Oracle® Analytics

Administering Oracle Analytics Server

F24224-51
April 2025

Oracle Analytics Administering Oracle Analytics Server,

F24224-51

Primary Author: Nick Fry

Contributing Authors: Peter Brownbridge, Suzanne Gill, Rosie Harvey, Christine Jacobs, Stefanie Rhone, Hemala Vivek

Contributors: Oracle Analytics Server development, product management, and quality assurance teams.

Contents

Preface

Part I Administer Oracle Analytics Server

1 Introduction to System Administration in Oracle Analytics Server

Part II Manage Your Installation

2 Start and Stop Your System

3 Take Snapshots and Restore

4 Perform Common Administration Tasks

Part III Scale and Deploy for High Availability and Performance

5 Scale Your Deployment

6 Deploy for High Availability

7 Manage Performance Tuning and Query Caching

Part IV Resolve Issues

8 Diagnose and Resolve Issues

Part V Configure Oracle Analytics Server

10 Use Tools to Manage and Configure the System

11 Use Scripts to Manage Oracle Analytics Server Instances

12 Configure Connections to External Systems

17 Configure Currency Options

18 Manage How Content Is Indexed and Searched

19 Configure and Manage the Presentation Catalog

Part VI Advanced Configuration Settings

20 Configure and Manage Analyses and Dashboards

21 Configure and Manage Agents

22 Configure Advanced Options for Maps and Spatial Information

23 Configure Resource Availability and URL Generation

Preface

Learn how to administer Oracle Analytics to manage processes, components, logging, usage tracking, performance, scalability, high availability, and configuration options.

Audience

This guide is intended for system administrators who are responsible for managing the Oracle Analytics system.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Conventions

Conventions used in this document are described in this topic.

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.

Part I

Administer Oracle Analytics Server

This part introduces administering the Oracle Analytics Server system.

- Introduction to System Administration in Oracle Analytics Server

1
Introduction to System Administration in Oracle Analytics Server

This chapter introduces system administration; explains what a system administrator does; describes where to get started with typical system administration tasks; describes the system architecture; lists the tools that can help you complete system administration tasks; and provides links to system requirements and certification information.

Topics:

- What Are the System Administration Tasks?
- Get Started to Manage the System
- What Is the Logical System Architecture?
- Key Directories
- What Are the System Administration Tools?
- Access the Console in Oracle Analytics Server
- Access the Classic Administration Page
- Download, Install, and Start the Client Tools for Oracle Analytics Server
- Oracle Analytics Publisher Integration
- About Oracle Fusion Middleware Feature Support in Oracle Analytics Server

What Are the System Administration Tasks?

System administrators need to take several steps to configure the system properly.

Administering the system involves the following tasks:

- **Configuring a system for deployment after installation**

  Configuring metadata and content, general preferences, and default system settings.
- **Starting and stopping the system when required**

  Bringing the system up and down during system maintenance tasks.
- **Configuring security**

  Securing access to system, metadata, and data, configuring Secure Sockets Layer (SSL) and Single Sign-On (SSO), and integration with identity management systems.
- **Scaling out and configuring for high availability**

  Configuring the system for linear scale-out (increasing capacity with more components on a machine) and identifying and removing single points of failure (adding more machines).
- **Managing performance and availability**

  Monitoring service levels and tuning performance.
- **Managing and resolving issues**

Diagnosing errors and establishing resolutions.

- **Moving a system from test to production**

  Managing the steps for moving from a test to a production environment.
- **Backing up and recovering data**

  Preparing for and recovering from unexpected events.

See Get Started to Manage the System.

Get Started to Manage the System

Use this section to identify a task to complete, then click the corresponding link to display the appropriate content.

The table below describes the typical system administration tasks that you perform and indicates where to find related information.

System Administration Task	More Information
Learning about system administration	See the topics in this section. Contains information about the system architecture, components, tools, links to other related topics, and certification information.
Viewing system status	Display Oracle Analytics Server Pages in Fusion Middleware Control Also contains information about using Fusion Middleware Control and using WebLogic Server Administration Console.
Configuring the system	Methods for Configuring System Settings Contains information about the available methods for updating configuration settings and where configuration files are located.
Starting and stopping the system	Start and Stop Your System Contains various topics on starting and stopping components, in addition to troubleshooting information.
Managing availability and capacity	Scale and Deploy for High Availability and Performance Contains chapters about scaling the environment, deploying for high availability, performance tuning, and query caching.
Diagnosing problems and resolving issues	Resolve Issues Contains chapters about diagnosing and resolving issues and about usage tracking.
Configuring the system	Configuring Oracle Analytics Server Contains chapters about required configuration such as configuring semantic models and connections to external systems.
Modifying advanced configuration settings	Advanced Configuration Settings Contains chapters about advanced and optional configuration settings for features such as analyses, dashboards, and maps.
Configuring Oracle BI Scheduler	*Scheduling Jobs Guide for Oracle Business Intelligence Enterprise Edition*

System Administration Task	More Information
Managing the life cycle	Manage the Life Cycle Contains chapters about life cycle management tasks such as patching, moving between environments, and backup and recovery.
Securing the system	• Defines administrative role membership • Secures middle-tier communications Secure Sockets Layer (SSL) and Single Sign-On (SSO) aren't described in this guide. See Enable SSO Authentication and Configure SSL in *Managing Security for Oracle Analytics Server*.

What Is the Logical System Architecture?

The logical system architecture comprises a single integrated set of manageable components called the BI domain which can be installed and configured to work together on a single host or can be clustered across multiple hosts for performance and availability.

> **Note:**
>
> You can improve the performance of your production system by using a web server with the system such as Oracle HTTP Server or Apache HTTP Server. A web server isn't included by default in the installer and isn't part of the logical system architecture. You must install and configure a web server separately.

This section contains the following topics:

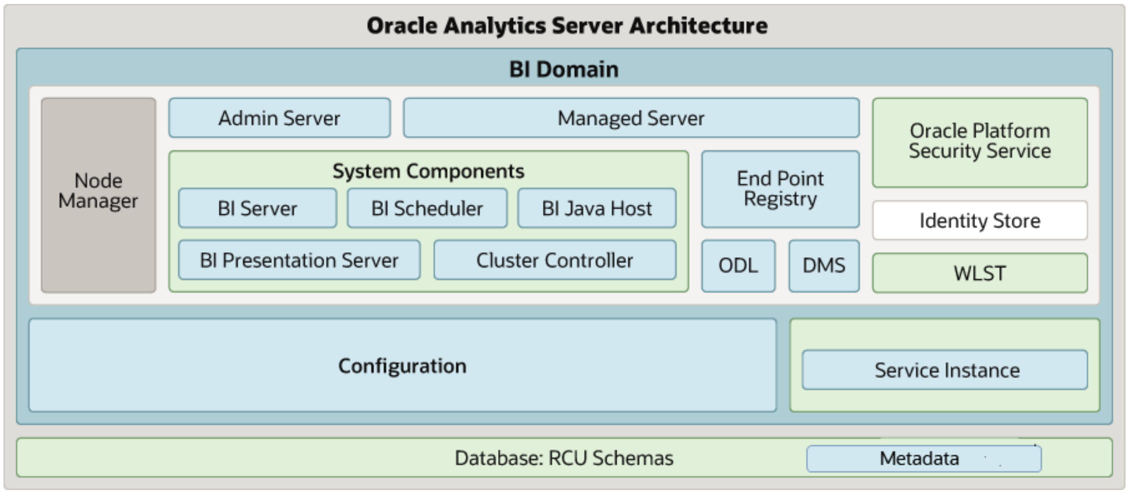
- Oracle Analytics Server System Architecture
- Oracle Analytics Server Components
- About the Administration Server, Managed Servers, and System Components

Oracle Analytics Server System Architecture

You install Oracle Analytics Server on a single host, but can subsequently scale out onto additional computers.

See Scale Your Deployment .

The figure below illustrates the system architecture on a single host. See Oracle Analytics Server Components.



You install Oracle Analytics on a single host but you can scale out onto multiple hosts. Java components (WebLogic server domain) and system components are clustered on each host as part of the single BI domain. The Administration Server exists on both hosts, but is active on only one host.

Oracle Analytics Server Components

When you install Oracle Analytics Server, you can install several components in the BI Domain on the host.

The BI Domain consists of Java components that are deployed into one or more Java EE (JEE) containers within a single WebLogic server domain ; system (non-JEE) components and processes; and required configuration files, semantic models, and infrastructure.

- **Administration Server** — Deployed as a JEE container that runs in a dedicated Java virtual machine that contains Java components for administering the system. These components include Oracle WebLogic Server Administration Console, Oracle Fusion Middleware Control, and JMX MBeans.
- **Managed Server** — Deployed as a JEE container that runs in a dedicated Java virtual machine that provides the runtime environment for the Java-based services and applications within the system. These services and applications include Publisher, Presentation Services.

  A BI domain contains one or more Managed Servers that are distributed across one or more host computers.
- **Node Manager** — Provides process management services for the Administration Server, Managed Server processes, and System Components.

  See Node Manager Overview in *Administering Node Manager for Oracle WebLogic Server*.
- **System Components** — Deployed as server processes that provide the core services.

  See About the Administration Server, Managed Servers, and System Components. For information about controlling system processes, see Process Control Commands.
- **Other Domain Contents** — Includes all the necessary software, configuration files, metadata, WLST commands, security, and connection and database configuration information that are required to run the system.

  For more information about:

  - Security configuration - See *Managing Security for Oracle Analytics Server*.
  - Metadata - See Manage Oracle Analytics Server Instances Using the WebLogic Server Scripting Tool and *Managing Metadata Repositories for Oracle Analytics Server*.

About the Administration Server, Managed Servers, and System Components

Oracle Analytics Server contains an Administration Server, Managed Servers, and system components which are described in this section.

See Getting Started Managing Oracle Fusion Middleware in *Administering Oracle Fusion Middleware*.

About the Administration Server and Managed Servers

The Administration Server and Managed Servers are Java components deployed as one or more Java EE applications and described in the following list:

- **Administration Server** — Manages configuration and runtime settings for a single or multi-node (distributed) BI domain, using:
  - **Fusion Middleware Control** — An administrative user interface that is used to manage the BI domain.
  - **WebLogic Server Administration Console** — An administrative user interface that provides advanced management for WebLogic, JEE components, and security.

  For more information, see What Are the System Administration Tools?.
- **Managed Server** — Manages the following components:
  - **Action Service** — This component provides the dedicated web services that are required by the Action Framework and that enable an administrator to manually configure which web service directories can be browsed by users when they create actions.
  - **Publisher** — This component provides an enterprise reporting solution for authoring, managing, and delivering all types of highly formatted documents to employees, customers, and suppliers.
  - **Security** — This component provides dedicated web services that enable the integration of the Oracle BI Server with the Oracle Fusion Middleware security platform.
  - **SOA Web Service** — This component provides dedicated web services for objects in the Presentation Catalog, to invoke analyses, agents, and conditions. These services make it easy to invoke functionality from Business Process Execution Language (BPEL) processes.
  - **Presentation Services** — This component is a JEE application that routes HTTP and SOAP requests to Presentation Services.

About System Components

System components are deployed as non-JEE components, such as processes and services written in C++ and J2SE, and are described in the following list:

- **BI Server (OBIS)** — This component provides the query and data access capabilities, and the services for accessing and managing the enterprise semantic model (stored in a file with an .rpd extension).

- **BI Scheduler (OBISCH)** — This component provides extensible scheduling for analyses to be delivered to users at specified times. (Publisher has its own scheduler.)
- **BI JavaHost (OBIJH)** — This component provides component services that enable Oracle BI Presentation Services to support various components such as Java tasks for Oracle BI Scheduler, Publisher, and graph generation. It also enables BI Server query access to Hyperion Financial Management and Hyperion Planning data sources.
- **BI Presentation Server (OBIPS)** — This component provides the framework and interface for the presentation of business intelligence data to web clients. It maintains a Presentation Catalog service for the customization of this presentation framework.
- **Cluster Controller (OBICCS)** — This component distributes requests to the BI Server, ensuring requests are evenly load-balanced across all BI Server process instances in the BI domain.

Key Directories

There are three key top-level directories.

- ORACLE_HOME - for binaries.

  There's one ORACLE_HOME for each host, or mounted from shared storage.
- DOMAIN_HOME - for configuration, and logs.

  There's one DOMAIN_HOME for each host (also referred to as BI_DOMAIN, or bidomain).
- SDD (Singleton Data Directory) - for cross-cluster files.

  There is one SDD for each domain.

  The SDD path (by default *DOMAIN_HOME*/bidata) is defined in the file bi-environment.xml, located in:

  `DOMAIN_HOME/config/fmwconfig/bienv/core/bi-environment.xml`

  > **Note:**
  >
  > If you've just created a domain on one host, then SDD is set to `DOMAIN_HOME/bidata`.
  >
  > If you've scaled-out, the SDD changes to use mounted shared storage. In this case, the SDD isn't `DOMAIN_HOME/bidata`.

  See Change the Singleton Data Directory (SDD).

What Are the System Administration Tools?

There are several tools that you can use to manage the system.

This section describes system administration tools that are available to help you to manage the system. The table outlines the tools and their purpose.

Tool	Purpose	More Information
Oracle Analytics Server Console	Manage configuration and administration of system settings, user accounts and permissions, information back up, safe domains, virus scanner, email server, deliveries, and more.	About the Oracle Analytics Server Console
Classic Administration	Most options on the Classic Administration page are exposed through the Console. Only use the Classic Administration page if you're familiar with on-premises products that use a similar page.	About the Classic Administration Page
Fusion Middleware Control	Monitor, manage, and configure system components.	Fusion Middleware Control
Oracle WebLogic Server Administration Console	Monitor and manage JEE Java components.	Oracle WebLogic Server Administration Console
Process control command line tool	Manage system components (for advanced users).	Process Control Commands
Scripts	Administer the system.	Scripts
Model Administration Tool	Manage the semantic model.	Model Administration Tool
Catalog Manager Command Line Interface	Manage the Presentation Catalog.	Catalog Manager Command Line Interface
Job Manager Command Line Interface	Manage the Oracle BI Scheduler	Job Manager Command Line Interface

About the Oracle Analytics Server Console

Use the Oracle Analytics Server Console unless you're specifically directed to do otherwise. Most administration options are exposed through the Console, so Oracle recommends that you configure your instance using the Console.

On the Home page click the **Navigator** and select **Console**.

Task	More Information
Advanced System Settings	Configure Oracle Analytics Server instance using a range of system settings. See About System Settings.
Content Management	View content in the catalog and change ownership. See Manage Content.
Content-Sharing Platforms	Configure social media platforms where users can share analytics content. See Integrate with Content-Sharing Platforms for Sharing Visualizations.
Data Enrichments	Update custom knowledge and system knowledge to enhance data enrichment. See Add Custom Knowledge for Data Enrichment.
Extensions	You can view and upload extensions from local folders.
Issue SQL	You can enter a SQL statement to issue directly against the Oracle Analytics Server. See Run Test SQL Queries.
Mail Settings	You can configure your email server. See Configure Email and Agents.
Maps	You can add custom map layers, make maps and map backgrounds available to users. See Apply Map Backgrounds and Map Layers to Enhance Visualizations in *Visualizing Data in Oracle Analytics Server*.

Task	More Information
Monitor Deliveries	You can monitor, schedule and edit deliveries. See Send Email Reports and Track Deliveries in *Visualizing Data in Oracle Analytics Server*.
Remote Data Connectivity	Not supported.
Roles and Permissions	You can manage users, roles, and application roles. See Manage Application Roles in *Managing Security for Oracle Analytics Server*.
Safe Domains	See Register Safe Domains.
Search Index	You can configure search indexing, semantic model crawls, Catalog crawls, and monitor crawl job status. See Manage How Content Is Indexed and Searched.
Session and Query Logs	You can manage sessions for your service instance and set the log level for this tenant. See Monitor Users and Activity Logs.
Snapshots	You can work with snapshots. See Take Snapshots and Restore .
Virus Scanner	You can configure a virus scanner for when you upload files. See Configure a Virus Scanner.
Workbook Translations	Localize workbook captions. See Localize Data Visualization Workbook Captions.

About the Classic Administration Page

Only use the Classic Administration page if you're familiar with on-premises products that use a similar page. Most options on the Classic Administration page are exposed through the Console, so Oracle recommends that you configure your service using the Console.

