that share the same policy.	Marker on the Association table.	Consider moving other devices that use the fast (EFD) storage to another policy.

Other information related to the Storage Device entities that can be read from the Objects tab are:

- The type of storage the application is using (Thin/Thick).
- The type of policy that is associated to each device.
- The pools the application is using.
- How full the pools are.
- Who else is using the pools.
- How much of each storage Tier is being used by each device.

About the Association area

In the Association area, you can select the entities from the dropdown menu. When choosing EMC Units to see all of the EMC units (storage arrays) that are connected to the monitored server (either physically or via a remote gatekeeper), two tabs are shown at the right side of the table, Statistics - statistical data about the storage units, and

Configuration - data regarding the unit's configuration. Each tab has its own columns in the table. The following tables describe the different column headings.

Table 7-2   Statistics

Column Name	Description
Type	Type of the device: Symmetrix or Clariion.
Unit ID	The ID of the unit
Used By Instance	Shows whether the Oracle database instance is using the storage unit (the database has files on at least one storage device that belongs to that unit).
Read/Write req.	The number of read and write requests.
Load	The amount of data that the unit handled within the timeframe.
Throughput	The amount of data per second (MB/Sec.).
I/O Rate	The amount of requests to the unit per second.
Seq. Reads	The amount of sequential requests.
Cache Hit Ratio	The amount of requests that were answered by the cache in relation to the total amount (only for Symmetrix).
Read Hit Ratio	The percentage of read hits in relation to the total cache hit ratio (only for Symmetrix).
Write Hit Ratio	The percentage of write hits in relation to the total cache hit ratio (only for Symmetrix).

.

Table 7-3   Configuration

Column Name	Description
Type	Type of the device: Symmetrix or Clariion.
Unit ID	The ID of the unit
Used By Instance	Shows whether the Oracle database instance is using the storage unit (the database has files on at least one storage device that belongs to that unit).
Unit Physical Disks	The amount of physical disks in the storage array.
Devices	The amount of storage devices.
Model	The unit's model
Microcode Version	The microcode version of the unit.
Cache Size	The size of the cache.
Max. Write Slots	The maximum amount of pending write slots allowed before starting delayed fast writes (only for Symmetrix).
Max. DA Write Slots	The maximum amount of pending write slots allowed for one DA (front director) before starting delayed fast writes (only for Symmetrix).
Max. Device Write Slots	The maximum amount of pending write slots allowed for one device before starting delayed fast writes (only for Symmetrix).

The following sections handle the different Storage Device entities:

■   About the EMC Device entity

■   About the Devices on the same RAID Group entity (only for Clariion)

- About the Front End Director entity

- About the Remote Director entity

- About the Disk Director entity

- About the Port entity (for Symmetrix devices only)

- About the Hyper entity (for Symmetrix devices only)

- About the Pools entity

About the EMC Device entity

This entity is only available, in the Objects tab, if you have installed Precise for Database & Storage. For more information on the supported platforms, see the *Precise Installation Guide.*

An EMC Device is a logical entity that defines the combination of ports, front-end directors, disk directors and hypers (disk partitions) that are presented to the host operating system as a single device. A device may be configured to store data for other applications (including other Oracle instances) in addition to the selected Oracle instance. The EMC Device entity displays configuration details and performance counters for the device.

To determine if the device is used by the selected Oracle instance, check the `Used by Instance` check box in the Association area. If the device is not used by Oracle but the physical name is present (and the number of ports is greater than zero), the selected device is used by another application on the server.

The following views are available:

- Overview

- Performance

- Load

- Throughput

- Tiering (only for Thin devices with FAST auto-tiering)

- Scalability (only when choosing a specific device)(only relevant for Symmetrix)

About the overview of an EMC device

The Overview, in the Objects tab, displays general details of the Symmetrix storage device, such as the device's logical name and unit ID.

About the performance of an EMC device

The Performance view, in the Objects tab, displays two overtime graphs that illustrate the following information:

- The number of sequential reads and the total number of read and write requests during the selected time period.

- The read, write and overall cache hit ratios over the selected time period.

About the load of an EMC device

The Load view, in the Objects tab, displays an overtime graph that illustrates the megabytes of data read and written during the selected time period.

About the throughput of an EMC device

The Throughput view, in the Objects tab, displays an overtime graph that illustrates the I/O rate (requests per second) and throughput (megabytes per second) during the selected time period.

About the tiering of an EMC device

The Tiering view, in the Objects tab (only for Thin devices with FAST auto-tiering), shows how the device's content is distributed over the different Tiers (Sata, Flash, Fiber Channel) over time, other devices which share the same policy and pool, and which devices (if there are any) "stole" the current device's Tier allocation.

About the scalability of an EMC device

The Scalability view, in the Objects tab (only when choosing a specific device), shows the device's response time in relation to the requests on that device. (Only relevant for Symmetrix)

Associating an EMC device with other entities

You can associate an EMC device in the Main area of the Objects tab with disk directors, front end directors, RAID groups, hypers, ports, and oracle files to find out which files have extents on the selected device. You can display additional entities by using the More... option.

About the Devices on the same RAID Group entity (only for Clariion)

This entity is only available, in the Objects tab, if you have installed Precise for Database & Storage. For more information on the supported platforms, see the *Precise Installation Guide*.

When viewing a Thick Clariion Device (in the configuration tab), there is an option called "Devices on the same Raid Group" in the dropdown menu, that shows all of the devices who share the same Raid group as the selected device.

Note: The graphs in the upper part of the screen (Overview, Performance, Load, and Throughput) - are the same graphs of that particular selected device - with no connection to the "Devices on the same Raid Group" option.

About the Front End Director entity

The Front End Director entity is only available, in the Objects tab, if you have installed Precise for Database & Storage. For more information on the supported platforms, see the *Precise Installation Guide*.

A Front End Director controls the I/O to and from the operating system devices using the connected ports.

The following views are available:

■ Overview

■ Performance

■ Load

■ Throughput

About the overview of a front end director

The Overview, in the Objects tab, displays general details of the front-end director, such as director ID and director type.

About the performance of a front end director

The Performance view, in the Objects tab, displays two overtime graphs that illustrate the following information:

■ The number of deferred writes, read requests, and write requests during the selected time period.

■ The read, write, and overall cache hit ratios over the selected time period.

About the Load of a front end director

The Load view, in the Objects tab, displays an overtime graph that illustrates the megabytes of data read and written during the selected time period.

About the Throughput of a front end director

The Throughput view, in the Objects tab, displays an overtime graph that illustrates the I/O rate (requests per second) and throughput (megabytes per second) during the selected time period.

Associating a front end director with other entities

You can associate a front end director in the Main area of the Objects tab with devices. You can display additional entities by using the More... option.

About the Remote Director entity

The Remote Director entity, in the Objects tab, is only available if you have installed Precise for Database & Storage. For more information on the supported platforms, see the *Precise Installation Guide.*

A Remote Director is a specially configured front-end director that controls I/O to a remote front-end director on another unit. This setup is used for SRDF or disaster recovery.

The following views are available:

- Overview
- Performance
- Load
- Throughput

About the overview of a remote director

The Overview, in the Objects tab, displays general details of the remote director, such as director ID and director type.

About the performance of a remote director

The Performance view, in the Objects tab, displays two overtime graphs that illustrate the following information:

- The number of deferred writes, read requests, and write requests during the selected time period.
- The read, write, and overall cache hit ratios over the selected time period.

About the Load of a remote director

The Load view, in the Objects tab, displays an overtime graph that illustrates the megabytes of data read and written during the selected time period.

About the Throughput of a remote director

The Throughput view, in the Objects tab, displays an overtime graph that illustrates the I/O rate (requests per second) and throughput (megabytes per second) during the selected time period.

Associating a remote director with other entities

You can associate a remote director in the Main area of the Objects tab with devices. You can display additional entities by using the More... option.

About the Disk Director entity

The Disk Director entity is only available, in the Objects tab, if you have installed Precise for Database & Storage. For more information on the supported platforms, see the *Precise Installation Guide*.

A Disk Director controls the I/O to a set of physical disks.

The following views are available:

- Overview

- Performance

- Load

- Throughput

About the overview of a disk director

The Overview, in the Objects tab, displays general details of the disk director, such as director ID and director type.

About the performance of a disk director

The Performance view, in the Objects tab, displays two overtime graphs that illustrate the following information:

- The number of sequential reads and the total number of read and write requests during the selected time period.

- The read, write, and combined cache hit ratios over the selected time period.

About the load of a disk director

The Load view, in the Objects tab, displays an overtime graph of the megabytes of data read and written during the selected time period.

About the throughput of a disk director

The Throughput view, in the Objects tab, displays an overtime graph that illustrates the I/O rate (requests per second) and the throughput (megabytes per second) during the selected time period.

Associating a disk director with other entities

You can associate a disk director in the Main area of the Objects tab with devices. You can display additional entities by using the More... option.

About the Port entity (for Symmetrix devices only)

The Port entity, in the Objects tab, is only available if you have installed Precise for Database & Storage. For more information on the supported platforms, see the *Precise Installation Guide*.

A Port describes a device's connection from a front-end director to an OS device.

If a port is selected in the Main area, the Overview displays configuration information for the port.

Associating a port with other entities

You can associate a port in the Main area of the Objects tab with devices. You can display additional entities by using the More... option.

About the Hyper entity (for Symmetrix devices only)

The Hyper entity, in the Objects tab, is only available if you have installed Precise for Database & Storage. For more information on the supported platforms, see the *Precise Installation Guide.*

A hyper is a disk partition on a physical disk where data for a device is stored. Data is usually striped or mirrored over a set of hypers. It is the lowest-level Symmetrix storage entity.

If a hyper is selected in the Main area, the Overview displays configuration information for the hyper.

Associating a hyper with other entities

You can associate a hyper in the Main area of the Objects tab with all the devices that are on the same disk as the hyper itself. You can display additional entities by using the More... option.

About the Pools entity

The Pools entity, in the Objects tab, is only available if you have installed Precise for Database & Storage. For more information on the supported platforms, see the *Precise Installation Guide.*

The following views are available:

■ Overview

■ Performance

■ Load

■ Throughput

About the overview of a pool

The Overview, in the Objects tab, displays general details of the pool.

About the performance of a pool

The Performance view, in the Objects tab, displays two overtime graphs that illustrate the following information:

■ The number of deferred writes, read requests, and write requests during the selected time period.

■ The read, write, and overall cache hit ratios over the selected time period.

About the load of a pool

The Load view, in the Objects tab, displays an overtime graph that illustrates the megabytes of data read and written during the selected time period.

About the throughput of a pool

The Throughput view, in the Objects tab, displays an overtime graph that illustrates the I/O rate (requests per second) and throughput (megabytes per second) during the selected time period.

Associating a pool with other entities

You can associate a pool in the Main area of the Objects tab with all the devices that are on the same disk as the hyper itself. You can display additional entities by using the More... option.

How the Objects tab can help you identify performance problems

The Objects tab helps you understand the relationship between database objects and statements.

You can identify a performance problem by doing the following:

■ Filtering Symmetrix device entries and criteria

Filtering Symmetrix device entries and criteria

For our example, the Objects tab reveals details for EMC Unit 000282503660. This unit has 48 physical disks, 17 of which are used by the sun4 server. 434 devices are configured for the unit. All of the devices listed in the Association area are used by Oracle instance ora920 (indicated by the column Used by Instance).

If there are many devices associated with this unit, you can limit your search by using the More option in the Association area.

To filter device entries from a More option

1   In the Association area, click **More** and the Associate with dialog box opens.

2   Click the **Entries** tab to define device entries to filter.

3   In the Populate with list, select the appropriate device entry.

4   In the Sort entries by list, select how you want to sort the entries.

5   Use the **A** (ascending) or **D** (descending) button to sort all selections.

6   In the Display top .... entries only list, select the number of entries to view for each search.

After defining the device entries, you can define criteria for each device search.

To filter device criteria from a More option

1   In the Association area, click **More** and the Associate with dialog box opens.

2   Click the **Criteria** tab to define device criteria for a filter.

3   From the lists on the left, select the rows to filter.

4   In the middle lists, select the appropriate variable.

5   In the lists on the right, type in the appropriate filter or filters.

6   Click **OK** to approve your selections, or **Cancel** to cancel your selections.

Evaluating explain plans

This section includes the following topics:

About the SQL tab

The process of explaining statements is a prerequisite for tuning. The explain process is designed to clarify the access path chosen for a statement and translate it into a visual medium. The SQL tab helps you identify bottlenecks and guides you through the steps required to tune SQL statements.

After a statement is explained, the explain results are stored in the PMDB. This information includes the objects referenced by the statement and the operations performed on these objects. The top statements are automatically explained every day. You can control how many statements to explain using a setting for the Explain Statements PMDB process in AdminPoint.

For more information, see the *Administration Guide*.

Statements are also parsed by a proprietary Precise for Oracle parser. Parsing allows Precise for Oracle for example to highlight statement objects from the execution plan and to format execution plans. The number of statements that are parsed is again governed by a setting in the PMDB. For more information, see the *Precise Administration Guide*.

Understanding the execution plan chosen by the Oracle Optimizer is extremely important when tuning your application. You can ensure optimal system performance by ensuring that the best plans are used for your queries.

Precise for Oracle provides you with a special tab only for this purpose, the SQL tab. This tab provides several different views of the same execution plan that makes analyzing long and complicated queries much easier.

If you are running Oracle 10g and later, you can also access the execution plan associated with a CREATE TABLE as SELECT statements. In addition to examining the plan, this lets you gain additional insight into a problem by analyzing the findings displayed for these statements.

Note: The statement will only explained. It will not be parsed.

Note: To analyze an execution plan, Precise for Oracle provides you with a full picture of the objects (table, indexes, and so on) participating in the plan. You can change a plan by modifying your query or changing the schema, for example by adding an index. To assist in this, Precise for Oracle can provide you with index recommendations for your statement.

One technique to identify the source of the problem is to view the historical information of statements, showing performance degradation. A change in the schema, volume, or execution plan may explain the impact on performance.

An important concept within the SQL tab is that of alternative or related SQL. You can take a statement and create different alternatives for it as part of the tuning process. There are a number of different ways of doing this. For example, you may re-write a statement yourself so that it gives the same result in a more efficient manner; or you may click **Actions>Generate Alternatives** to get Precise for Oracle to generate alternatives for you; or you may change a statement's execution plan by simulating the addition of an index in the What-If tab.

The related SQL is stored along with the original in the same cabinet and folder. However, it does not have an independent life of its own; it is always tied to the original. It does not have a resource consumption profile because it has not been captured by the Collector; it only exists in the SQL tab. The related SQL is intended to be transient until one of the alternatives is implemented as the replacing statement.

The following table lists the tabs and entities from which you can launch the SQL tab.

Table 8-1   Launching the SQL tab in context

Tab	Entities
Dashboard	Statement
Current	Statement, Active or Current Session that is currently executing a statement.
Activity	Statement, PL/SQL
Objects	Statement, PL/SQL
What-If	Statement that is affected by one or more index evaluations.

In the Current, Activity, and Objects tabs, you can either launch the SQL tab by clicking the SQL tab button when the selected entity in the Main area is a statement, or by selecting a statement in the Association area and clicking the Tune icon that appears before the SQL text.

Note: For a PL/SQL, the SQL tab offers only limited functionality.

If the SQL tab is opened with no statement in context, a message prompts you to open a statement.

If Precise has already collected the steps of a real plan, it appears highlighted in blue; otherwise, it appears highlighted in gray. If you click on a gray statement, Precise for Oracle will access the monitored instance and try to retrieve the selected plan's steps.

There can be statements with many execution plans. Only the latest 7 plans appear on the tree. Click `More...`, if available, to view additional execution plans.

see "About the Dashboard tab" on page 43.

see "About the Current tab" on page 51.

see "About the Activity tab" on page 61.

see "About the Objects tab" on page 87.*

How the SQL tab is structured

The SQL tab lets you analyze execution plans and explain results so that you can tune statements and achieve optimal results.

The following controls determine the information displayed in the SQL tab:

■   Related SQL selector

■   View tabs

■   Tuning Action buttons

About the Related SQL selector

The Related SQL selector appears only in the Plan view if a statement has a related SQL. The first time you visit the SQL tab, it will not be visible. It lets you choose one of the alternative statements that are associated with an original statement. An original statement is one captured by the Collector, imported, or entered manually. Full details on the related SQL are held in the Related SQL view.

Do any of the following to create a related SQL:

■   When you edit a statement and save an alternative.
    see "Editing an existing statement" on page 154.

■   When you click New Alternatives in the Related SQL view, and Precise for Oracle creates an alternative SQL for you.
    see "Generating new alternatives" on page 156.

■   When an index recommendation is made in the What-If tab and you click Compare to switch to the SQL tab, in which case a copy of the statement is saved along with its new virtual index and execution plan.
    see "About the What-If tab" on page 159.

About the View tabs

The SQL tab divides the information on the selected statement into different views. Each view has a different layout and presents different information.

The following views are available:

■   Plan

■   Run Alternatives

■   More ...

About viewing the execution plan of a statement

The Plan view displays the execution plan of a statement and related information, such as statistics, referenced objects, and operations performed in the execution plan. It lets you assess and tune statements based on real information.

Two types of execution plans are available when launching to the SQL tab:

Real execution plan	The plan which was collected from Oracles V$ tables. Additional real-time statistics are available for these plans. For example, you can view a breakdown of In Oracle time for a plan as compared to other plans, or a breakdown of its I/O when accessing Oracle objects.
	When launching to the SQL tab with a specific SQL statement in context, Precise for Oracle presents actual execution plan information, including actual plan steps, and information and statistics for multiple plans. For example, if a statement has several different execution plans, all are displayed.
	Note: This feature is only available for Oracle 10.1.0.4 monitored instances and later.
Estimated execution plan	Depending upon the version of Oracle running on your system (such as Oracle 8i), you may be only able to generate and run an estimated execution plan. At times, real execution plan information is lost (as when they are removed from the system before Precise for Oracle could access them). If this happens, all real-time statistics (such as, I/O, and In Oracle) are considered to belong to other plans.

The Plan view is divided into two panes. The Execution Plan tree is displayed in the left pane. The Details area is displayed in the right pane. The information displayed in the Details area is controlled by the information tabs (Highlights, Expanded Text, Objects, and More...) that are located above this area.

The following table describes the information displayed in the Plan view.

Table 8-2   Information tabs

Information Tabs	Description
Highlights	Displays the execution plan's formatted text and findings that enable you to identify the most probable cause of the problem your statement is experiencing.
Expanded Text	Shows all views and synonyms expanded inline allowing you to match the access steps to the version of SQL that Oracle is actually executing.
Objects	Displays information on all referenced objects in the execution plan, including their indexes and columns. Statistical details and general details are displayed for each object.
More ...	Shows the following additional information for a plan: ■ Statistics view 　Displays statistical information on all steps in the execution plan tree ■ Workshop view 　Displays details on the execution plan, where the statement is stored, and the SQL text of the statement

Note: If you use the Oracle rule-based Optimizer, cost information is not available. If tables and indexes are not analyzed, statistics changes are not available.

About the Execution Plan tree

The Execution Plan tree displays both the real execution plan (Oracle 10.1.0.4 and later) and the estimated execution plan, of the specified statement. When you analyze the access plan of a selected statement, you can examine the access path that was chosen by the Oracle Optimizer.

The execution plan's text or expanded text (with all referenced views expanded) appears both at the bottom of the execution plan tree (by clicking the plus sign (+) to view text) or at the top of the Details area (right pane). The text that relates to the selected step in the Plan tree is highlighted. This lets you view the text of the statement, the execution plan, and additional information, such as the objects referenced by the statement, all at once.

see "About expanded text of a statement in the Plan view" on page 135.

Actions that can be performed on the tree
You can perform the following actions on the execution plan tree:

■ The Playback controls, located at the top of the tree, enable you to freely move within the execution plan of an explained statement.

■ By moving the pointer over the execution plan steps, you can view more statistical information, such as Estimated Cost, Estimated Rows, and Estimated I/O Cost, and description of the specified step.

■ Selecting a specific step highlights nested steps and affects the information displayed in the information tabs: the relevant SQL text is highlighted on the Text tab; the referenced tables, indexes, and columns are shown on the Objects tab; and the relevant step is highlighted on the Statistics tab.

■ Clicking the plus sign (+), located at the bottom of the tree, displays and highlights the statement's formatted text or expanded text.

About the text of a statement in the Plan view

The Highlights tab, in the SQL tab, displays the execution plan's formatted text and findings that enable you to identify the most probable cause of the problem your statement is experiencing. The Findings table and Findings Details area provide additional information on what steps you can take to continue your analysis and lets you launch to the appropriate tab, with the statement in context.

The text that relates to the selected step in the execution plan tree (on the left) is highlighted. The highlight color is based on the selected step type. The structured text of the execution plan is displayed in the right pane.

When a table is highlighted, all table columns appearing in the text are highlighted in the same color as the table. The following is an explanation of the major step types.

About steps that access a table or index

The table and its columns are highlighted and color-coded. The columns that are accessed by an index are underlined.

About sort steps

All tables whose columns were used by sort steps are highlighted and color-coded. Each table and its corresponding columns have their own color. The columns that participate in the sort are underlined.

About join steps

The methodology for highlighting nested loop steps differs slightly from that of the merge join and hash join steps.

Nested loop steps are highlighted as follows: all tables and their corresponding columns participating in a sub-tree of the outer table (the first step that is a direct descendant of the nested loop step) are highlighted in blue; all tables and their corresponding columns participating in a sub-tree of the inner table (the second step that is a direct descendant of the nested loop step) are highlighted in red.

The highlight methodology for the other join steps (hash join and merge join) differs slightly. The first table and all its corresponding columns are highlighted in blue. The columns that are used for the join are underlined. The second table and all its corresponding columns are highlighted in red. The columns that are used for the join are underlined.

About expanded text of a statement in the Plan view

Oracle provides views and synonyms as a way of simplifying the complexity of an application. However, the Oracle Optimizer must generate an execution plan against the correct underlying tables even if they do not appear in the SQL text. The Expanded Text tab shows all views and synonyms expanded inline enabling you to match access steps to the version of SQL that Oracle is actually executing.

In the SQL tab, a statement with the original text appears in the bottom left corner and the expanded text on the right at the top. The expanded view text is shown in bold. The statement highlighting picks out the columns within the view referenced in the explain plan.

About the objects that are referenced by the execution plan

The Objects tab displays three tables (Tables in use, Indexes defined on table, Columns in table or index) that list all referenced objects in the execution plan, including their indexes and columns. Statistical details and general details are displayed for each object.

Each table has a title denoting the entities highlighted. The lower table could have a title denoting either the columns of a table or the columns of an index.

Note: If tables and indexes are not analyzed, statistics information is not available.

About tables in use

The Tables in use lists all the referenced tables in the tree.

The following table describes the information displayed for the tables that are accessed in the execution plan.

Table 8-3   Information on tables used in the plan

Column	Description
TUNE	Launches the Objects tab so that you can focus on the specified table.
🔍	Locates and highlights all the steps in the execution plan that access the specified table.

Table 8-3 Information on tables used in the plan

Column	Description
Used	Indicates whether the specified table is used in the selected step in the execution plan tree.
Table	Displays the table name. A ToolTip displays the full name in the following format: *Owner.Table_Name*.
Rows	Number of rows in the table based on data dictionary statistics.
Blocks	Total number of blocks in the table.
Non-Empty Blocks	Number of used blocks in the table.
Chained Rows	Number of chained rows in the table.
Partitioned	Indicates whether the table is a partitioned table.
Index Organized	Indicates whether the table is an index organized table.
Temporary	Indicates whether the table is a temporary table.
Object ID	ID of the object.
Owner	Owner of the table.
I/O Wait	Time of IO wait in the specific plan.

About the indexes defined for the selected table

The Indexes defined for the selected table lists all the indexes that are used to access the selected table, whether they were used in the explain plan or not.

The following table describes the information displayed for the indexes that are used to access the selected table.

Table 8-4 Information on the indexes used to access the selected table

Column	Description
	Launches the Objects tab so that you can focus on the specified index.
	Locates and highlights all the steps in the execution plan that access the specified index.
Used	Indicates whether the specified index is used in the selected step in the execution plan tree.
Index	Index name. A ToolTip gives the full name, such as *Owner.Index_name*.
Unique	Indicates whether the index is unique.
Type	Index type, such as Normal (B*Tree), Bitmap, and so on.
Partitioned	Indicates whether the index is a partitioned index.
Blocks	Total number of blocks in the index.
Leaf Blocks	Number of leaf blocks in the index.
Distinct Keys	Number of distinct keys or values in the index.
B-Level	Depth of a B*Tree index.

Table 8-4 Information on the indexes used to access the selected table

Column	Description
Clustering Factor	Clustering factor of the index.
	The clustering factor is an important factor in determining how efficiently an index range scan will retrieve data from the table. It measures the degree to which the data in the index and its table are in the same order or, put another way, the probability that the next row to be fetched from the table is in the same block as the current row. It can vary between the number of blocks in the table (the best case because they are in the same order) and the number of rows in the table (the worst case because they are completely out of sync). The clustering factor tends to become worse over time as data is inserted and deleted. Note that the clustering factor makes no difference for a unique index lookup.
I/O Wait	Time of IO wait for inex in the specific plan.
Last Analyzed	Time when the table was last analyzed.
Object ID	ID of the Object.
Owner	Owner of the table.
Locality	Indicates if the index is local or global.

About the columns in table or index

The Columns in table or index lists all columns in the selected table displayed in Tables Used in Plan. When an index is selected in Indexes of Table, the first column constitutes the index column sorted by the position of the column in the index, and marked with an ascending or descending arrow.

The following table describes the information displayed for the columns of the table.

Table 8-5 Information on the columns of a table

Column	Description
▤	Launch the Objects tab so that you can focus on the specified column.
Column	Column name.
Type	Physical storage type of the column.
Distinct Values	Number of distinct values.
Key Number	Position of the column in the index selected in Indexes of Table; otherwise blank.
Asc/Desc	Indicates whether the column is part of the selected index and whether it is sorted in ascending or descending order.
In Clause	Location of the column in the statement. The column may appear in the Select clause, Where clause, Group by clause, Having clause, and Sort by clause.
Indexable	Indicates whether the column could be used as part of an index. A column may be indexable even though it is not currently part of any index. Alternatively, a column may not be indexable, even though it is currently part of an index, if the column does not appear within the where clause (or group by, or order by clauses) or there is a function on the column (and there is no function-based index). The Optimizer will not be able to use an index if the leading columns in the index are non-indexable (unless it can employ a skip-scan search).
Indexes	List of indexes in which the column is used.

About statistical information on all operators in the execution plan

The Statistics tab displays statistical information on all steps in the execution plan tree. You can sort steps by their estimated cost or by any other column in the table.

The following table shows the information displayed in statistics table for the execution plan.

Table 8-6   Execution plan statistics

Column	Description
🔍	Locates and highlights the step in the Execution Plan tree that matches the selected step in the grid.
Step ID	Displays the number of the explain plan step.
Step	Provides a short description of the explain plan step.
Estimated Cost	Estimated cost of the current operation. A high cost value may indicate a problem in the current implementation of the operation. Check the Estimated I/O cost and Estimated CPU cost values to determine whether the operation is an I/O consuming operation or a CPU consuming operation (or both).
Estimated Rows	Estimated number of rows returned by this step.
Estimated Bytes	Estimated number of bytes returned by this step.
Estimated CPU Cost	The estimated CPU cost of the current operation. A high cost value may indicate a problem in the current implementation of the operation.
Estimated I/O Cost	Estimated I/O cost of the current operation. A high cost value may indicate a problem in the current implementation of the operation.
Partition ID	Name of the table or index partition, if the step involves a partitioned table or index.
Partition Start	Low value of the partition key.
Partition Stop	High value of the partition key.
Access Predicates	Predicates used to locate rows in an access structure; for example start or stop predicates for an index range scan (as defined in Oracle documentation).
Filter Predicates	Predicates used to filter rows before producing them (as defined in Oracle documentation).
#	Step #.
In Oracle	In Oracle time for the specific step.
Cost	Cost for the specific step.
Other Tag	Data as seen in Oracle for specific step.
Plan Hash	Plan hash value.

About information on statement plan, location, and text

The Workshop tab displays details on the execution plan, where the statement is stored, and the SQL text of the statement.

The following table describes how the information on the Workshop tab is structured.

Table 8-7   Structure of information on the Workshop tab

Item	Description
Explained on	Date the latest explain plan was first generated.
Cost	Estimated cost calculated by the Oracle cost-based Optimizer.
Parsing User	In the case of an imported or manually saved statement, this is the Oracle user that was specified at the time; in the case of an automatically captured statement, it is the first Oracle user that the Precise for Oracle Collector found running the statement.
Cabinet	Cabinet where the statement is stored.

