Oracle® Fusion Middleware

Oracle WebCenter Analytics Administrator's Guide (for Oracle WebLogic Portal)

10*g* Release 3 (10.3.0.1)

E14111-02

November 2009

Describes how to perform administration tasks for Oracle WebCenter Analytics running on Oracle WebLogic Portal.

ORACLE®

Oracle Fusion Middleware Oracle WebCenter Analytics Administrator's Guide (for Oracle WebLogic Portal), 10*g* Release 3 (10.3.0.1)

E14111-02

Contents

4 Managing the Oracle WebCenter Analytics Database

5 Synchronizing Oracle WebCenter Analytics

A Troubleshooting

Index

Preface

This book describes how to perform administration tasks for Oracle WebCenter Analytics 10*g* Release 3 (10.3.0.1).

Audience

This document is intended for Oracle WebCenter Analytics administrators who are responsible for configuring Oracle WebCenter Analytics reports, working with the Oracle WebCenter Analytics database, and maintaining the Oracle WebCenter Analytics system.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at `http://www.oracle.com/accessibility/`.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to

evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at `http://www.oracle.com/accessibility/`.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Deaf/Hard of Hearing Access to Oracle Support Services

To reach Oracle Support Services, use a telecommunications relay service (TRS) to call Oracle Support at 1.800.223.1711. An Oracle Support Services engineer will handle technical issues and provide customer support according to the Oracle service request process. Information about TRS is available at `http://www.fcc.gov/cgb/consumerfacts/trs.html`, and a list of phone numbers is available at `http://www.fcc.gov/cgb/dro/trsphonebk.html`.

Related Documents

For more information, see the following documents in the Oracle WebCenter Analytics 10*g* Release 3 (10.3.0.1) documentation set:

- *Oracle WebCenter Analytics Release Notes*

- *Oracle Fusion Middleware Installation and Upgrade Guide for Oracle WebCenter Analytics (for Oracle WebLogic Portal)*

- *Oracle Fusion Middleware Developer's Guide for Oracle WebCenter Analytics*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

Overview of Oracle WebCenter Analytics

Oracle WebCenter Analytics allows portal managers and business owners to track and analyze portal usage. Oracle WebCenter Analytics provides the following basic functionality:

- Usage Tracking Metrics: Oracle WebCenter Analytics collects and reports metrics of common portal functions, including desktop and portlet hits.

- Behavior Tracking: Users of Oracle WebCenter Analytics reports can analyze portal metrics to determine usage patterns, such as portal visit duration and usage over time.

- User Profile Correlation: Users of Oracle WebCenter Analytics reports can correlate metric information with user profile information. Usage tracking reports can be viewed and filtered by user profile data such as country, company or title.

1.1 Components of Oracle WebCenter Analytics

Oracle WebCenter Analytics is comprised of the following components:

Figure 1–1   Oracle WebCenter Analytics Architecture
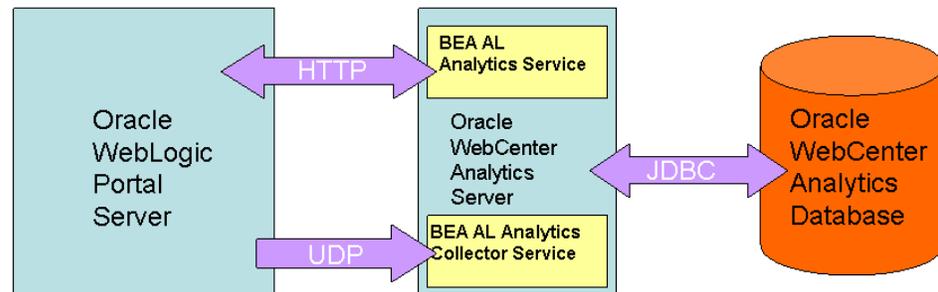
This graphic shows the Oracle WebCenter Analytics architecture. The Oracle WebLogic Portal Server communicates with the BEA AL Analytics Service using HTTP, and sends information to the BEA AL Analytics Collector Service using UDP. Both services are located on the Oracle WebCenter Analytics Server. The Oracle WebCenter Analytics Server communicates with the Oracle WebCenter Analytics database using JDBC.

****

The following table describes the components that are delivered with Oracle WebCenter Analytics. For a list of ports used by Oracle WebCenter Analytics, see

Oracle Fusion Middleware Installation and Upgrade Guide for Oracle WebCenter Analytics (for Oracle WebLogic Portal).