Task	More Information
Manage Privileges	Oracle recommends that you keep the default privileges because they’re optimized for Oracle Analytics. Editing privileges might result in unexpected behavior or access to features.
Manage Sessions	See which users are signed in and troubleshoot report queries. See Monitor Users and Activity Logs.
Manage Agent Sessions	Currently not available in Oracle Analytics Server.
Manage Device Types	Add devices that can deliver content for your organization. See Manage the Types of Devices that Deliver Content
Toggle Maintenance Mode	Indicates whether **Maintenance Mode** is on or off. In Maintenance Mode, you make the catalog read-only so that other users can't modify its content. Users can still view objects in the catalog, but they can't update them. Some features, such as the "most recently used" list aren't available.
Reload Files and Metadata	Use this link to reload XML message files, refresh metadata, and clear caches. You might want to do this after uploading new data, for example if you add or update a data model.
Reload Log Configuration	Oracle recommends that you keep the default log level. Oracle Support might suggest you change the log level to help troubleshoot an issue.

Task	More Information
Export Fallback Font	Oracle recommends that you configure Go Noto font as the fallback font in Classic reports and dashboards. Used when the default PDF fonts (such as Helvetica, Times-Roman, and Courier) can’t display non-Western characters included in the data when generating PDF output. See Open-Source Fonts to Replace Licensed Monotype Fonts.
Issue SQL	Test and debug SQL queries. See Run Test SQL Queries.
Scan and Update Catalog Objects That Require Updates	Use this link to scan the catalog and update any objects that were saved with earlier updates of Oracle Analytics.
Manage Themes	Change the default logo, colors, and heading styles for reporting pages, dashboards, and analyses. See Manage Themes.
Manage Captions	Localize the names (captions) of reporting objects that users create. See Localize Your Captions.
Manage Map Data	Define how users display their data on maps. See Manage Map Information For Analyses.
Manage Publisher	Set up data sources for pixel-perfect reports and delivery destinations. Configure the scheduler, font mappings, and many other runtime options. See Introduction to Publisher Administration.
Configure Crawl	This option is available through the Console. See Configure Data Model Search Indexing and Schedule Regular Content Crawls.
Monitor Crawl	This option is available through the Console. See Monitor Search Crawl Jobs.

Fusion Middleware Control

Fusion Middleware Control is a browser-based tool and the recommended method for monitoring, managing, and configuring system components.

Fusion Middleware Control is used principally for managing the system components of a BI domain and provides support for the following:

- Starting, stopping, and restarting system components.
- Configuring preferences and defaults.
- Viewing status of scaled out components.
- Managing performance and monitoring system metrics.
- Performing diagnostics and logging.

Fusion Middleware Control also provides access to Oracle WebLogic Server Administration Console, where you monitor and manage Java components.

Fusion Middleware Control is available only if the Administration Server is running, as described in Conditions to Start the System.

See Use Tools to Manage and Configure the System.

Oracle WebLogic Server Administration Console

Oracle WebLogic Server is a Java EE application server that supports the deployment of Java components in a robust, secure, highly available, and scalable environment.

See Use Tools to Manage and Configure the System.

Oracle WebLogic Server Administration Console enables you to monitor and manage a WebLogic Server domain. Its capabilities include the following:

- Monitoring the health and performance of JEE servers.
- Configuring WebLogic server domains.
- Stopping and starting JEE servers.
- Viewing JEE server logs.
- Managing user populations in the LDAP Server of the Oracle WebLogic Server.

See the Oracle Technology Network: http://www.oracle.com/technetwork/index.html

Process Control Commands

Process control commands enable you to manage Oracle Analytics Server system components, support both local and distributed process management, and the communication of process state (up, down, starting, and stopping).

> **Note:**
>
> You also use Fusion Middleware Control user interface to start, stop, and view status of system components.

Process control commands provide the following functionality to manage the Oracle Analytics Server system components:

- A command-line interface for advanced users to control Oracle Fusion Middleware components.

  For information, see Use Commands to Start, Stop, and View Status of Processes.
- An integrated way to manage Oracle Analytics Server component processes.

Scripts

Scripts are for advanced administrator use and enable you to programmatically administer Oracle Analytics Server.

The scripting environment is based on the Java scripting interpreter Jython. You use batch scripts that are supplied in a file (scripts invoke a sequence of commands without requiring your input), or embedded in Java code. See About the Scripts for Managing Service Instances. You can extend the scripting language by following the Jython language syntax. See WebLogic Server WLST Online and Offline Command Reference in *WLST Command Reference for Oracle WebLogic Server*.

Model Administration Tool

The Model Administration Tool is a Windows application that enables you to manage the semantic model.

To obtain Model Administration Tool, install Developer Client Tool onto a Windows machine in your environment. See Download, Install, and Start the Client Tools for Oracle Analytics Server.

See Before You Begin in *Metadata Repository Builder's Guide for Oracle Analytics Server*.

Catalog Manager Command Line Interface

The Catalog Manager Command Line Interface is a tool that's the interface with the Presentation Catalog.

To obtain the Catalog Manager Command Line Interface, install the Client Tools onto a machine in your environment. See Download, Install, and Start the Client Tools for Oracle Analytics Server.

You use the Catalog Manager Command Line Interface to manage Presentation Catalogs. See Configure and Manage the Presentation Catalog.

Job Manager Command Line Interface

The Job Manager Command Line Interface is a tool that is the interface with the Oracle BI Scheduler.

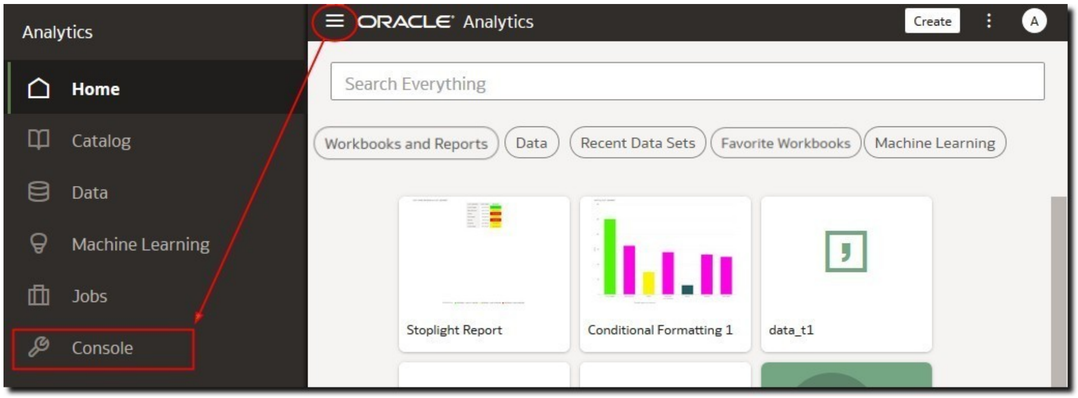
The Job Manager Command Line Interface is installed locally when you install Oracle Analytics Server.

Using the Job Manager Command Line Interface, you can connect to, start and stop the Oracle BI Scheduler, add and manage jobs, and manage job instances. See Perform Advanced Job Management Tasks.

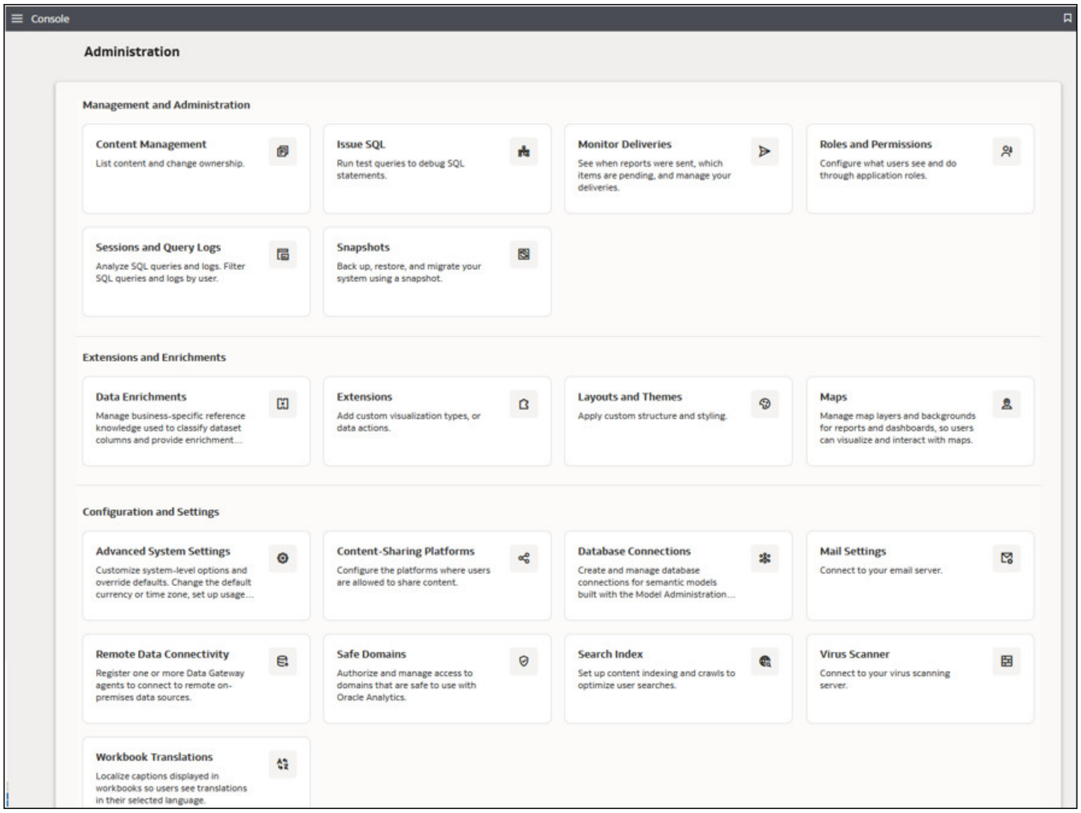
Access the Console in Oracle Analytics Server

Use the Console to manage user permissions, back up everyone's content to a snapshot, perform various configuration and administration tasks, and update system settings.

1. In the Home page, click the **Navigator** bar and click **Console**.



2. Click the option you want to configure.
   You must have the **BI Service Administrator** role to configure Oracle Analytics.



Console

Administration

Management and Administration

Content Management
List content and change ownership.

Issue SQL
Run test queries to debug SQL statements.

Monitor Deliveries
See when reports were sent, which items are pending, and manage your deliveries.

Roles and Permissions
Configure what users see and do through application roles.

Sessions and Query Logs
Analyze SQL queries and logs. Filter SQL queries and logs by user.

Snapshots
Back up, restore, and migrate your system using a snapshot.

Extensions and Enrichments

Data Enrichments
Manage business-specific reference knowledge used to classify dataset columns and provide enrichment...

Extensions
Add custom visualization types, or data actions.

Layouts and Themes
Apply custom structure and styling.

Maps
Manage map layers and backgrounds for reports and dashboards, so users can visualize and interact with maps.

Configuration and Settings

Advanced System Settings
Customize system-level options and override defaults. Change the default currency or time zone, set up usage...

Content-Sharing Platforms
Configure the platforms where users are allowed to share content.

Database Connections
Create and manage database connections for semantic models built with the Model Administration...

Mail Settings
Connect to your email server.

Remote Data Connectivity
Register one or more Data Gateway agents to connect to remote on-premises data sources.

Safe Domains
Authorize and manage access to domains that are safe to use with Oracle Analytics.

Search Index
Set up content indexing and crawls to optimize user searches.

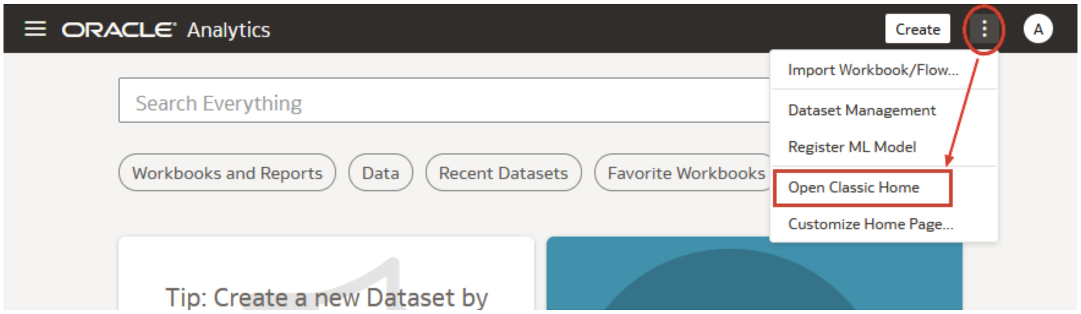
Virus Scanner
Connect to your virus scanning server.

Workbook Translations
Localize captions displayed in workbooks so users see translations in their selected language.

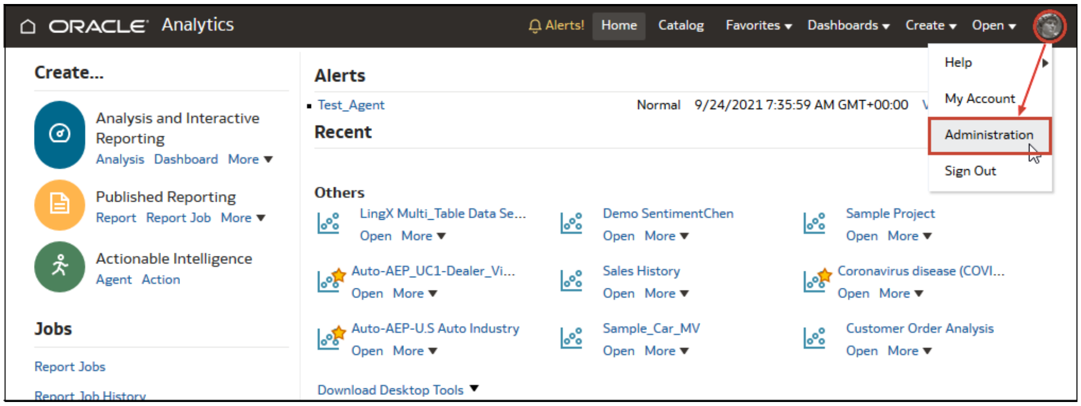
Access the Classic Administration Page

Use the Classic Administration page if you're familiar with on-premises products that use a similar page.

1. In the Home page, click the **Page Menu** and select **Open Classic Home**.



2. Click **My Profile**, and select **Administration**.
   You must have the **BI Service Administrator** role to see the Administration menu.



3. Click the link for the feature you want to configure.

Download, Install, and Start the Client Tools for Oracle Analytics Server

Use Client Tools for Oracle Analytics Server to manage and develop your data models and catalog.

Oracle updates Oracle Analytics Server Client Tools with each annual Oracle Analytics Server release. The Client Tools version you install must match your Oracle Analytics Server version. Using mismatched versions of Oracle Analytics Server and Client Tools leads to incompatibility issues.

You install Client Tools on a Windows machine in your Oracle Analytics Server environment. The Client Tools software pack installs Model Administration Tool and the Catalog Manager Command Line Interface.

1. Download the latest Oracle Analytics Server Client Tools.
   a. Navigate to the download page Download page for Oracle Analytics Server.
   b. Go to the **Optional** section and click **Oracle Analytics Client Installer**.
   c. In the Oracle Software Delivery Cloud page, click **All**. Then in the Software column leave Oracle Analytics Client Installer selected and deselect all other options.
   d. In the Oracle Software Delivery Cloud page, click the license agreement checkbox.
   e. Click **Download**.
2. Install Client Tools.
   a. Navigate to the downloaded installer (.exe) file on your computer.
   b. Run the installer file as Administrator.
3. Start the Client Tools.
   - For Windows, go to the Windows Start menu, click **Oracle Analytics Client Tools**, and select the name of the tool you want to use. For example, to edit your semantic model, click **Model Administration Tool**.
   - For Linux, navigate to the `BI_DOMAIN\bitools\bin` folder and run the `runcat.sh` and `datamodel.sh` command line utilities.

Oracle Analytics Publisher Integration

This guide assumes that Oracle Analytics Server and Publisher have been installed and configured to run as fully integrated components at your organization. If this isn't the case, then some mentions of Publisher in this guide might not be applicable to you.