Table 8-7   Structure of information on the Workshop tab

Item	Description
Folder	Folder within the cabinet where the statement is stored.
Origin	Source of the statement. The Precise for Oracle Collector will automatically capture most statements when they are executing in Oracle, but some may be imported from source files or saved manually. Possible values are: ■ Automatically collected ■ Generated as related SQL ■ Predicted plan (What-If) ■ Entered by user ■ Imported from source file ■ Generated as related SQL by user
Saved on	Date that the statement was saved.
Parsing Information	Indicates whether Precise for Oracle has performed an extra level of parsing above that performed by Oracle to support more detailed analysis and syntax color highlighting.
Comment	User-defined comment that can be entered against a statement when it is saved. see "Editing the properties of a statement" on page 31.

About the Recommend button

The Recommend button above the right pane is used to activate the Oracle Advisor (default) or Precise Advisor. The result of the recommendation is shown in the What-If Tier.

We recommend you to use the Oracle Advisor, as this provides an extended Oracle functionality. The recommendation van be evaluated by the What-If analysis.

Note: The Precise Advisor on the SQL tab will be used instead of the Oracle Advisor when: the Oracle Advisor is not installed, the Oracle version does not support it, or the usage of the Oracle Advisor was manually disabled in our application (for more information on how to manually disable the default Oracle Advisor, see "Disabling the Oracle Advisor" on page 139.

Disabling the Oracle Advisor

When you decide to disable the Oracle Advisor, you have to make changes in the registry file per instance. The result will be that the Precise Advisor will be used with the following limitation:

■   When clicking the Recommend link in the Finding on the Activity tab the following message will appear:

```
Unable to perform the Recommend function. Verify that the Oracle Advisor is enabled
in the Precise registry.
```

To disable the Oracle Advisor

1   For each instance, open the *<SID_name>*.xml file in the following location:

```
\products\i3fp\registry\products\indepth-oracle\hosts\<host_name>\oracle
```

2   Under the <recommend> tag, add/edit the following tag:

```
<oem-enabled>N</oem-enabled>
```

   Where N disables the Oracle Advisor.

3   Save the *<SID_name>*.xml file.

4   Restart the Precise for Oracle FocalPoint.

About the Run Alternatives view

The Run Alternatives view lets you run different statement alternatives under different conditions (bind variables and session parameters) and view details on how the statement was run.

Each row in the Alternatives table (displayed in the left pane) can represent details of one of the following:

■ The original statement.

■ Any of the statements alternatives.

■ If you decide to run the original statement or any of its alternatives, a new row, describing the details of the run, is added to the table.

The following table describes the information displayed for the alternative statements.

Table 8-8 Alternative Statements

Column	Description
TUNE	Launches to the Plan view with the statement in context and lets you view the statement's execution plan, in the Highlights tab.
🔍	Lets you compare the alternative statement with the original statement.
Status	Displays whether or not the statement was successfully run, or whether it is still running or has timed out.
Name	The information displays for the following statement names: ■ For an alternative statement, it displays the alternative statement's name. ■ For the original statement, it displays the original statement's name. ■ For an alternative run of a statement, it displays the name of the alternative or statement used as a base for running the alternative statement.
Timestamp	Displays the time the statement was run.
Plan Hash Value	Displays the hash value for the plan associated with the selected statement.
Statement ID	Displays the statement ID.
Duration (Avg.)	Displays how long the statement ran.
Cost	Displays the average cost of the plan.
Logical I/O (Avg.)	Displays the average number of logical I/O operations per executed statement.
Physical Reads	Number of physical reads from disk.
Hit Ratio	Number of physical reads or logical reads, expressed as a percentage.
Table Scans	Number of full table scans performed.
Write Requests	Number of requests to write data to disk, usually of temporary data during joins and sorts.
Sorts	Number of sorts performed.
Sorted Rows	Total number of sorted rows.
Table Scans by Row ID	Number of accesses to a table by Row ID. A Row ID contains the address of a row in a table and is the fastest way to gain access to an individual row, although not necessarily to multiple rows. Access via Row ID usually follows an index scan, because Row IDs are stored in the leaf blocks of an index.
Table Scan Rows Gotten	Number of rows retrieved from tables.
Table Scan Blocks	Number of blocks fetched from tables.

Table 8-8   Alternative Statements

Column	Description
Recursive Calls	Number of recursive SQL calls that Oracle made to the data dictionary while executing the statement.
Processed Rows	Number of rows processed during the execution. This includes rows retrieved from tables, indexes, and temporary segments. It may be many more than are returned.
Host Name	Displays the host name of the server that the statement is running on.
Description	If you chose to run an alternative of the statement, this field displays the description assigned to the run in the Run Statement dialog box.
Attempted Executions	Number of executions started.
Actual Executions	Number of executions completed.

About the text of an alternative statement

This view displays the text associated with the alternative statement.

About the plan tree of an alternative statement

This view displays the plan associated with the alternative statement, or the original statement.

About the Run Info view

The Run Info view displays bind variables and session parameters used during the run of the selected alternative statement.

About the Extended Statistics view

The Extended Statistics view displays additional statistics on the alternative runs.

The following table describes the additional statistics information displayed for each alternative.

Table 8-9   Extended Statistics table

Column	Description
#	Displays the sequential number of the step
Step ID	Displays the number of the explain plan step.
Step	Provides a short description of the explain plan step.
Rows	Displays the number of rows returned by this step.
Time	Displays when the alternative statement was last run.
Blocks	Displays the total number of blocks in the alternative statement.
Executions	Displays the number of executions of the alternative statement

About the History view of resource consumption

The History view displays the resource consumption over time vs. cost, and changes over time. This lets you determine which changes led to performance problems.

The History view is divided into two areas:

■   Main area

■   Change History area

Note: If you use the Oracle rule-based Optimizer, cost information is not available. If tables and indexes are not analyzed, statistics changes are not available.

About the Main area of the History view

The Main area displays resource consumption over time, estimated cost over time, changes over time, and executions over time.

The following views are available in the Main area:

- Overview

- Text

About the overview of history information

The Overview, in the SQL tab, displays the following overtime graphs:

Table 8-10 Graphs

Graph	Description
In Oracle (Avg.)	Illustrates the average resource consumption of the statement over time; does not detail the states or substates.
Cost	Displays the estimated cost calculated by the Oracle cost-based optimizer over time. This information is generated whenever you explain the statements using Precise for Oracle.
Changes	Marks significant events that affect the statement and the objects that it accesses over time, namely analyze statistics changes, schema changes, and execution plan changes.
	The execution plan changes information is generated whenever you explain the statements using Precise for Oracle, either using the PMDB process or the SQL tab.
	The statistics changes information is generated whenever you run the PMDB process Collect Schema Changes. It scans all the objects referenced by the statement.
	The schema changes information is generated whenever you run the PMDB process Collect Schema Changes. It scans all the objects referenced by the statement.
	see "About the Change History area" on page 143.
Executions	Displays the number of executions of the selected statement over time.

About the text of a statement in the History view

The Text view, in the SQL tab, displays the full SQL text of the statement and information on the statement and execution plan, as follows:

Table 8-11 Text view display

Information	Description
Cabinet	The cabinet where the statement is stored
Folder	The folder within the cabinet where the statement is stored.
Origin	The source of the statement. The Precise for Oracle Collector automatically captures most statements when they are executing in Oracle, but you can also import statements from source files or save them manually.
	A statement can be automatically collected, generated as related SQL, part of a predicted plan in the What-If tab, entered by users, imported from a source file, or generated as a related SQL by a user.
Saved on	The date that the statement was saved.

Table 8-11   Text view display

Information	Description
Parsing Information	Indicates whether Precise for Oracle has performed an extra level of parsing above that performed by Oracle to support more detailed analysis and syntax color highlighting.
	For statements that are explained in the background, parsing must be enabled in the Explain Statements PMDB process in AdminPoint (see the *Precise Administration Guide*) for statements that are explaining in the background. For statements that you explain in the SQL tab, parsing must be enabled from **Settings>General Settings>SQL**.
	By default, parsing is enabled. Some statements cannot be parsed.
	See the *Precise Administration Guide.*
	see "About SQL settings" on page 27.
Explained on	The date the latest explain plan was generated.
Cost	The estimated cost calculated by the Oracle cost-based optimizer.
Parsing User	In the case of an imported or manually saved statement, the Oracle user that was specified at the time; in the case of an automatically captured statement, the first Oracle user that the Precise for Oracle Collector found running the statement.

About the Change History area

The Change History area displays the changes that may have affected the performance of the statement over time.

Information on the following type of changes is displayed:

- ■ All changes
- ■ Schema changes
- ■ Statistics changes
- ■ Execution plan changes
- ■ Run statistics history

About viewing all changes

The All Changes view displays a list of all changes made. The changes can be statistics changes, schema changes, and execution plan changes. If there are many changes, you may want to use one of the other change types or filter your search using the More... option.

see "Associating entities with data that meets specific criteria" on page 30.

The following table describes the information displayed in the All Changes table.

Table 8-12   All Changes table

Column	Description
TUNE	Switches to the Plan view and displays the first access plan that was created by the Explain Statements process after the specified change was detected.
	If the object that was changed is one of the entities in the Objects tab (table, view, index, or column), this icon will launch the Objects tab with the selected object in context. Also, if the change type is Access Plan Changed, the icon will launch the Compare view with the specified access plan compared to the last access path, as detected by the Explain Statements process.
Timestamp	Indicates the date that the change was detected.

Table 8-12   All Changes table

Column	Description
Change Type	Displays the type of change detected. Can be one of the following values: ■ Table Created ■ Table Dropped ■ Table Altered ■ Index Created ■ Index Dropped ■ Index Altered ■ Table Statistics Changed ■ Initial Execution Plan ■ Different Execution Plan
Object	Indicates the object that was changed, such as table or index.
Change Details	Displays information about the change that was made. For example, if the Change Type is Different Execution Plan, the Change Details column gives the estimated cost.

About schema changes

The Schema Changes view displays a list of all schema type changes made.

The following table describes the information displayed in the Schema Changes table.

Table 8-13   Schema Changes table

Column	Description
	Switches to the Plan view and displays the first access plan that was created by the Explain Statements process after the specified change was detected.
	If the object that was changed is one of the entities in the Objects tab (table, view, index, or column), this icon will launch the Objects tab with the selected object in context.
Timestamp	Indicates the date the change was detected.
Change Type	Displays the type of change detected. It can be one of the following values: ■ Table Created ■ Table Dropped ■ Table Altered ■ Index Created ■ Index Dropped ■ Index Altered
Object	Indicates the object that was changed, such as table or index.
Columns	Indicates the columns in the object that were affected by the change.

About statistics changes

The Statistics Changes view displays a list of changes made to Oracle statistics on tables, indexes, and columns.

The following table describes the information displayed in the Statistics Changes table.

Table 8-14   Statistics Changes table

Column	Description
	Switches to the Plan view and displays the first access plan that was created by the Explain Statements process after the specified change was detected.

Table 8-14   Statistics Changes table

Column	Description
	Launches the Objects tab with the selected instance.
Timestamp	Indicates the date the change was detected.
Object	Indicates the object that was changed, such as table or index.
Type	Displays the type of object, such as table or index.
Rows	A change in the number of rows.
Non-Empty Blocks	A change in the number of non-empty blocks.
Free Space (Avg.)	A change in the average free space in a block.
Chained Rows	A change in the number of chained rows.
Distinct Keys	A change in the number of distinct keys in an index.
Leaf Blocks	A change in the number of leaf blocks in an index.
Clustering Factor	A change in the clustering factor of an index.
	The clustering factor is an important factor in determining how efficiently an index range scan will retrieve data from the table. It measures the degree to which the data in the index and its table are in the same order or, put another way, the probability that the next row to be fetched from the table is in the same block as the current row. It can vary between the number of blocks in the table (the best case because they are in the same order) and the number of rows in the table (the worst case because they are completely out of sync). The clustering factor tends to become worse over time as data is inserted and deleted. Note that the clustering factor makes no difference for a unique index lookup.

About execution plan changes

The Execution Plan Changes view displays a list of all the access plans of the statement as detected by the Explain Statement process.

The following table describes the information displayed in the Execution Plan Changes table.

Table 8-15   Execution Plan Changes table

Column	Description
	Switches to the Plan view and displays the first access plan that was created by the Explain Statements process after the specified change was detected.
	Opens the access plan in the Compare view and lets you compare the specified access plan with the last access plan, as detected by the Explain Statements process.
Timestamp	Indicates the date the specified access plan was detected by the Explain Statements process.
Cost	Estimated cost calculated by the Oracle cost-based Optimizer.
Nested Loops	Number of nested loop operations performed.
Hash Joins	Number of hash joins performed.
Merge Joins	Number of merge joins performed.
Sorts	Number of sorts performed.
Table Scans	Number of full table scans performed.
Index Scans	Number of index scans performed.

Table 8-15 Execution Plan Changes table

Column	Description
Steps	Bar graph showing the number of steps in the execution plan and a visual breakdown of the different types of operations performed.

About the run statistics history

The Run Statistics History view displays Oracle run statistics from when the statement has been run manually from within the tool. Any changes that it makes are rolled back. The only overhead is the execution time. A time-out may be specified in the Run dialog box to prevent the statement from taking too long.

See "Running a statement" on page 155.

The Run Statistics History is a useful way of testing the performance of different SQL alternatives to see which is the fastest before it is put into production. Until it is run, the Collector will not have captured its execution performance. Another use is to benchmark the performance of certain key statements over time.

The information in the Run Statistics History table is displayed on the following tabs: General and Statistics. The statistics information is the same as you would get from the V$SESSTAT table or if you set the autotrace function in SQL*Plus.

The following table explains the information shown on the General tab.

Table 8-16 Information shown on the General tab

Column	Description
Timestamp	Displays when the statement was run.
Duration	How long the statement took to run.
Processed Rows	Number of rows processed during the execution. This includes rows retrieved from tables, indexes, and temporary segments. It may be many more than are returned.
Physical Reads	Number of physical reads from disk.
Attempted Executions	Number of executions started.
Actual Executions	Number of executions completed.
Timed Out	Denotes whether the timeout specified in the Run dialog was reached. The timeout is for all executions.
Timeout (Sec.)	Timeout in seconds that was specified in the Run dialog box.
Status	One of the following states: Started, Completed, or an error message if an error has occurred.

The following table explains the information shown on the Statistics tab.

Table 8-17 Information shown on the Statistics tab

Column	Description
Timestamp	When the statement was run.
Logical Reads	Number of logical reads.
Hit Ratio	Number of physical reads or logical reads, expressed as a percentage.
Write Requests	Number of requests to write data to disk, usually of temporary data during joins and sorts.
Sorts	Number of sorts performed.
Sorted Rows	Total number of sorted rows.
Table Scans	Number of full table scans performed.

Table 8-17   Information shown on the Statistics tab

Column	Description
Table Scans by Row ID	Number of accesses to a table by Row ID. A Row ID contains the address of a row in a table and is the fastest way to gain access to an individual row, although not necessarily to multiple rows. Access using Row ID usually follows an index scan, because Row IDs are stored in the leaf blocks of an index.
Table Scan Rows Gotten	Number of rows retrieved from tables.
Table Scan Blocks	Number of blocks fetched from tables.
Recursive Calls	Number of recursive SQL calls that Oracle made to the data dictionary while executing the statement.

Comparing execution plans

The Compare view lets you compare a statement's text and its execution plan with one of its alternatives or with one of its historical execution plans. By default, the last execution plan is displayed in the left pane. The execution plan you want to compare it with is displayed in the right pane. The statement's text is displayed at the bottom of each execution plan pane, and the selected step is highlighted.

The Compare view is usually opened in context from other views within the SQL tab (such as History or Related SQL) or other tabs (such as What-If). To open the Compare view in the context of the selected execution plan and compare it with the latest execution plan, click the Compare icon.

    Click the Compare icon to display the list of execution plans available for a statement.

To compare the execution plans of a specified statement

1   Click the Compare icon.
2   From the list of access plans that belong to the statement and related SQL, choose the access plan or related SQL you want to use for a comparison from the list.
3   Click **OK**.
    The statement you selected is displayed in the right pane. The statement's text is displayed in the bottom pane.

About the All Plans view

The All Plans view displays information on all the available plans. Use this view to analyze and compare the plans to determine which is the best plan to run. Every row in the All Plans table (displayed in the left pane), describes a different plan belonging to the current statement.

The following table describes the information displayed in the All Plans table.

Table 8-18   All Plans table

Column	description
	Launches to the SQL tab with the selected plan in context.
	Launches to the Compare view so that you can compare the current plan and the selected plan.
Plan Hash Value	Displays the plan's hash value, as computed by Oracle.
Plan Type icon	Indicates whether a plan is a real or estimated execution plan.
First Detected	For actual plans, displays when the plan was first encountered. For estimated plans, displays when the statement was first explained and the plan was produced.

Table 8-18   All Plans table

Column	description
Last Detected	For actual plans, displays when the plan was last encountered.
	For estimated plans, displays when the statement was last explained and the plan was produced.
In Oracle	Displays the total time spent in Oracle by statements which were using this plan during the selected time frame.
Duration (Avg.)	Displays the average amount of time it took the plan to run.
Cost	Displays the last cost retrieved for the plan.
Executions	Displays the number of times the plan was executed, during the selected time frame.

About the all plans overview

The All Plans overview lets you analyze how plans changed over time. This information is displayed in the following graphs:

■   In Oracle (Avg.) vs. Executions

It displays the average time spent in Oracle vs. the number of executions, over time. This graph lets you analyze the scalability of the different plans displayed in the All Plans table.

It notices that at a certain point in time one of the plans was replaced by another and that the AVG in-oracle time of the second plan is considerably higher than the first one.

■   In Oracle

It displays the total time spent in Oracle by statements which were using this plan over time.

■   Cost

It displays the cost of the retrieved for the plan over time.

About the Plan tree

The Plan tree displays execution plan of the selected statement.

About the Bind Variables view

A bind variable is a placeholder in a SQL statement that must be replaced with a valid value (or address of a value) before the statement can be successfully executed. The execution program passes the value of the bind variable to Oracle when the statement is processed. Understanding which values were used when a statement was run, can be important to the tuning process.

Precise for Oracle can capture statements with bind variables if the following conditions exist:

■   Oracle 10.1.0.4 or later is running

■   The relevant PMDB process (Collect Bind Variables) is up and running.

■   The statement is a heavy statement that is run frequently.

Different values for the same bind variable can dramatically change the performance of a statement. By analyzing a statement's bind values, you can:

■   Run a statement with a real set of bind values, and observe how different binds sets influence the statement's execution).

■   Note the existence of different sets of bind values for a statement.

■   Determine if a correlation can be drawn between a statements performance to the bind values used during the period in question.

About the Bind Variables table

The following information is displayed for each bind set in the Bind Variables table in the left pane:

Table 8-19   Bind set

Bind Set	Description
Plan Match icon	Icon that indicates whether the plan's hash value matches the hash value of the real execution plan.
Estimated Cost	Displays the estimated cost of the plan.
Best Plan Hash Value	Displays the best hash value of the selected plan. This information is displayed after you run Get Best Plan. If different hash values are displayed, this may indicate that there is a matching problem.
Date Last Captured	Displays the date and time the bind set was last captured.
Bind Variables (B1B50)	Displays the values of bind variables B1 through B50. Precise for Oracle can display up to 50 bind variables.
Duplicated Sets	The number of times this specific bind set was collected.

Viewing information displayed for a particular bind set

You can view the text of a particular bind set, and copy an estimation of the text to another tool, in the Details area (right pane) of the Bind Variables view. You can view the following text for a bind set:

Table 8-20   Bind set

Bind Set	Description
Text, with bind variables replaced	Displays source text with bind variables replaced by the actual values run by the statement.
Text for Estimation	Displays source text and hints that instruct Oracle how to sample the table's data and create an optimal plan for the same bind variables.
Bind variables metadata	Displays additional information for the bind variables of the selected bind set (such as, bind name and type, and column name and type). This information can be useful in cases where the same bind name is used for different columns, with different data types.

To view the text of a bind set, with variables replaced

1   In the Bind Variables table, click on the plan whose text you want to view.

2   In the Details area, click on the `Text, with bind variables replaced` tab.

Note: You can select and copy this text to another tool for further analysis

---

To obtain an estimate of the bind set's text

1   In the Bind Variables table, click on the plan whose text you want to view.

2   In the Details area, click on the **Text for Estimation** tab.

3   You can select and copy this text to another tool for further analysis.

About Findings in the Details area

Whenever you select a step on the tree, the children appear as formatted text under Highlights in the Details area. At the top of the Details area, you find the date and time of the sampled execution plan, and its formatted text. Findings appear below the formatted text; it includes a table that lists the severity of each type of sort or other operation, and the name of each object that is referenced by the sampled execution plan.

About the findings table

The Findings table is the result of a proprietary tuning algorithm that recommends normal B*Tree indexes, bitmap indexes, index-only access, or statistics gathering on tables and/or indexes for the selected statement. Based on these recommendations, the Oracle Optimizer can choose a better access plan and improve the performance of the statement.

The following table describes the Findings table.

Table 8-21   Findings table

Column	Description
Severity	Indicates the severity alert that occurred during the sampled execution plan.
	Severity is displayed using the following colors:
	■ Top findings: red
	■ Medium findings: orange
	■ Other findings: yellow
	Note: An orange or yellow finding in an execution plan with many findings, may become a red finding in an execution plan with minimum findings because it becomes one of the top findings.
🔍	Launches to a tab in context with the chosen object.
Type	Indicates the type of operation. Notice that the given type may be underlined. This indicates a live link ToolTip. Select the live link type to view ToolTip recommendations to better access the execution plan and improve the performance of the statement.
Object	Lists the object in the type of operation that is referenced by the execution plan.
Impact (%)	Indicates the maximum theoretical saving, expressed as a percentage of the total In Oracle time consumed. Note that the actual saving that can be made depends on the activity.
In Oracle	Indicates the time used to perform this operation In Oracle for this execution plan.

About the Expanded view

The Expanded view displays recommendations that the Oracle Optimizer can use to create a better access plan and improve the performance of the statement.

The Expanded view is divided into two areas:

■ Highlights

■ What To Do Next

About the Highlights area

The Highlights area displays a brief description of the findings for this type of operation. It also contains a link which provides more help.

About the What To Do Next area

The What To Do Next area displays one or more recommended steps to identify the cause of the problem. Carefully review all data for the finding before continuing.

How to investigate Findings

When you start investigating the findings, it is good practice to start with the finding that has the highest severity rank in the Findings table.

To investigate a finding

1   Identify the finding with the highest severity rank in the Findings table.

2   Select the finding type to view the expanded information for this type of operation.

3   Read the Highlights and What To Do Next areas for the finding and perform the advice that best suits your needs.

4   Follow up on performance to verify that the problem was resolved.

About a statement's different versions

The Compare view lists the different versions of a statement that have been saved together. It lets you compare Oracle Optimizer cost and execution statistics for each alternative so that you can choose the most efficient one.

You can create related statements by doing any of the following:

■   Clicking Edit in the Action menu and saving a new version of the statement.

■   Clicking New Alternatives in the Action menu to generate alternatives.
    Statements that are generated this way are named Alternative-nnn.

■   Running a What-If simulation.
    Statements that are generated this way are named Predicted-nnn.

Note: The alternative versions of a statement only appear in the Activity or What-If tabs or in the Open Statement dialog box after they have been run and captured by the Collector.

The information displayed in the Related SQL view is controlled by the information tabs (Details, Text, Run Statistics) that are located above this area.

About Optimizer cost and more for an alternative

When you click the Run Alternative tab (on the right pane), the cost and more is shown in the Details tab (left pane) for each alternative.

The following table describes the information displayed on the Details tab.

Table 8-22   Details tab information

Column	Description
TUNE	Switches to `Plan` view, in context with the selected related SQL statement.
🔍	Available for related SQL statements but not for the original statement.
Name	Name of the statement.
Statement ID	ID of the statement.
Timestamp	Last run date and time of the statement.
Cost	Bar graph showing the Oracle Optimizer cost for the statement.
Plan Hash Value	Display plan hash value.
Duration (Avg.)	Average time of how long the statement took to run.
Logical I/O (Avg.)	Number of average logical I/Os from the disk.
Physical Reads	Number of physical reads from disk.
Hit Ratio	Number of physical reads or of logical reads, expressed as a percentage.
Table Scans	Number of full table scans performed.
Write Requests	Number of requests to write data to disk, usually of temporary data during joins and sorts.
Sorts	Number of sorts performed.
Sorted Rows	Number of sorted rows.

Table 8-22 Details tab information

Column	Description
Table Scans by Row ID	Number of accesses to a table by Row ID. A Row ID contains the address of a row in a table and is the fastest way to gain access to an individual row, although not necessarily to multiple rows. Access using Row ID usually follows an index scan, because Row IDs are stored in the leaf blocks of an index.
Table Scan Rows Gotten	Number of rows retrieved from tables.
Table Scan Blocks	Number of blocks fetched from tables.
Recursive Calls	Number of recursive SQL calls that Oracle made to the data dictionary while executing the statement.
Processed Rows	Number of rows processed during the execution. This includes rows retrieved from tables, indexes, and temporary segments. It may be many more than are returned.
Host Name	Name of the host.
Description	Description
Attempted Executions	Number of executions started.
Actual Executions	number of actual executions performed.

About the SQL text for each alternative

The Text tab displays the SQL text for each alternative.

About Tuning Actions

The Tuning Actions icon lets you perform additional tuning functions. These action options are accessed by clicking on their related button or by accessing them through the Actions menu.

It is possible to perform the following tuning actions:

Table 8-23 Tuning actions

Action	Description
Open	Open an existing statement.
New	Create a new statement.
Edit Properties	Edit the properties on an existing statement. see "Editing the properties of a statement" on page 31.
Edit Text	Edit an existing statement.
Run	Run a statement.
Re-Explain	Re-explain a statement.
Recommend	Generate new recommendations
Generate Alternatives	Generate new alternatives.
Get Best Plan	Get the best plan according to collected bind sets.

Opening an existing statement

The Open option lets you open an existing statement in the PMDB that was previously created by using the New button, collected by the Precise for Oracle Collector, or imported from the Activity tab. The SQL tab is populated with

the explain plan of your chosen statement. Alternatively, you can open a statement by selecting it in another tab and clicking the SQL tab button or the Tune icon.

You can open existing statements in the following views:

- Plan

- Recommend

- Run Alternatives

- History

- Compare

- All Plans

- Bind Variables

Note: You cannot open statements that were generated as related SQL. You must first open the original statement and then select the related SQL by using the Related SQL tab.

---

see "Editing the properties of a statement" on page 31.

To open an existing statement

1   Click **Actions>Open**.

2   In the Open Statement dialog box, choose the relevant properties for the statement that you want to view from the drop-down lists, as follows:

Instance	Indicates the name of the instance that the statement belongs to.
Cabinet	Indicates the name of the cabinet that the statement is saved in.
Folder	Indicates the name of the folder that the statement is saved in.
Statement	Indicates the ID or user-defined name of the statement.
Comment	Provides an optional comment that was previously associated with the statement.
Text	Provides a preview of the SQL text for the currently selected statement.

The maximum number of items displayed in the Cabinet, Folder, and Statement lists is limited.

3   Click **OK**.

Creating a new statement

The Create option lets you create a new statement and save it in the PMDB in a logical cabinet and folder hierarchy. However, you cannot create a new statement with the same combination of cabinet, folder, and name as an existing statement. You cannot create a statement with exactly the same text as an existing statement.

You can create a new statement in the following views:

- Plan
- Recommend
- Run Alternatives
- History
- Compare
- All Plans
- Bind Variables

Statements collected by the Collector are automatically saved in the default cabinet and folder, so it is recommended to choose another cabinet and folder.

see "Editing the properties of a statement" on page 31.

To create a new statement

1 Click **Actions>New**.

2 In the New Statement dialog box, choose the relevant properties for the new statement from the drop-down lists and fill in the fields, as follows:

Instance	Indicates the name of the instance that the statement belongs to.
User	Indicates the Oracle user name that will parse the statement.
Cabinet	Indicates the name of the cabinet that the statement is saved in.
Folder	Indicates the name of the folder that the statement is saved in.
Statement	Indicates the user-defined name of the statement.
Comment	Provides an optional tuning comment that is to be associated with the statement.
Text	Indicates the SQL text for the statement.

The maximum number of items displayed in the Cabinet, Folder, and Statement lists is limited. You may also type the names.

3 In the Text field, enter the text for the statement.
In the Comment field, type a tuning comment that is associated with the statement, if required.

4 Click **OK**.

5 The new statement is saved in the PMDB in the cabinet and folder hierarchy.

Editing an existing statement

The Edit Text option lets you edit a statement that was saved in the PMDB through the New option. Collected and imported statements cannot be edited. However, you can use the Edit dialog box to save a related SQL statement and then edit the related SQL statement.