Table 1–1   Oracle WebCenter Analytics Components

Component	Description
Analytics services	The Oracle WebCenter Analytics installer loads the Oracle WebCenter Analytics application and the following Oracle WebCenter Analytics services on a stand-alone server:
	■ BEA AL Analytics Collector service receives data and gathers data from the portal database.
	■ The BEA AL Analytics service provides Oracle WebCenter Analytics data to the end user through the Analytics Console.
	For details on installing Analytics services, see *Oracle Fusion Middleware Installation and Upgrade Guide for Oracle WebCenter Analytics (for Oracle WebLogic Portal)*.
Analytics database	The Analytics database component provides storage for metrics that are gathered from portal events. For details on configuring the Oracle WebCenter Analytics database, see *Oracle Fusion Middleware Installation and Upgrade Guide for Oracle WebCenter Analytics (for Oracle WebLogic Portal)*.

2

Working with Oracle WebCenter Analytics Reports

This chapter provides information on using and analyzing Oracle WebCenter Analytics reports, creating portlets, and configuring the portal for most favorable tracking. It includes the following topics:

- Section 2.1, "Overview of Oracle WebCenter Analytics Reports"
- Section 2.2, "Accessing Oracle WebCenter Analytics Reports"
- Section 2.3, "Tips and Techniques - Using Oracle WebCenter Analytics Reports"

2.1 Overview of Oracle WebCenter Analytics Reports

This section provides descriptions of the reports that are delivered with Oracle WebCenter Analytics.

The following table describes reports that are delivered with Oracle WebCenter Analytics.

Table 2–1   Descriptions of Oracle WebCenter Analytics Reports

Oracle WebCenter Analytics Report	Description
Summary Metrics - Traffic	The Traffic console page displays metrics for many common events within the portal including desktop views, page views, search events, and portlet views.
Summary Metrics - Pages	The Pages console page reports on portal page views. The report displays the most visited pages (top pages), the least visited pages (bottom pages), and all pages.
	The data can be summed by hits (total number of page views) or users (unique number of users who viewed pages). Additionally, the report can be filtered to show only pages from specific desktops.
Summary Metrics - Logins	The Logins console page displays counts of logins into the portal. The data can be summed by Logins (total logins into the portal) or Users (unique users who logged into the portal).
Summary Metrics - Duration	The Duration console page shows the maximum and/or average duration of all portal visits. A visit is defined as the time between a user's first click on any page within the portal and the same user's first click on a page outside of the portal, regardless of whether or not that user has logged in to the portal. Note that this report provides metrics for portal visits in general, not for specific web applications within the portal.

Table 2–1   (Cont.) Descriptions of Oracle WebCenter Analytics Reports

Oracle WebCenter Analytics Report	Description
Desktop Metrics - Views	The Desktop Traffic console page displays data regarding desktop page views. The report displays the most viewed desktops (top desktops), the least viewed desktops (bottom desktops), all desktops (which is only available in tabular view) and selected desktops.
	The report data can be viewed by Hits (page views), Visits (consecutive page views within a single desktops) and Users (unique users who viewed pages within the selected desktops).
Desktop Metrics - Response Time	The Response Time console page displays the length of time that has elapsed from when the portal server receives a request for a desktop page until the time the response is sent to the user. Because the tracking is done on the portal server, the length of time it takes for the response to reach the user's machine and be displayed on the browser is not included in this metric.
	This report displays the desktops with the fastest response time (top desktops), the desktops with the slowest response time (bottom desktops), all desktops (which is only available in tabular view) and selected desktops. The report shows maximum, average and minimum response times.
Portlet Metrics - Views	The Portlet Views console page tracks the number of times a portlet is displayed within the portal. That is, each time a page is viewed, all of the portlets on that page are counted as being viewed.
	The report displays the most viewed portlets (top portlets), the least viewed portlets (bottom portlets), all portlets (which is only available in tabular view), selected portlets and portlets within selected desktops. The report data can be viewed by Views and Users (unique users who viewed the selected portlets).
Portlet Metrics - Response Time	The Response Time console page displays the time from when the portal sends a request for a portlet until the time the remote server responds with the portlet content. This report is important because typically a desktop page response equals the response of the slowest portlet on that page. For this reason, when troubleshooting slow desktops it is important to find the worst performing portlet. The amount of time it takes for the response to reach the user's machine and be displayed on the browser is not included in this metric because the tracking occurs on the portal server.
	This report displays the portlets with the fastest response time (top portlets), the portlets with the slowest response time (bottom portlets), all portlets
	Note that this option is only available in tabular view), selected portlets and portlets within selected desktops. The report shows maximum, average or minimum response time.
Other Metrics - Search	The Search console page tracks portal searches. The report displays the top search terms.

2.2 Accessing Oracle WebCenter Analytics Reports

To access Oracle WebCenter Analytics reports:

1.   Navigate to WebLogic Portal Administration.

2.   Under Configuration & Monitoring, choose **Analytics**.

The Oracle WebCenter Analytics application appears.