For information about running Oracle Analytics Publisher, see Introduction to Pixel Perfect Reporting in *Using Oracle Analytics Publisher in Oracle Analytics Server*.

About Oracle Fusion Middleware Feature Support in Oracle Analytics Server

Oracle Analytics Server runs on Oracle Fusion Middleware. Not all Oracle Fusion Middleware capabilities are supported when running Oracle Analytics Server.

For example, due to technical restrictions only a subset of the authenticators supported by Oracle Fusion Middleware are certified against Oracle Analytics Server. The Oracle Analytics Server documentation including Enterprise Deployment Overview in *Enterprise Deployment Guide for Oracle Analytics Server* describes the certified Oracle Fusion Middleware configurations that have been tested with Oracle Analytics Server. Any Oracle Fusion Middleware or Oracle WebLogic Server configuration settings outside of what is described in the Oracle Analytics Server documentation might not be compatible.

Part II

Manage Your Installation

This part explains how to manage your installation offering visualization and business intelligence enterprise modeling services. The information is aimed at administrators whose primary job is to manage users and keep them productive. Administrators perform a long list of critical duties; start and stop the system, set up database connections for data modelers, manage data storage to avoid exceeding storage limits, keep regular backups so users don't risk losing their work, authorize access to external content by registering safe domains, troubleshoot user queries, and so much more.

This part includes the following chapters:

2

Start and Stop Your System

This chapter describes how to start and stop Oracle Analytics Server processes.

Topics:

About Managing Processes

System administrators start and stop system and component processes to perform a range of maintenance operations that require process downtime.

Understanding the state (that is, up, down, starting, and stopping) of each component in the system is an essential activity when diagnosing and resolving availability and performance issues, and when performing life-cycle and management operations. See Diagnose and Resolve Issues.

Oracle Analytics Server runs within Oracle WebLogic Server, and therefore Oracle WebLogic Server must be started before Oracle Analytics Server components can be started and maintained.

To make changes to server configuration settings, the Presentation Catalog, the semantic model (.rpd file offline), and other settings, you must restart the appropriate components before those changes can take effect.

When you stop the system, end users are logged out, and when ready, the system prompts you to log in again, ensuring session state consistency.

For information about the installed components, see Oracle Analytics Server Components.

Conditions to Start the System

The system starts with the Administration Server, then the Managed Servers, and then the system components.

If the computer that hosts the Administration Server isn't running or is rebooted, then you must ensure that the computer is running and you must start the system.

To start the system, the repository database (which contains Scheduler schemas) specified during installation must be running, and a network connection to it must be available. Otherwise, error messages are displayed.

The procedure to start the system differs slightly depending on the platform, as described in the following sections.

Start and Stop the System on Windows

On a Windows installation you can start and stop the system using either the Oracle Analytics Server Windows service or the start and stop commands, but you must avoid mixing both methods.

Using both methods interchangeably risks that the status of the Windows service won't always correctly reflect the status of the system.

On a Windows system, use only one of the following methods at a time to start and stop Oracle Analytics Server:

- Start and Stop commands - `start.cmd` and `stop.cmd`
  See Use Commands to Start, Stop, and View Status of Processes.
- Windows service - Oracle Analytics Server [<*DomainName*>]
  - The Windows service allows processes to automatically start when the machine starts up.
  - You can monitor the progress of system start and stop through the Windows service in `<`*`domain_home`*`>\bilogs\windowsService.log`.
  - When you start Oracle Analytics Server using the Windows service, although the service itself starts within a few seconds, the Oracle Analytics Server stack is initiated in the background and can take up to 20 minutes to start.

Use Commands to Start, Stop, and View Status of Processes

You can control software processes using script commands.

You use script commands to start, stop, and view component status.

- Start Component Processes in a Domain
- Stop Component Processes in a Domain
- View the Status of Components in a Domain

Start Component Processes in a Domain

Learn about how to start all component processes within a domain.

Assumptions

- The start command starts Node Manager locally if not already running.
- The start command runs only from the primary host.
- The start command doesn't complete until component processes are started or fail consecutively to start the specified number of times in the `restartMaxValue` parameter (-`m).`
- Component processes start in order.
- The command initially prompts for credentials and automatically creates a `boot.properties` file, so that subsequent runs don't require credentials.
- You must have file system permissions, and know the boot identity credentials.

- If the system is clustered across multiple servers, then the Node Manager on each server must be manually started (aside from the primary host) by running `DOMAIN_HOME/bin/startNodeManager.sh`. On Windows, don't start Node Manager as a service in a multi-node, clustered deployment because the system doesn't work in this case.

1. Enter an appropriate command to run the start script located in:

   `DOMAIN_HOME/bitools/bin`

   On Linux | Windows:

   ```
   ./start.sh | start.cmd{-noprompt} {-i <list of instances>} {-r
   <restartIntervalSeconds>} {-m <restartMaxValue>} {-r <restartIntervalSeconds>} {-c}

   For example, ./start.sh -i obis1,obips1 -r 10,000 -m 30 -c
   ```

   Use the optional arguments.

   - `-h<Domain home>`
     Use to specify the domain home, includes the *domainName* directory. The default is *DOMAIN_HOME* if set.
   - `-i <startServersList>`
     Use to specify instances to start up in a comma-separated list. You can use the Administration Server, a Managed Server or a system component instance name as the instance.
   - `-r<restartIntervalSeconds>`
     Use to specify the number of seconds during which the system components can be restarted. The default is `3600`, Maximum is *214748647*, Minimum is *300*.
   - `-m<restartMaxValue>`
     Use to specify the number of times that the Node Manager can restart the System Components within the interval specified in Restart Interval in Seconds. The default is *5000*, Maximum is *2147483647*, Minimum is 0. If set to *0*, then auto restart of system components is disabled.

     When you start components using start.sh|cmd and don't use the option -m<restartMaxValue>, the command uses the default value in start.servers.config.json, and sets the maximum number to 5000.
   - `-c<Clear cached credentials>`
     Use to clear and reset the cached Oracle WebLogic Server administrator credentials, prompts for user name and password. Use this argument after you change the default Oracle WebLogic Server administrator password. The password is encrypted and cached for later use in start/stop without user interaction. The default is `false`.

     The Node Manager credentials aren't changed.

   If you don't specify any instances as arguments in the command, the Administration Server, Managed Server, and all system components, start by default.

2. A list of the inactive components to start is displayed.

3. Components start.

   If you don't specify `-i`, then `start` starts all inactive processes. It doesn't fail if something is already running.

   The Administration Server, Managed Servers, local and remote node managers, and system components are started.

   The number of started components is displayed.

   The status of all components is displayed.

Permanently Change the Maximum Number of Attempts for Restarting System Components

You can permanently configure the number of times that the Node Manager tries to restart the System Components.

When System Components started by the Node Manager fail, the process automatically attempts to restart them until the maximum number of restart attempts is reached. The default maximum number of restart attempts is set to 5000, but this default value might be too large in cases when the process keeps failing, and you might want the process to fail sooner to avoid using unnecessary resource consumption in the restart attempts (for example, for CPU, memory, or hard disk generated core files disk consumption). You can permanently decrease the number of times that the Node Manager attempts to restart System Components.

You can permanently configure the startup script to use a maximum restart value that you specify instead of using the default value of 5000. You change the maximum restart value by customizing `max_restart_value` in the startup script.

1. Take a backup of the startup script in `<oas_home>/bi/modules/oracle.bi.sysman/scripts/start.sh|start.cmd`.
2. Open `start.sh|start.cmd` for editing and check the `max_restart_value`.

   When `max_restart_value` is set to -1, it uses the default value 5000, defined in the `start.servers.config.json` file.
3. Specify a new default maximum restart value, for example, set the value to 30 as shown below.

   ```
   # Default System Components Restart values. Indicates that the default
   should be used
   # max_restart_value=-1
   max_restart_value=30
   ```
4. Stop and start the System Components to reflect the change.

> **Note:**
>
> As the administrator, you're responsible for making backups and maintaining customizations. Unless specifically noted, patches to the system might overwrite your customizations. Ensure that you back up any files before editing them, and when editing, comment out original code rather than removing it.

Stop Component Processes in a Domain

Learn about how to stop running component processes within a domain.

Assumptions

- The stop command stops Node Manager locally and remotely on clustered servers in the command.
- The stop command runs only from the primary host.
- The stop command continues until all specified component processes are shut down.

- The stop command initially prompts for credentials and automatically creates a boot identity file, so that subsequent runs don't require credentials.
- Stopping specific process may cause failover, so familiarize yourself with Scale Your Deployment .

Node Manager must be running. The `nodemanager.properties` file must include `QuitEnabled=`*`true`*.

You must have file system permissions, and know the system administrator identity credentials to boot Oracle WebLogic Server.

1. Enter an appropriate command to run the stop script located in:

   `DOMAIN_HOME``/bitools/bin`

   On Linux | Windows:

   ```
   ./stop.sh | stop.cmd {-i <list of instances>} {-c}
   For example, ./stop.sh -i obis1,obips1
   ```

   Use the optional arguments.

   - `-h<`*`Domain home`*`>`
     Use to specify the domain home including the *domainName* directory. The default is *DOMAIN_HOME* if set.
   - `-i <`*`list of instances`*`>`
     Use to specify instances to shut down in a comma-separated list. An instance can be the Administration Server, a Managed Server or a system component instance name.
   - `-c<`*`Clear cached credentials`*`>`
     Use to clear and reset the cached Oracle WebLogic Server administrator credentials, prompts for username and password. Use this argument after changing the default Oracle WebLogic Server administrator password. The password is encrypted and cached, for later use in start/stop, without user interaction. The default is `false`.

     Node Manager credentials aren't changed.

   If you don't specify any instances as arguments in the command, the Administration Server, Managed Server and all system components shut down by default.
2. Components shut down.

View the Status of Components in a Domain

Use the status command to view a report displaying the status of components within a domain.

Assumptions

- The status command reports node manager status.
- The status command only runs from the primary host.
- The status command requires the local node manager process to be running.
- The first run prompts you for credentials, and automatically creates a boot identity file so that subsequent runs don't require credentials.

Prerequisites

You must have file system permissions, and know the boot identity credentials.

1. Enter an appropriate command to run the status script located in:

```
DOMAIN_HOME/bitools/bin
```

On Linux | Windows:

```
./status.sh | status.cmd {-v}

where {-v} is verbose
```

2. The command displays component name, type, status, and machine name.

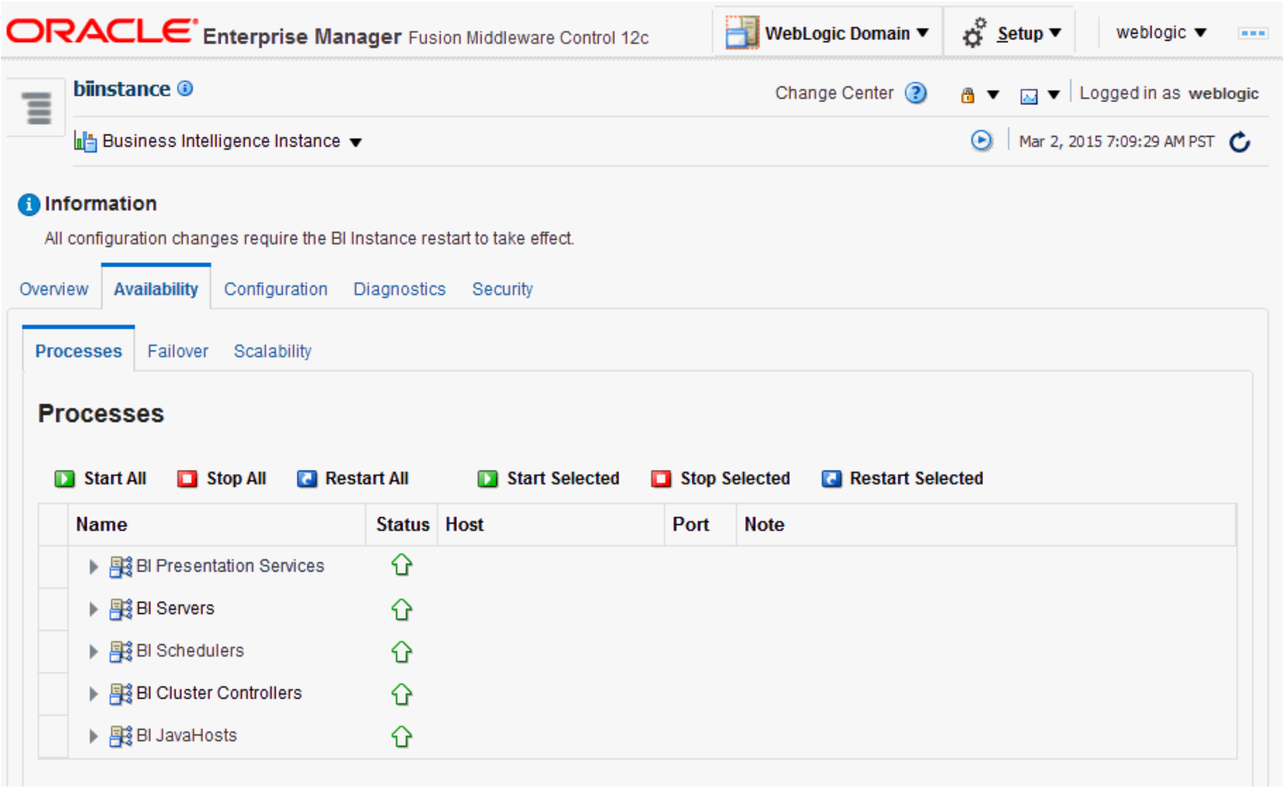
Use Fusion Middleware Control to Start and Stop System Component Processes

If the system processes are running, then you can start, stop, and restart the system component processes, using Fusion Middleware Control.

If Fusion Middleware Control isn't available, then see Use Commands to Start, Stop, and View Status of Processes.

1. Go to the Overview page, as described in Display Oracle Analytics Server Pages in Fusion Middleware Control.
2. Display the Processes tab of the Availability page, then either click **Start All**, or **Stop All**, **Restart All** buttons for all processes, or select a process row and use the appropriate button to start, stop, or restart an individual process as appropriate, as shown in the following illustration.

   You also use this page to view the status of system components.



Name	Status	Host	Port	Note
BI Presentation Services	⇧			
BI Servers	⇧			
BI Schedulers	⇧			
BI Cluster Controllers	⇧			
BI JavaHosts	⇧			

You can use other methods to start and stop processes:

- Use Commands to Start, Stop, and View Status of Processes
- Use Oracle WebLogic Server Administration Console to Start and Stop Java Components

Use Fusion Middleware Control to Start and Stop Java Components

Use this topic to monitor status and start and stop Java components (Administration Server and Managed Servers) using Fusion Middleware Control.

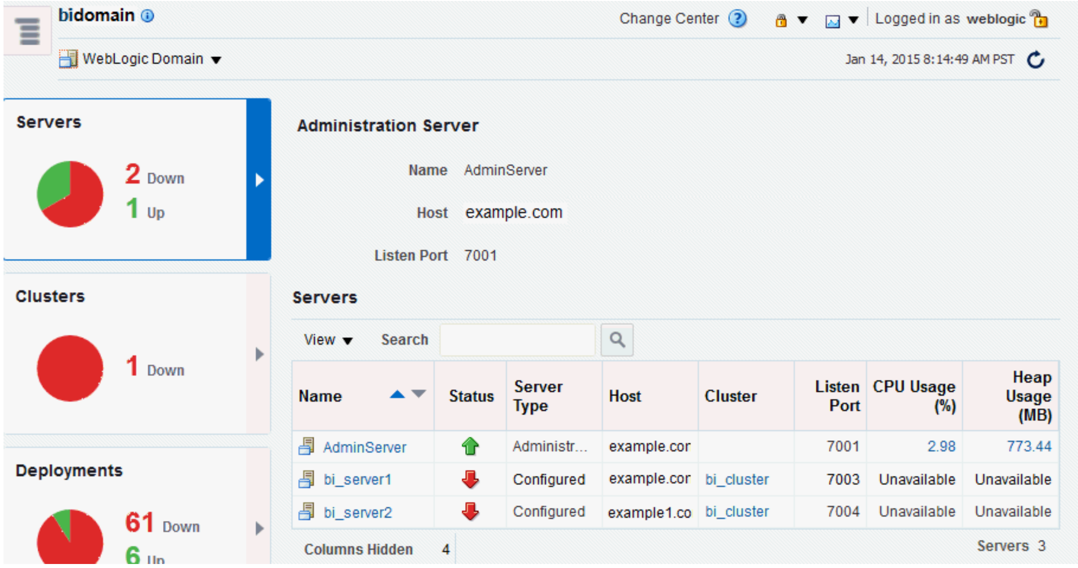
You can also display the WebLogic Server Administration Console to manage Java components by choosing a menu option on the WebLogic Domain menu.