Note: You cannot save a statement with the same combination of cabinet, folder, and name as an existing statement.

You can edit existing statements in the following views:

■ Plan

■ Run Alternatives

■ All Plans

see "Editing the properties of a statement" on page 31.

To edit an existing statement

1 Click **Actions>Edit Text.** The Edit Statement dialog box opens.

2 In the Edit Statement dialog box, do one of the following:

■ Enter a new name for the statement in the text field to rename the existing statement.

■ Enter a new name for the statement, check the **Save as alternative** option to save your changes under a different name.

Either choose or save it as a related statement by entering a new name.

If the statement was collected by the Collector, you can only save it under a new name. In this case, the Save as alternative option is pre-selected and cannot be unchecked.

see "About a statement's different versions" on page 151.

3 In the Text box, edit the SQL text for the statement.

4 Click **Save**.

The edited statement is saved in the same cabinet and folder as the original.

Running a statement

The Run option lets you run a statement in the monitored database so that you can gather its execution statistics. Any changes that it makes are rolled back. The only overhead is the execution time. A timeout may be specified in the Run dialog box to prevent the statement execution from running too long.

You can run a statement in the Run Alternatives view.

see "Editing the properties of a statement" on page 31.

To run a statement

1 Click **Actions>Run**.

2 On the General tab in the Run Statement dialog box, do the following:

- Select the database instance that the statement will run on. The maximum number of items displayed in the Cabinet, Folder, and Statement lists is limited. The database instance that the statement will run on

- In the Description box, type a short description of the run.

- Enter the Oracle user name and **password** with which to log in to Oracle.

- Choose one of the following options: Fetch all rows or Fetch first *n* rows

- Specify the number of times that you want to run the statement. You may want to run it more than once to reduce the effect of having to load the buffer cache on the first run. Precise for Oracle calculates and displays the average value for each statistic over the series.

- Check the **Time out after** *n minutes or hours*, if you want to configure a timeout after which the execution of the statement is aborted if the timeout is exceeded. If this is an alternative version of a statement that you have saved, a reasonable timeout is the duration of the original version because you are unlikely to be interested in slower alternatives.

- Check **Run in background**, if you want to run the statement in the background.

3 On the Bind variables tab in the Run Statement dialog box, choose the bind variables with which to run the bind set as follows:

- Click **Choose bind set**.

- In the Bind Sets dialog box, choose the bind set you want to use to run the statement.
  see "About bind sets" on page 156.

- Click **OK**.

4 On the Advanced tab in the Run Statement dialog box, specify the session parameters to be used when running the statement:

- Type in a value for each session variable that you want to run the statement with.

- Click Add New Row if you want to add a new session parameter. Choose a session parameter name from the list, or enter a new one. Type in the value you want to run the statement with.
  see "About session parameters" on page 156.

- If you want to delete a session variable from the statement run, select the session variable you want to delete and click **Delete**.

- Click **OK**.

- Check the **Extended run information (STATIC_LEVEL=ALL)** option if you want additional statistics to be collected. These statistics are displayed in the Extended Statistics tab in the Run Alternatives view.

- Check the **Array fetch** option and specify the number of rows you want to be fetched, if you want to use an array fetch operation.

5 Click **Fill Binds**.

6 Click **OK**.

About bind sets

Bind variables are placeholders in a statement. When you use bind variables, you increase the probability that statements will be stored in memory, making them more immediately available to the next operation that needs them. Bind variables obtain their values from the last statement that was run. You can change the values of bind sets and analyze the values used when a statement is run, to help improve your tuning process.

About session parameters

Session variables obtain their values from the last statement run. You can define a new session variable to run the statement with, in the Add Session Parameters dialog box.

Re-explaining a statement

The Re-Explain option instructs Precise for Oracle to re-explain a statement. Precise for Oracle generates a new execution plan if one or more of the following conditions are met:

■   The execution plan has changed—one or more of its steps or their order has changed.

■   The cost has changed.

■   One of the underlying objects has changed.

If the access plan is unchanged, the last explain time is updated.

You can set the parsing user from **Settings>General Settings>SQL** tab.

You can re-explain a statement in the following views:

■   Plan

■   All Plans

see "About configuring your settings" on page 26.

see "Editing the properties of a statement" on page 31.

To re-explain a statement

1 In the Plan or All Plans view, click **Actions>Re-Explain**.

Generating new alternatives

The Generate Alternatives option instructs Precise for Oracle to run the alternative SQL generator to see if it can find ways of re-writing the SQL so that the Oracle Optimizer comes up with a more efficient execution plan. It performs various mathematical transformations to the SQL, such as replacing EXISTS with IN, OR with UNION, but still so as to give an equivalent result set. The alternatives are saved in the same folder as the original and can be viewed in the Related SQL view.

You can control the settings for generating alternatives from **Settings>General Settings>SQL** tab, such as only generating alternatives that have a different execution plan or indexable columns.

You can generate new statement alternatives in the Run Alternatives view.

see "About configuring your settings" on page 26.

To generate a new alternative

1 In the Run Alternatives view, click **Actions>Generate Alternative**.

How the SQL tab can help you identify performance problems

After identifying a problematic statement that is slowing down the response time of a specific application, the first step in tuning the statement is to understand the access path that Oracle chose for the statement. The explain procedure is designed to clarify the access path chosen for a statement and translate it into a visual medium. Therefore, you can easily see whether the optimizer chose the proper execution plan. For example, you can see whether the optimizer performed an index seek as expected. In addition, you can see schema changes related to the statement's objects and compare these changes with previous access plans and In Oracle over time data for the selected statement to understand how the changes affected a statement's performance. You can also compare previous access plans and locate the steps that have been changed.

You can identify a performance problem by doing one or more of the following:

- Identifying problematic steps in the access plan

- Getting advice on a selected statement

- Creating a related SQL

- Comparing a related SQL

Identifying problematic steps in the access plan

The first step in tuning a statement is to identify problematic steps.

For Oracle 11 and higher, we collect the step activity from Oracle and know the time distributed between the steps. The highest step will be highlighted.

For versions lower than Oracle 11, the problematic steps are defined as steps where the estimated cost is high or where there are a large number of logical I/Os.

To identify problematic steps in the access plan

1   In the Time Frame list, choose the period of time you want to analyze.

2   Do one of the following:

- Launch to the SQL tab with a statement in context, for tuning.

- In the SQL tab, open the statement you want to analyze.

3   The SQL tab shows SQL text on the bottom left, and the execution plan on the top left of the SQL tab. For an example, we see that the SQL is accessing the ORDERS table using a full table scan. The top table on the right will show that the ORDERS table has 30000 rows in it. This is the problematic step.

Getting advice on a selected statement

In our example, we can identify in the statement text three conditions in the WHERE clause of the statement that restrict the amount of data required from the ORDERS table. However, the Oracle Optimizer has determined that every record in the ORDERS table must be read. There are several reasons for this occurrence. We can investigate this problem further by clicking the Recommend button.

Creating a related SQL

The Related SQL view lists the original statement along with any related statements that have been created. Initially there will only be the original statement.

For our example, the SQL tab shows the Related SQL view with just the original statement. We want to tune this statement because no alternate statements have been created yet either manually or automatically by Precise for Oracle.

To create a related SQL

1   In the Time Frame list, choose the period of time you want to analyze.

2 Do one of the following:

■ Launch to the SQL tab with a statement in context, for tuning.

■ In the SQL tab, open the statement you want to analyze.

3 Click **Run Alternatives**. It instructs Precise for Oracle to automatically suggest alternate ways of re-writing the SQL.

For our example, two alternatives have been suggested for this statement. The first alternative has a lower cost than the original.

4 Click the **Text** tab to see a second alternative.

The second alternative has no cost because it is using a `/*+ RULE */` hint to force the use of Oracle's rule-based Optimizer, rather than the cost-based Optimizer.

Comparing a related SQL

You can carry your analysis of the change further by selecting the Compare view.

The Compare view compares two access plans for the same statements.

To compare a related SQL

1 In the Time Frame list, choose the period of time you want to analyze.

2 Do one of the following:

■ Launch to the SQL tab with a statement in context, for tuning.

■ In the SQL tab, open the statement you want to analyze.

3 Click the **Compare** icon. The alternative statement opens both the original and the alternative SQL statement in the Compare view.

For our example, we can see that the alternate SQL is now able to use an index to access the CUSTOMER table, whereas before it was doing a Full Table Scan. The re-written SQL will return exactly the same records as the original, but Oracle can use a different access path to the data.

This is because an SQL function, such as SUBST on the C_LAST column, prevents Oracle from using an index on it (unless you have defined a specific function-based index).

The alternate SQL has no function around the C_LAST column. Therefore, Oracle is able to use the index CUSTOMER_I2 index to more efficiently access the records in the CUSTOMER table.

Simulating the effect of changes

This section includes the following topics:

- About the What-If tab

- How the What-If tab is structured

- How the What-If tab can help you identify performance enhancements

About the What-If tab

The What-If tab lets you simulate the effect of creating, modifying, or dropping a B*Tree or bitmap index on the performance of all statements that Precise for Oracle has collected in the selected context, without physically making the change.

You can either manually enter the list of recommendations or populate it when activating the Oracle advisor/Precise advisor as specified in the Recommend button section.

What-If scans statements, J2EE Caller Services, and Web Transactions in the PMDB. If the cost-based Optimizer is being used, Precise for Oracle can accurately calculate the revised cost of a statements, J2EE Caller Services, and Web Transactions by generating a new explain plan. Statements, J2EE Caller Services, and Web Transactions that have a measurable change in cost are marked as definitely affected. However, even if the revised cost is the same as before, there may still be an impact on response time. For example, Oracle does not change the cost of a DML statements, J2EE Caller Services, and Web Transactions to cater for the overhead of maintaining extra indexes. Such statements, J2EE Caller Services, and Web Transactions are marked as possibly affected because the effect is immeasurable. If the rule-based Optimizer is being used, Precise for Oracle assesses the likelihood of a change in execution plan and marks statements, J2EE Caller Services, and Web Transactions it finds as possibly affected.

As may be expected, adding an index may improve the response times of selected statements, J2EE Caller Services, and Web Transactions that can use the new index but decrease the response time of insert, update, and delete statements, J2EE Caller Services, and Web Transactions that have to maintain the extra index. The opposite effect is seen if an index is dropped. However, the effect will be immeasurable because usually the cost of these affected statements, J2EE Caller Services, and Web Transactions will not be changed. To help you make an informed choice about the desirability of the change, the What-If tab also displays the previous In Oracle resource consumption and execution figures for the statements, J2EE Caller Services, and Web Transactions.

Having selected impacted statements, J2EE Caller Services, and Web Transactions, you can launch the SQL tab in context, to compare the statements, J2EE Caller Services, and Web Transactions' execution plan before and after the simulation. The statements, J2EE Caller Services, and Web Transactions and their simulated indexes will be saved as alternative, in the same folder.

You should bear in mind the following things when using the What-If tab:

- If you are using the cost-based Optimizer, it is important that statistics are up to date so that the simulation is as accurate as possible.

- What-If is only as good as the data it has to work on. It is dependent on statements, J2EE Caller Services, and Web Transactions being loaded into the PMDB and explained by the Explain Statements PMDB process.

■ Until all functions within an application are exercised, Precise for Oracle cannot give a true picture of all affected statements, J2EE Caller Services, and Web Transactions.

■ The index recommendation feature in the SQL tab only examines a single statement, J2EE Caller Service, and Web Transaction. It may be the case that the What-If tab suggests that this recommendation should not be implemented because of its impact on other statements, J2EE Caller Services, and Web Transactions. It is always a good idea to use both tabs together.

How the What-If tab is structured

The What-If tab is slightly different in appearance from other tabs. It is divided into the following areas: Main area, Summary area, and Grid area.

About the Main area

The main area in the What-If tab lists index changes that you would like to make or that were recommended by the Oracle or Precise Advisor.

The following table describes the different index change parameters.

Table 9-1 Index change parameters

Parameter	Description
Evaluate	Check box to select the table row for manipulation.
Remove	Button to remove its table row from the Index list.
Edit	Button to edit the table row.
Action	Indicates the type of the change, which can be one of the following: Index Creation, Index Modifications, Index Deletion, Gather Statistics, Accept SQL Profile, or Rewrite Query.
Owner	Oracle owner.
Table	Name of the table that the index belongs to.
Object	Recommended Object name.
DDL/Comment	DDL command for recommendation implementation or comment regarding the recommendation.
Columns	List of columns in the index.
Type	Normal (B*Tree) or Bitmap.
Unique	Indicates whether the index is unique or not.
Leaf Blocks	Analysis of statistics information. This column is blank until you click Evaluate. If an index is deleted, this column will show N/A.
Distinct Keys	Analysis of statistics information. This column is blank until you click Evaluate. If an index is deleted, this column will show N/A.
Clustering Factor	Analysis of statistics information. This column is blank until you click Evaluate. If an index is deleted, this column will show N/A.
Partitioned	Indicates whether the index is partitioned or not.

About the Summary area

The Summary area in the What-If tab displays whether or not a statement, J2EE Caller Service, or Web Transaction would be effected by the change after clicking **Evaluate**, as follows:

■ Statements, J2EE Caller Services, and Web Transactions whose performance improve.
   The number of statements that are positively impacted.

About launching to SQL tab in context

Depending upon whether or not a statement was evaluated, you can launch to either the Plan tab or the Compare tab in the SQL tab. The following icons indicate to which tab in the SQL tab a statement will be launched, in context:

Table 9-2   Icons to launch in context

Icon	Description
TUNE	If the statement was evaluated, launches the SQL tab on the Plan tab.
🔍	Launches the SQL tab on the Compare tab so that you can compare the explain plan for this version with the original.

Evaluating index changes

The buttons below the Main area let you add and simulate an index change. The icons on the table rove let you edit index changes for evaluation or delete a suggested change from the evaluation table.

Adding or editing an index change for evaluation

If you want to test the consequences of an index change that is not yet listed in the evaluation table in the Main area, you can add a new entry. If you want to modify an index change that has already been suggested, you can edit it.

To add or edit an index change

1   Click **Add** or **Edit,** as required.

2   In the What-If... dialog box, select an owner and a table.

3   Select an operation: Index Creation, Index Modifications, or Index Deletion.

4   If indexes already exist for this table, select the name of the relevant index.

5   If you want the index to be a bitmap index or a unique index, select the relevant check box.

6   To add or remove columns to or from the index, use the left and right arrows.

7   Click **OK** to update the table in the Main area.

Simulating an index change

After you add or edit an index change, you should simulate the change to evaluate its effects.

Note: For a partitioned table, Precise for Oracle can only simulate global B*Tree indexes.

To simulate an index change

1   To simulate an index change, select the Evaluate check box of the relevant table rows and click **Evaluate.**

2   If the number of statements to evaluate exceeds the default setting (100 statements), you will be prompted to do one of the following:

■ Click **OK** to start the analysis.

■ Click **Top Statements** to evaluate the top 100 statements experiencing In Oracle activity only.

■ Click **Cancel** to cancel the operation.

The simulation takes into account all the changes listed in the Main area. When the simulation is complete, the Summary and Grid areas are populated with data.

Deleting a suggested index change

If you decide that a specific index change should not be evaluated, you can remove it from the evaluation table.

Note: Removing an entry from the table only means that the impact of this index change will not be evaluated. It is not the same as dropping an index. To drop an index, use the What if... dialog box.

To delete an index change

1 Click the **Remove** button of the relevant table row.

About the Grid area

After evaluating one or more indices, the Summary and the Grid area will be shown. As mentioned before, the Summary will show the performance improvement of the statements, J2EE Caller Services, and Web Transactions.

Note: The Summary and Grid area will only show statement information if J2EE is not installed.

Note: The number of statements that will be evaluatred is limited to Top *n.* A message with information about this will be displayed.

If J2EE is installed, the Grid area will show two tables:

■ The left table contains a list of J2EE Caller Services and Web Transactions.
   If a J2EE Interpoint is installed, the left table also contains statements.

■ The right table contains a list of statements that are related to the selected J2EE Caller Service or Web Transaction in the left table.

When a statement is chosen in the left table, the right table is removed.

The following tables describe the columns in the tables:

Table 9-3 Columns for J2EE Caller Service and Web Transaction table

Column	Description
J2EE Caller Service or Web Transaction	Names of either the J2EE Caller Services or Web Transactions.
In Oracle	Resource consumption of the impacted J2EE Caller Service or Web Transaction during the selected time frame. The format is: hh:mm:ss. Use this column to assess the heavy resource consuming statements that are either positively or negatively impacted.
Gain (%)	The amount in gain in percentage when performing the suggested index change.
Statement Executions	The amount of statement executions.
Affected Transaction	The number of transactions for the specific J2EE Caller Service that are affected by the index change (for J2EE Caller Services only).

Table 9-4   Columns for the Statement table

Column	Description
🔍	Lets you launch to the SQL tab on the plan tab, if the statement was not evaluated, or the SQL tab on Compare tab, to compare the explain plan for this version of the statement with the original. See "About launching to SQL tab in context" on page 161.
Text	Portion of the SQL text.
In Oracle	Resource consumption of the impacted statement during the selected time frame. The format is: hh:mm:ss. Use this column to assess the heavy resource consuming statements that are either positively or negatively impacted.
Cost	Current estimated cost of the Oracle Optimizer, as taken from the latest explain plan.
Predicted Cost	Predicted cost of the Oracle Optimizer for the new plan.
Gain (%)	The amount in gain in percentage when performing the suggested index change.
Executions	Number of times the statement was executed during the selected time frame.
Statement	Name of the impacted statement.
Statement ID	The ID of the statement.
statement-username	The user name which executed the statement
statement-explain-type	Display the explain type used for the recommendations.
Comment	Precise for Oracle may include its own comment on the simulated change, or why, for example, costs are not available.
Affected Transaction	The number of transactions for the specific statement that are affected by the index change.

How the What-If tab can help you identify performance enhancements

You can identify a performance problem by doing one or more of the following:

■   Adding an index

■   Removing an index

Adding an index

Simulating an index change is a two-step process.

To simulate an index change

1   Obtain a list of proposed index changes.
     A list of proposed index changes can be retrieved by one of the following options:

     ■   The list is produced automatically by following the link on the Advice tab in the SQL tab.

     ■   You can manually propose the addition, modification, or deletion of indexes.

     For our example, the What-If tab shows a proposed index addition to the ORDERS table. The change was recommended in the SQL tab, but it could just as easily have been entered manually by clicking **Add.**

2   Click **Evaluate** to run the impact analysis against all the SQL statements in this instance.

3   Click the **Explore** icon, next to the Statement name. You are taken to the SQL tab in context, and the original and the new execution plans are shown side-by-side on the Compare tab.

For our example, the What-If tab shows in the Summary area that two statements will be improved by the creation of the recommended index and none will have their performance deteriorate.

In the Association area on the right, you can see the two statements that will be affected.

The Oracle Optimizer is used to compare the cost of the original execution plan with the cost of the new execution plan. In our example, the cost will be reduced and the Change Ratio is less than 100 for both cases. The execution time of these statements and the resources they consume should therefore be reduced.

Removing an index

To simulate removing an index, you first need to add a proposed index change to the table in the Main area.

To simulate removing an index

1   Add a proposed index change to the table in the Main area.
A list of proposed index changes can be retrieved by one of the following options:

- ■   The list is produced automatically by following the link on the Advice tab in the SQL tab.
- ■   You can click **Add** to open the What-If dialog box.

For our example, the What-If tab proposes to remove the index IS_PWTR_01 from the table PS_PWTR_TABLE_RANGE. (This suggestion is not a good idea because it belongs to the PMDB.)

Having evaluated the proposal for our example, you can see in the Summary area that 31 statements will definitely be impacted, all having their performance worsened. No statements have their performance improved. You can also see that only two statements could not be evaluated.

Further information can be gathered from the change ratio, which varies from 113%—only slightly worse—to up to 700%. It is therefore unlikely that we implement the proposed change.

10

Examining Oracle instance statistics

This section includes the following topics:

- About the Statistics tab
- How the Statistics tab is structured
- About the entities you can examine
- How the Statistics tab can help you identify performance problems

About the Statistics tab

The Statistics tab displays statistical information on your Oracle instances. All information is derived from Oracle's own statistics and loaded into the PMDB to provide you with a long-term historical view. The Statistics tab provides hundreds of raw performance counters. While examining a single counter is very important, the real value of the Statistics tab lies in examining the ready-to-use sets of graphs that display several related counters.

The Statistics tab is primarily intended for instance tuning. You can also use the tab to periodically monitor the health of your instance. It lets you provide answers to the following types of questions: Is Oracle using resources efficiently? Is the SGA correctly sized? Do our systems have enough memory to enable us to keep adding dedicated server processes? Are we performing too many logical I/Os?.

With each release of the database, Oracle introduces new statistics. For example, operating system-related statistics were introduced in Oracle 10g. Precise for Oracle displays this information when available.

It is not the purpose of this section to define the meaning of each Oracle statistic or how to tune them. For additional information on Oracle statistics and how to tune them, see the relevant Oracle documentation. This section does describe which Oracle statistics are available in Precise for Oracle and how to use the Statistics tab.

How the Statistics tab is structured

The Statistics tab displays information on a selected entity and its associated entities. For example, it is possible to associate to all Wait Events that are related to a specific Instance, by selecting the Wait Events entity from the Association controls.

When you open the Statistics tab from another tab, you will either be focused on an individual instance, or on the entire Tier (if you selected the ALL row in the Dashboard, or All Oracle Instances in the Activity tab). The tab is always launched at the instance level, even if the currently selected entity is not an instance.

The selected entity is always reflected in the Tab heading, which serves as a point of orientation. The highest-level entity you can view information for in the Statistics tab is the Tier. You can view information on an instance by selecting it from the Instance list.

About the Main area in the Statistics tab

The Main area in the Statistics tab shows comprehensive information on the selected entity. You can choose from several views to examine the entity from different angles. You can for example focus on Oracle Wait Events or Oracle Memory Allocation, or I/O Activity.

About the Association area in the Statistics tab

The Association area in the Statistics tab provides corresponding information on the entities associated with the selected entity (displayed in the Main area). You can view information on one type of entity at a time, such as instances only, by selecting an item from the Association controls. The selection you make is reflected in the Association area only; the Main area remains unchanged.

From the Association area, you can also drill down to another entity by clicking on a row in the table. A drill-down affects the whole tab. When you drill down to another entity, the Tab heading displays the new selection; the Main area displays information on the newly selected entity, and the Association area displays the entities associated with the selected entity.

For example, when you want to view information on a specific active session, from the Association controls, click `Wait Events`. The Association area changes to display information on the wait events. Note that the Tab heading and the Main area remain unchanged.

If you want detailed information on a wait event, in the Association area, click the row of the event that you want to view detailed information for. The Tab heading indicates the newly selected entity, and the Main area displays over time graphs for the wait event you drilled down to. There is no Association Area data for a wait event.

See

About the entities you can examine

The Statistics tab displays information on different entities. This section provides an overview of all entities, their meaning, and their views. Entities that share the same set of views are grouped into one topic. They appear separately in the user interface.

The following entities are available:

■   Tier, Instance, and Database entities

■   Wait events entity

■   Block contentions entity

■   Latches entity

■   Statistics entity (Statistics tab)

■   Database time statistics entity (Oracle 10g and later)

■   Operating system statistics entity (Oracle 10g and later)

About the Tier, Instance, and Database entities

The Tier, Instance, and Database entities in the Statistics tab display the same type of information. To display information on the Tier level, click All Oracle Instances from the Instance list. This provides a view of all Oracle instances in the application. This is particularly useful in a Parallel Server or RAC application, where you want to observe combined instance statistics. To display information on the instance level, select a specific instance from the Instance list.

When you view information at the Tier level, some of the views show aggregated information (for example, Physical and Logical I/O) and some show averaged information (for example, Buffer Cache Hit Ratio and PGA Size).

The Database entity lets you compare databases.

The following views are available in the Main area:

■   Overview

- RAC Database Cache

- Instances in Database

- Load

- Logical I/O

- Physical I/O (Datafiles) and physical I/O (Temp files)

- Memory

- Buffer cache

- Parsing

- Sorting

- Redo activity

- Checkpoints

- SQL*Net

- Latching

- Operating system (Oracle 10g and later)

- System Time Model (Oracle 10g and later)

You can associate the Tier, Instance, or database entity with the following entities: Wait Events, Block Contentions, Latches, Statistics, Database Time Statistics (Oracle 10g and Later only), Operating System Statistics (Oracle 10g and later only). In addition, you can associate from the Tier to its instances.

About the overview of a Tier, instance, or database

The Overview, in the Statistics tab, displays the following pie charts:

- Top 10 Wait Events—displays the top 10 Oracle wait events that occurred during the selected time period, ranked by the amount of time spent waiting for these events (derived from `V$SYSTEM_EVENT`). The 10th ranked wait event is the sum of the time spent waiting for all the remaining wait events.

  The pie chart excludes wait events that are classified by Precise for Oracle as idle events, such as `sql*net message from client`, `rdbms ipc message` or `pmon timer`, where Oracle is effectively sleeping and not consuming any resources.

  To see a full list of Oracle wait events and their classifications, click `Wait Events` from the Association controls.

- Physical I/O Operations

  This pie chart illustrates the proportions and volume of read and write requests to datafiles and temp files during the selected time period.

About RAC database cache information

Oracle RAC handles the in-memory data caches of each computer belonging to the same cluster as a single, global cache. The RAC Database Cache view, in the Statistics tab, displays statistics that can help you determine the coherency of the global cluster cache.

The following graphs are displayed:

Table 10-1   Graphs of the RAC Database Cache view

Graph	Description
GC Blocks Transfer	It displays the total aggregated number of global cache current blocks and global cache CR blocks that have been transferred.

Table 10-1   Graphs of the RAC Database Cache view

Graph	Description
Avg. GC Blocks Access Latency (Msec.)	It displays the following average end-to-end elapsed time, or latency for a global cache block request: ■ Average current blocks receive time (in milliseconds) ■ Average CR blocks receive time (in milliseconds)
CR Blocks Request Latency (Avg.)	It displays the following breakdown of the average end-to-end elapsed time, or latency for a CR block request: ■ Average global cache CR block send time ■ Average global cache CR block flush time ■ Average global cache CR block build time
Current Blocks Request Latency (Avg.)	It displays the following breakdown of the average end-to-end elapsed time, or latency for a current block request: ■ Average global cache current block send time ■ Average global cache current block flush time ■ Average global cache current block pin time

About information on instances in the RAC database

The Instance in Database view, in the Statistics tab, displays the following additional RAC database information:

Table 10-2   Additional RAC database information

Information	Description
Database	Displays the logical name of the RAC database.
DB Name	Displays the Oracle database ID for the instance.
DB ID	Displays the Oracle database ID.
Instance Number	Displays the number of the instance in the RAC database.
Instances	Displays the number of instances that share the same RAC database with the selected instance.
Instance Sessions	Number of current instance sessions in the database.
Database Sessions	Number of total database sessions.
Time Non-idle Wait Event	Pie chart comparing the amount of non-idle wait time experienced by the selected instance vs. the entire database.
Latches	Pie chart comparing the amount of latch wait time experienced by the selected instance vs. the entire database.
Logical I/O	Pie chart comparing the amount of logical I/O experienced by the selected instance vs. the entire database.
Statement Executions	Pie chart comparing the number of statements executed by the selected instance vs. the entire database.

About the load of a Tier, instance, or database

The Load view, in the Statistics tab, displays the bar graphs for the selected time period.

The Load view displays the following bar graphs:

■   Logons

    It displays the number of user sessions.

■   Executions per Logon

    It displays the average number of statements executed by a session.

- Commits and Rollbacks
  It displays the number of transactions either committed or rolled back.

- Executions per Transaction
  It displays the average number of statements executed in a transaction.

About the logical I/O of a Tier, instance, or database

The Logical I/O view, in the Statistics tab, displays bar graphs for the selected time period.

The Logical I/O view displays the following bar graphs:

- Logical I/O Operations
  It displays the total number of logical I/O operations broken down into the number of DB Block Gets, Consistent Gets, DB Block Changes and Consistent Changes. The first two represent logical read operations, while the latter two represent logical write operations.

- Logical I/O Operations per Execution
  It displays the average number of logical I/O operations per executed statement.

About the physical I/O of a Tier, instance, or database

The Physical I/O (Datafiles) and the Physical I/O (Temp Files) views, in the Statistics tab, display bar graphs for the selected time period.

The Physical I/O (Datafiles and Temp Files) display the following bar graphs:

- Physical I/O Operations
  It displays the total number of physical read and write requests to all datafiles (or temp files).

- Avg. Physical Read Time (Msec.)
  It displays the average duration of a physical read.

- Avg. Physical Write Time (Msec.)
  It displays the average duration of a physical write.

About the memory of a Tier, instance, or database

The Memory view, in the Statistics tab, displays bar graphs for the selected time period.