3.   Click the **Reports** tab.

2.3 Tips and Techniques - Using Oracle WebCenter Analytics Reports

The following table provides tips and techniques for how you can more effectively use Oracle WebCenter Analytics reports to track activity and improve the performance of your portal.

Table 2–2   Tips and Techniques - Using Oracle WebCenter Analytics Reports

Tip	Technique
Combine filtering and grouping when viewing reports	Combining filtering and grouping lets you utilize Oracle WebCenter Analytics reports in some powerful ways. For example, you can view a breakdown of sales force activity by city, determine the most active city, then view the departments that are making that city so active. The following example provides steps that you might perform to make these determinations in a similar report of your own: 1. Filter the report by the Department user property, which contains the Sales property value. 2. Group the report by the City user property. The report displays a breakdown of sales force activity by city. 3. Determine the most active city in the report. 4. Filter the report by the most active city. 5. Group the report by the Department user property. The report displays the departments that are making that city so active.
Use the Analytics Console as a portal clean up tool	Analyzing reports that track desktop usage can help you to improve the performance of your portal. You can use Oracle WebCenter Analytics reports to: ■ Find the least-active desktops and portlets in your portal. Once you have done this, determine if these desktops and portlets are valuable. If they are not valuable, remove them from your portal. ■ Find the desktops and portlets that have the slowest performance in your portal, and determine whether you should fix or remove these desktops or portlets from your portal; then take appropriate action.

2.4 Exporting Oracle WebCenter Analytics Reports

You can export the report you are viewing to an exteternal file.

■ To export the report you are viewing, in the upper left corner, click **Export Report**.

■ To export the report you are viewing, grouped by user (including the first and last event for each user), in the upper left corner, click **Export User Details**.

3

Using Analytics Administration

This chapter provides information on accessing and working with the Analytics Administration console. It includes the following topics:

- Section 3.1, "Accessing Analytics Administration"
- Section 3.2, "Using Analytics Administration"

3.1 Accessing Analytics Administration

To access Analytics Administration:

1. Navigate to WebLogic Portal Administration.

2. Under Configuration & Monitoring, choose **Analytics**.

   The Oracle WebCenter Analytics application appears.

3. Click the **Administration Console** tab.

3.2 Using Analytics Administration

This section contains the following topics:

- Section 3.2.1, "Configuring Runtime Settings"
- Section 3.2.2, "Configuring Partition Settings"
- Section 3.2.3, "Registering Events"

3.2.1 Configuring Runtime Settings

The Runtime Settings page lets you configure timeout periods, and choose the format of exported reports. The timeout settings that you specify on the Runtime Settings page are only for Oracle WebCenter Analytics reporting purposes; there is no relationship between these timeout settings and other portal timeout settings.

The following table describes the timeout settings that you can configure.

Table 3–1   Timeout Settings

Setting	Description
Visit Timeout (seconds)	Type the maximum number of seconds that a user must remain inactive during a portal visit in order for Oracle WebCenter Analytics to report subsequent activity as a new portal visit. Oracle WebCenter Analytics reports this data on the Duration console page. The default is 120 seconds.

In the Preferred Locale area of the Runtime Settings page, use the drop-down list to select the locale (language) in which you want Oracle WebCenter Analytics to display report data. The locale that you select determines the object names that Oracle WebCenter Analytics requests from your portal. For example: if you select *France - France [fr_FR]*, Oracle WebCenter Analytics requests the French names—localized for France—of objects from your portal.

The Preferred Locale setting only determines the language of the data in Oracle WebCenter Analytics reports. Individual users must configure their own browser's locale setting to specify their language preference for non-data strings that appear on Oracle WebCenter Analytics console pages. Examples of non-data strings are text in dialog boxes, drop-down lists, buttons, and tabs.

> **Note:** You must ensure that strings in the preferred locale that you select exist in your portal. If your portal does not return object names in your preferred locale, Oracle WebCenter Analytics performs one of the following:
>
> - If your preferred locale is not country-specific (for example, if your preferred locale is *French - [fr]*), Oracle WebCenter Analytics requests object names from your portal using the locale setting of the machine on which Oracle WebCenter Analytics is installed. If your portal does not return object names in this locale, Oracle WebCenter Analytics displays object IDs in reports instead of localized strings.
>
> - If your preferred locale is country-specific—and your portal does not return object names in this locale—Oracle WebCenter Analytics attempts to find strings in the most appropriate locale that is not country-specific. If your preferred locale is not country-specific (for example, if your preferred locale is *France - France [fr_FR]*) and your portal does not return object names in this locale—Oracle WebCenter Analytics requests object names from *French - [fr]*. If your portal does not return strings in this locale, Oracle WebCenter Analytics requests object names using the locale setting of the machine on which Oracle WebCenter Analytics is installed. If your portal does not return object names in this locale, Oracle WebCenter Analytics displays object IDs in reports instead of localized strings.