1. Log in to Fusion Middleware Control.

   See Log into Fusion Middleware Control.

2. Expand the **WebLogic Domain** folder and select the bidomain node.

   Fusion Middleware Control displays the WebLogic Domain home page, as shown in the following illustration.



Administration Server

Name AdminServer

Host example.com

Listen Port 7001

Servers

Name	Status	Server Type	Host	Cluster	Listen Port	CPU Usage (%)	Heap Usage (MB)
AdminServer	⬆	Administr...	example.cor		7001	2.98	773.44
bi_server1	⬇	Configured	example.cor	bi_cluster	7003	Unavailable	Unavailable
bi_server2	⬇	Configured	example1.co	bi_cluster	7004	Unavailable	Unavailable

Columns Hidden 4

Servers 3

   The WebLogic Domain home page is the starting point for monitoring status of servers, clusters, deployments, and partitions and for starting and stopping Java components using Fusion Middleware Control. You can also click a menu option to display the WebLogic Server Administration Console, where you can manage and configure Java components. See Manage Java Components in the Oracle WebLogic Server Administration Console.

3. Using the WebLogic Domain home page, you can perform the following management tasks:

   - View the status of Administration Server (AdminServer) and Managed Servers (for example, bi_server1).
   - Start and stop selected Java components (for example, AdminServer or bi_server1) using the WebLogic Domain menu **Control** option.

     See Use Fusion Middleware Control to Start and Stop Java Components.

- Manage or configure the WebLogic server domain using the WebLogic Server Administration Console by clicking a link on the WebLogic Domain menu.

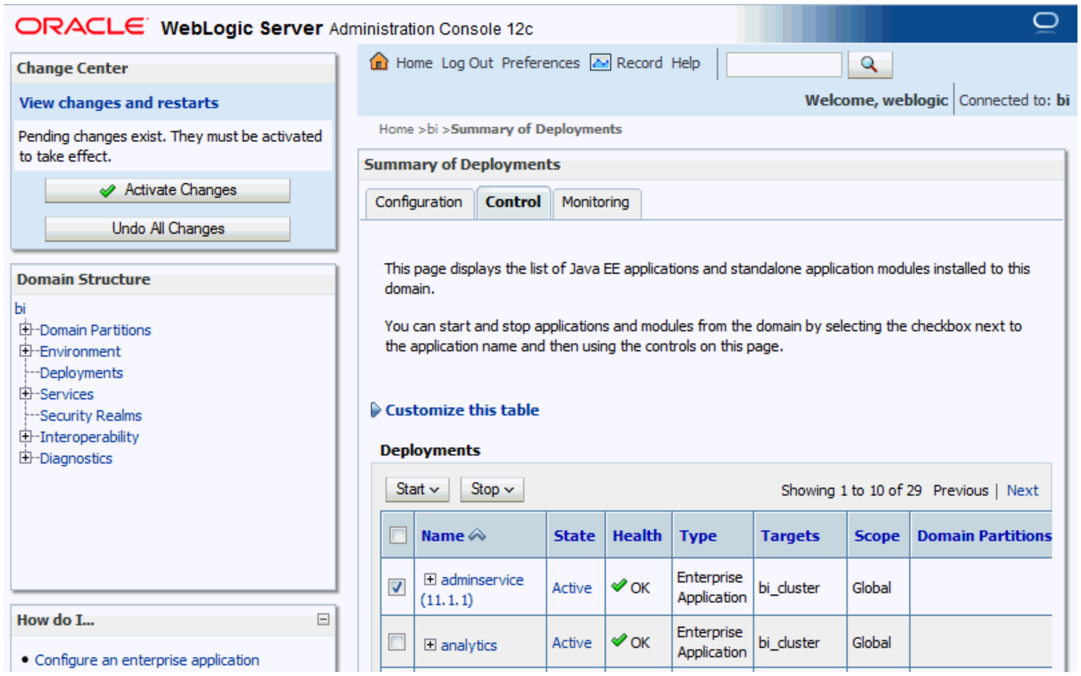
  See Manage Java Components in the Oracle WebLogic Server Administration Console.

Use Oracle WebLogic Server Administration Console to Start and Stop Java Components

In the event that you can't use the standard methods to start and stop Java components, you can use the Oracle WebLogic Server Administration Console.

Don't use Oracle WebLogic Server Administration Console to start and stop Java components. You can use Fusion Middleware Control to start and stop Java components (see Use Fusion Middleware Control to Start and Stop System Component Processes).

1. Start the Oracle WebLogic Server Administration Console.
2. In the Domain Structure region, click **Deployments**.
3. The Oracle WebLogic Server Administration Console displays the Summary of Deployments page.
4. Display the Control tab.
5. Select a check box for each component to start or stop.
6. Click **Start** or **Stop** to start or stop the selected components, as shown here:



Pending changes exist. They must be activated to take effect.

This page displays the list of Java EE applications and standalone application modules installed to this domain.

You can start and stop applications and modules from the domain by selecting the checkbox next to the application name and then using the controls on this page.

□	Name	State	Health	Type	Targets	Scope	Domain Partitions
☑	⊞ adminservice (11.1.1)	Active	✓ OK	Enterprise Application	bi_cluster	Global	
□	⊞ analytics	Active	✓ OK	Enterprise Application	bi_cluster	Global	

3
Take Snapshots and Restore

This topic describes how to back up and restore content using a file called a snapshot or BAR file.

Topics:

- Typical Workflow to Take Snapshots and Restore
- About Snapshots
- Take Snapshots and Restore Information
- Export and Import Snapshots

Typical Workflow to Take Snapshots and Restore

Here are the common tasks to back up and restore your content using snapshots.

Task	Description	More Information
Take a snapshot	Capture content and settings in your environment at a point in time.	Take a Snapshot
Restore from a snapshot	Restore the system to a previously working state.	Restore from a Snapshot
Delete a snapshot	Delete unwanted snapshots.	Delete Snapshots
Download a snapshot	Save a snapshot to a local file system.	Export Snapshots
Upload a snapshot	Upload content from a snapshot that is stored on a local file system.	Import Snapshots

About Snapshots

A snapshot captures the state of your environment at a point in time. Snapshots don’t include data that's hosted on external data sources.

Backup and Restore

Take a snapshot of your environment before people start using the system and again at suitable intervals so you can restore the environment if something goes wrong. You can export and store snapshots on your local file system or cloud storage and import them back to your system if they’re required to restore content. The snapshot file that you download is a compressed archive file (BAR file).

Content Migration

Snapshots are also useful if you want to migrate your content to another environment. For example, you might want to :

- Migrate content you created in a development or test environment to a production environment.

- Migrate content you created in a different Oracle product and exported to a snapshot (BAR file).
  You can generate and migrate BAR files from several Oracle products.
  - Oracle Analytics Cloud
  - Oracle Analytics Server
  - Oracle BI Enterprise Edition

When you restore a snapshot taken from a different environment:

- The snapshot must be taken from an environment at the same version (or earlier version) as the target environment.
  For example, if you take a snapshot of an Oracle Analytics environment that includes the May 2022 update, you can restore it on other Oracle Analytics environments that include the May 2022 update or a later update (such as July 2022). You can't restore this snapshot on an Oracle Analytics environment that includes an earlier update, such as March 2022.
- Catalog objects that your target environment doesn’t support aren't migrated.
- In most cases, you must upload the data associated with your datasets on the target environment.

Exclusions

There are a few items that aren't included in a snapshot:

- Data files - XLSX, XLS, CSV, or TXT files that users upload to create datasets. You can include references to data files but not the actual files.
- Map layers and backgrounds - Custom map layers and map backgrounds that users upload to enhance their visualizations and reports.
- Snapshot list - The list of snapshots that you see on the Snapshot page.

Options When You Take a Snapshot

When you take a snapshot you choose the content you want to include in it. You can take a snapshot of your entire environment (everything) or specify only specific content that you want to back up or migrate (custom).

- **Everything** - Saves your environment in the snapshot. This option is useful if you want to:
  - Back up everything in case something goes wrong.
  - Migrate everything to a new environment.
  - Clone an existing environment.

  File-based data, plug-ins, or extensions that users upload aren’t included in the snapshot.
- **Custom** - You select which content to save in the snapshot. Some content types are always included while others are optional.

Snapshot Option	Description	Optional?
Data	Data visualization content that users create (Data tab).	
– Datasets	Datasets that users create for data visualizations and data flows.	Always included

Snapshot Option	Description	Optional?
– Connections	Data connections that users create so they can visualize their data.	Always included
– Data Flows	Data flows that users create for data visualization.	Always included
– Sequences	Sequences that users create for data visualization.	Always included
Machine Learning	Machine learning models that users create from data flows.	Always included
Jobs	Jobs that users schedule for data flows, sequences, and pixel-perfect reports.	Optional
Configuration and Settings	Service configuration and settings configured through the Console. For example, mail settings, safe domains, and so on. **Note**: System settings aren't included in the snapshot.	Optional
Application Roles	– User-defined application roles that administrators create through the Console. – Membership details for each application role, that is, the users, groups, and other application roles assigned to each application role.	Always included
Credentials	– **Data connections:** Credentials and other connection parameters, such as host, port, user name, and password. If you exclude credentials, you must reconfigure the connection details after you restore the snapshot. – **Cloud storage:** Credentials required to access cloud storage where file-based data that users upload is stored. If you include file-based data in your snapshot, include the storage credentials if you plan to migrate the content to another environment. If you exclude credentials, you can use the Data Migration utility to download and then upload your data files separately.	Optional
Classic Content	Content that users create using Classic tools, such as analyses, dashboards, and pixel-perfect reports.	Always included

Snapshot Option	Description	Optional?
– Semantic Model and Subject Areas	Semantic models and subject areas that users create.	Always included
– Catalog Content	Catalog containing content that users create and save for future use, such as analyses, dashboards, reports, deliveries, agents, and so on.	Always included
– Shared Folders and Workbooks	Content that is being shared, that is, content that everyone with access to Classic tools can see. This includes any data visualization workbooks saved in the shared folders.	Always included
– User Folders and Workbooks	Content stored in user folders. Content that users create and store for their private use. This includes any data visualization workbooks that users saved in their private folders.	Optional

Options When You Restore a Snapshot

When you restore content from a snapshot you have several options. You can restore only the content that's inside the snapshot, restore everything in your environment, or restore a specific set of items in the snapshot (custom).

- **Replace Snapshot Content Only** - Everything in the snapshot that's supported in your environment is restored. Any content type excluded from the snapshot remains unchanged in your environment.
- **Replace Everything** - Replaces your entire environment using information in the snapshot.
  Any content type excluded from the snapshot is restored to its default state, that is, "no content". For example, if you chose not to include jobs in the snapshot, any jobs that exist on your system are deleted when you restore the snapshot and the jobs feature is restored with default settings. There are some exceptions; any file-based datasets, plug-ins, or extensions in your environment remain unchanged.

  This option is useful if you want to:

  - Replace everything after something went wrong.
  - Migrate from another service.
  - Clone an existing service.
- **Custom** - You select the content you want to restore. If you don’t want to restore certain content types, exclude them before you restore.
  In most cases, the options on restore are the same as the options when you take a snapshot. Some content types are always restored, while others are optional.

> **Note:**
>
> When you restore *catalog content* from a snapshot, delivery schedules aren’t automatically restored or activated. This is so you can restore and activate deliveries at a time that suits you. See Restore and Enable Delivery Schedules.

If your snapshot contains items that your environment doesn’t support, you see the message "*Not supported in this environment*".

Restoring a Snapshot Taken from a Different Product

You can take snapshots in several Oracle products; Oracle BI Enterprise Edition 12c, Oracle Analytics Cloud, and Oracle Analytics Server.

- **Unsupported Content**

  If you take a snapshot in one product and try to restore it in a different Oracle product, you might find the snapshot contains some items that the target environment doesn’t support. When Oracle Analytics detects unsupported content, warning icons display on the Custom page to highlight unsupported items in the snapshot that won't be restored.

  Not supported in this environment.

  For example, you take a snapshot in Oracle Analytics Cloud and include data replications, file-based datasets, plug-ins and extensions in the snapshot. When you restore the snapshot in Oracle Analytics Server, you notice that these items are marked *not supported*. Oracle Analytics Server doesn’t allow you to include data replications, file-based datasets, plug-ins and extensions in an Oracle Analytics Server snapshot or import them from snapshots you created in other products.

- **Granting Permissions to Access New Features in Oracle Analytics Server**

  If you restore a snapshot taken from Oracle BI Enterprise Edition 12c (or 11g), Oracle Analytics Server grants additional permissions to your existing application roles so that users automatically get access to some new features in Oracle Analytics Server. After restoring your application roles from the snapshot, you can fine-tune permission assignments to suit your users using two scripts: `grantPermissionSetsToBIRole` and `revokePermissionSetsFromBIRole`. See Grant or Revoke Permissions to Perform Tasks in Oracle Analytics Server.

Take Snapshots and Restore Information

You can take a snapshot of your system at any time.

Topics:

- Take a Snapshot
- Restore from a Snapshot
- Track Who Restored What and When
- Edit Snapshot Descriptions
- Delete Snapshots

Take a Snapshot

Administrators can take a snapshot of the system at any time.

1. Click **Console**.
2. Click **Snapshots**.
3. Click **Create Snapshot**.
4. Enter a short description for the snapshot to help you remember later why you took it.

   For example, why you created the snapshot and what it contains.
5. Select the content you want to include, **Everything** or **Custom**.
   - **Everything** - Include everything about your environment in the snapshot except for file-based data, plug-ins, and extensions.
   - **Custom** - Select only the content types you want to save in the snapshot. Deselect any items that you don't want.
6. Click **Create**.

The latest content is saved to a snapshot.

Restore from a Snapshot

If something goes wrong, you can easily restore your content to a previous working state from a snapshot. You also restore snapshots when you migrate content between environments.

Before you start, read these tips about restoring snapshots.

- As you start to restore the snapshot, users currently signed in have their session terminated.
- After you restore from a snapshot, allow time for the restored content to refresh (for example, approximately 15 to 30 minutes for a large snapshot).
- Delivery schedules aren’t automatically restored or activated when you restore *catalog content* from a snapshot. This is so you can restore and activate deliveries at a time that suits you.
- You can restore snapshots taken at the same version (or earlier version) as the target environment.

  You might experience unexpected results if you try to restore from a snapshot that was taken from a more recent update of Oracle Analytics.
- When you restore a snapshot taken from a different environment, you must upload the data associated with your file-based datasets to the target environment.
- If you restore a snapshot taken from Oracle BI Enterprise Edition 12c (or 11g), Oracle Analytics Server grants additional permissions to your existing application roles so that users automatically get access to new features in Oracle Analytics Server. After restoring the snapshot, you can fine-tune permission assignments to suit your users using two scripts: `grantPermissionSetsToBIRole` and `revokePermissionSetsFromBIRole`. See Grant or Revoke Permissions to Perform Tasks in Oracle Analytics Server.
- If you installed Oracle Analytics Server 2024 or later and then you import a snapshot from an earlier version of Oracle Analytics Server, you must grant the BI Data Model Author application role the `semanticmodeler.author` and `obis.administrator` permissions sets.

This is so the users with the application role can access Semantic Modeler. See Grant Semantic Modeler Permissions Assignments.

To restore a snapshot:

1. Click **Console**.
2. Click **Snapshots**.
3. Select the snapshot that you want to use to restore your system.
4. Click **Snapshot Actions** ⋮.
5. Click **Restore** to return your system to the state when this snapshot was taken.
6. In the Restore Snapshot dialog, select only those elements you want to restore.

   For example, you may not want to include application roles if you're restoring a snapshot taken from a pre-production environment, to a production environment. Pre-production roles often have different members to the production environment. If so, select **Custom** and deselect **Application Roles** before you restore.

   a. Select the **Restore** option you want.