The Memory view displays the following bar graphs:

- SGA Size
  It displays the size of the fixed and variable parts of the SGA together with the amount used and available. This information is derived from V$SGASTAT. From Oracle 9i onwards, the size of the buffers in the SGA may vary dynamically over time as Oracle tries to adjust to the workload. Using this graph, it may be possible to return some of the SGA memory to the operating system. Alternatively, it is possible to offset an increase in one area with a decrease another area, without increasing the total SGA size. For example, you may be able to increase the size of the Buffer Cache at the expense of the Java Pool.

- PGA Size
  It displays the total amount of PGA memory allocated to all dedicated server processes (derived from V$PGASTAT from Oracle 9i onwards), plus the Aggregated PGA Target (defined by the pga_aggregate_target Oracle initialization parameter). If you observe that the total PGA allocated has exceeded the PGA target many times, consider increasing the PGA target.

Note: PGA size data is only displayed for Oracle 9.2 and later.

About the buffer cache of a Tier, instance, or database

The Buffer Cache view, in the Statistics tab, displays bar graphs for the selected time period.

The Buffer Cache view displays the following bar graphs:

■ Buffer Cache Hit Ratios

It displays the buffer cache hit ratio, expressed as a percentage, for each of the buffer pools. From Oracle 9i onwards, different tablespaces may be defined with different block sizes. Each block size has its own buffer pool.

■ Waits on Buffer Cache

It displays the number of waits on all pools, divided into the number of buffer busy waits and free buffer waits.

About parsing information for a Tier, instance, or database

The Parsing view, in the Statistics tab, displays bar graphs for the selected time period.

The Parsing view displays the following bar graphs:

■ Hard Parse Ratio

It displays the percentage of statements that required a hard parse as a proportion of total parses. A high figure may indicate that Oracle is unable to share statements in the shared pool, for example, if the application uses literals instead of bind variables.

■ Soft Parses

It displays the number of statements requiring a soft parse.

■ Library Cache Hit Ratio

It displays the percentage of statements that were already in the shared pool when Oracle began parsing them. A low figure may indicate that the shared pool is too small, or that the applications are not using bind variables or cursor sharing.

■ Parse Times (Sum.) - Oracle 10g

It displays the total duration of all parse times for Oracle 10g. This information is taken from V$SYS_TIME_MODEL.

About sorting information for a Tier, instance, or database

The Sorting view, in the Statistics tab, displays bar graphs for the selected time period.

The Sorting view displays the following bar graphs:

■ Disk Sorts

It displays the number of sorts that required a write to disk and could not be done entirely in memory.

■ Disk Sorts Ratio

It displays the percentage of sorts requiring a write to disk compared with the total number of sort operations (in-memory and to disk). It is much faster to perform sorts entirely in memory if possible and so a high ratio may indicate a performance problem.

■ Rows Sorted

It displays the total number of rows sorted.

About the redo activity of a Tier, instance, or database

The Redo Activity view, in the Statistics tab, displays bar graphs for the selected time period.

The Redo Activity view displays the following bar graphs:

■ Generated Redo Log

It displays total amount of redo log data generated.

■ Redo Wait Time

It displays the amount of time that applications waited on Redo Log Buffer wait, Log Switch, and Clear wait. Note that these are Precise for Oracle wait categories—the underlying Oracle waits are log file sync, log file switch completion, and so on.

■ Redo Writes

It displays the total number of writes to the redo log buffer.

- Commits and Rollbacks
  It displays the total number of transactions committed or rolled back.

About the checkpoints of a Tier, instance, or database

The Checkpoints view, in the Statistics tab, displays bar graphs for the selected time period.

The Checkpoints view displays the following bar graphs:

- Background Checkpoints
  It displays the total number of checkpoints completed or interrupted.

- Physical Writes
  It displays the total number of checkpoint physical writes and other physical writes to disk. This parameter provides an understanding of the checkpoint contribution to total write activity.

About SQL*Net information for a Tier, instance, or database

The SQL*Net view, in the Statistics tab, displays bar graphs for the selected time period.

The SQL*Net view displays the following bar graphs:

- SQL*Net Client Throughput
  It displays the amount of data sent from the Oracle server process to the client process.

- SQL*Net Client Roundtrips
  It displays the total number of client requests. This is mainly the number of times a cursor is opened and executed, and the number of fetches required to retrieve all the data.

- SQL*Net DB Link Throughput
  It displays the amount of data sent from the Oracle server process to a remote Oracle server process using a database link.

- SQL*Net DB Link Roundtrips
  It displays the total number of messages sent/received from the Oracle server process to a remote Oracle server process using a database link.

If these figures are unusually high, it may be an indication that queries are returning more data than necessary, rather than the minimum set of rows and columns. Another possibility is that the application may be inefficient in terms of networking; you may want to examine the SQL*NET parameters, such as the Session Data Unit (SDU) size.

About latching information for a Tier, instance, or database

The Latching view, in the Statistics tab, displays bar graphs for the selected time period.

The Latching view displays the following bar graphs:

- Wait Times (Sum.)
  It displays the total amount of time that Oracle has spent waiting for a latch to become available. The time is broken down into various categories according to latch type. To see the full list of latches and their categorization, click `Latches` from the Association controls in the Association Area. Note that this categorization is defined within Precise for Oracle.

About operating system information for a Tier, instance, or database

The Operating System (Oracle 10g and Later) view, in the Statistics tab, displays bar graphs for the selected time period.

The Operating System view displays the following bar graphs:

- CPU Utilization
  It displays the percentage of all the available CPUs on the server used by Oracle, broken down into system time and user time.

- Paging

It displays the average number of bytes paged-in and paged-out per second by Oracle.

About system time model information for a Tier, instance, or database

The System Time Model (Oracle 10g and Later) view, in the Statistics tab, displays graphs for the selected time period.

The System Time Model view displays the following graphs:

- Oracle (DB and Background) Time
  It displays Oracle time broken down into DB CPU time, elapsed CPU background time, and elapsed non-CPU background time

- DB Elapsed Time
  It displays database time only, broken down into sequence load elapsed time, parse time, SQL execute time, connection management time, PL/SQL execution time, PL/SQL compilation time, inbound PL/SQL RPC time and Java execution time.

About the Wait Event entity

The Wait Event entity, in the Statistics tab, displays Oracle wait event information that is derived from the V$SYSTEM_EVENT table.

Examining a wait event over time

The Over Time view, in the Statistics tab, displays graphs for the selected time period.

The Over Time view displays the following graphs:

- Event Waits vs. Timeouts
  It displays the total number of occurrences of the wait event vs. the number that exceeded the timeout.

- Times Waited (Sum.)
  It displays the total time waited.

You cannot associate a wait event with any other entity.

About information on wait events

When you view information on wait events in the Association area of the Statistics tab, the information is displayed in a table.

The following table shows Wait event information in the Association area.

Table 10-3   Wait event information in the Association area

Column	Description
Wait Event	Name of the Oracle wait event.
In Oracle Sub-State	Equivalent Precise for Oracle wait sub-state. See "Wait States" on page 35.*
Idle	Indicates whether Precise for Oracle categorizes the wait event as an idle event. Precise for Oracle considers certain events, such as sql*net message from client, rdbms ipc message or pmon timer, as idle because Oracle is effectively sleeping and not consuming any resources. The categorization affects the top 10 wait events displayed in the Wait Event view of the Tier or an instance.
Waits	Total number of occurrences of the wait event.
Timeouts	Total number of occurrences of the wait event that exceeded the timeout. Some events have a timeout associated with them.
Times Waited (Sum.)	Total time waited, displayed as a duration and bar graph.

About the Block Contentions entity

The Block Contention entity, in the Statistics tab, displays information that is derived from the V$WAITSTAT table.

About a block contention over time

The Over Time view, in the Statistics tab, displays a graph for the selected time period.

The Over Time view displays the following graph:

■ Block Contention Waits vs. Time Waited

It displays the total number of block contentions vs. the total time waited.

You cannot associate a block contention with any other entity.

About information on block contentions

When you view information on block contentions in the Association area, the following information is displayed in the table:

Table 10-4 Block contention information in the Association area

Column	Description
Block Contention	Name of the block contention.
Waits	Total number of block contention waits.
Times Waited (Sum.)	Total time waited, displayed as a duration and bar graph.

About the Latch entity

The Latch entity, in the Statistics tab, displays information that is derived from the V$LATCH table.

About a latch over time

The Over Time view, in the Statistics tab, displays graphs for the selected time period.

The Over Time view displays the following graphs:

■ Wait Times (Sum.)

It displays the total time Oracle spent waiting for the latch to become available.

■ Latch Miss Ratio

It displays the number of requests to get a latch which were unsuccessful. It includes Willing to Wait Miss Ratio and No Wait Miss Ratio.

■ Latch Misses

It displays the total number of latch misses. It includes the number of Willing to Wait Misses and No Wait Misses.

You cannot associate a latch with any other entity.

About information on latches

When you view information on latches in the Association area, the following information is displayed in the table:

Table 10-5 Latch information in the Association area

Column	Description
Latch	Name of the latch.
Class	Precise for Oracle categorization of the latch. Precise for Oracle uses these categories to display a breakdown of latch types in the Latching view of a Tier or instance.

Table 10-5 Latch information in the Association area

Column	Description
Misses	Bar graph showing the number of Willing to Wait misses and No Wait misses. Willing to Wait misses is equivalent to MISSES in V$LATCH; No Wait misses is equivalent to IMMEDIATE_MISSES in V$LATCH.
Willing to Wait Miss Ratio	Percentage of requests to get a latch that were unsuccessful but the caller was willing to wait. Equivalent to the ratio of MISSES to GETS in V$LATCH.
No Wait Miss Ratio	Percentage of requests to get a latch that were unsuccessful but the caller was not willing to wait. Equivalent to the ratio of IMMEDIATE_MISSES to IMMEDIATE_GETS in V$LATCH.
Wait Times (Sum.)	Total time Oracle spent sleeping, waiting on the latch to become available, displayed as a duration and bar graph.

About the Statistics entity

The Statistics entity, in the Statistics tab, displays information that is derived from the V$SYSSTAT table.

About statistics over time

The Over Time view, in the Statistics tab, displays a bar graph for the selected time period.

The Over Time view displays the following bar graph:

■ Statistics

Displays the value of the statistic.

You cannot associate statistics with any other entity.

About statistics information

When you view statistics information in the Association area, the following information is displayed in the table:

Table 10-6 Statistics information in the Association area

Column	Description
Statistic	Name of the Oracle statistic.
Class	Oracle class of statistic.
Value	Statistic value.

You can use the More... option in the Association controls to populate the association area with a subset of statistics based on class. For example, you can click **Statistics>Cache** or **Statistics>Debug**. Some statistics are classified more than once. For example, Buffer is Pinned Count is classified as both Cache and SQL.

About the Database Time Statistics entity

The Database Time Statistics (Oracle 10g and later) entity, in the Statistics tab, displays information that is derived from the V$SYS_TIME_MODEL table.

About database time statistics over time

The Over Time view, in the Statistics tab, displays a graph for the selected time period.

The Over Time view displays the following graph:

■ Times Waited (Sum.)

Displays the total time waited.

You cannot associate database time statistics with any other entity.

About database time statistics information

When you view database time statistics information in the Association area, the following information is displayed in the table:

Table 10-7   Database time statistics information in the Association area

Column	Description
Statistic	Name of the statistic.
Times Waited (Sum.)	Total time waited, displayed as a duration and bar graph.

About the Operating System Statistics entity

The Operating System Statistics (Oracle 10g and later) entity, in the Statistics tab, displays information that is derived from the V$OSSTAT table.

About operating system statistics

The Overview, in the Statistics tab, displays processor-related and paging-related statistics for the selected time period.

Note: Precise for Oracle only provides operating system statistics as reported by Oracle; some operating systems may not necessarily report accurate statistics, and Oracle may provide different statistics depending on the specific operating system.

You cannot associate operating system statistics with any other entity.

When you view database time statistics information in the Association area, the following information is displayed in the table:

Table 10-8   Operating system statistics information in the Association area

Column	Description
Statistic	Name of the statistic.
Type	Type of statistic. Values are Processor or Paging
Processor Time	Amount of CPU time consumed. This and the Utilization column are only available for Processor-related statistics.
Utilization	CPU utilization.
Paged Bytes	Total number of bytes paged in or out. This and the Paged Bytes/sec column are only available for Paging-related statistics
Paged Bytes/Sec.	Average number of bytes paged in or out per second.
Description	Provides a description of the statistic.

How the Statistics tab can help you identify performance problems

To determine whether Oracle is performing optimally, it is necessary to monitor the health of your instances, and at times to tune your instances. The Statistics tab provides many performance counters grouped into several predefined graphs that display related counters that enable you to locate performance problems in your system.

You can identify a performance problem by doing one or more of the following

■   Examining top 10 wait events

- Examining latches

- About examining logical I/Os

- Examining physical I/Os

- Examining memory

- Examining parse times

Examining top 10 wait events

We should begin our analysis with the Instance overview. This view displays the Top 10 Wait Events in a pie chart.

Note: The pie chart will not display what are considered to be idle events.

See

To examine top 10 wait events

1 In the Time Frame list, choose the period of time you want to analyze.

2 In the Instance list, choose the item you want to analyze.

3 In the Association area, click **Wait Events** to observe the wait events that are associated with the selected instance and their wait duration.

Examining latches

After we have examined the top 10 wait events, we now need to determine which latch or latches are contributing to the latch free event.

To examine latches

1 In the Association area, click **Latches** from the Association controls. The Statistics tab displays the latches that are associated with this instance.

About examining logical I/Os

One of the recommended ways to tune statements is to reduce the number of logical I/Os, because logical I/Os tend to dominate response time. It is important to understand how the number of logical I/Os varies over time. The instance statistics in the Statistics tab re-enforce the resource consumption figures observed in the Activity tab.

Examining consistent gets

We can tune statements by examining the current number of consistent gets for a period of time.

To examine consistent gets

1 In the Time Frame list, choose the period of time you want to analyze.

2 Open the statement you want to analyze in the Statistics tab. For our example, the Statistics tab shows the logical I/O for All Oracle Instances. We can see that the number of logical I/Os remained fairly constant on the evening of September 22nd. The logical I/Os are nearly all made up of Consistent Gets (that is, fetching read-consistent copies of data).
In the Association area, you see that the logical I/O count is dominated by ora920 on linuxi4o2. This is the instance we should look at when considering tuning of logical I/Os. Observe the number of logical/Os per execution. Notice the upward trend. This, together with the total number of logical I/Os, indicates that fewer statements are being run, but each statement is processing more data. This may indicate that the application is I/O bound because it cannot handle more than a certain number of I/Os.

Examining DB block gets

We can also tune statements by examining the number of DB block gets.

To examine DB block gets

1   In the Time Frame list, choose the period of time you want to analyze.

2   Open the statement you want to analyze in the Statistics tab. In our example, we observe two major peaks in I/O activity, one on the 13th of September and one on the 17th. For our example, the Logical I/O Operations graph shows an increase on the 17th and that the logical I/O was split fairly evenly. The logical I/O was split between DB Block Gets, which are fetching the current version of the data, and DB Block Changes, which are updating the current version of the data. This indicates that there was heavy DML activity on these two days.
The Logical I/O Operations per Execution graph shows that the number of logical I/O operations per execution remained roughly constant on these two days. Together, both graphs will imply that approximately twice as many statements were executed on the 17th.

Examining physical I/Os

It is usually good practice to examine logical I/Os first because it is a more reliable measure than physical I/Os, due to caching effects that can complicate an analysis of physical I/Os.

To examine physical I/Os

1   First examine logical I/Os.
See "Examining consistent gets" on page 176.

2   Now examine physical I/Os. For our example, the Statistics tab shows sampled data from the same time period as that shown for consistent gets over time. We can see that the average read and write time follows the number of physical I/O operations reasonably close, particularly the first peak on September 22nd. This indicates that there is some contention between I/O operations, perhaps because they are all reading from different points on the same disk, causing a rise in seek time.

Examining memory

If you are looking to reduce the amount of memory used by Oracle, or looking to increase one of the memory buffers without increasing the overall memory footprint, you can use the memory view.

To examine memory

1   In the Time Frame list, choose the period of time you want to analyze.

2   Open the statement you want to analyze in the Statistics tab. For our example, the Statistics tab shows data that was gathered over a 16-day period. All 32MB of the Java pool is free; all 8 MB of the large pool is free, and at least 15 MB of the shared pool is free.
The Association area lists all Oracle events that are in the Shared Pool Wait category. As would be expected (for a shared pool that is not full), there has been very little wait time.
For this example, the Aggregated PGA Target has been set to 24 MB and a maximum of 19 MB has been used. This leaves us with very little scope by which to reduce total PGA memory.

Examining parse times

Oracle 10g provides even more statistics than were provided by previous versions. Precise for Oracle displays graphs of some of the more important statistics, such as CPU utilization and parse times.

To examine parse times

1   In the Time Frame list, choose the period of time you want to analyze.

2   Open the statement you want to analyze in the Statistics tab.
For our example, we will examine how a number of hard parses affects the parse time. If we look at the Hard Parse Ratio graph in the Main area, we can observe that the hard parse ratio increased to around 50% at 6 PM, meaning that some new statements were executed or required reloading. Consequently, the total parse time increased to around 15 minutes. Then the number of statements that required parsing remained fairly constant (shown on the Soft Parse graph), but the hard parses fell away and so did the parse times. This example explains why it is a good idea to use bind variables or cursor sharing.

Examining concurrent managers and requests

This section includes the following topics:

- About the Oracle Applications tab

- How the Oracle Applications tab is structured

- About the information you can examine in the Oracle Applications tab

- Defining the settings of overtime graphs

- How the Oracle Applications tab can help you identify performance problems

About the Oracle Applications tab

The Oracle Applications tab displays details about the current status of the Concurrent Managers and Requests (batch jobs and reports) that have been submitted. It also displays historical information on the availability of the managers and completion of requests. It has a structure similar to the Dashboard tab, allowing you to quickly monitor the health of your Oracle Applications instances.

This tab lets you answer the following types of questions: Are all the Concurrent Managers up?; Is there a backlog of pending requests?; Which are the slow-running requests?; How many requests fail?

Information on the Concurrent Managers includes their status, their availability, and the number of requests that are pending, running, or completed. Information on the Requests includes their status, priority, session details, requesting user and historical data on the program executed, and the programs arguments. All the information on the Concurrent Managers and Requests is based on data from the Oracle Applications tables in the Oracle database. To see information on interactive sessions, use the Current or Activity tabs. You can also see the database component of a request's run time performance in the Current or Activity tabs.

This section describes the available Oracle Applications parameters and how to use the tab. It does not, however, define the meaning of each Oracle Applications parameter or how to tune them; for this kind of information, see the appropriate Oracle documentation.

Note: The Oracle Applications tab is only available if Interpoint for Oracle Applications has been installed. This is an optional extension to Precise for Oracle. For more information, see the *Precise Installation Guide.*

See "Components of Precise for Oracle" on page 17.

See the *Precise Installation Guide.*

See "About the Dashboard tab" on page 43.

See "About the Current tab" on page 51.

See "About the Activity tab" on page 61.

How the Oracle Applications tab is structured

The Oracle Applications tab is divided into two views: Managers and Requests. Each view is also divided into two sections: the Main area on the left and the Details area on the right. Both areas can have additional tabs, depending on which view is selected.

The Time Frame selector that is available in other tabs is not present here because most of the information is current and other historical data is kept for a maximum of 14 days. In addition, the Instance selector lists only Oracle Applications instances in this application.

When the Requests tab is selected, additional Action controls are displayed: a Concurrent Manager selector and a Request type selector. The Concurrent Manager selector lets you choose a specific Concurrent Manager or All Managers. The currently selected Concurrent Manager is retained when you move between the Manager view and the Requests view. If, for example, you select the Standard Manager in the Manager view and then click `Requests`, it will only display Requests submitted to the Standard Manager.

The Request Type selector lets you choose All Requests, Current Requests, Past Requests and so on. By default, all Requests are shown.

About the information you can examine in the Oracle Applications tab

The Oracle Applications tab displays information on Concurrent Managers and Requests.

Viewing information about Concurrent Managers

On the Mangers tab, the Main area displays a table that provides information on all Concurrent Managers.

The following table describes the information displayed.

Table 11-1   Concurrent Manager information in the Main area

Column	Description
🏃	Indicates when the state of the Concurrent Manager is `Stale`. This column is empty for all other states.
Status	State of the Concurrent Manager, which can be `Active`, `Running`, `Terminated`, `Inactive`, or `Stale`.
Concurrent Manager	Name of the Concurrent Manager.
Pending Requests	Number of pending requests that are queued.
Running Requests	Number of requests that are currently running.
Running Processes	Number of Concurrent Manager processes that have been started.
Available Processes	Number of Concurrent Manager processes that have not been started.

The Details area shows specifics on the Concurrent Manager that is selected in the Main area. The information in this area is displayed on the Overview and the Processes tabs.

About the overview of Concurrent Managers

The Overview tab identifies the Concurrent Manager and gives general information about it. It also shows a graph of the number of pending and running requests over time. A request normally starts as Pending and changes later to Running, so it should contribute to both totals. If there is a backlog, the pending total will be greater than the running total for each time interval.

About Concurrent Manager processes

The Processes tab displays a table that provides details on the different processes.

The following table describes the information displayed.

Table 11-2 Concurrent Manager table on the Processes tab

Column	Description
🏃	Indicates when the state of the process is `Fault`. This column is empty for all other states.
Process ID	Process ID of the Concurrent Manager process.
Start At	Date the process was started.
Oracle Process ID	Process ID of the Oracle server process. If the Concurrent Manager process is not connected and there is a fault, this column may provide extra information.

In addition, this tab displays a History graph that illustrates the availability of the Concurrent Manager processes over time, as follows:

■ Processes Target—the total number of Concurrent Manager processes on a particular day.

■ Processes Target OK and Processes Target Missed—represent the average number of running processes on that day and whether the target was hit or not.

Viewing information about Requests

On the Requests tab, the information in the Main area is displayed on the Requests and the Arguments tabs. Both show a list of Requests filtered by the currently selected Oracle Applications instance, Concurrent Manager, and Request type. The difference lies in the columns that are displayed on each tab. The Requests tab provides the execution context; the Arguments tab shows the program that is executed, and its related arguments.

The following table describes the information displayed.

Table 11-3 Request information in the Main area

Column	Description
ID	Request ID
Program	Oracle Applications program that this Request executes.
Manager	Requests tab only. Concurrent Manager handling this Request.
User	Requests tab only. Oracle Applications user name.
Phase	Requests tab only. Execution phase of the Request. The Request can be `Inactive`, `Scheduled`, `Pending`, `Running`, or `Completed`.
Status	Requests tab only. Completion or success status of the Request. The Request can be `Normal`, `Warning`, `Error`, or `Cancelled`.
Elapsed Time	Requests tab only. Time the request has spent in the Pending and Running states.
Argument 1 to 5	Argument to the Oracle Applications program.

The Details area shows specifics on the Request that is selected in the Main area. The information is this area is displayed on the following tabs:

■ Overview—shows the status of the Request, its identification details, and its executed program details.

■ Last 14 Day History—displays the following pie charts:

Total Time (Same program, various arguments)	Proportion of time that all Requests with the same program name as this Request have spent either pending or running during the last 14 days.

Total Time (Same program, same arguments)	Proportion of time that all Requests with the same program name and arguments as this Request have spent either pending or running during the last 14 days.
Completions (Same program, various arguments)	Proportion of successful and unsuccessful executions of all programs with the same name as this Request during the last 14 days. It also displays general information about the executed program.
Completions (Same program, same arguments)	Proportion of successful and unsuccessful executions of all programs with the same name and arguments as this Request during the last 14 days. It also displays general information about the executed program.

■   Details—more specifics on the request, such as whether it has a sub-request, its completion code, its schedule, and its output destination.

Defining the settings of overtime graphs

You can customize the time frame that is covered by the overtime graphs on the Managers and Requests tabs.

See "About configuring your settings" on page 26.

To define the graph settings

1   On the Precise toolbar, click **Settings>General Settings.**

2   Specify for how many days you want to view the Concurrent Manager history.

3   Specify for how many hours you want to view the Concurrent Manager request history.

4   Click **OK**.

How the Oracle Applications tab can help you identify performance problems

You can identify a performance problem by doing one or more of the following:

■   Examining Concurrent Manager backlog

■   Determining Concurrent Manager availability

■   Examining requests

Examining Concurrent Manager backlog

The Concurrent Manager backlog shows us the number of requests (submitted to the Standard Manager) during a period of time. For our example, the Oracle Applications tab shows us that on Sept. 15, 33 requests were submitted to the Standard Manager in the first hour of September 15 and were therefore pending. Also 33 requests were run. In this case, even at the peak workload around midnight each day, it would appear that the Standard Manager processes were able to keep up with the workload. They at least ran the requests within an hour of being submitted.

To examine the Concurrent Manager backlog

1   In the Time Frame list, choose the period of time you want to analyze.

2   Open the Oracle Applications tab to view the number of submitted requests.

Determining Concurrent Manager availability

Ideally, all Concurrent Manager processes should be available to do work.

To determine Concurrent Manager availability

1   In the Time Frame list, choose the period of time you want to analyze.

2   Open the Oracle Applications tab to view the number of available processes.

3   Click the **Processes** tab. Examining the Processes tab, for the Standard Manager, shows that the number of available processes has not always hit the target figure. On September 15, the average number of processes available was 7.21 out of 16. Looking at the current state of the processes identifies that they cannot connect to Oracle, perhaps because it is unavailable.

Examining requests

The Oracle Applications tab shows the performance of related requests.

To examine requests

1   In the Time Frame list, choose the period of time you want to analyze.

2   Open the Oracle Applications tab to view the number of submitted requests.

In the Main area, for our example, we see that Request 2160224 completed normally. The Details area displays the past performance of related requests.

The top two pie charts reveal that related requests spent approximately twice as long executing as they did waiting to be executed, regardless of which arguments were passed. The bottom two pie charts show the proportion of successful to unsuccessful executions.

The program has exited with an error code on 3 occasions, but has always been successful when executed with exactly the same arguments as the selected request. In other words, it has only failed when passing different arguments to the current request.

12

Oracle Findings

This section includes the following topics:

- About the Oracle Findings view
- About Instance findings
- About Activity tab findings
- About SQL tab findings
- About Object Findings

About the Oracle Findings view

The Findings view displays recommendations that can be used to create a better execution plan and improve the performance of a statement. The Findings vary according to the type of operation in a statement, the Precise product and the technology.

Precise for Oracle is capable of providing additional insight into a problem by helping you focus on the item or object that is causing the problem. When you click on certain findings in the SQL tab or the Object tab, and you launch to the tab that can provide additional information on the problem, the row representing the item or object that is most likely causing the problem is highlighted. This provides a further indication as to where you should focus your analysis.

About the Highlights area

The Highlights area displays a brief description of the findings for the selected type of operation.

The What to do next area displays one or more recommended steps to identify the cause of the problem. Carefully review all data for the finding before continuing.

About the Advice area

The Advice area displays one or more recommended options to resolve or reduce the problem for the selected finding. Carefully review all data for the finding and then perform the advice that best suits your needs.

How to investigate findings

When you start investigating the findings, it is good practice to start with the finding that has the highest severity ranking in the Findings table.

To investigate a finding

1   Identify the finding with the highest severity ranking in the Findings table.

2   Select the finding type to view additional information on the selected type of operation.

3   Read the Highlights and What to do next areas for the finding.

4   After you have studied all of the information provided, read the Advice area and perform the recommendation that best suits your needs.

5   Follow up on performance to verify that the problem was resolved.

About Instance findings

Several Dashboard tab findings exist in the table to help the user.

The Dashboard tab for an instance has the following findings:

- High CPU Wait
- High Other Host Wait
- High Memory Wait
- High Shared Pool Wait
- High Rollback Segment Wait
- High Redo Log Buffer Wait
- High Log Switch and Clear Wait
- High RAC/OPS Wait
- High Other Lock Wait
- High Background Process Wait
- High Parallel Query Server Wait
- High Buffer Wait
- High Other Wait
- High Remote Query Wait
- High Client Communication Wait
- High Resource Manager Wait
- High MTS Wait
- Heavy Statement
- Frequently Executed Statement
- Heavily Accessed Object
- Locked Object
- High Sorts on Disk
- High Undo Activity
- Heavily Accessed Cluster
- Locked Cluster
- Storage Contention on Device (Clariion)
- Storage Contention on Device (Symmetrix Thick)
- Storage Contention on Device (Symmetrix Thin)
- Storage Contention on Device (Symmetrix F.A.S.T. VP)
- Storage Contention on Redo Logs and DB Files
- Storage Contention on Temporary Objects
- Heavy Storage Device Holding Undo Objects
- Unbalanced Storage Devices Activity
- Heavy J2EE Caller Service
- High SQL Executions for J2EE Caller Service

High CPU Wait

Your instance has spent much of its In Oracle time waiting for CPU.