The following table describes the export report settings that you can configure.

Table 3–2 Export Report Settings

Setting	Description
Export Report	Select to export reports in one of the following formats:
	- Excel (10,000 row limit): Exporting to Excel is limited to 10,000 rows. If you choose this option, reports that contain more than 10,000 rows in Oracle WebCenter Analytics contains only the first 10,000 rows when exported to Excel.
	- TSV: Tab Separated Values is a text-based file format in which exported columns are separated by tabs. TSV files are typically opened by Excel.
	Note that although TSV reports have no limitation on numbers of rows, exported reports in excess of 10,000 rows might take an extended period of time to generate.

3.2.2 Configuring Partition Settings

This section discusses the configuration of Analytics Administration's Partition Settings page. For guidelines on archiving and restoring partitions in your Oracle WebCenter Analytics database, see Section 4.2, "Archiving and Restoring Partitions."

Oracle WebCenter Analytics data is segregated into month-by-month partitions in the database. The Partition Settings page lets Portal Administrators specify the number of data partitions that are accessible to Oracle WebCenter Analytics reports. The Analytics Engine regularly scans each data partition that is accessible. For this reason, you can have more control over system performance by limiting the number of accessible partitions to only those that are needed.

At the beginning of each month, the system creates a new partition and stores all new Oracle WebCenter Analytics data in that partition until the month ends.

> **Note:** You also use the Partition Settings page to refresh database views after archiving or restoring partitions. To do so, click **Finish**. If you do not refresh the database views, Oracle WebCenter Analytics reports fail.

3.2.2.1 Working with the Scrolling View Window

The Scrolling View Window, which you configure on the Partition Settings page, is a rolling database view that changes month-to-month. For example, suppose that on August 5, you enable the Scrolling View Window and set its size to 4 months. The following occurs:

The Scrolling View Window makes data from 4 months available: May, June, July, and August. (Note that even though the month of August has not completed, the Scrolling View Window makes data from the August partition available so that the most current data appears in Oracle WebCenter Analytics reports).

On September 1, the Scrolling View Window scrolls one month forward and makes data from the June, July, August, and September partitions accessible. Though the Scrolling View Window removes the May partition from view, the partition and its data is still stored in the database. In this example, if in December of the same year the Portal Administrator wanted to make the data from the May partition available to reports, he or she would increase the Scrolling View Window size to 8 months so that it includes the May to December time frame.

To set the number of months, that are accessible to Oracle WebCenter Analytics reports, first enable the Scrolling View Window, then specify the number of months: between 3 and 60. Data from monthly partitions that do not fall within the Scrolling View Window size is made unavailable to Oracle WebCenter Analytics reports.

> **Note:** If you do not enable the Scrolling View Window, all data from all partitions is accessible to Oracle WebCenter Analytics reports.

3.2.2.2 Previewing Partitions

At any time, you can click **Preview** on the Partition Settings page to view a chart that displays a representation of all partitions and indicates whether they are available or unavailable to Oracle WebCenter Analytics reports:

- Green squares indicate partitions whose data is available to Oracle WebCenter Analytics reports.

- Red squares with an X indicate partitions whose data is unavailable to Oracle WebCenter Analytics reports; these partitions do not exist in the database.

- Grey squares indicate partitions for months that do not currently fall within the Scrolling View Window's date range. These partitions may or may not exist in the database.

3.2.3 Registering Events

The Event Registration feature lets Portal Administrators and developers register custom portal and non-portal events that are sent to Oracle WebCenter Analytics using the OpenUsage API. Event data is saved to the Oracle WebCenter Analytics database, which can then be queried for reporting to a non-portal application. This feature's user interface includes the Event Registration and Create Event pages.

Use the Event Registration page to:

- Launch the Create Event page, which you use to create events and their parameters.

- Enable and disable the storage of data generated by individual events. Once event data is stored in the database, you can query the data for reporting in a portal or non-portal application.

- Create dimensions, which are discussed below.

This topic only discusses the Event Registration user interface. To utilize the full capabilities of this feature, you must use the OpenUsage API. For additional details on using the OpenUsage API, see *Oracle Fusion Middleware Developer's Guide for Oracle WebCenter Analytics*.