      - **Replace Snapshot Content Only -** Replace all the content types included in snapshot (listed in the description field) with the content inside the snapshot. The restore process replaces entire content types on the target. For example, if your target includes workbooks A and B and the snapshot contains workbook A, only workbook A will exist on the target after you restore the snapshot.

        Select this option if you don't want to replace or remove any other content types that exist on the target, that is, only replace the content types inside the snapshot.

      - **Replace Everything -** Overwrite all your existing content. Replace your existing content with the content included in this snapshot (listed in the description field).

        Any content types not included in the snapshot, excluding file-based datasets, plug-ins and extensions, are removed and restored with default settings.

      - **Custom -** Select only the content types you want to restore. You can restore with content saved inside the snapshot or restore content with default settings if that content is missing from the snapshot.

        – Content saved inside the snapshot is listed in the description field.

        – Content not included in the snapshot is marked with a warning icon ⚠. Only restore content marked with a warning icon if you want to restore that content with default settings.

        If you don't want to restore everything, deselect all the items you want to keep.

   b. If you select **Custom**, select only those items you want to restore.

7. For auditing purposes, enter the reason why you're restoring.

   It's good practice to include a restore reason. Later on you might want to analyze the restore history, and this information can help you remember why you restored the snapshot.

8. Click **Restore**.

   A warning message is displayed because restoring a snapshot can be very disruptive.

9. Click **Yes** to restore the selected snapshot, or click **No** to abandon the restore.

10. Wait for the restore to complete, and then wait a few more minutes for the restored content to refresh through your system.

    The time it takes to restore your system depends on the size of your snapshot. For a large snapshot, allow approximately 15 to 30 minutes.
11. Sign out and then sign back in to see the restored content and inherit newly restored application roles, if any.

Track Who Restored What and When

You can check the restore history to learn exactly when and what content was restored, and to check for any errors during the restore process. This might be useful if you experience issues during or after you restore a snapshot.

1. Click **Console**.
2. Click **Snapshots**.
3. Click the Page menu ⋮ and select **Show Restore History**.

Edit Snapshot Descriptions

You can add or update the description for any snapshot.

1. Click **Console**.
2. Click **Snapshots**.
3. Select the snapshot you want to edit.
4. Click **Snapshot Actions** ⋮.
5. Click **Edit Name**.
6. Update the description, and click **OK**.

Delete Snapshots

From time to time, delete snapshots that you don’t need.

1. Click **Console**.
2. Click **Snapshots**.
3. Select the snapshot that you want to delete.
4. Click **Snapshot Actions** ⋮.
5. Click **Delete** to confirm that you want to delete the snapshot.

Export and Import Snapshots

You can save snapshots to your local file system and import them. Exporting and importing snapshots enables you to back up and restore your content or migrate content between development, test, and production environments.

Topics:

- Export Snapshots
- Import Snapshots

Export Snapshots

Use the export option to save a snapshot to your local file system. Exporting allows you to store and manage any snapshots you might take of your system.

The snapshot exports as an archive file (`.bar`). The time it takes to export your snapshot depends on the size of the `.bar` file.

If you haven't taken the snapshot yet, you'll need to do that first.

1. Click **Console**.
2. Click **Snapshots**.
3. Select the snapshot that you want to export.
4. Click **Snapshot Actions** ⋮.
5. Click **Export**.
6. In **Snapshot Password**, enter and confirm a password for the snapshot.

   The password must be between 14 and 50 characters long and contain at least one numeric character, one uppercase letter, and one lowercase letter.

   Don't forget this password. You'll be asked for this password when you try to import the file in the future. For example, if want to restore or migrate the content stored in the snapshot.
7. Click **Export**.

   The time it takes to export depends on the size of the file.

Import Snapshots

You can import a snapshot that you previously saved on your local file system. The time it takes to import the snapshot depends on the size of the snapshot `.bar` file.

When you import a snapshot, the file itself is uploaded to your system but the artifacts stored inside the snapshot aren't immediately available in your environment. Snapshots you import display in the snapshot list. When you're ready to do so, you can overwrite your current artifacts, such as your catalog, by restoring the snapshot.

1. Click **Console**.
2. Click **Snapshots**.
3. Click the **Page actions** menu ⋮ and select **Import Snapshot**.

4. Click **Select** to locate the snapshot that you want to upload.

   Select the snapshot file (`.bar`) that contains your snapshot. You can upload snapshots taken from Oracle Analytics Server and Oracle Analytics Cloud excluding file-based data, plug-ins, and extensions.

5. Enter the snapshot password.

   This is the password that you specify whenever you export a snapshot to your local file system.

   > **Note:**
   >
   > If you enter an incorrect password too many times, the system automatically locks the snapshot as a safety precaution. Wait for approximately 30 minutes, then try to upload the snapshot again with the correct password.

6. Click **Import**.

4 Perform Common Administration Tasks

This chapter describes common tasks performed by administrators managing Oracle Analytics Server. Read the first topic to see the full list.

Topics:

- Typical Workflow of Common Administration Tasks
- Configure a Virus Scanner
- Register Safe Domains
- Integrate with Content-Sharing Platforms for Sharing Visualizations
- Delete Unused Datasets
- Monitor Users and Activity Logs
- Run Test SQL Queries
- Apply Custom Logos and Dashboard Styles
- Enable Custom Java Script for Actions
- Switch to a Different Language
- Configure Advanced System Settings
- Manage Content
- Enable Users to Export to Excel

Typical Workflow of Common Administration Tasks

Here are the common tasks for administrators managing data visualization and enterprise modeling services.

Task	Description	More Information
Back up and restore content	Back up and restore the data model, catalog content, and application roles using a file called a snapshot.	Take Snapshots and Restore
Set up virus scanning	Connect to your virus scanning server.	Configure a Virus Scanner
Set up social channels for content sharing	Enable users to share content on Twitter, Slack, Oracle Cloud Storage, and Oracle Content Management Service.	Integrate with Content-Sharing Platforms for Sharing Visualizations
Set up email deliveries	Connect to your email server.	Configure Email and Agents Track the Reports You Distribute By Email or Through Agents

Task	Description	More Information
Enable agents to deliver content	Allow users to use agents to deliver their content.	Enable and Customize Content Delivery Through Agents Suspend and Resume Deliveries Restore and Enable Delivery Schedules
Manage the types of devices that deliver content	Configure devices for your organization.	Manage the Types of Devices that Deliver Content
Free up storage space	Delete data sources on behalf of other users to free up storage space.	Delete Unused Datasets
Manage how content is indexed and searched	Set up how content is indexed and crawled so users always find the latest information when they search.	Manage How Content Is Indexed and Searched
Manage maps	Manage map layers and background maps.	Manage Map Information For Analyses Make Maps Available to Users in *Visualizing Data in Oracle Analytics Server* Make Map Backgrounds Available to Users in *Visualizing Data in Oracle Analytics Server* Add Custom Map Layers in *Visualizing Data in Oracle Analytics Server* Update Custom Map Layers in *Visualizing Data in Oracle Analytics Server*
Register safe domains	Authorize access to safe domains.	Register Safe Domains
Manage session information	Monitor who is signed in and troubleshoot issues with analyses by analyzing the SQL queries and logs.	Monitor Users and Activity Logs
Change the default reporting page and dashboard styles	Change the default logo, page style, and dashboard style.	Apply Custom Logos and Dashboard Styles
Localize reporting dashboards and analyses	Localize the names of catalog objects (known as captions) into different languages.	Localize Presentation Catalog Captions
Track usage	Track the user-level queries to the content in Oracle Analytics Server.	Track Usage
Set up write-back	Enable users to update data from analyses and dashboards.	Enable Write-back in Analyses and Dashboards
Set up custom JavaScript for actions	Enable users to invoke browser scripts from analyses and dashboards.	Enable Custom Java Script for Actions
Configure advanced options	Enables administrators to set more advanced service-level options using the System Settings page in the Console.	About System Settings

Configure a Virus Scanner

To keep Oracle Analytics virus-free, Oracle highly recommends that you set up the virus scanning servers used by your organization, to scan any files that are uploaded to Oracle Analytics. Once set up, all files are checked. This includes data files that users might upload for analysis, and snapshots that you might upload to restore content or migrate content from another environment.

Oracle supports virus scanners that use the Internet Content Adaptation Protocol (ICAP) protocol to communicate.

1. In the Oracle Analytics Home page, click the **Navigator**, and then click **Console**.
2. Click **Virus Scanner**.
3. Enter the host and port of the virus scanning server.

   For example, `my.virus.scanning.serverexample.com`.
4. Click **Save**.
5. To remove the current virus scanner configuration, click **Delete**.

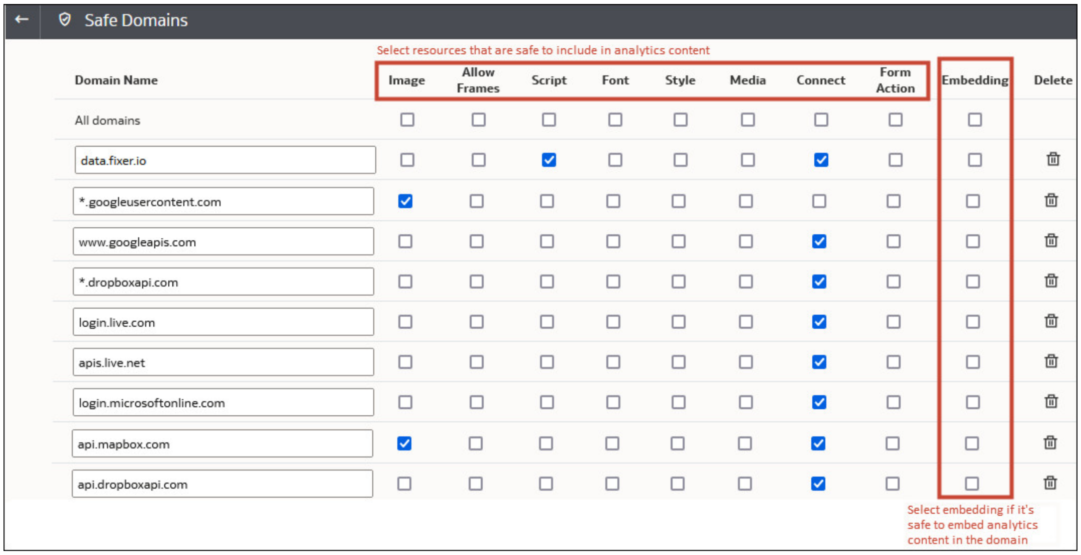
Register Safe Domains

You can register or approve access to specific content. For security reasons, you’re not allowed to add external content to reports or embed your reports in other applications unless your administrator considers it safe to do so. Only administrators can add safe domains to the list.

Only authorized users may access the content. Users are prompted to sign in when they access content on these safe domains, unless your service is set up with Single Sign On (SSO).

After you've added a safe domain to the list, users need to sign out and sign back in to access content from that source.

1. Click **Console**.
2. Click **Safe Domains**.
3. Click **Add Domain** to register a safe domain.
4. Enter the name of the safe domain. Use formats such as:
   - www.example.com
   - *.example.com
   - https:
5. Specify the types of resources users can embed in their projects, reports, and dashboards.
   - Select the types of resources you want to allow, for example, images, scripts, and so on.
   - Deselect to block any resource types you don't consider safe.
6. If you want to allow users to embed their projects, reports, and dashboards in external content located on the domain, select **Embedding**.



Select resources that are safe to include in analytics content

Domain Name	Image	Allow Frames	Script	Font	Style	Media	Connect	Form Action	Embedding	Delete
All domains	□	□	□	□	□	□	□	□	□	
data.fixer.io	□	□	☑	□	□	□	☑	□	□	🗑
*.googleusercontent.com	☑	□	□	□	□	□	□	□	□	🗑
www.googleapis.com	□	□	□	□	□	□	☑	□	□	🗑
*.dropboxapi.com	□	□	□	□	□	□	☑	□	□	🗑
login.live.com	□	□	□	□	□	□	☑	□	□	🗑
apis.live.net	□	□	□	□	□	□	☑	□	□	🗑
login.microsoftonline.com	□	□	□	□	□	□	☑	□	□	🗑
api.mapbox.com	☑	□	□	□	□	□	☑	□	□	🗑
api.dropboxapi.com	□	□	□	□	□	□	☑	□	□	🗑

Select embedding if it's safe to embed analytics content in the domain

7. To remove a domain, select it and click the **Delete** icon.

Add Custom Knowledge for Data Enrichment

Add custom knowledge to Oracle Analytics to augment the system knowledge. For example, you might add a custom knowledge reference that classifies prescription medication into USP drug categories Analgesics or Opioid.

Tutorial

Custom knowledge enables the Oracle Analytics semantic profiler to identify more business-specific semantic types and make more relevant and governed enrichment recommendations.

Before you start, download your custom knowledge reference files (in CSV format) and make them available locally for upload. The maximum file size you can upload is 250 MB. You can also create your own custom knowledge reference files in CSV or XLSX format. See Custom Knowledge Recommendations.

1. In the Oracle Analytics Home page, click the **Navigator**, and then click **Console**.
2. Click **Data Enrichments**.
3. Under **Custom Knowledge**, click **Add Custom Knowledge**.
4. In the Open dialog, navigate to and select your custom knowledge CSV file, then click **Open**.
5. In the Create Custom Knowledge from dialog, specify a name, verify the upload options, then click **OK**.

   The Custom Knowledge page lists the new file with the **Include** option selected. When content authors enrich datasets, Oracle Analytics presents enrichment recommendations based on this data.

Working with Digit-only Keys

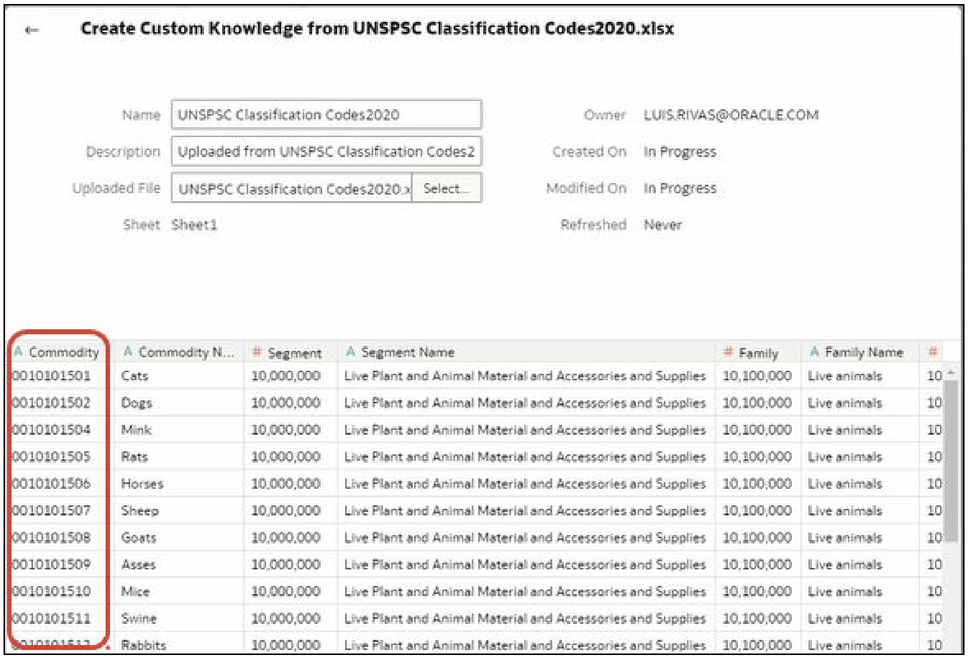
When you add Custom Knowledge to Oracle Analytics, you might want to profile digit-only or numeric keys without removing leading zeros, which is how Oracle Analytics usually ingests numbers. For example, you might want Oracle Analytics to ingest the UNSPSC classification code 0010101501 as 0010101501 (that is, retaining "00" at the start of the code) rather than

10101501. By retaining the full key in Reference Knowledge, workbook designers can access recommendations to enrich their data, which in this example provides UNSPSC data such as name, family, and class.

Tips on Adding Digit-only Keys

In the source file, define the key column as text and make it the first column. You don't have to change the format of the other columns in the file.

For example, in the UNSPSC classification codes dataset the Commodity column holds the key identifier for each row. The Commodity keys are numbers with leading zeros. Oracle Analytics treats the values in the Commodity column as an attribute.