Table 12-1 High CPU Wait

	Description
What to do next	Perform one of the following options: ■ Examine high wait for CPU statements in the Activity tab. ■ Examine high CPU statements usage in the Activity tab. ■ Examine CPU utilization in the Statistics tab. ■ Try to identify the system processes consuming CPU resources using the Insight Savvy for OS tool.
Advice	Perform one of the following options: ■ Identify other processes in the system. ■ The instance has spent much of its In Oracle time waiting for CPU; every process running in your system affects the available CPU resources. Effort spent tuning non-Oracle factors can improve Oracle performance. ■ Identify heavy statements using CPU or Waiting for CPU and try to tune them.

High Other Host Wait

Your instance has spent much of its In Oracle time in Other Host Wait.

Table 12-2 High Other Host Wait

	Description
What to do next	Perform one of the following options: ■ Examine heavy wait for Other Host Wait statements in the Activity tab. ■ Try to identify the system processes consuming OS resources using the Insight Savvy for OS tool.
Advice	Other Host Wait can result from any of the following causes: asynchronous I/O, gateways, or the use of NFS and TP monitors. Check the statements and programs suffering from this state and check whether the above resources are being utilized efficiently.

High Memory Wait

Your instance has spent much of its In Oracle time waiting for memory.

Table 12-3 High Memory Wait

	Description
What to do next	Perform one of the following options: ■ Examine high Memory Wait statements in the Activity tab. ■ Try to identify the system processes consuming memory using the Insight Savvy for OS tool.

Advice	Perform one of the following options:
	■ Identify other processes in the system.
	  The instance has spent much of its In Oracle time waiting for memory; every process running in your system affects the available memory.
	  Effort spent tuning non-Oracle factors can improve Oracle performance.
	  For example: the result of setting a high number of MAX_PARALLEL_SERVERS when using a parallel query option.
	■ Identify heavy statements using High Memory Wait and try to tune them.

High Shared Pool Wait

Your instance has spent much of its In Oracle time waiting for the group event Shared Pool Wait.

Table 12-4   High Shared Pool Wait

	Description
What to do next	Perform one of the following options: ■ Examine the Oracle events that are grouped into the Wait for Shared Pool in the Statistics tab. Determine the dominant Oracle event and follow the tuning scenario set by this event. ■ Examine high Shared Pool Wait statements in the Activity tab.
Advice	Common scenarios for this wait occur when the shared pool is either too small or too big. Verify that your shared pool is sized according to the type of application being used (cursor sharing, literals usage, and so on.)

High Rollback Segment Wait

Your instance has spent much of its In Oracle time waiting for the group event Rollback Segment Wait.

Table 12-5   High Rollback Segment Wait

	Description
What to do next	Perform one of the following options: ■ Examine the Oracle events that are grouped into the Wait for Rollback segment. Determine the dominant Oracle event and follow the tuning scenario set by this event. ■ Examine the statements or objects with the highest values of Rollback Segment Wait and determine which applications are creating this wait.
Advice	To reduce contention on the rollback segment, consider one of the following solutions: ■ Add rollback segments, moving them into a less busy tablespace. ■ Change the application flow or change the rollback policy (using no logging on specific objects).

High Redo Log Buffer Wait

Your instance has spent much of its In Oracle time waiting for the Redo Log.

Table 12-6   High Redo Log Buffer Wait

	Description
What to do next	Perform one of the following options: ■ Examine the related Oracle events (lower area), Redo Activity (upper area), to determine the problem type in the Statistics tab. ■ Examine high Redo Log Buffer Wait statements in the Activity tab.

Table 12-6 High Redo Log Buffer Wait

	Description
Advice	Use any one of the typical problem scenarios described below. ■ If the log buffer size is too small, this usually results in long waits for the Log Buffer Space event. Consider increasing the Log_buffer parameter. ■ If the log buffer size is too big, this usually results in a low number of user commits, high redo wastage statistics, and long waits for the Log File Sync event. Consider decreasing the Log_buffer parameter and/or the hidden LOG_I/O_SIZE parameter. ■ If there are too many commits, this usually results in long waits for the Log File Sync event and the number of user commits is very high. Consider changing the application flow and logic (by decreasing the commit frequency or using bulk commits [resulting in larger transactions]). ■ If the LGWR is too slow, check whether Log File Sync is still the dominant event. This may be due to high values for Log File Parallel Write, or because there are not many commits. This may mean that the LGWR is underperforming. Consider moving the log file to a faster, dedicated device. Whenever the Log Buffer Space and Log File Sync events occur together, consider changing the hidden LOG_I/O_SIZE parameter.

High Log Switch and Clear Wait

Your instance has spent much of its In Oracle time waiting for the group event Log Switch and Clear.

Table 12-7 High Log Switch and Clear Wait

	Description
What to do next	Perform one of the following options: ■ Examine the Oracle events that are grouped into the Wait for Log Switch and Clear. Determine the dominant Oracle event and follow the tuning scenario set by this event in the Statistics tab. ■ Examine the statements with the highest values for this wait and determine which applications are creating this wait in the Activity tab.
Advice	If the related Oracle events show too many log switches, try and reduce them by one of the following options: ■ Increase the Redo log size (when the Log File Switch (checkpoint incomplete) and/or Log File Switch Completion event is dominant). ■ Change the application flow or logging policy (by changing the commit frequency or using No Logging on specific objects). There can be other reasons for a high Log Switch and Clear wait, such as an LGWR delay where the files cannot be switched until ARCH archiving is completed. This is usually caused by the Log File Switch (archiving needed) event.

High RAC/OPS Wait

Your instance has spent much of its In Oracle time waiting for RAC or OPS.

Table 12-8 High RAC/OPS Wait

	Description
What to do next	Perform one of the following options: ■ Examine the Oracle events that are grouped in the Oracle RAC/OPS Wait. Determine the dominant Oracle event and follow the tuning scenario set by this event in the Statistics tab. ■ Examine Load balancing between RAC instances in the Activity tab. ■ Examine Heavy objects suffering from RAC Waits in the Activity tab.

Table 12-8 High RAC/OPS Wait

	Description
Advice	There are two typical scenarios relevant to RAC Waits. Launch to the Dashboard RAC Database view. This view compares the selected instance with other instances in the same database. From the RAC Database view, examine each of the following issues: ■ If there is a balancing instances issue, launch to the Activity tab and identify the root cause for this unbalanced issue. ■ If there is a RAC Wait event, compare them to other events in the database in the Statistics tab. ■ If there are Objects suffering from RAC Waits, launch to the Activity tab and identify their use across Database instances.

High Other Lock Wait

Your instance has spent much of its In Oracle time waiting for a latch.

Table 12-9 High Other Lock Wait

	Description
What to do next	Perform one of the following options: ■ Examine Latching view overtime and the Oracle latches that are grouped in the Other Lock Wait. Then determine the dominant Oracle latch or enqueue and follow the tuning scenario set by this latch in the Statistics tab. ■ Examine high Other Lock Wait statements in the Activity tab.
Advice	Examine the Oracle latches that are grouped into the Other Lock Wait. Determine the dominant Oracle latch or enqueue and follow the tuning scenario set by this event.

High Background Process Wait

Your instance has spent much of its In Oracle time waiting for the group event Background Process Wait.

Table 12-10 High Background Process Wait

	Description
What to do next	Perform one of the following options: ■ Examine the Oracle events that are grouped in the Background Process Wait. Determine the dominant Oracle event and follow the tuning scenario set by this event in the Statistics tab. ■ Examine high Background Process Wait statements in the Activity tab.
Advice	Examine the Oracle events that are grouped into the Background Processes Wait. Determine the dominant Oracle event and follow the tuning scenario set by this event.

High Parallel Query Server Wait

Your instance has spent much of its In Oracle time waiting for the group event Parallel Query Server Wait.

Table 12-11 High Parallel Query Server Wait

	Description
What to do next	Perform one of the following options: ■ Examine the Oracle events that are grouped in the Parallel Query Server Wait. Determine the dominant Oracle event and follow the tuning scenario set by this event in the Statistics tab. ■ Examine high Parallel Query Server Wait statements in the Activity tab.
Advice	Examine the Oracle events that are grouped into the Parallel Query Server Wait. Determine the dominant Oracle event and follow the tuning scenario set by this event.

High Other Wait

Your instance has spent much of its In Oracle time waiting for the group event Other Wait.

Table 12-12   High Other Wait

	Description
What to do next	Perform one of the following options: ■ Examine the Oracle events that are grouped as Other Wait. Determine the dominant Oracle event and follow the tuning scenario set by this event in the Statistics tab. ■ Examine high Other Wait statements in the Activity tab.
Advice	In the Statistics tab, examine the Oracle events that are grouped as Other Wait. Determine the dominant Oracle event and follow the tuning scenario set by this event.

High Buffer Wait

Your instance has spent much of its In Oracle time waiting for database buffers.

Table 12-13   High Buffer Wait

	Description
What to do next	Perform one of the following options: ■ Examine the Oracle events that are grouped in the Buffer Wait. Determine the dominant Oracle event and follow the tuning scenario set by this event in the Statistics tab. ■ If the dominant Oracle event is buffer busy, launch into the Activity tab and locate the objects with the highest buffer wait.
Advice	There are two typical scenarios relevant to buffer wait that are determined by the dominant Oracle event: ■ Buffer Busy wait event There is high contention on specific table blocks. To reduce contention, increase the Freelists or the Pctfree for the table. ■ Free Buffer Wait event There are no available buffers in the buffer cache. This is usually an I/O wait related problem. Do the following: Try and tune the heaviest statements or objects. If this event persists after statement tuning, try and increase DBWR throughput by adding more DBWR processes or DBWR_I/O_SLAVES. Increase the buffer cache size.

High Remote Query Wait

Your instance has spent much of its In Oracle time waiting for remote queries to complete.

Table 12-14   High Remote Query Wait

	Description
What to do next	Perform one of the following solutions: ■ Examine DBLink relevant statistics in the Statistics tab. ■ Try to identify the statement suffering from a high wait for Oracle Comm Wait in the Activity tab.

Table 12-14   High Remote Query Wait

	Description
Advice	To reduce remote access wait time, consider the following options: ■ Launch Precise for Oracle on the remote instance, locate the remote query, and tune the statement. ■ Control the driving instance executing the statement by using the DRIVING_SITE hint. ■ Tune SQL*Net throughput by checking SDU and TDU settings. ■ Include TCP.NODELAY=yes parameter in the SQLNET.ORA configuration file. ■ Specify the parameters TDU and SDU in the connection description on the application client. For example: in the TNSNAMES.ORA configuration file and in the LISTENER.ORA configuration file on the Oracle database server.

High Client Communication Wait

Your instance has spent much of its In Oracle time waiting for data from the Oracle server.

Table 12-15   High Client Communication Wait

	Description
What to do next	Perform one of the following options: ■ Examine SQL*Net relevant statistics in the Statistics tab. ■ Identify top statements that suffer from the high Oracle Client Comm Wait in the Activity tab.
Advice	To reduce the amount of data transferred using SQL*Net, consider the following solutions: ■ Change the SQL statements so that only needed data is transferred to the client. For example, select only columns that are actually used in the client application and only retrieve required rows. ■ Check that you are not suffering implicit delays in your TCP/IP network by specifying TCP.NODELAY=yes in the SQLNET.ORA files for the Oracle listener and for the application client. ■ Specify the parameters TDU and SDU in the connection description on the application client. For example: in the TNSNAMES.ORA configuration file and in the LISTENER.ORA configuration file on the Oracle database server.

High Resource Manager Wait

Your instance has spent much of its In Oracle time waiting for the group event Resource Manager Wait.

Table 12-16   High Resource Manager Wait

	Description
What to do next	Perform one of the following options: Examine the top statements that suffer from high Resource Manager Wait, and the influence of the wait on the instance in the Activity tab. Try to identify the most dominant events related to Resource Manager Wait in the Statistics tab.
Advice	To reduce the high Resource Manager Wait, consider the following solutions: Examine the Overtime graph for the instance, to measure the severity of the Resource Manager Wait. Also check the top statements that suffer from Resource Manager Wait. Checking the statements can reveal whether the problem relates to a specific statement. In the Statistics tab, examine the Oracle events that are grouped to Resource Manager Wait. Determine the dominant Oracle event and follow the tuning scenario set by this event.

High MTS Wait

Your instance has spent much of its In Oracle time waiting for MTS Wait.

Table 12-17   High MTS Wait

	Description
What to do next	Perform one of the following options: ■ Examine high MTS wait by programs in the Activity tab. ■ Try to identify the most dominant events related to MTS Wait in the Statistics tab.
Advice	To reduce high MTS wait, consider the following solutions: ■ Examine the Overtime graph for the instance, to measure the severity of the MTS Wait. Look for top programs suffering from MTS Wait. ■ Examine the memory usage of MTS connections by querying v$sesstat and v$sessions. Measure the maximum amount of UGA memory used at any given moment and divide that amount by the number of current user sessions. This determines the average amount of memory each connection allocates. ■ If your application is not suited to MTS, use "dedicated" connections which create a separate server process for each user connection.

Heavy Statement

The statement is a major consumer of Oracle resources. By tuning the statement, you may free resources needed by other statements and processes.

Table 12-18   Heavy Statement

	Description
What to do next	Try to determine what is causing the statement's high resource consumption. In the SQL tab, examine the text of the relevant statement, and its findings, execution plan, change data and statistics.

Advice	For resource consumption, these are the possible scenarios:
	■ Object wait on statement objects.
	  Use SQL findings to identify the heaviest waiting object.
	■ High CPU consumption without object wait.
	  Check the number of executions (to identify possible infinite loops) or intensive statements (with a low average time but a high aggregated CPU time).
	■ Instance-related wait (such as: internal lock wait, shared pool wait, and redo log wait).
	  In this case, switch to the Statistics tab and examine the breakdown of this state in Oracle events.
	■ A change in the execution plan.
	  If this is responsible for the statement's performance degradation, check the History tab and the All Plans view to identify the actual change that caused the slowdown.
	■ Check the Binds tab for possible offensive values resulting in differing execution plans and performance.

Frequently Executed Statement

The statement is a major consumer of Oracle resources. This statement is frequently executed with a low In Oracle time average.

Table 12-19   Frequently Executed Statement

	Description
What to do next	Go to the Activity tab and examine the statement executors (programs and users).

Table 12-19   Frequently Executed Statement

	Description
Advice	Examine the Activity tab for statement exaggerated usage patterns.
	Try tuning scenarios for resource consumption from the following list:
	■ Object wait on the statement objects. Use SQL findings to identify the heaviest waiting object.
	■ High CPU consumption without object wait. Check the number of executions (to identify possible infinite loops) or intensive statements (with a low average time but a high aggregated CPU time).
	■ Instance related wait (such as: Internal lock wait, Shared pool wait, Redo log wait). In this case, switch to the Statistics tab and examine the breakdown of this state in Oracle events.
	■ A change in the execution plan. This is responsible for the statements performance degradation. Use history and all plans to identify the actual change that caused the slowdown.
	■ Check the binds tab for possible offensive values resulting in differing execution plans and performance.

Heavily Accessed Object

Much of the instance In Oracle time was spent on waits (lock, I/O, Buffer, and so on) for the object.

Table 12-20   Heavily Accessed Object

	Description
What to do next	Perform one of the following options:
	■ To tune the object, go to the Tune Object tab.
	■ Examine the waits for the object in the Activity tab.
Advice	To reduce the Waits for the object, follow the findings instructions in the Object tab.
	In the Object tab, you can examine the following object data:
	■ Access path pattern to the object (full scans, range scans, and so on)
	■ Read-Write access patterns
	■ Partition usage (such as whether there are extreme cases in use)
	■ Object changes versus performance changes

Locked Object

Much of the instance In Oracle time is spent waiting for a lock on the table.

Table 12-21   Locked Object

	Description
What to do next	Perform one of the following options:
	■ To extensively tune the object, go to the Tune Object tab.
	■ Examine the Lock for the statement in the Activity tab.
Advice	To reduce the lock wait for the table, consider the following solutions:
	■ Check if the lock appears in the Current tab. If so, examine the lock chain to discover which statement is holding the lock.
	■ Try to identify the locking statement in the Activity tab, using narrow time frames that match the lock periods. Focus on the locked table and its associated statements. The immediate suspect is the DML statements (and update queries) that are not waiting for locks.

High Sorts on Disk

The result table for a sort operation could not be completed in memory and was performed on a temporary tablespace.

Table 12-22   High Sorts on Disk

	Description
What to do next	In the Activity tab, examine temporary tablespace overtime I/O consumption, and the statements using temporary tablespace I/O.

Advice	To reduce the I/O consumption for the sort operation, consider the following solutions:
	■ Change the SORT_AREA_SIZE to a higher value.
	   If you are using PGA settings, you can change the PGA_Aggregated_Target, so as to avoid sort and hash area size problems. You can either change the values for a specific session, using the `Alter Session` command, or for the entire instance.
	■ Try to identify statement-originated sorts.
	   If there are many sorts located in a few statements, try to solve the problem at the statement level by doing one of the following:
	   •Add an index to prevent a sort.
	   If your statement has an Order by clause that has columns for a single table, check whether you can add an index. In some cases, you may get an index recommendation that prevents a sort. If you are considering adding an index, check the effect of adding that index in the What-If tab.
	   •Identify the heavy sort or hash consumer step.
	   Run a statement with statistics_level=all. Click the Run and Compare tab. Examine LAST_TEMPSEG_SIZE and MAX_TEMPSEG_SIZE in the extended section of the run results. Change the SORT_AREA_SIZE to a higher value.

High Undo Activity

Much of the instance I/O is spent waiting for the Undo object.

Table 12-23   High Undo Activity

	Description
What to do next	Examine Undo activity over time and the statement accessing it, in the Activity tab.
Advice	Examine Undo behavior over time, identify the statement accessing it, and try to tune them.

Heavily Accessed Cluster

Much of the instance I/O is spent waiting for the cluster.

Table 12-24   Heavy Cluster Activity

	Description
What to do next	Examine Cluster activity over time and the statement accessing it, in the Activity tab.
Advice	Examine Cluster behavior over time, identify the statement accessing it, and try to tune them.

Locked Cluster

Much of the instance time is spent waiting for a lock on the cluster.

Table 12-25   Cluster Locks

	Description
What to do next	Examine the lock for the statement in the Activity tab.

Table 12-25   Cluster Locks

	Description
Advice	To reduce the lock wait for the table, consider the following solutions: ■ Check if the lock appears in the Current tab. If so, examine the lock chain to discover the statement holding the lock. ■ Try to identify the locking statement in the Activity tab, using narrow time frames that match the lock periods. Focus on the locked table and associated statements. The immediate suspect is the DML statements (and update queries) that are not waiting for locks.

Storage Contention on Device (Clariion)

The fact that a storage device (LUN) is causing a lot of I/O waits could be caused from an intensive load or as a result of two sorts of contentions: a logical contention (e.g. imbalanced activity of the database) or a physical contention (e.g. one of the underlying physical devices is being shared with another heavy I/O consuming activity).

Table 12-26   Storage Contention On Device

	Description
What to do next	■ Examine the device activity over time and database files contention. ■ Examine storage device statistics and contention on the Raid Group and Physical Disks.
Advice	■ If the device is loaded by the monitored database only and by a singular entity (e.g. a file, object, or partition), consider splitting this load (e.g. separating the objects in the file, partitioning the object, etc). ■ To relieve inter application logical contention, check if the database's I/O activity is balanced. Spread heavy I/O consuming files across the storage devices, to avoid a situation in which few heavy files reside on the same storage device. ■ To relieve intra application logical contention, check whether there are additional applications using the storage device. For example, if the number of I/O requests processed by the storage device is significantly higher that the requests sent by the database, it means that the storage device is being used by an additional application. ■ To relieve physical contention, check whether there is significant I/O activity in the underlying shared physical disks and raid group. Another potential cause of contention are the EMC adapters (front director and disk director). If the load is imbalanced, consult with the storage administrator about relocating the information to other disks which reside on a more vacant location. ■ Consider storage tiering - a faster device may reduce the I/O wait time significantly.

Storage Contention on Device (Symmetrix Thick)

The fact that a storage device (LUN) is causing a lot of I/O waits could be caused from an intensive load or as a result of two sorts of contentions: a logical contention (e.g. imbalanced activity of the database) or a physical contention (e.g. one of the underlying physical devices is being shared with another heavy I/O consuming activity).

Table 12-27   Storage Contention On Device

	Description
What to do next	■ Examine the device activity over time and database files contention. ■ Examine storage device statistics and contention on the Raid Group and Physical Disks.

Table 12-27 Storage Contention On Device

	Description
Advice	■ If the device is loaded by the monitored database only and by a singular entity (e.g. a file, object, or partition), consider splitting this load (e.g. separating the objects in the file, partitioning the object, etc).
	■ To relieve inter application logical contention, check if the database's I/O activity is balanced. Spread heavy I/O consuming files across the storage devices, to avoid a situation in which few heavy files reside on the same storage device.
	■ To relieve intra application logical contention, check whether there are additional applications using the storage device. For example, if the number of I/O requests processed by the storage device is significantly higher that the requests sent by the database, it means that the storage device is being used by an additional application.
	■ To relieve physical contention, check whether there is significant I/O activity in the underlying shared physical disks and raid group. Another potential cause of contention are the EMC adapters (front director and disk director). If the load is imbalanced, consult with the storage administrator about relocating the information to other disks which reside on a more vacant location.
	■ Consider storage tiering - a faster device may reduce the I/O wait time significantly.

Storage Contention on Device (Symmetrix Thin)

The fact that a storage device (LUN) is causing a lot of I/O waits could be caused from an intensive load or as a result of two sorts of contentions: a logical contention (e.g. imbalanced activity of the database) or a physical contention (e.g. one of the underlying physical devices is being shared with another heavy I/O consuming activity).

Table 12-28 Storage Contention On Device

	Description
What to do next	■ Examine the <u>device activity</u> over time and database files contention.
	■ Examine storage device statistics and contention on the <u>Raid Group</u> and <u>Physical Disks</u>.
Advice	■ If the device is loaded by the monitored database only and by a singular entity (e.g. a file, object, or partition), consider splitting this load (e.g. separating the objects in the file, partitioning the object, etc).
	■ To relieve inter application logical contention, check if the database's I/O activity is balanced. Spread heavy I/O consuming files across the storage devices, to avoid a situation in which few heavy files reside on the same storage device.
	■ To relieve intra application logical contention, check whether there are additional applications using the storage device. For example, if the number of I/O requests processed by the storage device is significantly higher that the requests sent by the database, it means that the storage device is being used by an additional application.
	■ To relieve physical contention, check whether there is significant I/O activity in the underlying shared physical disks and raid group. Another potential cause of contention are the EMC adapters (front director and disk director). If the load is imbalanced, consult with the storage administrator about relocating the information to other disks which reside on a more vacant location.
	■ Consider storage tiering - a faster device may reduce the I/O wait time significantly.

Storage Contention on Device (Symmetrix F.A.S.T. VP)

The fact that a storage device (LUN) is causing a lot of I/O waits could be caused from an intensive load or as a result of two sorts of contentions: a logical contention (e.g. imbalanced activity of the database) or a physical contention (e.g. one of the underlying physical devices is being shared with another heavy I/O consuming activity).

Table 12-29   Storage Contention On Device

	Description
What to do next	■ Examine the <u>device activity</u> over time and database files contention. ■ Examine storage device statistics and contention on the <u>Raid Group</u> and <u>Physical Disks</u>.
Advice	■ If the device is loaded by the monitored database only and by a singular entity (e.g. a file, object, or partition), consider splitting this load (e.g. separating the objects in the file, partitioning the object, etc). ■ To relieve inter application logical contention, check if the database's I/O activity is balanced. Spread heavy I/O consuming files across the storage devices, to avoid a situation in which few heavy files reside on the same storage device. ■ To relieve intra application logical contention, check whether there are additional applications using the storage device. For example, if the number of I/O requests processed by the storage device is significantly higher that the requests sent by the database, it means that the storage device is being used by an additional application. ■ To relieve physical contention, check whether there is significant I/O activity in the underlying shared physical disks and raid group. Another potential cause of contention are the EMC adapters (front director and disk director). If the load is imbalanced, consult with the storage administrator about relocating the information to other disks which reside on a more vacant location. ■ Consider storage tiering - a faster device may reduce the I/O wait time significantly.

Storage Contention on Redo Logs and DB Files

Redo/Transaction Log files are frequently accessed by the database. The majority of the operations performed are writing commands, which cause a heavy load on the underlying disks.

As these files are considered heavy I/O consumers, it is highly recommended to place them on a separate disk without other database files. Separating the Redo/Transaction Logs files by placing them on different volumes (e.g. E:/ and F:/) may not be enough, as the storage devices (LUNs) and physical disks may be shared between several file systems and volumes.

Table 12-30   Storage Contention on Redo Logs and DB Files

	Description
What to do next	■ Examine the <u>device activity</u> over time and database files contention. ■ Examine storage device statistics and contention on the <u>Raid Group</u> and <u>Physical Disks</u>.
Advice	It has been detected that the Redo/Transaction Log files share the storage devices (LUNs) with other database files. Consult the storage administrator about provisioning the storage devices (LUNs) better to avoid this.

Storage Contention on Temporary Objects

Temporary tablespace files are frequently accessed by the database. The majority of the operation performed are writing commands, which cause a heavy load on the underlying disks.

As these files are considered heavy I/O consumers, it is highly recommended to place them on a separate disk without other database files. Separating the temporary tablespace files by placing them on different volumes (e.g. E:/ and F:/)

may not be enough, as the storage devices (LUNs) and physical disks may be shared between several file systems and volumes.

Table 12-31   Storage Device on Temporary Objects

	Description
What to do next	■ Examine the <u>device activity</u> over time and database files contention. ■ Examine storage device statistics and contention on the <u>Raid Group</u> and <u>Physical Disks</u>.
Advice	It has been detected that the temporary tablespace files share the storage devices (LUNs) with other database files. Consult the storage administrator about provisioning the storage devices (LUNs) better to avoid this.

Heavy Storage Device Holding Undo Objects

Undo tablespace files are frequently accessed by the database. The majority of the operation performed are writing commands, which cause a heavy load on the underlying disks.

As these files are considered heavy I/O consumers, it is highly recommended to place them on a separate disk without other database files. Separating the undo tablespace files by placing them on different volumes (e.g. E:/ and F:/) may not be enough, as the storage devices (LUNs) and physical disks may be shared between several file systems and volumes.

Table 12-32   Heavy Storage Device Holding Undo Objects

	Description
What to do next	■ Examine the <u>device activity</u> over time and database files contention. ■ Examine storage device statistics and contention on the <u>Raid Group</u> and <u>Physical Disks</u>.
Advice	It has been detected that the undo tablespace files share the storage devices (LUNs) with other database files. Consult the storage administrator about provisioning the storage devices (LUNs) better to avoid this.

Unbalanced Storage Devices Activity

There are several storage devices (LUNs) allocated to the instance. However, the I/O activity is not spread evenly across these storage devices. The contention on the heavy storage devices increases the response time for the activities run on them. Such a situation can be caused by imbalanced internal database activity, contention on the storage device by other applications or an inefficient RAID policy.

Table 12-33   Unbalanced Storage Devices Activity

	Description
What to do next	■ Compare the <u>storage devices activity</u> over time. ■ Examine the <u>storage devices statistics</u>.
Advice	■ In the Activity tab, check which database files are the most I/O consuming and spread them evenly across the storage devices. ■ Consult with the storage administrator and check for other applications using the same storage devices or their underlying physical disks. ■ Consult with the storage administrator about the RAID policy. A different striping may spread the I/O load across the storage devices.

Heavy J2EE Caller Service

The J2EE Caller Service is a major consumer of Oracle resources. By tuning it, you may free resources needed by other Caller Services and processes.

Table 12-34   Heavy J2EE Caller Service

	Description
What to do next	■ Try to determine what is causing the J2EE Caller Service's high resource consumption. ■ In the Activity tab, examine its findings, relevant statements, objects, and the Oracle resources
Advice	For resource consumption, these are the possible scenarios: ■ Object bottleneck – Examine the resource consumption of objects in context of this Caller Service in the Activity tab. ■ Statement bottleneck – Examine the resource consumption of SQL statements in context of this Caller Service in the Activity tab. ■ Instance-related wait (such as: internal lock wait, shared pool wait, and redo log wait). In this case, switch to the Statistics tab and examine the breakdown of this state in Oracle events.

High SQL Executions for J2EE Caller Service

The J2EE Caller Service is a major consumer of Oracle resources, issuing and exceptionally high number of SQL Statements executions. By tuning it and reducing the number of executions, you may free resources needed by other Caller Services and processes.