3.2.3.1 Overview of Events, Event Parameters, and Dimensions

An event typically defines one user action that you would like to capture. Each event is composed of several *event parameters*, which defines the various kinds of data that is generated by the event. By default, each event includes the USERID, TIMEID, VISITID, and OCCURRED event parameters (for more details on these parameters, see Section 3.2.3.3.4, "Delivered Parameters").

Event Registration also lets you create your own event parameters to capture data that is not defined by the delivered defaults. To capture data of type Date, Integer, or Float, simply create an event parameter for that data type. To capture data of type String, you must create a dimension to define this data (a dimension defines data of type String). After you create the dimension, you must create a new event parameter of type String, and associate it to the dimension that you created.

> **Note:** We recommend that you do not create too many dimensions, since they slow down the speed of data collection and reporting.

Event Registration also lets you use dimensions from your own application's tables. To do so, create a parameter of type Integer. Then, using your database schema, map the parameter to the ID column in your dimension table.

3.2.3.2 Creating Custom Dimensions

The Managed Dimensions section of the Event Registration page lets you create custom dimensions to define data of type *String*.

To use a dimension, you must associate it to an event parameter on the Edit Event page. You can associate a dimension to multiple event parameters and use that dimension in multiple events.

Dimension names can be up to 20 characters in length and can only include letters, numbers, spaces, and underscores.

The names that you create only define how dimensions appear on the Event Registration page. In the database, custom dimension tables use the following naming convention: ascdim_*dimension_name* (spaces in dimension names are replaced by underscores).

> **Note:** You cannot remove or change the name of a dimension after clicking **Finish** on the Event Registration page.

To store only unique values in this dimension's table, check **Unique**.

> **Note:** In the database, dimension tables using the naming convention asdim_*dimension_name* store events that are generated by the portal; they do not store custom events. For descriptions of these tables, see the Oracle WebCenter Analytics Database Schema document.

3.2.3.3 Creating Events

The Create Event page lets you define the event name and parameters that are included in the event. After you define the event and its parameters, click **Finish** to save an event and create its fact table in the database. Access the Create Event page by clicking **Add** on the Event Registration page.

> **Note:** After you save an event, you cannot change the names of the event and its parameters; you also cannot delete parameters from an event. You can, however, add parameters after saving an event.

3.2.3.3.1 Event Names Event names can be up to 14 characters in length and can only include letters, numbers, spaces, and underscores.

> **Note:** You must set the OpenUsage API to send data to the fact table using the event name that you specified on this page, not its column name in the database.

3.2.3.3.2 Table Names Table names are read-only, and define the name of the fact table that will store the event's data. Custom event tables use the following naming convention in the database: ascfact_*event_name* (spaces in event names are replaced with underscores).

3.2.3.3.3 Parameter Names Parameter names can be up to 14 characters in length and can only include letters, numbers, spaces, and underscores. Use this parameter name when passing parameters with the OpenUsage API.

> **Note:** You cannot remove or change the name of a parameter after clicking **Finish** on the New Event page.

3.2.3.3.4 Delivered Parameters The following table describes the parameters that are delivered with Oracle WebCenter Analytics which are, by default, included in every event:

Table 3–3

Parameter	Description
USERID	The ID of the user who triggers the event. You must use the OpenUsage API to set the User ID.
TIMEID	The unique ID number that is created for each occurrence of the event. This value is set by Oracle WebCenter Analytics.
VISITID	The portal visit ID of the user who triggered the event.
	Note: This parameter is only compatible with events that occur in the portal. This value is set by Oracle WebCenter Analytics.
OCCURRED	The date and time when the event was generated. The format of the date/time stamp is determined by your database type. This value is set by Oracle WebCenter Analytics.

3.2.3.4 Saving Event Data

To begin saving an event's data, select the check box next to an event on the Event Registration page and click **Enable**. By default, the BEA AL Analytics Collector service starts saving the event's data 30 minutes after you click **Enable**.

3.2.3.5 Stopping Event Data Storage

To stop saving an event's data, select the check box next to an event on the Event Registration page and click **Disable**. You can edit events even when they are disabled. By default, the BEA AL Analytics Collector service stops saving the event's data 30 minutes after you click **Disable**.

4

Managing the Oracle WebCenter Analytics Database

This chapter includes the following topics:

4.1 Sizing the Oracle WebCenter Analytics Database

This section provides information and recommendations that should help you manage the size and growth of your Oracle WebCenter Analytics database to, which should increase performance. It includes the following topics:

4.1.1 Overview of Oracle WebCenter Analytics Database Growth

The majority of growth in the Oracle WebCenter Analytics database occurs in the fact tables that are delivered with Oracle WebCenter Analytics. The rest of the delivered Oracle WebCenter Analytics tables -- including dimension tables -- generate negligible growth in the Oracle WebCenter Analytics database. This section provides details on the growth of fact and dimension tables.