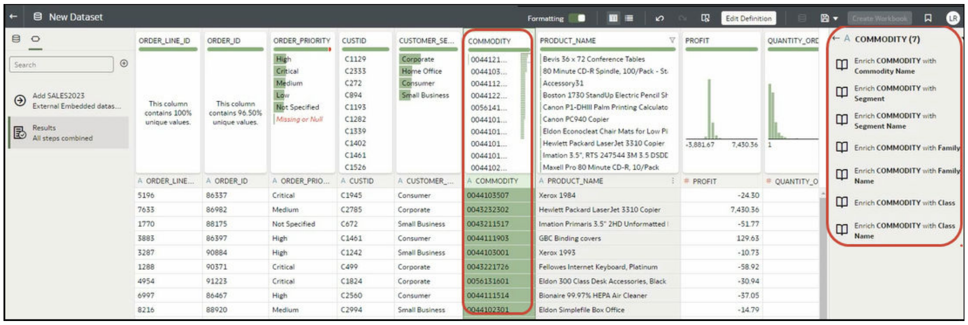


Create Custom Knowledge from UNSPSC Classification Codes2020.xlsx

Name	UNSPSC Classification Codes2020	Owner	LUIS.RIVAS@ORACLE.COM
Description	Uploaded from UNSPSC Classification Codes2	Created On	In Progress
Uploaded File	UNSPSC Classification Codes2020.x	Modified On	In Progress
Sheet	Sheet1	Refreshed	Never

Commodity	Commodity N...	Segment	Segment Name	Family	Family Name	#
0010101501	Cats	10,000,000	Live Plant and Animal Material and Accessories and Supplies	10,100,000	Live animals	10
0010101502	Dogs	10,000,000	Live Plant and Animal Material and Accessories and Supplies	10,100,000	Live animals	10
0010101504	Mink	10,000,000	Live Plant and Animal Material and Accessories and Supplies	10,100,000	Live animals	10
0010101505	Rats	10,000,000	Live Plant and Animal Material and Accessories and Supplies	10,100,000	Live animals	10
0010101506	Horses	10,000,000	Live Plant and Animal Material and Accessories and Supplies	10,100,000	Live animals	10
0010101507	Sheep	10,000,000	Live Plant and Animal Material and Accessories and Supplies	10,100,000	Live animals	10
0010101508	Goats	10,000,000	Live Plant and Animal Material and Accessories and Supplies	10,100,000	Live animals	10
0010101509	Asses	10,000,000	Live Plant and Animal Material and Accessories and Supplies	10,100,000	Live animals	10
0010101510	Mice	10,000,000	Live Plant and Animal Material and Accessories and Supplies	10,100,000	Live animals	10
0010101511	Swine	10,000,000	Live Plant and Animal Material and Accessories and Supplies	10,100,000	Live animals	10
0010101512	Rabbits	10,000,000	Live Plant and Animal Material and Accessories and Supplies	10,100,000	Live animals	10

When workbook designers add data based on this custom knowledge, the enrichment recommendations are appropriate for the data. In this example, enrichment recommendations for UNSPSC classification codes in the Commodity column enable you to enrich your visualization with commodity data such as name, family, and class.



ORDER_LINE...	ORDER_ID	ORDER_PRIO...	CUSTID	CUSTOMER_...	COMMODITY	PRODUCT_NAME	PROFIT	QUANTITY_O
5196	86337	Critical	C1945	Consumer	0044103507	Xerox 1984	-24.30	
7633	86982	Medium	C2785	Corporate	0043232302	Hewlett Packard LaserJet 3310 Copier	7,430.36	
1770	88175	Not Specified	C672	Small Business	0043211517	Imation Primaris 3.5" 2HD Unformatted I	-51.77	
3883	86397	High	C1461	Consumer	0044111903	GBC Binding covers	129.63	
3287	90884	High	C1242	Small Business	0044103001	Xerox 1993	-10.73	
1288	90371	Critical	C499	Corporate	0043221726	Fellowes Internet Keyboard, Platinum	-58.92	
4954	91223	Critical	C1824	Corporate	0056131601	Eldon 300 Class Desk Accessories, Black	-30.94	
6997	86467	High	C2560	Consumer	0044111514	Bionaire 99.97% HEPA Air Cleaner	-37.05	
8216	88920	Medium	C2994	Small Business	0044102301	Eldon Simplefile Box Office	-14.79	

COMMODITY (7)

- Enrich COMMODITY with Commodity Name
- Enrich COMMODITY with Segment
- Enrich COMMODITY with Segment Name
- Enrich COMMODITY with Family
- Enrich COMMODITY with Family Name
- Enrich COMMODITY with Class
- Enrich COMMODITY with Class Name

Integrate with Content-Sharing Platforms for Sharing Visualizations

Integrate with content-sharing platforms and social channels (such as Slack and X) so it's easy for users to share their visualizations with others.

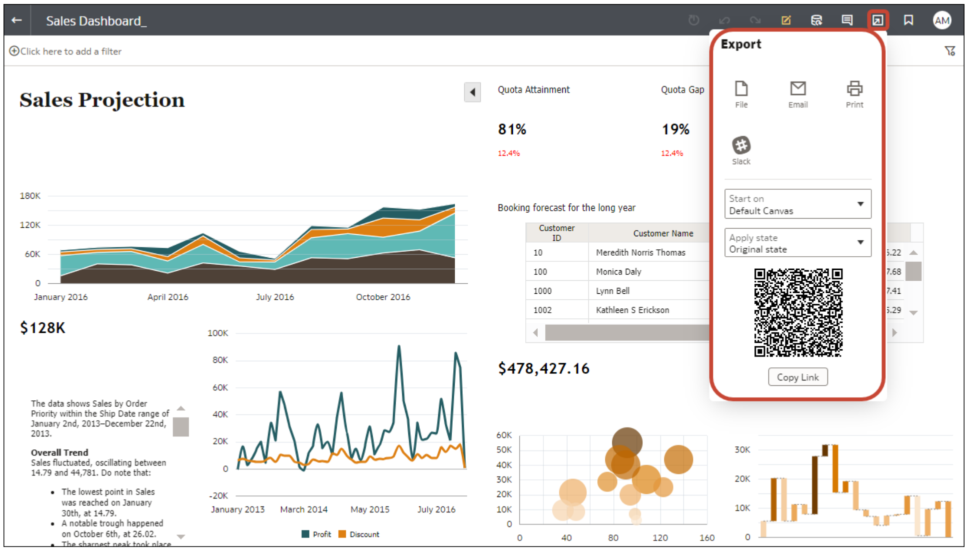
Topics:

- About Sharing Visualizations on Other Platforms
- Enable Workbook Users to Share Visualizations on Slack
- Enable Workbook Users to Share Visualizations on X (formerly Twitter)

About Sharing Visualizations on Other Platforms

If your organization uses content-sharing platforms and social channels such as X or Slack, administrators can configure access to such platforms so it's easy for content authors to share their data visualizations with others.

Once set up, additional content-sharing options are listed on the **Export** dialog for visualizations. For example, if you configure and activate Slack, users see an option to export their visualization to Slack when they click the **Export** icon.



The data shows Sales by Order Priority within the Ship Date range of January 2nd, 2013–December 22nd, 2013.

Overall Trend
Sales fluctuated, oscillating between 14.79 and 44,781. Do note that:

- The lowest point in Sales was reached on January 30th, at 14.79.
- A notable trough happened on October 6th, at 26.02.
- The sharpest peak took place

Customer ID	Customer Name
10	Meredith Norris Thomas
100	Monica Daly
1000	Lynn Bell
1002	Kathleen S Erickson

Start on: Default Canvas

Apply state: Original state

Some platforms are displayed in inactive mode by default, for example, Slack, and others are hidden by default. When you set up a content-sharing platform, you can set the status to one of the following:

Status	Description
Active	Display the content-sharing platform on the Export dialog and enable users to share content using it. For example, you might display Slack as an Export option.

Status	Description
Inactive	Display the content-sharing platform on the Export dialog, for example, Slack, but don't enable users to share content using it. When users select an inactive option, they see a message that advises them to contact their administrator.
Hidden	Don't display the content-sharing platform on the Export dialog, whether it's configured or not. For example, you might configure it ready for rollout but keep it hidden until a future date.

Enable Workbook Users to Share Visualizations on X (formerly Twitter)

Administrators can set up an X (formerly Twitter) channel in Oracle Analytics, so that content authors can share their data visualizations as a tweet on their organization's X feed.

To enable Oracle Analytics to share data visualization workbooks through your organization’s X app:

1. Obtain the client ID and client secret values for the X app that you want to use to share data visualizations.
    a. Open X Application Manager, for example `developer.twitter.com`.
    b. Click the app that you want to use for tweets.
    c. On the **Keys and Tokens** tab, obtain the **Consumer Key** and **Consumer Secret Key** values.
    d. On the **Permissions** tab, select **Read, write, and direct messages**.
2. Configure the X channel in Oracle Analytics.
    a. In the Oracle Analytics Home page, click the **Navigator**, click **Console**, and then click **Content-Sharing Platforms**.
    b. For **Service**, select **X**.
    c. Change **Status** to **Active**.
    d. For **Application Name**, enter the name of the app that you set up in X Application Manager.
    e. For **Client ID** and **Client Secret**, enter the Consumer Key and Consumer Secret values that you obtained in X Application Manager (Step 1).
    f. Click **Update**.
    g. Click **Copy to Clipboard** to copy the redirect URL for Oracle Analytics.
3. In X Application Manager, configure the callback URL for Oracle Analytics.
    a. In X Application Manager, click the app to use for tweets.
    b. On the App Details tab, click **Edit** and paste the clipboard content in the **Callback URL** field.
    c. Click **Save**.
4. Verify you can share a visualization on the X channel.
    a. In Oracle Analytics, open a workbook.
    b. On the Visualize or Narrate canvas, click the **Export** icon.
    c. Click **Twitter**.

If you set up and activate the channel correctly, **Twitter** displays as an option on the **Export** menu.

Enable Workbook Users to Share Visualizations in LinkedIn

Administrators can set up a LinkedIn channel in Oracle Analytics, so that content authors can share their data visualizations in the organization's LinkedIn feed.

Before you start, make sure that you have a public web storage container on Oracle Cloud that Oracle Analytics can use to share visualizations on LinkedIn. See Set Up a Public Container to Share Visualizations.

1. Obtain the client ID and client secret values for the LinkedIn app that you want to use to share data visualizations.
   a. Open LinkedIn Developer Portal, that is `linkedin.com/developers/apps`.
   b. Click the app that you want to use.
   c. On the Authentication page, obtain the **Client ID** and **Client Secret** values.
2. Configure the LinkedIn channel in Oracle Analytics.
   a. In the Oracle Analytics Home page, click the **Navigator**, click **Console**, and then click **Social**.
   b. For **Service**, select **LinkedIn**.
   c. Change **Status** to **Active**.
   d. For **Application Name**, enter the name of the app that you set up in LinkedIn Developer Portal.
   e. For **Client ID** and **Client Secret**, enter the values that you obtained in LinkedIn Developer Portal (Step 1).
   f. Click **Update**.
   g. Click **Copy to Clipboard** to copy the redirect URL for Oracle Analytics.
3. In LinkedIn Developer Portal, configure the redirect URL for Oracle Analytics.
   a. Select the app that you want to use.
   b. On the App Details tab, click **Edit** and paste the clipboard content in the **Authorized Redirect URLs** field.
   c. Click **Save**.
4. Verify you can share a visualization on the LinkedIn channel.
   a. In Oracle Analytics, open a workbook.
   b. On the Visualize or Narrate canvas, click the **Export** icon.
   c. Click **LinkedIn**.

   If you set up and activate the channel correctly, **LinkedIn** displays as an option on the **Export** menu.

Enable Workbook Users to Share Visualizations on Slack

Administrators can set up a Slack channel in Oracle Analytics, so that content authors can share their data visualizations on their organization's Slack app.

1. Obtain the client ID and client secret values for the Slack app that you want to use to share data visualizations.

   a. Open the Your Apps page in Slack, that is, `https://api.slack.com/apps`.
   b. Select the app that you want to use or create a new one.
   c. On the **Basic Information** tab, navigate to the **App Credential** section and obtain the **Client ID** and **Client Secret** values.

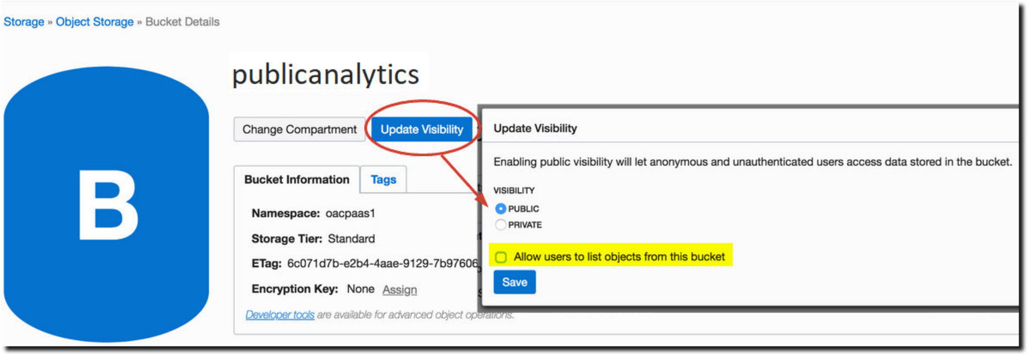
2. Configure the Slack app in Oracle Analytics.
   a. In the Oracle Analytics Home page, click the **Navigator**, click **Console**, and then click **Content-Sharing Platforms**.
   b. For **Service**, select **Slack**.
   c. Change **Status** to **Active**.
   d. For **Application Name**, enter the name of the app that you set up in Slack.
   e. For **Client ID** and **Client Secret**, enter the values that you obtained in Slack (Step 1).
   f. Click **Update**.
   g. Click **Copy to Clipboard** to copy the redirect URL for Oracle Analytics.
3. In Slack, configure the callback URL for Oracle Analytics.
   a. Open the Your Apps page in Slack.
   b. Select the app that you want to use.
   c. On the **Basic Information** tab, click **OAuth and Permissions**.
   d. Click **Add New Redirect URL**, paste the clipboard content in the **Redirect URL** field, and click **Add**.
   e. Click **Save URLs**.
4. Verify you can share a visualization on the Slack channel.
   a. In Oracle Analytics, open a workbook.
   b. On the Visualize or Narrate canvas, click the **Export** icon.
   c. Click **Slack**.

   If you set up and activate the channel correctly, **Slack** displays as an option on the **Export** menu.

Set Up a Public Container to Share Visualizations

Administrators can set up a public web storage container in Oracle Cloud so that content authors can share their data visualizations with others.

1. Create the public container in Oracle Cloud.
   a. In Oracle Cloud Infrastructure Console, navigate to **Object Storage**.
   b. On the Object Storage tab, click **Create Bucket**, and create a container with a suitable name, such as `publicanalytics`.
   c. Select the bucket, and click **Update Visibility**.
   d. Select **Public**, and verify that **Allow users to list objects from this bucket** *isn't* selected.



Namespace: oacpaas1

Storage Tier: Standard

ETag: 6c071d7b-e2b4-4aae-9129-7b97606

Encryption Key: None Assign

Developer tools are available for advanced object operations.

Update Visibility

Enabling public visibility will let anonymous and unauthenticated users access data stored in the bucket.

VISIBILITY

● PUBLIC

○ PRIVATE

□ Allow users to list objects from this bucket

e. Click **Save**.

2. Configure the public web store in Oracle Analytics.

   a. In the Oracle Analytics Home page, click the **Navigator**, click **Console**, and then click **Social**.

   b. For **Service**, select **Public Web Store**.

   c. To specify a public container for the first time or change the existing container, click **Edit**.

   d. Enter **Storage Container URL**.

      Use the REST endpoint URL format:

      `https://swiftobjectstorage.`*`region`*`.oraclecloud.com/v1/`*`object-storage-namespace`*`/`*`public-bucket-name`*

      For example: `https://swiftobjectstorage.us-ashburn-1.oraclecloud.com/v1/oacpaas1/publicanalytics`

      See Oracle Cloud Infrastructure documentation, Ways to Access Object Storage.

   e. For **Storage User** and **Storage Password**, enter the user name and password of a user with read and write access to the public container.

   f. Click **Save**.