Table 12-35   High SQL Executions for J2EE Caller Service

	Description
What to do next	■ Try to determine what is causing the J2EE Caller Service's high resource consumption and high number of executions. ■ In the J2EE application Expert view examine the entry point logic to detect redundant or infinite loop executions. ■ In the Activity tab, examine its findings, relevant statements, objects, and the Oracle resources.
Advice	For resource consumption, these are the possible scenarios: ■ Object bottleneck – Examine the resource consumption of objects in context of this Caller Service in the Activity tab. ■ Statement bottleneck – Examine the resource consumption of SQL statements in context of this Caller Service in the Activity tab. ■ Instance-related wait (such as: internal lock wait, shared pool wait, and redo log wait). In this case, switch to the Statistics tab and examine the breakdown of this state in Oracle events.

About Activity tab findings

Several Activity tab findings exist in the table to help the user. These findings are available in Service Caller and Web Transactions entities.

The Activity tab has the following findings:

■ Slow Statement

■ Heavy Full Scan

■ High Redo Log Buffer Wait

■ High Buffer Wait

■ High Temporary I/O

■   Heavy Statement

Slow Statement

The statement is identified with a high average time in the chosen context. By tuning the statement, you may improve the application response time.

Table 12-36   Slow Statement

	Description
What to do next	Try to determine what is causing the statement's high average time. In the SQL tab, examine the text of the relevant statement, and its findings, execution plan, change data and statistics. The Recommend is executed as explain in the Recommend tab.

Advice	For resource consumption, these are the possible scenarios:
	■ Object wait on statement objects.
	     Use SQL findings to identify the heaviest waiting object.
	■ High CPU consumption without object wait.
	     Check the number of executions (to identify possible infinite loops) or intensive statements (with a low average time but a high aggregated CPU time).
	■ Instance-related wait (such as: internal lock wait, shared pool wait, and redo log wait).
	     In this case, switch to the Statistics tab and examine the breakdown of this state in Oracle events.
	■ A change in the execution plan.
	     If this is responsible for the statement's performance degradation, check the History tab and the All Plans view to identify the actual change that caused the slowdown.
	■ Check the Binds tab for possible offensive values resulting in differing execution plans and performance.

Heavy Full Scan

The J2EE Caller Service in the chosen context, spent a high amount of its resources on performing disk related full scans on the referenced object.

Table 12-37   Heavy Full Scan

	Description
What to do next	Try to determine what is causing the J2EE Caller Service to spent a high amount of its resources on performing disk related full scans on the referenced objects: ■ Get index recommendations for the SQL statements of the Caller Service. When activating the Recommend, Precise generates a filtered statements workload in a context of the time frame and caller service/Web transaction. Oracle advisor provides enhanced recommendations based on the provided workload. The recommendation can be evaluated by the What-if analysis. ■ Examine the Scattered and Direct I/O, translated into full scan access, over time and related objects. ■ Examine the Scattered and Direct I/O, translated into full scan access, over time and related devices.
Advice	To reduce the I/O consumption for the J2EE Caller Service, consider the following solutions: ■ Create an index matching the statement's predicates. ■ Change the DB_FILE_MULTIBLOCK_READ_COUNT parameter in INIT.ORA to a higher value. ■ Partition the objects according to the best predicates existing in the J2EE Caller Service's statements. ■ Use a parallel query option. ■ Move the objects to another tablespace with a higher block size.

High Redo Log Buffer Wait

The J2EE Caller Service in the chosen context, spent much of its In Oracle time waiting for the Redo Log events.

Table 12-38   High Redo Log Buffer Wait

	Description
What to do next	Perform one of the following options: ■ Examine the top statements waiting for the Redo Log Buffer events. ■ Examine the Redo Log Buffer events for the entire instance.
Advice	Use any one of the typical problem scenarios described below. ■ If the log buffer size is too small, this usually results in long waits for the Log Buffer Space event. Consider increasing the Log_buffer parameter. ■ If the log buffer size is too big, this usually results in a low number of user commits, high redo wastage statistics, and long waits for the Log File Sync event. Consider decreasing the Log_buffer parameter and/or the hidden LOG_I/O_SIZE parameter. ■ If there are too many commits, this usually results in long waits for the Log File Sync event and the number of user commits is very high. Consider changing the application flow and logic (by decreasing the commit frequency or using bulk commits [resulting in larger transactions]). ■ If the LGWR is too slow, check whether Log File Sync is still the dominant event. This may be due to high values for Log File Parallel Write, or because there are not many commits. This may mean that the LGWR is underperforming. Consider moving the log file to a faster, dedicated device. Whenever the Log Buffer Space and Log File Sync events occur together, consider changing the hidden LOG_I/O_SIZE parameter.

High Buffer Wait

The J2EE Caller Service in the chosen context, spent much of its In Oracle time waiting for database buffer events.

Table 12-39   High Buffer Wait

	Description
What to do next	Perform one of the following options: ■ Examine the top objects waiting for the Buffer events. ■ Examine the Buffer events for the entire instance.
Advice	There are two typical scenarios relevant to buffer wait that are determined by the dominant Oracle event: ■ Buffer Busy wait event There is high contention on specific table blocks. To reduce contention, increase the Freelists or the Pctfree for the table. ■ Free Buffer Wait event There are no available buffers in the buffer cache. This is usually an I/O wait related problem. Do the following: Try and tune the heaviest statements or objects. If this event persists after statement tuning, try and increase DBWR throughput by adding more DBWR processes or DBWR_I/O_SLAVES. Increase the buffer cache size.

High Temporary I/O

The J2EE Caller Service in the chosen context, spent much of its In Oracle time waiting for the waiting on sort or hash operations performed on a temporary tablespace.

Table 12-40   High Temporary I/O

	Description
What to do next	Perform the following: ■ Examine the temporary I/O activity over time and related statements
Advice	To reduce the I/O consumption for the sort operation, consider the following solutions: ■ Change the SORT_AREA_SIZE to a higher value. If you are using PGA settings, you can change the PGA_Aggregated_Target, so as to avoid sort and hash area size problems. You can either change the values for a specific session, using the `Alter Session` command, or for the entire instance. ■ Try to identify statement-originated sorts. If there are many sorts located in a few statements, try to solve the problem at the statement level by doing one of the following: •Add an index to prevent a sort. If your statement has an Order by clause that has columns for a single table, check whether you can add an index. In some cases, you may get an index recommendation that prevents a sort. If you are considering adding an index, check the effect of adding that index in the What-If tab. •Identify the heavy sort or hash consumer step. Run a statement with statistics_level=all. Click the Run and Compare tab. Examine LAST_TEMPSEG_SIZE and MAX_TEMPSEG_SIZE in the extended section of the run results. Change the SORT_AREA_SIZE to a higher value.

Heavy Statement

The statement is identified as a major consumer of the selected J2EE Caller's Oracle resources. By tuning the statement, you may improve the application response time.

Table 12-41   Heavy Statement

	Description
What to do next	Perform one of the following options: ■ Tune the statement using SQL Tune methodology ■ Examine the statement activity over time ■ Tune the caller service using J2EE application expert views
Advice	For resource consumption, these are the possible scenarios: ■ Object wait on statement objects. Use SQL findings to identify the heaviest waiting object. ■ High CPU consumption without object wait. Check the number of executions (to identify possible infinite loops) or intensive statements (with a low average time but a high aggregated CPU time). ■ Instance-related wait (such as: internal lock wait, shared pool wait, and redo log wait). In this case, switch to the Statistics tab and examine the breakdown of this state in Oracle events. ■ A change in the execution plan. If this is responsible for the statement's performance degradation, check the History tab and the All Plans view to identify the actual change that caused the slowdown. ■ Check the Binds tab for possible offensive values resulting in differing execution plans and performance.

About SQL tab findings

Several SQL tab findings exist to help the user. The SQL tab has the following findings:

- Sorts Performed on Disk
- No Parallel Processes Available
- Bottleneck in Remote Access
- Heavy Scattered I/O on Index
- Heavy Sequential I/O on Index
- Heavy Scattered I/O on Table
- Heavy Sequential I/O on Table
- Heavy I/O Due to Direct Access
- Heavy I/O Due to Other Access
- Statement State Row Lock
- Buffer Wait Contention
- Redo Log Activity
- Undo Activity
- RAC Wait
- Bind Variables Were Collected
- More Than One Real Plan Was Detected
- Costs Have Changed Over the Last Month
- Frequently Executed Statement
- The Average Execution Uses CPU Heavily
- Heavy Index Overhead
- Preferable Plan Detected by Oracle
- Low End of Fetch Count
- Major Difference Between Plans
- Newer Execution Plan Exists
- Cartesian Join Used
- CPU Used for Sorts
- CPU Used for Index Scattered Read
- CPU Used for Index Sequential Read
- CPU Used for Table Scattered Read
- CPU Used for Table Sequential Read
- Heavy Table Full Scan
- Heavy Step
- Heavy Sort
- Heavy Hash
- Heavy Merge
- Heavy Index Full Scan

- Heavy Index Range Scan

- Heavy Index Skip Scan

- Heavy Cartesian Join

Sorts Performed on Disk

The result table for a sort operation could not be completed in memory and was performed on a temporary tablespace.

Table 12-42   Sorts Performed on Disk

	Description
What to do next	In the Activity tab, examine temporary tablespace overtime I/O consumption for the statement, and the programs activating the statement.
Advice	To reduce the I/O consumption for the sort operation, consider the following solutions: ■ Change the SORT_AREA_SIZE to a higher value. If you are using PGA settings, you can change the PGA_Aggregated_Target, so as to avoid sort and hash area size problems. You can either change the values for a specific session using the Alter Session command, or for the entire instance. ■ Try to identify statement originated sorts. If there are many sorts located in a few statements, try to solve the problem at the statement level by one of the following options: •Add an index to prevent a sort. If your statement has an Order by clause that has columns for a single table, check whether you can add an index. In some cases, you may get an index recommendation that prevents a sort. If you are considering adding an index, you must check the effect of that index. To analyze that effect, launch to the What-If tab. •Identify the heavy sort or hash consumer step. Run a statement with statistics_level=all. Click the Run & Compare tab. Examine the LAST_TEMPSEG_SIZE and MAX_TEMPSEG_SIZE in the extended section of the run results, and set parameters according to the first advice.

No Parallel Processes Available

Some of the executions for the statement were not run in parallel; they worked serially. Oracle has reached the threshold of the MAX_PARALLEL_SERVERS and was not able to allocate parallel processes for the statement.

Table 12-43   No Parallel Processes Available

	Description
What to do next	Select the finding type to see the minimum and maximum parallel processes available for the statement in the Activity tab.
Advice	To prevent this problem, consider one of the following solutions: ■ Increase the MAX_PARALLEL_SERVERS parameter in INIT.ORA while closely monitoring the effect on wait for memory. This will lower the chances for a serial execution, but may cause a high memory consumption. ■ Set PARALLEL_AUTOMATIC_TUNING parameter in INIT.ORA to TRUE. This will make Oracle use more sophisticated algorithms in determining the number of PQ processes for each session.

Bottleneck in Remote Access

Your statement has spent much of its In Oracle time waiting for a remote query to complete.

Table 12-44 Bottleneck in Remote Access

	Description
What to do next	Perform one of the following options: ■ Select the findings type to see the SQL text sent to the remote database on the Statistics tab, Other column. ■ Examine DBLink relevant statistics in the Statistics tab.
Advice	To reduce remote access wait time, consider the following options: ■ Launch Precise for Oracle on the remote instance, locate the remote query, and tune the statement. ■ Control the driving instance executing the statement by using the DRIVING_SITE hint. ■ Tune SQL*Net throughput by checking SDU and TDU settings. ■ Include TCP.NODELAY=yes parameter in the SQLNET.ORA configuration file. ■ Specify the parameters TDU and SDU in the connection description on the application client. For example: the TNSNAMES.ORA configuration file and in the LISTENER.ORA configuration file in the Oracle database server.

Heavy Scattered I/O on Index

Statement I/O is spent on scattered I/O (usually representing a full scan) on the index specified in the Object column.

Table 12-45 Heavy Scattered I/O on Index

	Description
What to do next	Perform one of the following options: ■ Click the Locate icon to find the relevant step in the execution plan ■ Select the findings type to investigate the objects used and their structure in the Objects tab ■ Select the Activity tab, locate the statement associated with objects, and drill to the index consumption for the statement in the Activity tab
Advice	To reduce the index I/O consumption, consider the following solutions: ■ If the index is not used to prevent a sort operation, consider adding the INDEX_FFS hint and change DB_FILE_MULTIBLOCK_READ_COUNT parameter to a higher value. ■ Switch to another index or add a new index. ■ Use a parallel query option. ■ Move the index to another tablespace with a higher block size. Findings refer to the whole statement - not to a specific execution plan. If a step does not exist in the selected execution plan (unless this is due to an index overhead), switch to another plan and locate the relevant step.

Heavy Sequential I/O on Index

Statement I/O is spent on sequential I/O (usually representing a range scan) on the index specified in the Object column. If the statement is DML and the index is not used in the execution plans, then the I/O represents the index maintenance overhead, caused by fetching the index blocks for update to memory.

Table 12-46   Heavy Sequential I/O on Index

	Description
What to do next	Perform one of the following options: ■ Click the Locate icon to see if the index is used in the execution plan. ■ Select the findings type to investigate the objects used and their structure in the Objects tab. ■ Select the Activity tab, locate the statement associated with objects, and drill to the index overtime consumption for the statement in the Activity tab.
Advice	If this is the result of a range scan, consider one of the following solutions: ■ If the index is not used to prevent a sort operation, consider adding the INDEX_FFS hint and change DB_FILE_MULTIBLOCK_READ_COUNT (now standing on X) to a higher value. ■ Switch to another index, add a new index, or change the index structure. ■ Add columns to the index to enable index only access. ■ Switch to a full table scan ■ Partition the table or the index ■ Create a cluster with just one table in it and a cluster key used as the index key. If this is due to an index overhead, consider reducing the index overhead by deleting unused indices, or unused columns in used indices. Findings refer to the whole statement - not to a specific execution plan. If a step doesn't exist in the selected execution plan (unless this is the result of an index overhead), switch to another plan and locate the relevant step.

Heavy Scattered I/O on Table

Statement I/O is spent on scattered I/O (usually representing a full scan) on the table specified in the Object column.

Table 12-47   Heavy Scattered I/O on Table

	Description
What to do next	Perform one of the following options: ■ Click the Locate icon to find the relevant step in the execution plan. ■ Select the findings type to investigate the objects used and their structure in the Objects tab. ■ Select the Activity tab, locate the statement associated with objects, and drill to the table consumption for the statement in the Activity tab
Advice	To reduce the I/O consumption for the table, consider the following solutions: ■ Create an index matching the statement's predicates. ■ Change the DB_FILE_MULTIBLOCK_READ_COUNT parameter in INIT.ORA to a higher value. ■ Partition the table according to the best predicates existing in the statement. ■ Use a parallel query option. ■ Move the table to another tablespace with a higher block size. Findings refer to the whole statement - not to a specific execution plan. If a step doesn't exist in the selected execution plan (unless this is the result of an index overhead), switch to another plan and locate the relevant step.

Heavy Sequential I/O on Table

Statement I/O is spent on sequential I/O (usually representing table access by rowid following an index range scan) on the table specified in the Object column.

Table 12-48   Heavy Sequential I/O on Table

	Description
What to do next	Perform one of the following options: ■ Click the Locate icon to find the relevant step in the execution plan. ■ Select the findings type to investigate the objects used and their structure in the Objects tab. ■ Select the Activity tab, locate the statement associated with objects, and drill to the table consumption for the statement in the Activity tab.
Advice	To reduce the I/O consumption for the table, consider the following solutions: ■ Change the index structure by adding a better filtering column, or change the column sequence to improve the matching and screening performed in the index tree. ■ Add columns to the index to enable index only access. ■ Switch to a full table scan. ■ Partition the table or the index. ■ Create a cluster with just one table in it and a cluster key used as the index key. Findings refer to the whole statement - not to a specific execution plan. If a step doesn't exist in the selected execution plan (unless this is the result of an index overhead), switch to another plan and locate the relevant step.

Heavy I/O Due to Direct Access

Statement I/O is spent on direct I/O (usually representing the SQL Loader in the direct path), on the object specified in the Object column.

Table 12-49   Heavy I/O Due to Direct Access

	Description
What to do next	Perform one of the following options: ■ Click the Locate icon to find the relevant step in the execution plan. ■ Select the findings type to investigate the objects used and their structure in the Objects tab. ■ Select the Activity tab, locate the statement associated with objects, and drill to the object (table/index) consumption for the statement in the Activity tab.
Advice	To reduce the I/O consumption for the object, consider the following solutions: ■ If this is the result of an SQL loader utility in a direct path, verify that the temporary segments for the index updates are on a fast device. ■ Use the SINGLEROW option. Findings refer to the whole statement - not to a specific execution plan. If a step doesn't exist in the selected execution plan (unless this is the result of an index overhead), switch to another plan and locate the relevant step.

Heavy I/O Due to Other Access

Statement I/O is spent on another I/O on the object specified in the Object column.

Table 12-50   Heavy I/O Due to Other Access

	Description
What to do next	Perform one of the following options: ■ Click the Locate icon to find the relevant step in the execution plan. ■ Select the findings type to investigate the objects used and their structure in the Objects tab. ■ Select the Activity tab, locate the statement associated with objects, and drill to the object (table/index) consumption for the statement in the Activity tab. Findings refer to the whole statement - not to a specific execution plan. If a step doesn't exist in the selected execution plan (unless this is the result of an index overhead), switch to another plan and locate the relevant step.

Statement State Row Lock

Much of the statement I/O is spent on waiting for a lock on the table specified in the Object column.

Table 12-51   Statement State Row Lock

	Description
What to do next	Select the findings type to examine lock for the statement in the Activity tab.
Advice	To reduce the lock wait for the table, consider the following solutions: ■ Check to see if the lock appears in the Current tab. If so, examine the lock chain to identify the statement holding the lock. ■ Try to identify the locking statement in the Activity tab using the narrow time frames matching the lock periods. Now focus on the locked table and associated statements. The DML statement (and update queries) that are not waiting for locks are the immediate suspects. Findings refer to the whole statement - not to a specific execution plan. If a step doesn't exist in the selected execution plan (unless this is the result of an index overhead), switch to another plan and locate the relevant step.

Buffer Wait Contention

Your statement has spent much of its In Oracle time on Buffer Wait.

This usually occurs because one of two possible scenarios:

Contention on a table buffer in Insert statements (Buffer Busy Wait), or

Lack of free buffers when trying to load blocks from a disk (Free Buffer Wait)

Table 12-52   Buffer Wait Contention

	Description
What to do next	Perform one of the following options: ■ Select the findings type to identify the tables suffering from Buffer Wait in the Objects tab, or ■ Examine Buffer Wait sub-state events in the Statistics tab.

Table 12-52   Buffer Wait Contention

	Description
Advice	When Buffer Busy Wait is the more dominant Oracle event, consider the following options: ■ Increase the free lists for the table to reduce the chances for contention. ■ Increase the PCTFREE parameter or decrease the block size for the table to spread the data among many blocks and reduce the chances for contention. When Free Buffer Wait is the more dominant Oracle event, consider the following options: ■ Tune the object's access, so as to reduce the number of blocks fetched. ■ If this is a global instance problem, increase the number of DBWR processes or I/O slaves.

Redo Log Activity

Your statement has spent much of its In Oracle time waiting for Redo Log Wait.

Table 12-53   Redo Log Activity

	Description
What to do next	Perform one of the following options: ■ Examine Redo Log Wait overtime I/O consumption for the statement in the Activity tab, or ■ Examine Redo Log Wait relevant statistics in the Statistics tab.
Advice	To reduce Redo Log Wait, consider the following solution: ■ Check Dashboard findings and overtime graphs to see the overall Redo Log activity for the instance.

Undo Activity

Your statement has spent much of its In Oracle time waiting for undo.

Table 12-54   Undo Activity

	Description
What to do next	Perform one of the following options: ■ Examine Undo Wait overtime I/O consumption for the statement in the Activity tab, or ■ Examine Undo Wait relevant statistics in the Statistics tab.
Advice	To reduce Undo Wait time, consider the following solutions: ■ Check Dashboard findings and overtime graphs to see the overall undo activity for the instance.

RAC Wait

Your instance has spent much of its In Oracle time waiting for a RAC activity to complete on the object specified in the Object column.

Table 12-55   RAC Wait

	Description
What to do next	Select the findings type to see the instances consuming object RAC Wait in the Activity tab.
Advice	The object is suffering from a RAC Wait because several instances are using it simultaneously. To solve the problem, identify all programs currently accessing the object and try to avoid accessing it concurrently.

Bind Variables Were Collected

Bind sets were collected for the statement.

Table 12-56 Bind Variables Were Collected

	Description
What to do next	Perform one of the following options: ■ Select the findings type to examine the bind values in the Bind Variables tab. ■ Use the `Estimate Cost` button to check whether the Oracle optimizer identified a better execution plan for a provided set of binds.
Advice	If the Oracle optimizer finds a better execution plan, try to evaluate one plan in relation to another for a set of captured binds and try to stabilize it by using outlines.

More Than One Real Plan Was Detected

More than one real plan was collected for the statement during the selected time frame.

Table 12-57 More Than One Real Plan Was Detected

	Description
What to do next	Select the findings type and use the All Plans tab to examine the different captured execution plans and their real resource consumption.
Advice	Try to identify the source for the different execution plans. To resolve the multiple plans, consider the following solutions: ■ Go to the History tab to see whether there were any changes (such as schema or statistics changes), or ■ go to the Activity tab to see whether it is being run by different programs.

Costs Have Changed Over the Last Month

The cost for the statement has changed over the last month.

Table 12-58 Costs Have Changed Over the Last Month

	Description
What to do next	Select the findings type and use the History tab to examine the different costs caught for the statement.
Advice	Try to identify the source for the different execution plan costs. Go to the History tab to check whether there were any changes (such as schema or statistics changes).

Frequently Executed Statement

Much of the statements In Oracle time is spent on CPU usage, even though the average In Oracle time is low (indicating executions run many times).

Table 12-59 Frequently Executed Statement

	Description
What to do next	Perform one of the following options: ■ Launch to the Activity tab to examine the scalability of the statement. ■ Examine the activity of the program enabling the statement.
Advice	Try to determine if the large number of executions is valid or if it is derived from redundant executions as a result of inefficient program design (e.g. if the statement is enabled in an inefficient loop).

The Average Execution Uses CPU Heavily

Much of the stats In Oracle time is spent on CPU usage, and the average In Oracle time is high.

Table 12-60 The Average Execution Uses CPU Heavily

	Description
What to do next	Run the statement with statistics_level=ALL.
Advice	Perform one of the following: ■ Examine run results in the Extended Statistics tab. Analyze problematic data (such as many logical reads, table scans and more). ■ Explain the statement and use the SQL tab tools to improve performance.

Heavy Index Overhead

Most I/O wait on indexes is due to fetching index pages from the disk, reflecting changes made by the DML statement. The indexes do not appear in the execution plan.

Table 12-61 Heavy Index Overhead

	Description
What to do next	Perform one of the following options: ■ In the Objects tab, observe the I/O breakdown between the different indexes. Focus on the heaviest index and launch to the Tune Object tab. ■ Check if this index is being used in other access types other than Index Overhead. If the only access type is Index Overhead, this may indicate that the index is not being used to filter table rows. ■ In the Statements tab, examine the different statements using the index, to learn about index usage patterns. ■ In the Read/Write Operations tab, in the In Oracle graph, try to identify index update patterns (such as, day or night).
Advice	Perform one of the following options: ■ When inserts are part of a load, batch, or night activity, consider dropping the index before performing the activity, and recreating it afterwards. ■ If the index is not used in execution plans, consider dropping the index or unused columns from the index to reduce index overhead. If the index is used in execution plans, launch to the What-If tab to see which statements may be effected by this change. **Note:** The Statements tab shows the activity and execution plans that were detected during the selected time frame and may not reflect the activity of all the statements and execution plans executed during this time frame. Proceed with caution when determining whether to drop an index or delete a column from an index.
Example	Table: INSERTED_TABLE (C1 number,C2 date, C3 varchar2(128), C4 number) Indexes on table: IX1 (C1,C2) IX2(C4,C3) Statement: Insert into INSERTED_TABLE values (:h1,:h2,:h3,:h4) In this example, Oracle fetches the relevant index blocks of the two indexes, for the new rows, even though the indexes do not appear in the execution plan. The I/O wait accumulated while fetching these index blocks is considered to be an index update.

Preferable Plan Detected by Oracle

This finding can only appear after running the "Get Best Plan" command in the Bind Variables tab.

It indicates that some bind values may lead Oracle to choose a different execution plan than others. This does not mean that when those bind values are used for the statement that their relevant plan will be used. This depends on

which version of Oracle is being used, whether the statement's plan exists in memory, and whether the init.ora parameter "_optim_peek_user_binds" is set to TRUE or FALSE.

Table 12-62   Preferable Plan Detected by Oracle

	Description
What to do next	Perform one of the following options: ■ Check the PLAN_HASH_VALUE field that appears in the left pane in the Bind Variables tab and examine the difference between plans of different bind sets. ■ Consider running the statement using a different bind set leading to a different plan, and compare their run time.
Advice	If different execution plans result in a fluctuation in run time consider the following: ■ Fixing an optimal plan using outlines or profiles. ■ Disabling bind peeking by setting the "_optim_peek_user_binds" parameter to FALSE ■ Consider using literals instead of bind variables. **Note:** Setting "_optim_peek_user_binds" to FALSE will influence all statements running in the selected instance.

Low End of Fetch Count

The query activator doesn't fully read the query result set.

Consider speeding up the first set of rows fetch time. This is usually done by using the "FIRST_ROWS" hint.

Verify that the application is not performing a huge data scan for no reason. Unnecessarily large scans are expensive (in terms of CPU and I/O time).

The end of fetch count is identified by the number of times the specified cursor was fully executed since the cursor was brought into the library cache. Its value is not incremented when the cursor is partially executed, either because it failed during execution or because only the first few rows produced by this cursor were fetched before the cursor was closed or re-executed.

Table 12-63   Low End of Fetch Count

	Description
What to do next	■ Examine the statement's execution plan and verify that unnecessary scans are not being performed. ■ Observe which program is calling the statement in the Activity Tab, and check your Application to verify that there is a true need to scan all data and fetch only part of it.
Advice	■ Consider tuning the statement so that it scans fewer rows, by adding a "FIRST_ROWS" hint. Compare the result execution plan to the original, and run both statements to evaluate the influence of the change. ■ Consider changing the application's logic to improve filtering. This will lead to smaller scans.

Major Difference Between Plans

This finding may appear when the difference between the best and worse plans, within a selected time frame, is significant in terms of in Oracle time. This indicates that some plans were used, and the best plan consumed significantly less resources of Oracle than the worse plan.

Table 12-64   Significant Differentiation between Best and Worse Plans

	Description
What to do next	Perform one of the following options: ■ Compare the execution plan of the best plan with the execution plan of the worse plan (also includes a comparison of costs). In the Bind variables tab, check if the difference can be attributed to using a different bind set. If not, check if the significant difference can be derived from a change of one of the objects. ■ If you find that the difference between the best and worse plans is derived from a different bind set, consider running the statement using a different bind set and compare their run times.
Advice	If the significant difference between the best and worse plans was derived from using a different bind set consider: ■ Fixing an optimal plan using outlines or profiles. ■ Disabling bind peeking by setting the "_optim_peek_user_binds" parameter to FALSE. ■ Using literals instead of bind variables. If the significant difference between the best and worse plans were derived from a change of one of the objects within the plan: ■ Explore the objects that are being accessed inefficiently in the worst plan, in the Objects tab. Check if the table grew significantly or if one of the indexes was dropped and can be rebuilt. ■ Consider using outlines or profiles for checking improvements to the worst plan. **Note:** Notice that setting the "_optim_peek_user_binds" parameter to FALSE will influence all statements running in this instance.

Newer Execution Plan Exists

A newer execution plan than the one you are viewing was collected for the statement during the selected time frame, or during a later time frame.

Table 12-65   Newer Execution Plan Exists

	Description
What to do next	Perform one of the following options: ■ Expand the time frame you are viewing, using the time frame list box, to identify the new execution plan. Expand the time frame until it includes the current time so that the newer execution plan is loaded. ■ Examine the different execution plans that were captured, and their real resource consumption.
Advice	Try to identify the source of the different execution plans. To resolve the multiple plans, consider the following solutions: ■ In the All Changes section view, observe whether any changes took place (such as schema or statistics changes). ■ In the Activity tab, observe whether the execution plan is being run by different programs.

Cartesian Join Used

The use of a merge join Cartesian is very expensive for Oracle. Cartesian joins can be caused by a missing Table Join condition to the WHERE clause.