4.1.1.1 Fact Table Growth

Fact tables capture event parameter data of types Date, Integer, and Float.

Both Oracle and SQL Server databases grow at approximately the same rate: every one million events that are stored in the database use approximately 150 megabytes of disk space. The following table lists numbers of events and their corresponding estimated database sizes.

Table 4–1   Numbers of Events and Corresponding Estimated Database Sizes

Number of Events	Estimated Database Size
1,000,000	150 megabytes
2,000,000	300 megabytes
5,000,000	750 megabytes
10,000,000	1.5 gigabytes

Use these size estimates to calculate your own database growth requirements. As the number of events in your system continues to grow, query performance ultimately starts to decline. For this reason, in high-volume environments you should monitor the growth of your database and take appropriate measures to prevent performance degradation.

4.1.1.2 Dimension Table Growth

Oracle WebCenter Analytics uses dimensions to capture event parameter data of type String. Dimension tables do not grow as quickly as delivered fact tables because dimension data does not change at nearly the same rate as events occur.

Note that if an object is removed from the application on which Oracle WebCenter Analytics is reporting, the record for that object's dimension data remains in the Oracle WebCenter Analytics database. For this reason, Oracle WebCenter Analytics continues to report the events that occurred on this object before it was removed from the application.

4.1.2 Tuning the Oracle WebCenter Analytics Database - Oracle

To obtain the best performance with Oracle, we recommend that you:

- Edit the create_analytics_tablespaces.sql script and include appropriate sizing information for your database. The create_analytics_tablespaces.sql script is located in a platform-specific subdirectory within the Oracle WebCenter Analytics database's host computer.

  > **Note:** The default values in the create_analytics_tablespaces.sql script are acceptable for a development or staging database. You should, however, change these values accordingly in a production environment.

- Move the ANALYTICSTABLE tablespace to a different drive than the one used for the ANALYTICSINDEX tablespace.
- Configure the Oracle settings described in this table:

Table 4–2   Oracle Settings and Configuration Recommendations

Oracle Setting	Configuration
Buffer Cache	Increase to 250 megabytes
PGA Aggregate Target	Increase to 150 megabytes
Data Block Size	Increase to 16K

> **Note:** The recommendations in this table are for use with the Oracle WebCenter Analytics database only. You might want to change these configurations slightly to more appropriately suit your environment.

4.2 Archiving and Restoring Partitions

This section provides guidelines for archiving and restoring the partitions within your Oracle WebCenter Analytics database. For details on using Analytics Administration's Partition Settings page, see Section 3.2.2, "Configuring Partition Settings."

To maintain a steady size of your Oracle WebCenter Analytics database and keep your queries performing quickly, we recommend archiving partitioned data that is greater than six months old. You can identify partitioned tables by their date/year suffix. For example: _08_2006. You archive and restore partitions using database commands. See documentation for database for more details. It is up to your organization to decide the media to which you want to back up your data.

After archiving or restoring partitions, you must refresh the database views by clicking **Finish** on the Partition Settings page. If you do not refresh the database views, Oracle WebCenter Analytics reports fail.

> **Caution:** Never remove the current fact table, which is not partitioned. Also, never remove database views. Instead, use the scrolling view window to set the number of partitions that are accessible to Oracle WebCenter Analytics reports. Additionally, if you delete archived partitions from your database, you will not be able to restore them. As a result, customers will no longer have access to the deleted data.

5

Synchronizing Oracle WebCenter Analytics

This chapter includes the following topics:

- Section 5.1, "Overview of Oracle WebCenter Analytics Synchronization"
- Section 5.2, "Strategizing Synchronization Scheduling"

5.1 Overview of Oracle WebCenter Analytics Synchronization

Oracle WebCenter Analytics synchronization is a process by which object dimension data is sent from the remote database to the dimension tables of the Oracle WebCenter Analytics database, where the data is stored. Dimension data is transferred through a JDBC connection.

If you are using Streaming Portal Mode, use the command line to run Analytics sync jobs. To schedule jobs, create scheduled tasks (Windows) or cron jobs (UNIX/Linux). An Analytics sync job exists for each type of portal object. For the commands you use to run sync jobs, see Section 5.1.1, "Analytics Sync Jobs." For synchronization scheduling guidelines, see Section 5.2, "Strategizing Synchronization Scheduling."

You can find details of sync jobs in the portal's job history log or in Oracle WebCenter Analytics's sync.log file. For the location of the Oracle WebCenter Analytics sync.log file, see Section A.1, "Overview of Logs."