      If you decide to use a different public container in the future, links to content that people have already shared through the existing public container continue to work but they can’t be updated. Newly shared content is stored in the new location.

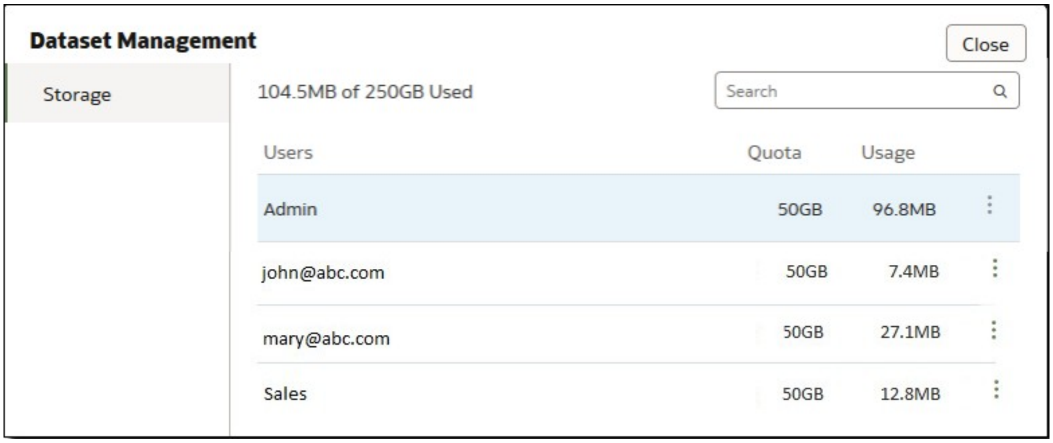
   g. Change **Status** to **Active**.

After you set up and activate the channel, **Public Web Storage** displays as an option on the Export menu.

Delete Unused Datasets

From time to time, administrators might need to delete datasets on behalf of other users to free up storage space. For example, a user uploads data files and then their account is disabled when they leave the company.

1. Click the **Page Menu** on the Home page, and select **Dataset Management**.
2. To free up some space, click the **Options** menu for a user with files you want to delete.



Dataset Management

104.5MB of 250GB Used

Users	Quota	Usage
Admin	50GB	96.8MB
john@abc.com	50GB	7.4MB
mary@abc.com	50GB	27.1MB
Sales	50GB	12.8MB

3. Select one of the following options:
   - **Delete Private** to delete non-shared (private) data files.
   - **Delete All** to delete all data files.

Monitor Users and Activity Logs

You can see information about any users who are currently signed in and troubleshoot report queries from the Manage Session page.

Topics:

- Monitor Users Who Are Signed In
- Analyze SQL Queries and Logs

Monitor Users Who Are Signed In

You can filter SQL queries and logs by user from the Manage Sessions page.

- **User ID**: Name that the user entered when they signed in.
- **Browser Info**: Information about the browser used to sign in.
- **Logged On**: Time when the user signed in.
- **Last Access**: Time stamp for the last activity for this user. This can be any kind of activity, such as switching from one page to another.

1. In the Oracle Analytics Home page, click the **Navigator**, and then click **Console**.
2. Click **Sessions and Query Logs**.
3. Locate the **Sessions** sections.

   The Sessions section at the top of the page shows how many users are currently signed in (Total Number of Sessions) and detailed information about these users.
4. To monitor a particular user, select **Filter Cursors by Session**.

   Information for this user displays in the Cursor Cache table.

   Click **Clear Filter** to show information for all users.
5. To change how messages are logged for a particular user, select a **Log Level** from the list.

   By default, logging is disabled.

Analyze SQL Queries and Logs

Administrators can examine the underlying SQL query requests that are run as people use the service.

1. In the Home page, click the **Navigator**, and then click **Console**.
2. Click **Sessions and Query Logs**.
3. Locate the **Cursor Cache** section, and review the query information recorded there. See Query Information Recorded in the Cursor Cache Table.
4. Optional: Click **Close All Cursors** to remove information in the Cursor Cache table.
5. Optional: Click **Cancel Running Requests** to cancel all requests that are running for analyses.

Query Information Recorded in the Cursor Cache Table

Administrators can examine the underlying SQL query requests that are run as people use the service.

These options apply only to analyses and dashboards. They don't apply to data visualizations.

Field	Description
ID	A unique internal identifier that is assigned to each entry.
User	The name of the user who ran the analysis and last placed it into the cache.
Refs	The number of references to this entry since it was placed into the cache.
Status	The status of the analysis that is using this cache entry: • **Starting** — The analysis is starting to run. • **Waiting on Parent** — A view in the analysis is waiting for data to be returned for the query. • **Running** — The analysis is currently running. • **Finished** — The analysis has finished. • **Queued** — The system is waiting for a thread to become available so the analysis can be processed. • **Canceling** — The application is in the process of canceling the analysis. • **Error** — An error occurred during the processing or running of the analysis. Look in the Statement column for information about the error.
Time	The time taken to process and run the analysis, displayed in one-second increments. A value of 0s (zero seconds) indicates that the analysis took under 1 second to complete.
Action	Links that you can click to affect the analysis: • **Cancel** — Terminates the analysis. Is displayed for in-progress analyses. The user running the analysis receives an informational message indicating that the analysis was canceled by an administrator. • **Close** — Clears the cache entry associated with this analysis. Is displayed for completed analyses. • **View Log** — Displays the log of a query run for this analysis. • **Diagnostic** — Displays an HTML page of diagnostic information that you can share with Oracle Customer Support.
Last Accessed	The time stamp of the last time the cache entry for this analysis was used to satisfy an analysis.

Field	Description
Statement	The logical SQL statement that was issued for the analysis; or if the analysis resulted in an error, information about the nature of the error.
Information	Usage tracking information (for example, what analysis contained the query).
Records	The number of records in the result set that have been seen (for example, 50+ to indicate that 50 records have been seen but there are additional records to be fetched or 75 to indicate that 75 records have been seen and there are no more records to be fetched).

Run Test SQL Queries

Administrators can enter a SQL statement directly to underlying data sources. This feature is useful for testing and debugging.

1. In the Oracle Analytics Home page, click the **Navigator**, and then click **Console**.
2. Click **Issue SQL**.
3. Enter the SQL statement. For example:

```
SELECT
    XSA('weblogic'.'SalesTargets')."Columns"."E1 Sales Rep Name" s_1
FROM XSA('weblogic'.'SalesTargets')
```

4. Change the **Logging Level** if required.
5. Select **Use Oracle Analytics Presentation Services Cache**.
6. Click **Issue SQL**.

Apply Custom Logos and Dashboard Styles

Administrators use themes to apply custom logos and dashboard styles.

Topics:

- About Custom Logo and Dashboard Styles
- Change the Default Style for Analyses and Dashboards
- Manage Themes
- Customize Links on the Classic Home Page
- Add Localized Custom Links to the Classic Home Page - Example

About Custom Logo and Dashboard Styles

As an administrator you can customize your reporting environment by creating a theme that displays custom logo, branding text, page style and so on.

When working with themes:

- You can create multiple themes, but only one theme can be active at one time.
- If you deactivate a theme, you revert to the default Oracle theme, unless you select a different one.

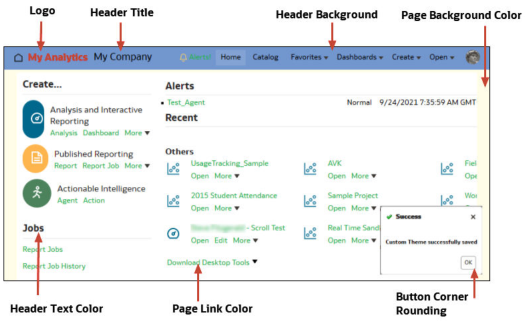
- Themes are applied on pages with analyses and dashboards, but not to visualization workbooks.
- You manage themes using the **Manage Themes** page available from the Classic Administration page.
- When you activate a theme, you apply it to the browser session of the currently signed-in administrator and to the browser sessions of end-users as they sign in.
- If Oracle Analytics is running on multiple instances, then duplicate and activate them for each instance.

Change the Default Style for Analyses and Dashboards

Administrators create themes to change the default logo, colors, and heading styles for analyses and dashboards.

1. In the Classic Home page, click the user profile icon and then click **Administration**.
2. Click **Manage Themes**.
3. To apply an existing dashboard style, select one from the **Theme** list, click **Active**, then click **Save**.
4. To create a new dashboard style, in the **Theme** list, click **New Theme** to display the New Theme dialog.
5. In **Theme Name**, the name that you specify here is displayed in the **Style** list on the Dashboard Properties dialog.
6. In **Logo**, specify the page logo that you want displayed in the top left hand corner. To replace the default Oracle logo, click **Select Logo** and navigate to and select a different logo in PNG, JPG, or JPEG format. Logos can't exceed 136 pixels in width by 28 pixels in height.
7. In **Header Title**, specify the branding information that you want displayed in the top left hand corner next to the logo.
8. In **Active**, click to apply the currently displayed theme when you click **Save**. If you click **Active**, then click **Back** without saving changes, the new theme isn't applied.

This diagram shows you what theme options affect different areas of the reporting environment.



Success

Custom Theme successfully saved

Manage Themes

Administrators manage themes to change the default logo, colors, and heading styles for reporting pages, dashboards, and analyses.

In Oracle Analytics Server the user interface defaults to the Alta style, which is controlled by elements in the `instanceconfig.xml` file. You can manage themes only when the `UI` element is configured for Alta, as shown in this code example from the instanceconfig.xml file.

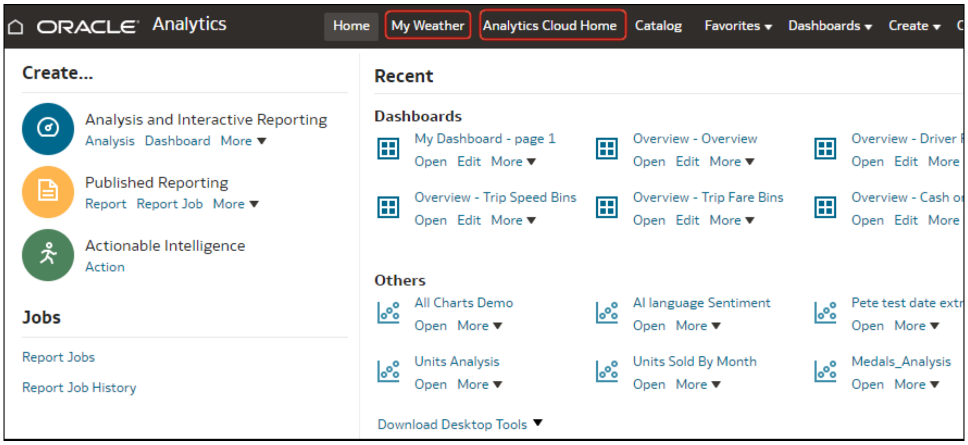
```
<ServiceInstance>
    <UI>
        <DefaultSkin>Alta</DefaultSkin>
        <DefaultStyle>Alta</DefaultStyle>
    </UI>
</ServerInstance>
```

1. In the Classic Home page, click the user profile icon and then click **Administration**.
2. Click **Manage Themes**.
3. Optional: To apply a previously created theme, select the theme you want from the Theme list, click **Active**, then click **Save**, then click **Back**.
4. Optional: To revert to the default Oracle theme, clear the **Active** option, click **Save**, then click **Back**.
5. Optional: To remove a theme completely, select the theme you want to remove, click **Delete**, then click **Back**.

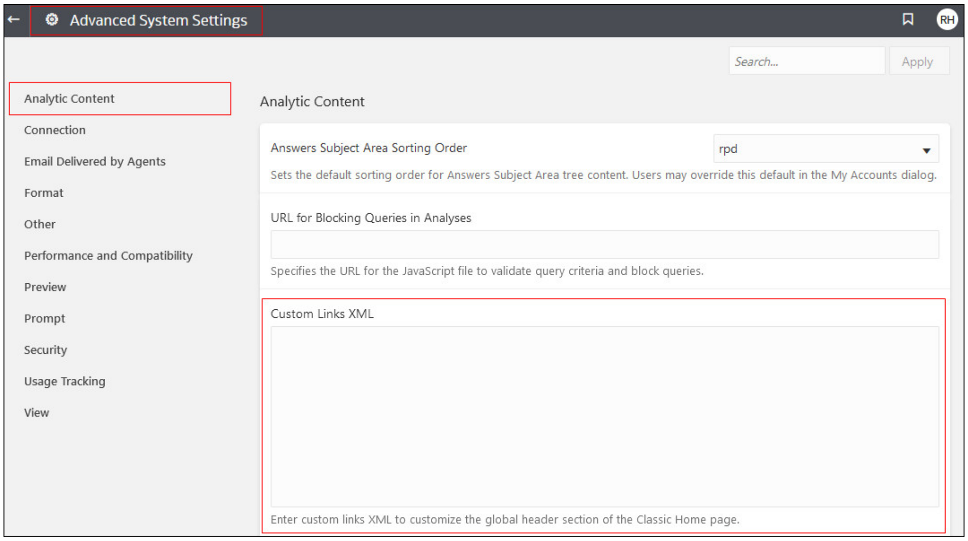
Customize Links on the Classic Home Page

You can configure the Classic home page to display custom links. For example, you might add a link to a website showing the local weather, or a link to the Oracle Analytics home page to enable business analysts to navigate from the Classic home page to workbooks and visualizations.

In this example, links are added for "My Weather" and "Analytics Cloud Home".



To add custom links, add XML code to the **Custom Links XML** system setting. To access system settings, go to the Oracle Analytics home page, click **Navigator**, then **Console**, then **Advanced System Settings**, then **Analytic Content**).



Analytic Content

Answers Subject Area Sorting Order: rpd

Sets the default sorting order for Answers Subject Area tree content. Users may override this default in the My Accounts dialog.

URL for Blocking Queries in Analyses

Specifies the URL for the JavaScript file to validate query criteria and block queries.

Custom Links XML

Enter custom links XML to customize the global header section of the Classic Home page.

You can use XML code to specify links and attributes, including the following:

- The text for the link (either a static string or a message name to use for localization).
- A target URL.
- Whether the target link opens in the current page or opens in a new tab or window.
- The relative ordering of links in the header.
- An optional icon to use with the link.

This example displays two custom links to the left of the **Catalog** link in the global header of the Classic home page.

```
<?xml version="1.0" encoding="utf-8"?>
<customLinks xmlns="com.siebel.analytics.web/customlinks/v1">
   <link id="1" name="My Weather" description="Local weather site"
src="https://www.example.com/weather" target="blank" >
       <locations>
           <location name="header" insertBefore="catalog"/>
       </locations>
   </link>
   <link id="1" name="Analytics Cloud Home" description="OAC Viz Home Page"
src="https://<OAC example URL>.analytics.ocp.oraclecloud.com/ui/dv/?
pageid=home" target="blank" >
       <locations>
           <location name="header" insertBefore="catalog"/>
       </locations>
   </link>
</customLinks>
```

> **Note:**
>
> To obtain the link for Oracle Analytics home page, log into Oracle Analytics, copy the URL, and paste it into the `src="<target link>"` element (as shown in the example XML code).

If you use this Console method, then you shouldn't have any CustomLinks elements in the `instanceconfig.xml` file.

This table describes the elements and attributes that you can specify for custom links.

Element or Attribute	Optional?	Data Type	Description
`link: accessibility`	Optional	Boolean	Specifies that in accessibility mode, the link is available only when the accessibility attribute is set to **true**. Values are **true** and **false**, and **false** is the default. In previous updates, the `vpat` attribute served the same purpose as the accessibility attribute. The `vpat` attribute has been deprecated.
`link: description`	Optional	String	Specifies the description of the link (not translated).
`link: iconSmall`	Optional	String	Specifies the file name of an icon to display with the link in the global header. The display of icons is controlled by the fmap syntax.
`link: id`	Required	String	Use as a unique ID that specifies the position of the link. You can include IDs for custom links to position them relative to default links.
`link: localizedDesc`	Optional	String	Specifies the message ID of the link (translated), which takes precedence over the non-translated description.
`link: localizedName`	Optional	String	Specifies the message ID of the link that's translated, which takes precedence over the non-translated name.
`link: name`	Required	String	Specifies the name of the link that isn't translated.
`link: privilege`	Optional	String	Specifies the name of privileges that a user must be granted to see the link. The privileges are indicated as an expression, as shown in the following example: `privileges.Access['Global Answers']&amp;&amp; privileges.Access['Global Delivers']`
`link: src`	Required	String	Specifies the URL for the link.
`link: target`	Optional	String	Specifies the browser window in which to open the link. The values are: **self:** Opens in same window in which Oracle Analytics is running. **blank:** Opens in a new window. *any-name*: Opens in a window with the specified name.