A Cartesian Join is used when one or more of the tables does not have any join conditions to any other tables in the statement. The optimizer joins every row from one data source with every row from the other data source, creating the Cartesian product of the two sets.

Table 12-66   Cartesian Join Used

	Description
What to do next	Examine the predicates in the statement's WHERE clause.
Advice	To avoid a Cartesian Join verify that you have provided the proper join conditions in the statement's WHERE clause.

CPU Used for Sorts

I/O found on temporary tablespace may indicate sort operations are consuming CPU time of the statement.

Table 12-67   CPU Used for Sorts

	Description
What to do next	In the Activity tab, examine temporary tablespace overtime I/O consumption for the statement and for the programs activating the statement.
Advice	To reduce the I/O consumption for the sort operation, consider the following solutions: ■ Change the SORT_AREA_SIZE to a higher value. If you are using PGA settings, you can change the PGA_Aggregated_Target, so as to avoid sort and hash area size problems. You can either change the values for a specific session using the Alter Session command, or for the entire instance. ■ Try to solve the problem at the statement level by one of the following options: Add an index to prevent a sort. If your statement has an Order by clause that has columns for a single table, check whether you can add an index. In some cases, you may get an index recommendation that prevents a sort. If you are considering adding an index, you must check the effect of that index. To analyze that effect, launch to the What-If tab. Identify the heavy sort or hash consumer step. Run a statement with statistics_level=all. Click the Run & Compare tab. Examine the LAST_TEMPSEG_SIZE and MAX_TEMPSEG_SIZE in the extended

CPU Used for Index Scattered Read

Statement 's I/O may indicate Index Scattered read operation (often full scan) on the index specified in the Object column.

Table 12-68   CPU Used for Index Scattered Read

	Description
What to do next	Perform one of the following options: ■ Click the Locate icon to find the relevant step in the execution plan. ■ Select the findings type to investigate the objects used and their structure in the Objects tab. ■ Select the Activity tab, locate the statement associated with objects, and drill to the index consumption for the statement in the Activity tab.

Table 12-68   CPU Used for Index Scattered Read

	Description
Advice	To reduce the CPU consumption, consider the following solutions: ■　If the index is not used to prevent a sort operation, consider adding the INDEX_FFS hint and change DB_FILE_MULTIBLOCK_READ_COUNT parameter to a higher value. ■　Switch to another index or add a new index. ■　Use a parallel query option. ■　Move the index to another tablespace with a higher block size. Findings refer to the whole statement - not to a specific execution plan. If a step does not exist in the selected execution plan (unless this is due to an index overhead), switch to another plan and locate the relevant step.

CPU Used for Index Sequential Read

Statement 's I/O may indicate Index Sequential read operation (often range scan) on the index specified in the Object column. If the statement is DML and the index is not used in the execution plans, then the I/O represents the index maintenance overhead, caused by fetching the index blocks for update to memory.

Table 12-69   CPU Used for Index Sequential Read

	Description
What to do next	Perform one of the following options: ■　Click the Locate icon to see if the index is used in the execution plan. ■　Select the findings type to investigate the objects used and their structure in the Objects tab. ■　Select the Activity tab, locate the statement associated with objects, and drill to the index overtime consumption for the statement in the Activity tab.
Advice	If this is the result of a range scan, consider one of the following solutions: ■　Switch to another index, add a new index, or change the index structure. ■　Add columns to the index to enable index only access. ■　Switch to a full table scan. ■　Partition the table or the index. ■　Create a cluster with just one table in it and a cluster key used as the index key. If this is due to an index overhead, consider reducing the index overhead by deleting unused indices, or unused columns in used indices. Findings refer to the whole statement - not to a specific execution plan. If a step does not exist in the selected execution plan (unless this is the result of an index overhead), switch to another plan and locate the relevant step.

CPU Used for Table Scattered Read

Statement 's I/O may indicate table scattered read operation (often full scan) on the table specified in the Object column.

Table 12-70   CPU Used for Table Scattered Read

	Description
What to do next	Perform one of the following options: ■　Click the Locate icon to find the relevant step in the execution plan. ■　Select the findings type to investigate the objects used and their structure in the Objects tab. ■　Select the Activity tab, locate the statement associated with objects, and drill to the table consumption for the statement in the Activity tab.

Table 12-70   CPU Used for Table Scattered Read

	Description
Advice	To reduce the CPU consumption for the table, consider the following solutions:
	■ Create an index matching the statement's predicates.
	■ Change the DB_FILE_MULTIBLOCK_READ_COUNT parameter in INIT.ORA to a higher value.
	■ Partition the table according to the best predicates existing in the statement.
	■ Use a parallel query option.
	■ Move the table to another tablespace with a higher block size.
	Findings refer to the whole statement - not to a specific execution plan. If a step does not exist in the selected execution plan (unless this is the result of an index overhead), switch to another plan and locate the relevant step.

CPU Used for Table Sequential Read

Statement 's I/O may indicate a Table Sequential read operation (often ROWID access) on the table specified in the Object column.

Table 12-71   CPU Used for Table Sequential Read

	Description
What to do next	Perform one of the following options:
	■ Click the Locate icon to find the relevant step in the execution plan.
	■ Select the findings type to investigate the objects used and their structure in the Objects tab.
	■ Select the Activity tab, locate the statement associated with objects, and drill to the table consumption for the statement in the Activity tab.
Advice	To reduce the CPU consumption for the table, consider the following solutions:
	■ Change the index structure by adding a better filtering column, or change the column sequence to improve the matching and screening performed in the index tree.
	■ Add columns to the index to enable index only access.
	■ Switch to a full table scan.
	■ Partition the table or the index.
	■ Create a cluster with just one table in it and a cluster key used as the index key.
	Findings refer to the whole statement - not to a specific execution plan. If a step does not exist in the selected execution plan (unless this is the result of an index overhead), switch to another plan and locate the relevant step.

Heavy Table Full Scan

Statement resources are spent performing Full table scans on the table specified in the Object column.

Table 12-72   Heavy Table Full Scan

	Description
What to do next	Perform one of the following options:
	■ Click the **Locate** icon to focus on the relevant step in the execution plan.
	■ Get Index Recommendations for the statement.
	■ Select the finding type to investigate the objects used and their structure in the Objects tab.
	■ Select the finding type to investigate the step statistics, step resource consumption and step vs. plan over time.

Table 12-72   Heavy Table Full Scan

	Description
Advice	To reduce the Full Scan resource consumption consider: ■ Creating an index matching the statement's predicates. ■ If full scan is using I/O heavily consider changing the DB_FILE_MULTIBLOCK_READ_COUNT init.ora parameter to a higher value. ■ Partition the table according to the best predicates existing in the statement. ■ Use a parallel query option. ■ Move the table to another tablespace with a higher block size.

Heavy Step

Statement resources are spent performing the specified step.

Table 12-73   Heavy step

	Description
What to do next	Perform one of the following options: ■ Click the Locate icon to focus on the relevant step in the execution plan. ■ Select the finding type to investigate step statistics and resource consumption in the Statistics tab.

Heavy Sort

Statement resources are spent performing sort operation.

Table 12-74   Heavy Sort

	Description
What to do next	Perform one of the following options: ■ Click the **Locate** icon to focus on the relevant step in the execution plan. ■ Select the finding type to investigate step statistics and resource consumption in the Statistics tab.
Advice	Try to solve the problem at the statement level by one of the following options: ■ Add an index to prevent a sort. ■ If your statement has an Order by clause that has columns for a single table, check whether you can add an index. In some cases, you may get an index recommendation that prevents a sort. If you are considering adding an index, you must check the effect of that index. To analyze that effect, launch to the What-If tab. ■ If the sort involves I/O on the temporary tablespace, you can reduce the I/O consumption for the sort operation by changing the SORT_AREA_SIZE to a higher value. If you are using PGA settings, you can change the PGA_Aggregated_Target, so as to avoid sort area size problems. You can either change the values for a specific session using the Alter Session command, or for the entire instance.

Heavy Hash

Statement resources are spent performing hash operation.

Table 12-75   Heavy Hash

	Description
What to do next	Perform one of the following options: ■ Click the **Locate** icon to focus on the relevant step in the execution plan. ■ Select the finding type to investigate step statistics and resource consumption in the Statistics tab.

Table 12-75   Heavy Hash

	Description
Advice	Try to solve the problem at the statement level by one of the following options: ■ If the hash operation involves I/O against temporary tablespace, you can reduce the I/O consumption for the hash operation by changing the HASH_AREA_SIZE to a higher value. If you are using PGA settings, you can change the PGA_Aggregated_Target, so as to avoid hash area size problems. You can either change the values for a specific session using the Alter Session command, or for the entire instance.

Heavy Merge

Statement resources are spent performing merge operation.

Table 12-76   Heavy Merge

	Description
What to do next	Perform one of the following options: ■ Click the **Locate** icon to focus on the relevant step in the execution plan. ■ Select the finding type to investigate step statistics and resource consumption in the Statistics tab.
Advice	Try to solve the problem at the statement level by one of the following options: ■ If the hash operation involves I/O against temporary tablespace, you can reduce the I/O consumption for the hash operation by changing the SORT_AREA_SIZE to a higher value. If you are using PGA settings, you can change the PGA_Aggregated_Target, so as to avoid sort area size problems. You can either change the values for a specific session using the Alter Session command, or for the entire instance.

Heavy Index Full Scan

Statement resources are spent performing Full index scans on the index specified in the Object column.

Table 12-77   Heavy Index Full Scan

	Description
What to do next	Perform one of the following options: ■ Click the Locate icon to focus on the relevant step in the execution plan. ■ Select the finding type to investigate the objects used and their structure in the Objects tab. ■ Select the finding type to investigate step statistics and resource consumption in the Statistics tab.
Advice	To reduce the Full Scan resource consumption consider: ■ Creating an index matching the statement's predicates. ■ If the full scan is using I/O heavily and the step name is a fast full scan, consider changing the DB_FILE_MULTIBLOCK_READ_COUNT init.ora parameter to a higher value. ■ Partition the table according to the best predicates existing in the statement. ■ Use a parallel query option. ■ Move the table to another tablespace with a higher block size.

Heavy Index Range Scan

Statement resources are spent performing Range index scans on the index specified in the Object column.

Table 12-78   Heavy Index Range Scan

	Description
What to do next	Perform one of the following options: ■ Click the **Locate** icon to focus on the relevant step in the execution plan. ■ Select the finding type to investigate the objects used and their structure in the Objects tab. ■ Select the finding type to investigate step statistics and resource consumption in the Statistics tab.
Advice	To reduce the Range Scan resource consumption consider: ■ Checking the matching level of the Index to discover if there is a mismatch between existing Where predicates and operators with column order in the Index. ■ Changing column order in the Index for better matching level or adding a new index. Check the global effect of this change by launching to WhatIF tab.

Heavy Index Skip Scan

Statement resources are spent performing Range index skip scans on the index specified in the Object column.

Table 12-79   Heavy Index Skip Scan

	Description
What to do next	Perform one of the following options: ■ Click the **Locate** icon to focus on the relevant step in the execution plan. ■ Select the finding type to investigate the objects used and their structure in the Objects tab. ■ Select the finding type to investigate step statistics and resource consumption in the Statistics tab.
Advice	To reduce the Range Skip Scan resource consumption consider: ■ Changing column order in the Index for better matching level or adding a new index. Check the global effect of this change by launching to WhatIF tab.

Heavy Cartesian Join

Statement resources are spent performing Cartesian join.

Table 12-80   Heavy Cartesian Join

	Description
What to do next	Perform one of the following options: ■ Click the **Locate** icon to focus on the relevant step in the execution plan. ■ Select the finding type to investigate step statistics and resource consumption in the Statistics tab.
Advice	To avoid a Cartesian Join verify that you have provided the proper Join conditions in the statement's WHERE clause.

About Object Findings

Several Object tab findings exist to help the user. The Objects tab includes the following findings:

■ Heavy Index Overhead

■ Extensive Full Table Scan Access

■ Full Scan Reading Deleted Blocks

- Index Clustering Factor Very High

- Buffer Wait Contention

- Object or Row Lock Contention

- Bottleneck in RAC Wait

- Many Chained Rows

- Statistics Not Updated on Object

- Changes Detected in Object Structure

- Table Grew Considerably

- Partition Is Accessed Extensively

- Segment Hit Ratio Very Low

- Extensive Activity on Non-explained Statements

- Extensive Index Range Scan Access

- Extensive Full Index Scan Access

- Extensive Fast Full Index Scan Access

- Extensive Index Skip Scan Access

Heavy Index Overhead

Most of the I/O wait on indexes is due to the fetching of index pages from disk that reflect changes made by INSERT, DELETE, UPDATE, or MERGE statements. The index does not appear in the execution plan.

Table 12-81   Heavy Index Overhead

	Description
What to do next	Perform one of the following options: ■ Examine the DML statements causing index updates in the Statements tab (see example below). Focus on Index Update access types in the Access Types table. ■ Check if the index is being used in execution plans, in the Access Types table, in the Statements tab. If the only access type is Index Update, this may indicate that the index is not being used. ■ Try to identify index update patterns (such as, day or night) in the In Oracle graph, in the Read/Write Operations tab.
Advice	Perform one of the following options: ■ When inserts are part of load, batch, or night activity, consider dropping the index before performing the activity, and recreating it afterwards. ■ If the index is not used in execution plans consider dropping the index or unused columns from the index to reduce index overhead. If the index is used in execution plans, launch to the What-If tab to see which statements may be effected by this change. The Statements tab shows the activity and execution plans that were detected during the selected time frame and may not reflect the activity of all the statements and execution plans executed during this time frame. Proceed with caution when determining whether to drop an index or delete a column from an index.
Example	Table: INSERTED_TABLE (C1 number,C2 date, C3 varchar2(128), C4 number) Indexes on table: IX1 (C1,C2) IX2(C4,C3) Statement: Insert into INSERTED_TABLE values (:h1,:h2,:h3,:h4) In this example, Oracle fetches the relevant index blocks of the two indexes, for the new rows, even though the indexes do not appear in the execution plan. The I/O wait accumulated while fetching these index blocks is considered to be an index update.

Extensive Full Table Scan Access

Table extensively accessed using full table scans.

Table 12-82   Extensive Full Table Scan Access

	Description
What to do next	Perform the following options: ■ Examine associated statements in the Statements tab. Focus on Full Table Scan access types in the Access Types table. ■ Examine column usage for each statement in the Columns table. ■ Get Index Recommendations for the object's statements.
Advice	Perform one of the following options: ■ Eliminate Full Table Scan by: Trying to identify common high selectivity columns for the top statements. Create an index matching the statements predicates. Partitioning the table according to the best predicates existing in the statements. ■ Improve Full Table Scan by: Moving the table to another tablespace with a higher block size. Increasing the value of the DB_FILE_MULTIBLOCK_READ_COUNT init.ora parameter.

Full Scan Reading Deleted Blocks

Table containing many deleted blocks is extensively accessed using full table scans.

Table 12-83   Full Scan Reading Deleted Blocks

	Description
What to do next	Perform the following options: ■ Examine associated statements in the Statements tab. Focus on Full Table Scan access types in the Access Types table. ■ Examine column usage for each statement in the Columns table.
Advice	Perform one or more of the following options: ■ Eliminate full table scan by: Creating an index matching the statements predicates. Partitioning the table according to the best predicates existing in the statements. ■ Treat deleted blocks: Consider exporting the table data, truncating the table, and then reloading the data. If the dirty blocks problem is repeated, the reorganization solution may be less relevant and the user can use an index using range scan or even full index scan to reduce access to the table. ■ Improve full table scan by: Moving the table to another tablespace with a higher block size. Increasing the value of the DB_FILE_MULTIBLOCK_READ_COUNT init.ora parameter.

Index Clustering Factor Very High

Intensive I/O wait activity on table due to a range scan carried out by an index with a bad clustering factor (mismatch between physical order of rows in table and order of ROWIDs from the index range scan leads to re-reading of table blocks).

Table 12-84   Index Clustering Factor Very High

	Description
What to do next	Perform the following options: ■ Examine associated statements in the Statements tab. Focus on Index Range Scan access types in the Access Types table. ■ Examine column usage for each statement in the Columns table.
Advice	Perform one of the following options: ■ Enhance filtering of the table data blocks by adding columns to the index or ensure that index only accesses top statements. ■ If no primary key exists and the index is the most used or essential index for the table, consider sorting the table data according to the index key.

The following example shows the effect that a bad clustered index can have on performance when an index is scanned:

The figure below shows an example of an index with a good clustering factor. In this example, the root is read first, followed by the first leaf page. Then the first data block that serves the first three keys matching the three rows in the data block is fetched. In this way the keys and data blocks that follow are read. The I/O operations required by this scan include five index blocks and six data blocks, which is the equivalence of 11 I/O operations.

Figure 12-1   Index with good clustering factor (low = number of table blocks)
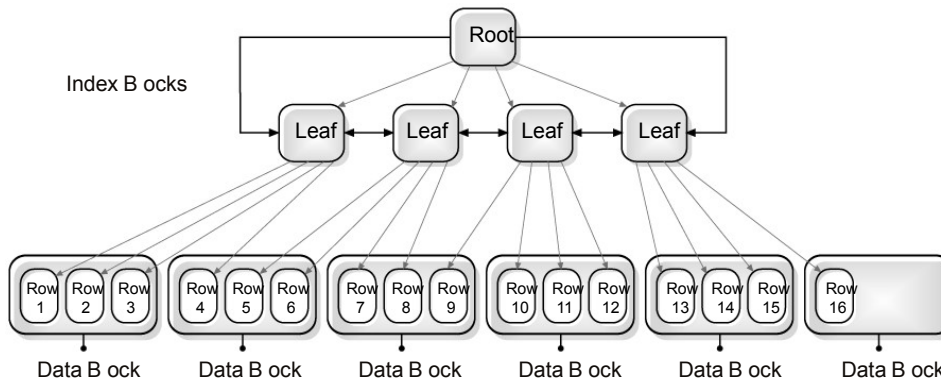
The following figure shows an example of an index with a bad clustering factor.

The index with the bad clustering factor starts in the same way as the index with the good clustering factor. However, when the second key in the index is read, the row for the second key in the first data block has not yet been fetched, so another block must be fetched. By the time Oracle accesses the index key matching the second row in the first table block, it has already been swapped out of memory and needs to be re-read. In the worse case scenario, I/O for the table blocks will be required for every index key. The I/O operations required by this scan include five index blocks and 16 table blocks, which is equivalence of 21 I/O operations. When the difference between the number of blocks and number of rows is great, performance can be greatly impacted.

Figure 12-2   Index with bad clustering factor (high = number of rows)
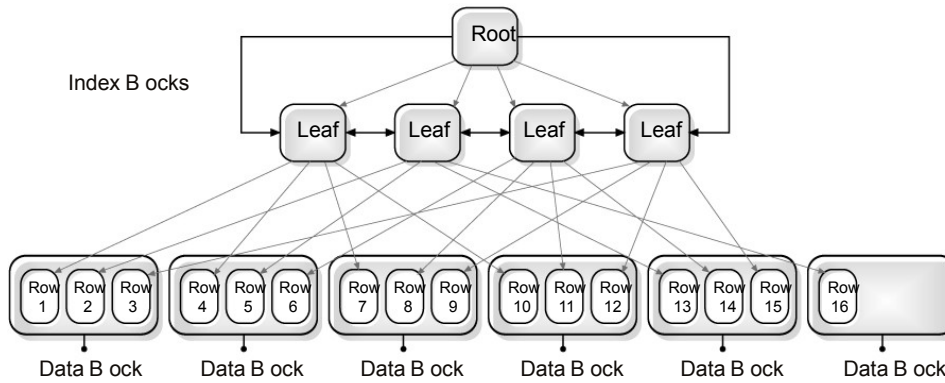
Buffer Wait Contention

Object (table and indexes) spent much of its In Oracle time on Buffer wait. This usually occurs as a result of one of the following:

Contention on a table or index buffer in Insert statements (Buffer Busy wait).

Lack of free buffer space when trying to load blocks from a disk (Free Buffer wait).

Table 12-85   Buffer Wait Contention

	Description
What to do next	Perform the following options: ■ Examine buffer wait over time, in the Activity Tab. ■ Examine buffer wait substate events in the Statistics tab.
Advice	Perform one of the following options: If Buffer Busy wait is the more dominant Oracle event, consider doing the following: ■ Increase the free lists for the table to reduce the chances of contention. ■ Increase the PCTFREE parameter or decrease the block size for the table, to distribute data among many blocks and reduce the chances for contention. If Free Buffer wait is the more dominant Oracle event, consider doing the following: ■ Tune access to the object, to reduce the number of blocks fetched. ■ If the problem is a global instance problem, increase the number of DBWR processes or I/O slaves.

Object or Row Lock Contention

Much of the objects (table and indexes) I/O time is spent waiting for a lock on the object specified in the Object column.

Table 12-86   Object Or Row Lock Contention

	Description
What to do next	Examine the statement in the Activity tab.
Advice	To reduce the lock wait for the object, perform one of the following: ■ Check to see if the lock appears in the Current tab. If so, examine the lock chain to identify which statement is holding the lock. ■ Try to identify the locking statement in the Activity tab using smaller time frames that match the lock periods. Focus on the locked table and associated statements. The DML statements (and update queries) that are NOT waiting for locks should be the immediate suspects.

Bottleneck in RAC Wait

The object (table and indexes) spent much of its In Oracle time waiting for a RAC activity to complete on the object specified in the Object column.

Table 12-87   Bottleneck in RAC Wait

	Description
What to do next	Examine the statement in the Activity tab
Advice	The object is suffering RAC wait because several instances are using it simultaneously. To solve this problem, identify all programs currently accessing the object and try to avoid accessing it concurrently.

Many Chained Rows

Access to table deteriorated as a result of chained rows.

Table 12-88   Many Chained Rows

	Description
What to do next	Examine table dictionary information.
Advice	The object is suffering because it is accessing an object that suffers from chained rows. Chained rows are typically caused by the Insert operation. To solve the problem, perform one of the following: ■ Increase the PCTFREE parameter (can be by alter or move table). ■ Reorganize table (export/import or manually reorganize table). ■ Move table to tablespace with higher block size.

Statistics Not Updated on Object

A significant block change occurred since the last time the object was analyzed, for at least one of the objects related to table.

Table 12-89   Statistics Not Updated on Object

	Description
What to do next	Examine the dictionary details of the object in the Details section. Look at the Read/Write Operation Tab to quantify the magnitude of the block change.
Advice	To reduce potential access type problems resulting from statistics that are not up-to-date, consider analyzing the table and checking it periodically.

Changes Detected in Object Structure

Changes were made to the table or index structure. Possible changes include:

Index was added or dropped.

Partitions or subpartitions were added or dropped.

Table was altered (columns were added).

Table 12-90   Changes Detected in Object Structure

	Description
What to do next	Examine the changes in the Changes graph in the Read/Write Operations tab.

Table 12-90   Changes Detected in Object Structure

	Description
Advice	Perform one of the following options: ■ Try to determine whether the changes in object structure and changes in performance are related by comparing the In Oracle graph and the Changes graph in the Read/Write Operations tab. If it seems that performance deteriorated as a result of the change in table or index structure, consider rolling back the change. ■ Consider adjusting index structure and execution plans.

Table Grew Considerably

The table is considerably larger than it was at the start of the time frame.

Table 12-91   Table Grew Considerably

	Description
What to do next	Examine if there is a correlation between table growth and performance degradation in the Read/Write Operations tab.
Advice	Verify that full scans are not widely used for the table, that the existing indexes correlate with the table growth, and that no new indexes are required. You can also check Materialized Views usage.

Partition Is Accessed Extensively

A large percentage of In Oracle time for the object is spent accessing one partition.

Table 12-92   Partition Is Accessed Extensively

	Description
What to do next	Perform the following options: ■ Examine the In Oracle activity of the partition, in the Partitions tab. Check if the massive activity spent accessing the partition is abnormal. ■ Examine the activity of statements accessing the partition in the Activity tab.
Advice	Perform one of the following options: If the partition is a table partition: ■ Create local or global indexes for statements that access the partition. ■ Subpartition the partitions. ■ If the partitions are not balanced well, consider building the partitioned table with new partition keys. If the partition is an index partition: ■ Subpartition the partitions. ■ Add more columns to the index to improve filtering.

Segment Hit Ratio Very Low

The hit ratio, for at least one of the objects related to the table, is very low.

Table 12-93   Segment Hit Ratio Very Low

	Description
What to do next	Perform the following options: ■ Examine associated statements in the Statements tab. Focus on the All access type in the Access Types table. ■ Examine buffer cache usage in the Statistics tab. Check if there is an overall wait on the Free Buffer event.
Advice	If there are no outstanding contentions on the buffer cache consider moving the object into Keep or Recycle buffer cache pools.

Extensive Activity on Non-explained Statements

Extensive activity on statements that were not explained.

Table 12-94   Extensive Activity on Non-explained Statements

	Description
What to do next	Examine associated statements in the Statements tab. Focus on non-explained statements in the Access Types table.
Advice	Perform an explain on the non-explained statements.

Extensive Index Range Scan Access

Extensive I/O wait was experienced, as a result of range scans on the index. Although this may be normal, it can often indicate a matching level problem, indicating that the structure of the index can be improved.

Table 12-95   Extensive "Index Range Scan" Access

	Description
What to do next	Examine statements using the index in the Statements tab. Examine column usage, selectivity and matching level (see example) for the top statements, in the Columns table to assess the efficiency of the index.
Advice	If the index structure does not fit the Where predicates of the top statements consider doing one of the following: ■ Add columns to the index in the right sequence (columns with high selectivity and equal predicates should be first) to improve the matching level. This leads to a better filtering of leaf pages. ■ Change the sequence of columns in the index to the optimal sequence (columns with high selectivity and equal predicates should be first) to improve the matching level. This leads to a better filtering of leaf pages.
Example	Table: TAB1 (C1 number, C2 number, C3 number, C10 Date) Index: IX1 (C1,C2,C5) Statement: select * from TAB1 where C1=:h and C5=10; Execution plan uses IX1 in Index Range Scan In this statement the matching level of the index is 1. This means that Oracle uses only C1 to filter index leaf pages, it cant match C5=10 against the index tree because of the absence of a C2= predicate. Because C1 is not selective, many irrelevant index leaf pages can be read. Oracle will apply the C5=10 predicate on the index keys to screen irrelevant table ROWIDs. An index on C1 followed by C5 would be more efficient for the query.

Extensive Full Index Scan Access

Index is extensively accessed using full index scans. This is sometimes done to avoid sorts, when the sort order matches the leading portion of the index key, or to avoid accessing table blocks, when all the columns required by the query exist in the index key.

Table 12-96   Extensive "Full Index Scan" Access

	Description
What to do next	Perform the following: ■　Examine associated statements in the Statements tab. ■　Focus on the Full Index Scan access type in the Access Types table. ■　Examine column usage, for each statement in the Columns table.
Advice	Perform one of the following options: ■　Full index scan, can be eliminated by: Identifying common high selectivity columns for the top statements. Create an index matching the statements predicates. Partitioning the table according to the best predicates existing in the statements. ■　Full index scan, can be improved by switching to Fast Full Index Scan when the index is not used to save sorts (a fast full scan retrieves the rows according to the index key). Perform one of the following options to enable this access path: Ensure that the Fast_Full_Scan_Enabled parameter = yes. Use the "Index_ffs" hint for major statements that use the full index scan, to check if the fast full scan improves performance. When the full index scan doesn't save sort results, performance may improve considerably.
Example	Table: TAB1 (C1 number, C2 number, C3 number, C10 Date) Index: IX1 (C1) Statement: select C1 from TAB1 Order by C1; Execution plan uses IX1 in Full Index Scan In this case a full index scan is the best option. Because there are no filtering predicates, there is no need to access the table blocks and the sort operation is avoided.

Extensive Fast Full Index Scan Access

Index is extensively accessed using fast full index scans.

Table 12-97   Extensive "Fast Full Index Scan" Access

	Description
What to do next	Perform the following: ■　Examine associated statements in the Statements tab. ■　Focus on the Fast Full Index Scan access type in the Access Types table. ■　Examine column usage for each statement in the Columns table.
Advice	Perform one of the following options: ■　Fast Full index scan can be eliminated by: Identifying common high selectivity columns for the top statements. Create an index matching the statements predicates. Partitioning the table according to the best predicates existing in the statements. ■　Fast Full Index scan can be improved by: Moving the index to another space with a higher block size. Increasing the DB_FILE_MULTIBLOCK_READ_COUNT init.ora parameter.