5.1.1 Analytics Sync Jobs

Oracle WebCenter Analytics is delivered with sync jobs that synchronize your portal's objects with the Oracle WebCenter Analytics database. You can either run all three of the sync jobs at the same time using one command, or run each sync job separately. Separate sync jobs exist for synchronizing desktop, page, and portlet objects with Oracle WebCenter Analytics.

5.1.1.1 Analytics Sync Jobs (Windows)

To synchronize desktop, page, and portlet objects at the same time, type the following command in the Windows Command Prompt:

- *install_dir*\ptanalytics\10.3.0.1\bin\AnalyticsRunJobs.bat

By default, *install_dir* is c:\bea\alui.

Following are the commands you use to run each sync job separately:

- *install_dir*\ptanalytics\10.3.0.1\bin\AnalyticsRunJobs.bat
  RUN_COMMUNITY_SYNC

- *install_dir*\ptanalytics\10.3.0.1\bin\AnalyticsRunJobs.bat
  RUN_COMMUNITY_PAGE_SYNC

- *install_dir*\ptanalytics\10.3.0.1\bin\AnalyticsRunJobs.bat
  RUN_PORTLET_SYNC

To schedule a sync job, create a scheduled task and use one of the above commands appropriate to the sync job you are scheduling.

5.1.1.2 Analytics Sync Jobs (UNIX/Linux)

To synchronize desktop, page, and portlet objects at the same time, type the following command in the command line:

- *install_dir*/ptanalytics/10.3.0.1/bin/AnalyticsRunJobs.sh

By default, *install_dir* is /opt/bea/alui.

Following are the commands you use to run each sync job separately:

- *install_dir*/ptanalytics/10.3.0.1/bin/AnalyticsRunJobs.sh RUN_
  COMMUNITY_SYNC

- *install_dir*/ptanalytics/10.3.0.1/bin/AnalyticsRunJobs.sh RUN_
  COMMUNITY_PAGE_SYNC

- *install_dir*/ptanalytics/10.3.0.1/bin/AnalyticsRunJobs.sh RUN_
  PORTLET_SYNC

To schedule a sync job, create a cron job and use one of the above commands appropriate to the sync job you are scheduling.

5.2 Strategizing Synchronization Scheduling

Because some dimension data in you environment might change more frequently than other dimension data, we recommend that you adopt a job scheduling strategy that reflects your environment's unique circumstances.

Following are some examples of scheduling strategies that you can adopt:

- If your portal contains objects that rarely change, run the appropriate Analytics sync job manually. For example, if your portal's desktop objects rarely change, run the desktop sync job manually.

- If your portal contains objects that change only once per week, schedule the appropriate Analytics sync job to run once per week. For example, if new portlets are added to your production portal once per week, schedule the portlet sync job to run once per week.

- If your portal contains data that changes on a constant basis, schedule the appropriate Analytics sync job to run frequently. For example, if new pages are added to your production portal throughout the day, schedule the pages sync job to run once per hour.

A
Troubleshooting

This appendix provides information on troubleshooting problems that occur in Oracle WebCenter Analytics runtime. It includes the following topics:

- Section A.1, "Overview of Logs"

- Section A.2, "Troubleshooting Common Runtime Problems"

> **Note:** For details on troubleshooting the installation and configuration of Oracle WebCenter Analytics, see the *Oracle Fusion Middleware Installation and Upgrade Guide for Oracle WebCenter Analytics (for Oracle WebLogic Portal)*.

A.1 Overview of Logs

The following table provides the descriptions and locations of logs that you can use to troubleshoot Oracle WebCenter Analytics during runtime. Individual log files are generated for each day's activity.

Table A–1   Logs Used to Troubleshoot Oracle WebCenter Analytics in Runtime

Log	Description	Location and Platform
analyticsui.log	Provides activity and error details for the Analytics Console user interface/BEA AL Analytics service.	■ Windows: *install_ dir*\ptanalytics\10.3.0.1\logs ■ UNIX/Linux *install_ dir*/ptanalytics/10.3.0.1/logs
wrapper_analyticsui.log	Provides additional activity and error details for the BEA AL Analytics service, including details for these user interfaces: ■ Analytics Console ■ Analytics Administration ■ Analytics Configurator	■ Windows *install_ dir*\ptanalytics\10.3.0.1\logs ■ UNIX/Linux *install_ dir*/ptanalytics/10.3.0.1/logs
asmanagerui.log	Provides activity and error details for the Analytics Administration user interface/BEA AL Analytics service.	■ Windows: *install_ dir*\ptanalytics\10.3.0.1\logs ■ UNIX/Linux: *install_ dir*/ptanalytics/10.3.0.1/logs

Table A–1   (Cont.) Logs Used to Troubleshoot Oracle WebCenter Analytics in Runtime