Extensive Index Skip Scan Access

Index extensively accessed using index skip scans which often means that the index structure does not fit the query in the best possible way, and leads Oracle to perform heavy activity against the index

Table 12-98   Extensive "Index Skip Scan" Access

	Description
What to do next	Perform the following: ■ Examine associated statements in the Statements tab. ■ Focus on the Index Skip Scan access type in the Access Types table. ■ Examine column usage for each statement in the Columns table.
Advice	Perform one of the following options: ■ Change the columns sequence within the index that use the index skip scan. Check the effect of this change in the What-If tab. ■ Identify common high selectivity columns for top statements. Create an index matching the statements predicates. ■ If you cannot create a new index, use hints, such as, "full" and "index_ffs" to determine if by using these access types you can achieve better performance results.
Example	Table: TAB1 (C1 number, C2 number, C3 number, C10 Date) Index: IX1 (C1,C2) (C1 has two distinct values Yes and No) Statement: select * from TAB1 where C2=10; Execution plan uses Index Skip Scan on IX1. In this case Oracle has to perform two range scans on the index—one with a key of (Yes,10) and another with a key of (No,10)and then unite the results. The more distinct values defined for C1, the more index scanning required. Defining a new index on (C2), or changing the column sequence in IX1 to be (C2,C1), enables a more efficient access path for the Index Range Scan.

abandonment rate	In Web, a counter that keeps track of the percentage of users that abandon the loading of the Web page before it completes downloading.
action	An operation that Alerts FocalPoint automatically performs when detecting a warning or critical status for a specific metric. According to the defined action, Alerts FocalPoint opens a message box, sends an email or SNMP trap, or executes a program.
action item (manual)	Some installation steps cannot be executed automatically by Precise Installer. The administrator is required to execute them manually. Action Items are presented (when necessary) at end of installation or product update.
AdminPoint	The central administration console of Precise that facilitates the maintenance, configuration, and management of all installed Precise components, such as monitoring the status of all Precise agents and PMDB processes, getting license information, starting and stopping the agents, getting log data on agents and events, changing PMDB settings, and installing patches. See also PMDB. See also agent.
advice	In Oracle, an algorithm that is designed to recommend on gathering statistics, creating new indexes, change existing indexes, and add or delete hints to make Oracle's Optimizer choose a better access path and make the statement perform better. Can be activated from any DML (Data Manipulation Language) statement.
agent	A program that runs on a server machine to collect, process, or communicate performance information. The Precise installation consists of multiple agents.
alert	The state of an Alerts metric that has reached a near-critical or critical status. An alert is issued by Alerts, triggering an action and informing of a problem that has occurred or is likely to occur within the area sampled by the specific metric. See also action.
Alerts	These product provide alerts to problematic conditions before they turn into performance problems, based on predefined metrics and thresholds. Alerts can automatically perform an action, such as displaying a pop-up message, sending an email message or SNMP trap, or running a program.
Alerts FocalPoint	An agent that receives data from the InformPoint agents, stores it, and performs any action that has been user-defined for that specific alert, such as displaying a pop-up message, sending an email message or SNMP trap, or running a program. See also InformPoint.
alert type	In Alerts, the status of all metrics belonging to a metric group or a monitored instance, indicating the current performance level through colors. See also alert. See also metric.
application server metric	In J2EE, a metrics that is provided by the application server or by customer code. This can include metrics published by the Java Management Extension (JMX) APIs or vendor-specific APIs, such as IBM's Performance Management Interface (PMI).
AppTier	The abbreviation for an application tier in a Precise environment. An AppTier contains one or more instances of the same technology and purpose. Application tiers do not necessarily correspond to distinct physical servers: in many cases, the tiers are logical, with one server running multiple AppTiers or one AppTier spans a cluster of servers. A Precise environment may contain multiple AppTiers on the same technology. For example, you may group J2EE instances (JVMs) into a J2EE Presentation AppTier and a J2EE Business Logic AppTier. Segmenting application service time into the contribution of individual application tiers is helpful in identifying the source of performance problems. Analyzing the performance and behavior of each tier separately is crucial for isolating the root causes of performance problems.
cabinet	In Oracle, SQL Server, DB2, and Sybase, the highest logical level in the SQL workspace hierarchy. A cabinet contains folders and, within folders, statements.

call path	A subset of a stack trace including only those methods that have been instrumented.
client cache	In Web, a counter that keeps track of the percentage of requests taken from the client cache (http status 304).
client-side collection	See Web client browser-side.
cluster	A group of servers or instances that are configured to be treated as a single entity. In Oracle, all Oracle instances of the same Oracle database (RAC configuration) are treated as a cluster and presented in the Precise for Oracle UI as a Database. In J2EE, a group of JVMs that serve the same application can be defined as Cluster (even if they are not defined as a cluster in the application server configuration). All JVMs of the same cluster share the same configuration.
collapsed access plan	In SQL Server and Sybase, the access plan of a unique group of statements or batches that belong to the same collapsed statement or batch but have different access plans. This can differ due to the constants in the text of the original (not collapsed) statements.
	See also explained statement.
	See also collapsed statement.
collapsed statement	In Oracle, SQL Server, Sybase, and DB2, a statement whose constants are replaced with parameters. Each collapsed statement can have several access plans, according to the occurrences of its statements.
	See also collapsed access plan.
Collector	The program that runs on the monitored server to collect performance information.
	Some technologies allow a single Collector (agent) to serve multiple instances running on the same server. Other technologies require a dedicated Collector per monitored instance.
	In most cases, it must be installed on the monitored server. Collectors for SAP, SQL Server, and Sybase may reside on a remote server.
	See also agent.
collector slice size	In J2EE, the smallest unit of data presentation and storage. It can be configured as 30 seconds or five minutes. If SmartLink is configured, relevant Web and Microsoft .NET Collectors will report to SmartLink using the same collector slice size. Regardless of the defined collector slice size, the most granular slice size presented on both the Precise for Web UI and the Precise for Microsoft .NET UI will be 15 minute intervals.
completion rate	In Web, a counter that keeps track of the percentage of users that completed the loading of a Web page.
count	The number of occurrences observed during a measurement interval.
CPU time	The average amount of time consumed by the operating system actively processing instructions on behalf of a running activity.
critical status	In Alerts, the status represented by a red bullet indicating that the value of the sampled metric has exceeded the near-critical and critical threshold values.
	See also metric.
Cross-AppTiers	In Insight, Report Manager, and Alerts, a perspective that provides high-level information about the performance of all AppTiers in an environment, including operating system data. It provides a holistic view of the entire environment and helps understand how the AppTiers interact.
	See also AppTier.
current data	In SmartLink, the end-to-end activity, detected by Precise over the last five minutes, including activities that are still in progress. In all databases products, a workspace that presents Current Sessions and the SQL statements that were executed including statements that are still in progress.
custom	Represents invocations of a type not specifically defined.
	See also work time.
customized metric	In Alerts, a user-defined metric that measures site-specific parameters.
drilldown	Within Precise products and Insight SmartLink, the filtering of analyzed data by clicking a specific entity. Then, additional information about the selected entity is presented, plus a breakdown of its activity by another entity. For example, drilldown on a transaction and get an overtime graph of the selected transaction, plus a list of top users that executed the selected transaction.
entry point	Usually a top-level HTTP or EJB request. An entry point can originate when a user clicks in a browser or an E-commerce server executes a remote EJB. J2EE makes a distinction between service request invocations and other invocations. The first HTTP or EJB invocation within a call path is designated as an entry point.

environment	The highest Precise logical group. It may contain multiple AppTiers of various technologies that serve an application together. For example, a Payroll Production environment may contain all Web servers, application servers, transaction managers, databases, and servers that serve this application. Alternatively, it may contain any set of instances that form together an administrative group. Since Precise version 8.5, an instance can belong to more than one environment. Let's say a single Oracle database serves two different applications plus the DBA wants to associate this database with a group of other databases under his responsibility regardless of served applications. In this case the Oracle database will be associated to three different Precise environments. See also AppTier. See also instance.
ERP Extension	In Oracle, DB2, and SQL Server, an extension to the Collector that provides detailed information on the activities and resource consumption of packaged application components. It correlates the database information, and the packaged application information and lets you see users, transactions, reports, and other elements of ERP applications, such as Oracle Applications, SAP, PeopleSoft, and Siebel.
event	In Alerts, the occurrence of a sampling or progress. A sampling occurrence occurs every time a metric samples. A progress occurrence occurs when a metric's progress status is changed or when the investigated status reaches the end of the given investigation time. In AdminPoint, all occurrences reported by Precise agents, including informational events, warnings, and errors related to one of the agents. All events are shown in the Events view. See also metric. See also progress.
executions	In Oracle, SQL Server, DB2, and Sybase, the number of times a SQL statement was executed during the selected time frame. In SmartLink and SAP, the number of times a transaction was executed during the selected time frame.
explained statement	In Oracle, SQL Server, DB2, and Sybase, a statement whose access path (chosen by the RDBMS Optimizer) is clarified and translated into a visual display. Explained results include information on the objects referenced by the statement and the operations performed on these objects.
extended collection	In Oracle, a function that proactively specifies a future period during which Oracle activity data is collected and organized for subsequent analysis. Extended collections are an easy means to view collected information, assess application resource consumption, and identify bottlenecks that are inhibiting application performance and end-user productivity.
Federated Precise	Federated Precise is a version of Precise (starting with version 8.5) that can manage multiple Precise installations within unified StartPoint and AdminPoint screens, displaying and managing all environments, instances, and installations.
findings	A ranked list of top performance problems in the selected context. They are presented in Oracle, SQL Server, .NET, and J2EE. Each finding provides: a short explanation of the problem, detailed background information, and most important: one or more links to further investigate the problem and possibly get more granular findings on the selected context.
first byte time	In Web, a counter that keeps track of the time that it takes from the moment a new Web page is called until the first byte arrives back from the Web server.
FocalPoint	An agent that communicates with the Listeners installed on the monitored servers, receives data from the Collectors, periodically processes and stores this data in the PMDB, and serves UI requests.
folder	In SQL Server, Oracle, DB2, and Sybase, the intermediate logical level in the SQL workspace hierarchy. Folders are grouped into cabinets and contain SQL statements. See also cabinet.
Framework Installer	The application that facilitates the installation of Precise framework components. It can be invoked from the installation DVD to install a new Precise deployment. It can also be invoked from an existing Precise deployment to install an additional framework node and attach it to the originating Precise deployment. See also Framework node.
Framework node	A set of FocalPoint agents that are installed together and manages a set of monitored instances. The performance data of these instances will be loaded into a dedicated PMDB. A single Precise deployment may contain multiple framework nodes (using a separate PMDB for each node). An environment cannot span over multiple framework nodes.

garbage collection	An automatic process in the Java runtime environment that periodically reclaims memory used by objects that are no longer referenced. The process can impact an application's performance while memory is being reclaimed. Java programmers may initiate garbage collection explicitly.
grouper	In Web, the identifier that is used to group other identifiers, such as sites or URLs.
hint	In Oracle, an instruction directed at the Oracle Optimizer that includes considerations for an execution plan. The Oracle Optimizer will build an execution plan based on the hint, ignoring its own set of considerations.
hour group	A unit that reflects the type and level of activity within the system at different times. By defining the times of the day that are peak and off-peak, or day and night, the performance analysis can be focused on those particular times of the day. If, for example, most performance problems occur within nighttime and weekend batches, it can be useful to focus only on them.
InformPoint	In Alerts, an agent that retrieves performance data from all installed Precise products, analyzes it, and sends an alert to the Alerts FocalPoint when the predefined thresholds are exceeded. See also agent.
Insight	The Precise product family that facilitates the process of monitoring and correlating system performance. It consists of Insight. See also Report Manager. See also FocalPoint.
Insight FocalPoint	An agent that receives performance information from Insight Savvies, which monitor the environment. Insight FocalPoint then correlates, processes, and stores this information in a centralized location. The Insight performance history is stored in the PMDB. See also Savvy. See also PMDB.
instance	A monitored object of a specific technology. The following list specifies what constitutes an instance for the various supported technologies: J2EE - a Java Virtual Machine (JVM - a logical name set by the user), Microsoft .NET - a Common Language Runtime (CLR - a logical name set by the user), Oracle Applications - an Oracle Applications Form server, Oracle - an Oracle instance, SAP - a SAP system, SQL Server - an SQL instance, Sybase - a Sybase instance, Tuxedo - a Tuxedo domain, Web - a Web server, WebSphere MQ - an IBM WebSphere Queue Manager, DB2 - a DB2 non-partitioned database or a DB2 database partition. During installation, the instance is associated with one AppTier and environment. An Instance can be moved to a different environment or associated with multiple environments without re-installation nor losing historical data. See also environment.
instance statistics	See performance counter.
instance/database changes	In SQL Server and Oracle, the part of the Collect Schema Changes process that captures instance definition changes and database option changes and saves them in the PMDB.
instrumentation	The process of inserting fault-tolerant recording hooks in Java byte code, .NET MSIL, HTML pages, or other monitored components, resulting in the capture of performance metrics. In J2EE, a mechanism that enables collecting performance information when an application is executed. The process involves inserting special fault-tolerant recording hooks into application class objects. In WEB, the insertion of recording hooks into HTML pages can either be in memory (Dynamic Instrumentation) or file-based (Static Instrumentation).
instrumentation context	See invocation context.
internal invocation	The process of invoking a request from an HTTP request (Servlet or JSP) or an EJB. J2EE displays an internal invocation like any other invocation.
invocation	An execution of a J2EE entity (a Servlet, EJB, SQL Statement, method, and so on). When mentioned in plural (Invocations), means the amount of times that the entity was executed.
invocation context	The context within which a method is invoked. For example, if Method A is invoked both from Method B and from Method C, there will be two different invocation contexts for the performance metrics collected for Method A, one for when it is invoked from Method B and one for when it is invoked from Method C. See also call path and instrumentation context.
JRE (Java Runtime Environment)	As the runtime part of the Java software, the combination of the components that enable the execution of a Java program: a Java virtual machine, the core class libraries, and the files that form the Java platform.

JSP (Java Server Page)	An HTML page with special tags for Java scripting. An application server processes the tags and generates a Servlet.
JVM (Java Virtual Machine)	An instance of a JRE that executes Java programs. A server-side Java application server is itself a Java program that runs inside a JVM. Servlets, JSPs and EJBs are Java programs (applications) that run within the application server's JVM. J2EE monitors the JVM running the application server and the server-side Java applications within the application server.
JVM Heap Memory	The amount of real computer memory that is allocated to the JVM for executing Java programs.
key metric	In Alerts, a parameter that monitors a very important performance aspect. The status of the Oracle instance (up/down), for example, is crucial for system performance. If the instance is down, this is the first problem that needs to be solved. Marking a metric as key metric ensures that a critical alert raised for this metric receives top priority by the person that is responsible to handle alerts at any time. It can also help determine which alert to handle first in case of multiple alerts. In Alerts, key metrics are always displayed at the top of a metric table when it is sorted by alerts so that they get immediate attention.
Listener	The agent that facilitates the communication between the various Precise agents across different servers must be installed on every server where Collectors or FocalPoint agents are installed. The Listener allows communication with all other agents installed on the monitored server, while only the Precise Listener port is known by other servers.
login name	In SQL Server and Sybase, the session identifier that represents the credential used to connect to the database. When an ERP extension is installed, the user name of the packaged application's client overrides the login name. For example, when SAP extension is installed, the SAP user name overrides the login name.
machine	A session identifier. A machine as sampled, for example, by the SQL Server Collector is the identification of the machine where the client process executes. Machine is also sampled by Oracle, DB2 and Sybase. In Insight terminology Machine is called Client machine.
Main framework node	A main framework node is the single point for login and also serves as the Precise FocalPoint for the entire deployment.
major collection count	Number of estimated major garbage collection events that occurred during the last J2EE collector slice. A major garbage collection can stop the application while JVM heap memory is being reclaimed. See also garbage collection.
major garbage collection time	Percentage of time spent by the JVM executing major garbage collection events during the displayed interval. See also garbage collection.
MBean	A Java object that represents a manageable resource. In J2EE, MBeans, or Managed Beans, are used for application server metrics.
memory logger interval	The interval at which the J2EE Collector gathers JVM heap memory data. All snapshots of the memory logger's data collected according to the memory logger interval are summarized with counters in the current aggregation interval. The memory logger interval's time span is typically a small fraction of the aggregation interval.
metric	In Alerts, a query that helps measure performance in the environment. Three types of metrics are available: System metrics relate to the internal resources, operations, and objects of the monitored infrastructure; application metrics reflect the way the applications perform; user-defined metrics can be customized to specifically relate to a site. When a metric's value exceeds one of the defined thresholds, its status changes to near-critical (yellow bullet) or critical (red bullet). See also key metric.
metric set	In Alerts, a unit that groups metrics that measure related performance aspects. The following metric sets exist: Status: includes metrics that alert to functional problems related to the instance. Performance: includes metrics that alert to performance problems related to the instance. Load: includes metrics that alert to instance-related load problems that may later cause errors or crashes. Service: includes metrics that alert to instance-related SLA breaches. The metrics in this set are sampled by Insight. Performance Trending: includes metrics that alert to potential future performance problems. The metrics in this set are sampled by Report Manager.

Load Trending: includes metrics that alert to potential future load problems. The metrics in this set are sampled by Report Manager.

Customized: includes user-defined metrics.

Precise Status: includes metrics that alert to the near-critical or critical status of the installed Precise environment.

module	A session identifier. In Oracle it contains the value of MODULE column in V$SESSION table. In DB2 it contains the command name of non-SQL statements or the package name for SQL statements.
near-critical status	In Alerts, a status indicating that the value of the sampled metric has exceeded the defined near-critical threshold. A near-critical status is indicated by a yellow bullet.
network time	In Web, a counter that keeps track of the time spent on network activity from the server side perspective. This includes the time to read the request from the network and the network time to send the response back to the client.
node	See Framework node.
	Note: when creating a new installation from DVD, it would be called "Framework." In the UI screens (columns in tables, choosing a system for a new environment etc), we would call them "Nodes." For example: choose a node for the new environment.
	When adding a new system within AdminPoint, it would be called a "Framework node".
packaged application	An application that is created and/or maintained by a third party and is not custom-built to one's specific needs. The following packaged applications have special treatment by Precise: SAP, Oracle Applications, PeopleSoft, and Siebel. They are harder to modify because the application code is either not available or hard to understand.
	See also ERP Extension.
page size	In Web, a counter that keeps track of the amount of data loaded from the server to display the page.
page views	In Web, a counter that keeps track of the number of Web pages viewed at a specific Web site during a selected time period.
parent metric	In Alerts, a joining of several child metrics. Each time a parent metric samples, it gathers data from a set of child metrics and presents it as a single metric query. The individual child metric values are displayed on the Thresholds tab of the Properties dialog box in Alerts.
	See also submetric.
performance counter	In SQL Server, a Windows performance counter as reported by the operating system. In Oracle, an instance level statistic as gathered from some Oracle V$ views. In Sybase, a sysmonitors counter collected by Sybase. In Web, an operating system or a Web server performance counter.
PMDB	The Precise data warehouse of performance and availability data. It can be hosted on an Oracle or SQL Server database.
portal server	An application server for Web-based applications that commonly provide personalization, single sign-on, and content aggregation from different sources and that host the presentation layer of Information Systems. Aggregation is the action of integrating content from different sources within a Web page. A portal may have sophisticated personalization features to provide customized content to users. Portal pages may have a different set of portlets creating content for different users.
portlet	A Java-based Web component, managed by a portlet container, that processes requests and generates dynamic content. Portlets are used by portals as pluggable user interface components that provide a presentation layer to Information Systems.
Precise deployment	An independent Precise system. It contains and manages various agent types and provides centralized monitoring and administration. A Precise deployment may contain multiple framework nodes. One of them is defined as the main framework node and it manages all other nodes of the Precise deployment.
Precise FocalPoint	See Precise FocalPoint agent.
Precise FocalPoint agent	An agent that manages all agents in a single Precise deployment. Additional product FocalPoint agents manage specific technologies and resources.
Precise Proxy agent	When multiple Framework Nodes are managed by a single Precise deployment, the Precise FocalPoint agent manages all of the agents of the main Framework node, while every other framework node has a Precise Proxy agent to manage all its agents.
Precise_root	This is the term used in a path for the Precise installation directory. The terms *<i3_root>* or i3 root can appear in text too.

program	A session identifier in Oracle, DB2, SQL Server, Sybase, Insight, and Alerts. A program as sampled, for example, by the SQL Server Collector, is the name of the executable that connects to the database. Applications that do not set the application name have N/A as program. When a SAP extension is installed, the SAP transaction overrides the program, and it may change during the application's lifetime. When a Siebel extension is installed, the Siebel views override the program, and it may change during the application's lifetime. In Insight, program is the name of an executable as recognized by the operating system. If an executable is invoked from a script (a batch or a shell), the script is displayed as the command entity. In Alerts, program is part of the customized metrics definition and is the name of the executable or stored procedure executed in the database that runs when the metric is sampled. In action definitions, program is the name of the executable that will run if the metric exceeds its threshold.
progress	In Alerts, the management state of a metric for which an alert has been raised. The following statuses exist:
	Open: An alert is raised.
	Investigated: The alert is taken care of.
	Closed: The problem has been solved.
Proxy FocalPoint	See Precise Proxy agent
recommended index	In SQL Server, a function that uses the Microsoft Index Tuning Wizard to recommend on adding indexes or statistics for a selected statement, batch, or table. Based on the results of this function, the Optimizer will choose a better access plan and make the respective statement or batch perform better. For statements and batches, recommendations are based on the content of the statement or batch. For tables, recommendations are based on all the statements that are stored in the PMDB, are executed during the selected time period, and have an average duration time greater than the value defined in the registry (where 0 is the default).
	See also advice.
related SQL	In Oracle, a generated statement that uses alternative syntax to access the database in different ways and returns the same output as the original statement.
relative frequency	Number of contributor invocations per entry point. For example, if a service request calls three methods each time it is invoked, the Relative Frequency for the method is three invocations per service request execution. Similarly, if a service request calls one method every other time it is invoked (half of the time), the Relative Frequency for the method is 0.5 invocations per service request execution.
rendering time	In Web, a counter that keeps track of the time that it takes for a Web page to be loaded from the moment the first byte arrives until the Web page is fully loaded or the user interrupts or abandons the loading process.
report	A collection of queries, programming code, and layout settings that Report Manager executes to generate graphical results like tables and charts.
Report Manager	Uses historical information to identify problematic conditions, track long-term performance, volume trends and patterns, view availability problems over long periods of time and on different levels, compare the performance of similar systems, correlate between performance metrics of different products, assist in capacity planning, and generate demand-driven, user-defined reports.
Report Manager FocalPoint	An agent that examines the PMDB tables to produce scheduled performance reports.
	See also PMDB.
report parameter	In Report Manager, a keyword used in a report. Its value is set during the report execution. A parameter's value can be updated either permanently or for the current execution only.
report property	In Report Manager, the attributes that define a specific report, consisting of report parameters and scheduling information.
	See also report parameter.
request	In Web, a counter that keeps track of the number of HTTP requests sent for a viewed entity. **request**
error	In Web, a counter that keeps track of the percentage of requests completed with an HTTP error.
sampling	In Alerts, the process during which a metric queries a Precise product for a specific instance, retrieves values, and calculates the metric's alert level.
sampling base	In Alerts, the start time of a scheduled sampling process (by default Sunday, 00:00 AM).
sampling period	In specific Alerts metrics, the time frame during which statistical data is returned from other Precise products. Such a metric is for example Oracle's General Behavior metric, which returns database behavior for a certain period of time.

sampling rate	In Alerts, the frequency of a metric's regular sampling schedule. The sampling rate is measured in minutes.
Savvy	An Insight agent that collects AppTier-specific performance data.
scalability	A system's ability to withstand load. For example, positive scalability means that the system continues to function properly even when it is called upon to service a larger number of users.
schema changes	In Oracle and SQL Server, a process that captures schema changes and saves them in the PMDB. In SQL Server, instance configuration parameters and database option changes are also captured.
Script Installer	A component of Precise Agent Installer that enables adding a small JavaScript script, also known as a Web browser-side agent, to the Web pages of your Web site.
server	Also called a host machine or server machine. The combination of a computer and associated operating system software that is accessed by multiple clients to perform requested functions and typically returns results to the clients.
server-side collection	In Web, the collecting of performance data from the Web server instance through the Web server agent.
service + network time	In Web, a counter that keeps track of the service and network time, including the total amount of time the request took to reach the server.
Service Level Agreement	See SLA (Service Level Agreement).
service time	In Web and SmartLink, the time elapsed from when a request is received by the server to the time a response is sent back to the computer that made that request. The service time is measured on the server side. In J2EE, the time it takes an invocation to complete execution. In other words, service time is the average length of time between the start time and end time of a Java method execution. For example, the SQL service time is the time it takes the JDBC method executing the SQL statement to be completed. The service time includes CPU and wait time. The service times are reported in the interval in which they complete execution. Though a contributor's average service time may be very small, the contributor may be called many times. As a result, a contributor's overall contribution to performance may be large even though its average service time is low.
servlet	A class that is loaded only once and for which the application server uses multithreading to process requests. The servlet generates an HTML page that is sent back to the Web browser.
size	In Web, a counter that keeps track of the bytes sent and received.
SLA (Service Level Agreement)	A formal definition of an information system's performance goals. Within Precise, an SLA consists of clauses corresponding to various system activities. Once a system's SLA is defined, its SLA compliance can be analyzed, and breaches can be isolated to identify their causes.
SmartLink	In Insight, a component that provides transactive, correlated information across AppTiers from the user's perspective. Insight SmartLink functions in PeopleSoft and Web applications environments.
SmarTune	In Oracle, SQL Server, and .NET, an algorithm that analyzes performance metrics, identifies and ranks potential problem areas, and provides advice for correcting the problems.
socket	An I/O abstraction layer that enables processes to communicate with each other, regardless of whether they are on the same server. Sockets are bi-directional FIFO pipes that look like regular file I/O to the developer with the abstraction layer handling all of the low-level communication details.
StartPoint	The opening page of Precise. It provides a quick overview of the environment status and links to launch any of the Precise products.
Statistics workspace	In Oracle, DB2, SQL Server, Sybase, J2EE, and Web this workspace presents instance level statistics such as hit ratios, utilization, JMX metrics, logging, and more.
submetric	Also called child metric. In Alerts, the subquery of a parent metric. A child metric gathers its own data and combines it with the data gathered by other child metrics to form the result of the parent metric. Each child metric has its own thresholds and may be enabled or disabled individually. See also parent metric.
summary table	A container that stores the information collected by the Precise agents and loads it into the PMDB. Summary tables store the same data at different levels of granularity: time slice, hourly, daily, weekly, and monthly. By storing data in multiple summary tables, Precise can present a detailed view and progressively higher-level views of the same data. Summary tables are particularly useful for data aging. A data purging policy can be implemented for each summary table so that detailed data is retained for short-term historical analyses while more summarized data is used for long-term analyses and trending.

technology	A technology identifies the monitored object. For example, Oracle, SQL Server, Sybase, and DB2 are different technologies, while all Web servers (such as: Apache, IIS, and WebSphere) are defined as a single Web technology. A single monitored object can be monitored by two different technology's Collectors. For example, WebLogic server can be monitored by both a Web Collector and a J2EE Collector.
throughput	The number of average completions per second that are observed during an interval.
time slice	A unit used to break up long sessions into smaller time periods. The length of a time slice is preset and cannot be changed. It represents the maximum time that passes before the data collected can be displayed. For example, if the length of a time slice is 15 minutes, the collection is only updated at 15-minute intervals.
	The length of a time slice is different for each technology: In Oracle, SQL Server, Sybase, DB2, Web, .NET, MQ, Tuxedo, Network, and OS, a time slice is 15 minutes. In J2EE and SmartLink, a time slice is either 30 seconds or 5 minutes. In Precise for SAP, a time slice is 5 minutes.
	See also collector slice size.
URI (Uniform Resource Identifier)	The relative path to a resource after the location (network node) is found.
URL mapping	In Web, a function that defines rules that map URLs (Uniform Resource Locators) with dynamic parameters originating from a specific domain to a format that identifies the Web pages and prevents them from having different URLs.
Web client browser-side	In Web, the gathering of performance data from the browser running on the desktop of the Web application's user, through the static or dynamic instrumentation of Web pages.
Web server	A program that receives client requests for Web pages, retrieves the static pages and/or issues a request for dynamic page creation to an application server, and sends the pages to the client. In Web, a computer that delivers (serves up) Web pages. Every Web server has an IP address and possibly a domain name.
work time	The time spent in the specific entity excluding time spent by called (other) entities. In J2EE, work time of a method is calculated as the service time of that method minus the service time of the methods it invoked.
work type	In Web, a counter that keeps track of the type of the Web instance, such as Web, PeopleSoft, SAP, or Siebel.
workspace	A display unit in Precise products. All workspaces display data from different perspectives. For example, in Oracle, SQL Server, and Sybase, the Current workspace shows information on the sessions currently active in an application, and the Objects workspace displays information on Oracle or SQL Server database objects that can be used to understand relationships and associations between database schema objects.