Log	Description	Location and Platform
collector.log	Provides activity and error details for: ■ BEA AL Analytics Collector service. ■ Routine partitioning activity that occurs during Oracle WebCenter Analytics runtime. The partition.log file provides activity and error details for initial partitioning of the Oracle WebCenter Analytics database during upgrade.	■ Windows: *install_dir*\ptanalytics\10.3.0.1\logs ■ UNIX/Linux: *install_dir*/ptanalytics/10.3.0.1/logs
wrapper_collector.log	Provides additional activity and error details for the BEA AL Analytics service, including details for these user interfaces: ■ Analytics Console ■ Analytics Administration ■ Analytics Configurator	■ Windows: *install_dir*\ptanalytics\10.3.0.1\logs ■ UNIX/Linux: *install_dir*/ptanalytics/10.3.0.1/logs
sync.log	Provides activity and error details for Analytics sync jobs.	The sync.log file is located on the Automation Service host machine in the following directories: ■ Windows: *install_dir*\ptanalytics\10.3.0.1\logs ■ UNIX/Linux: *install_dir*/ptanalytics/10.3.0.1/logs

A.2 Troubleshooting Common Runtime Problems

The following table describes common runtime problems and provides solutions to them.

Table A–2   Common Runtime Problems and Solutions

Problem Description and Details	Cause and Solution
Problem: No data is being generated in the report. Details: Oracle WebCenter Analytics reports are not being populated with data.	Cause 1: The BEA AL Analytics Collector service is not running. Solution 1: Verify that the BEA AL Analytics Collector service is started. For details, see the *Oracle Fusion Middleware Installation and Upgrade Guide for Oracle WebCenter Analytics (for Oracle WebLogic Portal)*. Cause 2: You did not refresh the database views after archiving or restoring partitions. Solution 2: After archiving or restoring partitions, you must refresh the database views by clicking **Finish** on the Partition Settings page in Analytics Administration. Cause 3: You did not correctly configure the analytics-config.xml file. Solution 3: Check your settings in analytics-config.xml and make any necessary changes. Cause 4: The fully-qualified domain name is not correctly specified beneath the report-url and admin-url nodes in the analytics-config.xml file. Solution 4: Correctly specify the fully-qualified domain name beneath the report-url and admin-url nodes in the analytics-config.xml file. Cause 5: The WebLogic Portal host and the Oracle WebCenter Analytics host are not in the same domain. Solution 5: Ensure that the WebLogic Portal host and the Oracle WebCenter Analytics host are in the same domain. Cause 6: You did not enter fully-qualified domain names in the analytics-config.xml file. Solution 6: You may have entered host names into the analytics-config.xml file, instead of fully-qualified domain names. Edit the file and replace the host names with fully-qualified domain names.
Problem: When using filtering options, reports do not query on all data. Details: Reports only query on data that is associated to properties included in the filter. Example: A company employs 10,000 employees. Only the 200 newest employees are assigned to Departments in their user profiles. The administrator runs a report and queries on a date range, and filters on the Department property, containing the `Human Resources` property value. The report only filters through the 200 newest employees, since only those employees are assigned to a Department. The report does not filter through the other 800 employees, since those employees are not assigned to a Department.	Solution: This behavior works as designed. If you want your filtered reports to query on all data, make sure that all data is associated to properties that are included in the filters. Using the example, if you want the report to filter through all 10,000 employees, then you must make sure that all 10,000 employees are assigned to a Department.

Table A–2   (Cont.) Common Runtime Problems and Solutions

Problem Description and Details	Cause and Solution

Problem: The text in Japanese reports does not appear correctly.
 	
Details: The Y axis does not display Japanese characters.
	Cause: The analytics.war file is not configured to display Japanese characters in the Y axis of Oracle WebCenter Analytics reports.
	
	Solution: Configure the AnalyticsUIParams.properties file to include font settings that are appropriate to your locale and platform:
	
	1. Unpack the analytics.war file.
	
	2. Open the AnalyticsUIParams.properties file for editing. You can find this file in the analyticsui.jar file.
	
	3. Add these locale entries:
	
	`LocaleCode_font_OSName=LocalFont`
	`LocaleCode_fontSize=14`
	`LocaleCode_fontSmallSize=10`
	
	4. Save and repackage .war file.
	
	The locale entries should include these values:
	
	■ *LocaleCode* should include the appropriate two letter language code
	
	■ *OSName* should include your operating system: Windows, Solaris, or Linux
	
	■ *LocalFont* should include the name of the font that is appropriate to your locale and platform
	
	Windows example:
	
	`ja_font_Windows=MS PGothic`
	`ja_fontSize=14`
	`ja_fontSmallSize=10`

Index