Element or Attribute	Optional?	Data Type	Description
`location: insertBefore`	Optional	String	Specifies the ID of an existing link to the left of which you want to add the custom link. For example, to add a custom link to the left of the **Catalog** link, specifiy `<location name="header" insertBefore="catalog"/>`. Valid IDs: • `admin` • `catalog` • `dashboard` • `favorites` • `help` • `home` • `logout` • `new` • `open` • `user` If you make a mistake and specify an invalid ID, the link is inserted in a default location.
`location: name`	Required	String	Use this attribute if you include the locations parent element. The values are: **header:** Specifies to include the link in the global header.
`locations`	Optional	Not Applicable	Use as the parent element for specifying the locations of the links to add. If you don’t specify a location, then by default links are included before the Help link in the global header and at the end of the Get Started section.

Add Localized Custom Links to the Classic Home Page - Example

You can use this example to add localized custom links to the Classic Home page.

1. Use these commands to create a custom message directory in the Domain_Home and copy the messages files into it.

   On Linux | Windows:

   ```
   cd DOMAIN_HOME /bi/bidata/components/OBIPS
   mkdir custommessages
   cd custommessages
   cp $OAS_HOME/bi/bifoundation/web/msgdb/* . -r
   ```

2. Add English localized messages for kmsgWeather and kmsgOasHome to the `bieehomemessages.xml` file in the English folder:

   a. Backup `$DOMAIN_HOME/bi/bidata/components/OBIPS/custommessages/l_en/messages/bieehomemessages.xml` and open it for editing.

   b. Add the following messages to `bieehomemessages.xml` in the English folder.

   ```
   <WebMessage name="kmsgWeather"><TEXT>The Weather</TEXT></WebMessage>
   <WebMessage name="kmsgWeatherDesc"><TEXT>Click to check the weather </
   TEXT></WebMessage>
   ```

```
<WebMessage name="kmsgOasHome"><TEXT> The OAS Home Page</TEXT></
WebMessage>
<WebMessage name="kmsgOasHomeDesc"><TEXT>Click to check The OAS Home
Page </TEXT></WebMessage>
```

3. Add French localized messages for kmsgWeather and kmsgOasHome to the `bieehomemessages.xml` file in the French folder:
   a. Backup `$DOMAIN_HOME/bi/bidata/components/OBIPS/custommessages/l_fr/messages/bieehomemessages.xml` and open it for editing.
   b. Add the following messages to `bieehomemessages.xml` in the French folder.

   ```
   <WebMessage name="kmsgWeather"><TEXT>La Météo</TEXT></WebMessage>
   <WebMessage name="kmsgWeatherDesc"><TEXT>Cliquer pour vérifier la météo
   </TEXT></WebMessage>
   <WebMessage name="kmsgOasHome"><TEXT> La Page d'accueil d'OAS</TEXT></
   WebMessage>
   <WebMessage name="kmsgOasHomeDesc"><TEXT>Cliquer pour vérifier la page
   d'accueil d'OAS </TEXT></WebMessage>
   ```

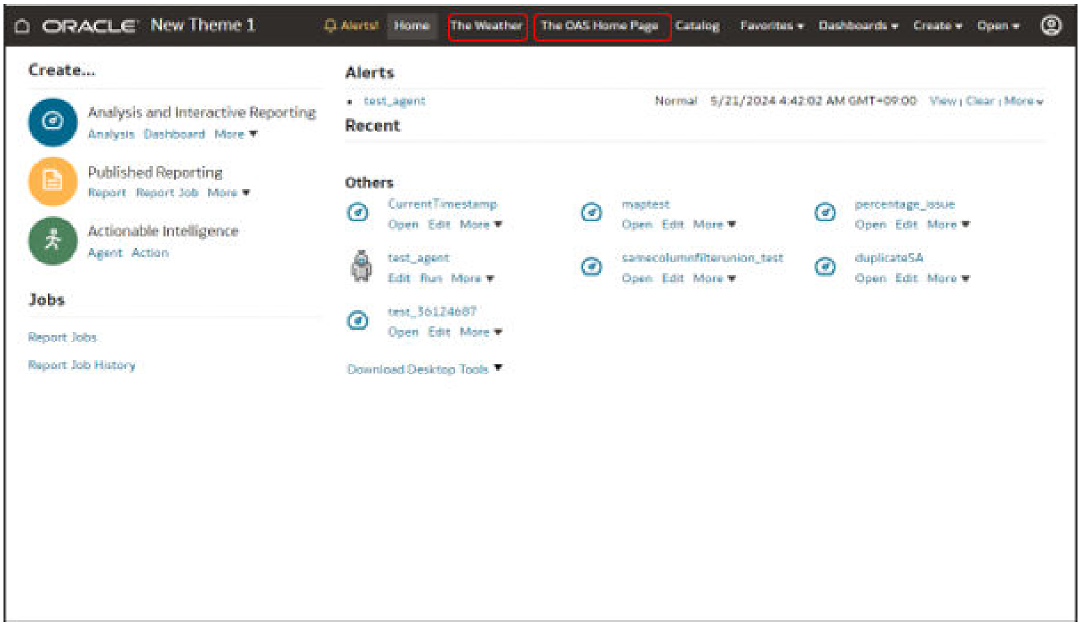
4. Log in to Oracle Analytics Server, then from the **Navigator**, click **Console**, and **System Settings**.
5. Click **Analytic Content**, and in **Custom Links XML** add the following code and click **Apply**:

   ```
   <?xml version="1.0" encoding="utf-8"?>
   <customLinks xmlns="com.siebel.analytics.web/customlinks/v1">
   <link id="1" name="My Weather" localizedName="kmsgWeather"
   description="Local weather site" localizedDesc="kmsgWeatherDesc"
   src="https://www.example.com" target="blank">
   <locations><location name="header" insertBefore="catalog"/></locations>
   </link>
   <link id="1" name="Analytics Server Home" localizedName="kmsgOasHome"
   description="The OAS Home Page" localizedDesc="kmsgOasHomeDesc"
   src="https://www.example.com" target="blank">
   <locations><location name="header" insertBefore="catalog"/></locations>
   </link>
   </customLinks>
   ```

   This XML code enables you to display localized custom links that you define in the language-specific `bieehomemessages.xml` files.
6. Stop and start Oracle Analytics Server. See Use Commands to Start, Stop, and View Status of Processes.
7. Log in to the Oracle Analytics Server Classic Home page using an English language URL:

   ```
   http://<HOST>:<PORT>/analytics/saw.dll?bieehome&lang=en
   ```

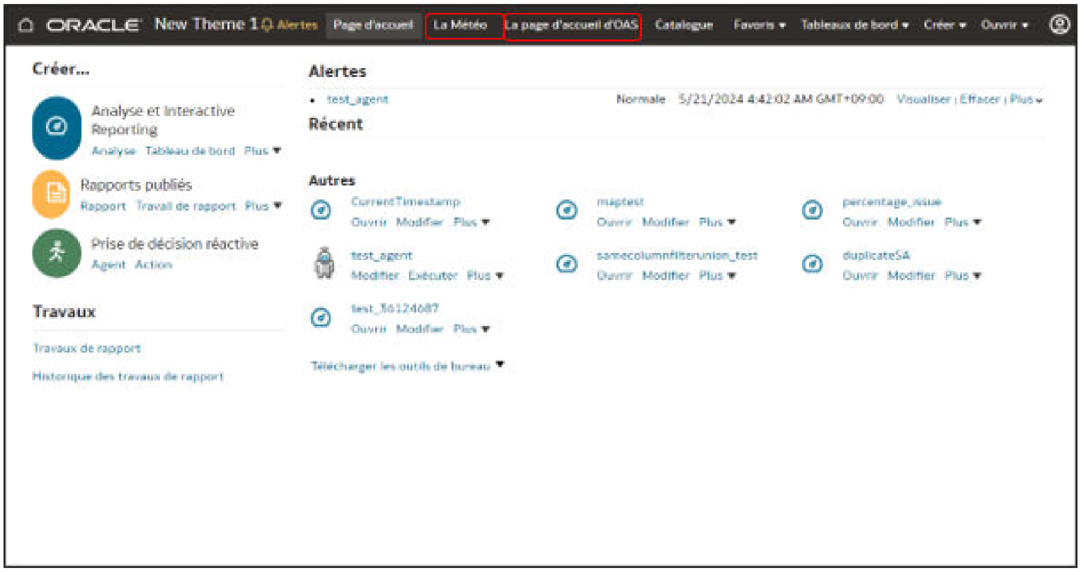
   Observe the newly added buttons, labeled The Weather and The OAS Home Page.



8. Log in to the Oracle Analytics Server Classic Home page using a French language URL:

```
http://<HOST>:<PORT>/analytics/saw.dll?bieehome&lang=fr
```

Observe the newly added buttons, labeled La Météo and La page d'accueil d'OAS.



Enable Custom Java Script for Actions

Users working with analyses and dashboards can add action links that invoke custom JavaScript accessible through a web server. To enable this feature, administrators specify the URL of the web server in advanced system settings page and register the web server as a safe domain.

1. Develop your scripts in JavaScript, store them in a web server, and make a note of the URL pointing to the JavaScript (*.JS) file containing the custom scripts.

   For example, you might develop a currency conversion script named `mycurrencyconversion` that you store in `myscripts.js`, and the URL might be:`http://example.com:8080/mycustomscripts/myscripts.js`.

2. Specify the URL of your web server in System Settings:
   a. Click **Console**, then click **Advanced System Settings**.
   b. In **URL for Browser Script Actions**, enter the URL that you noted in Step 1.
   c. If prompted to, click **Apply**.

3. Register the web server as a safe domain:
   a. Click **Console**, then click **Safe Domains**.
   b. Add an entry for the domain in the URL you specified in Step 2.

      For example, you might add: `example.com:8080`.
   c. For options, select **Script** and **Connect**.

4. Test your configuration:
   a. In Classic Home, open or create an analysis.
   b. Display the Column Properties for a column, click **Interaction**, then **Add Action Link**.
   c. Click **Create New Action**, then **Invoke a Browser Script**.
   d. Under **Function Name** enter the name of a script in your JavaScript (*.JS) file.

      For example, `USERSCRIPT.mycurrencyconversion`.
   e. Save the details, and open the analysis.
   f. Click the column to which you added the action, then click the action.

Switch to a Different Language

Oracle Analytics supports a range of languages.

- What languages does Oracle Analytics support?
- What's translated?
- What isn't translated?
- How do I select my language?
- How do I find documentation in my language?

What languages does Oracle Analytics support?

Oracle Analytics supports 28 languages:

Arabic, Chinese (Simplified), Chinese (Traditional), Croatian, Czech, Danish, Dutch, English, Finnish, French, French (Canada), German, Greek, Hebrew, Hungarian, Italian, Japanese, Korean, Norwegian (Bokmål), Polish. Portuguese. Portuguese (Brazil), Romanian, Russian, Slovak, Slovenian, Spanish, Swedish, Thai, Turkish.

What's translated?

- **User Interface:** Oracle Analytics translates text in menus, buttons, messages, and other elements of the user interface.

- **Auto-generated text:** Some auto-generated text in content that you create is translated too. For example, automatically generated titles and filters displayed in visualizations, analyses, dashboards, pixel-perfect reports, and so on.
- **User guides:** Several user guides are translated.

What isn't translated?

A few features are available only in English.

- Analyses, dashboards, and pixel-perfect reports:
  - User-defined titles and text in your workbooks, unless you choose to translate them. See Localize Catalog Captions.
  - Column names coming from your data sources, unless you set up column name translation in your semantic model.
- Data visualization workbooks:
  - User-defined titles and text in your workbooks.
  - Column names coming from your data sources, such as "Revenue". Unless your workbook is based on a subject area and you set up column name translation in your semantic model.
  - Default names for your workbooks. If English is your selected language, the default name for workbooks is *Untitled*. If you use a different language such as Italian, the default name when you save a workbook is the equivalent of *Untitled* in Italian. However, after you save a workbook, the name is fixed in that language. Workbook names don't change if you sign-in with a different language.
- Datasets:
  - Column names in Microsoft Excel spreadsheets that you upload.
  - Column names from your data sources.

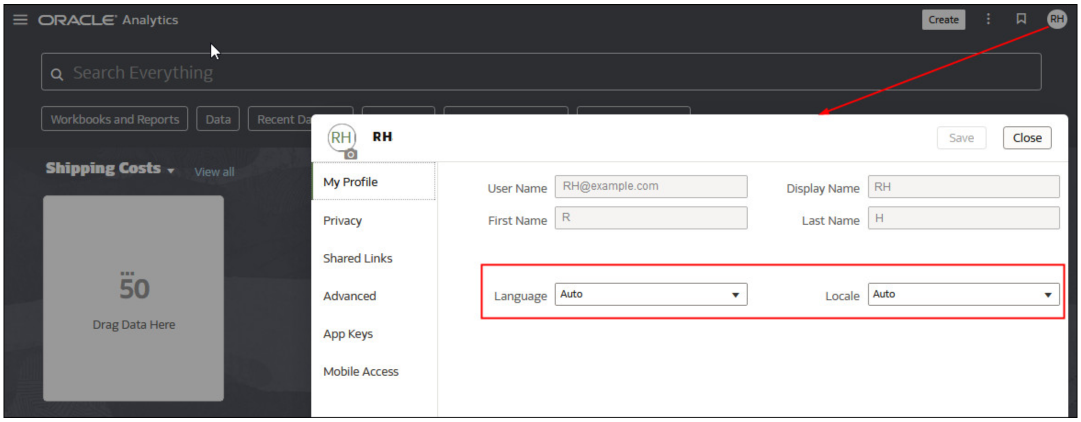
How do I select my language?

Several options are available:

- Select your language in your browser settings.

  Refer to the documentation for your browser.
- Data visualization pages - Select your language in the My Profile tab, available from the Home page.

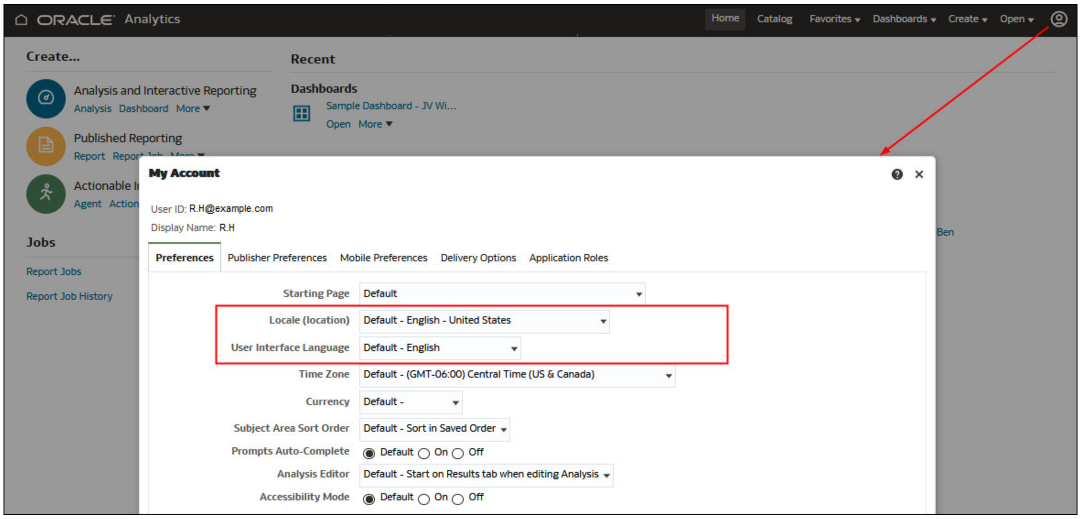
  See Localize the User Interface for Data Visualization.



User Name RH@example.com
Display Name RH
First Name R
Last Name H
Language Auto
Locale Auto

- **Classic pages** - Select your language in the My Account Preferences tab, available from the Classic Home page.

  See Set Your Preferences.



My Account

User ID: R.H@example.com

Display Name: R.H

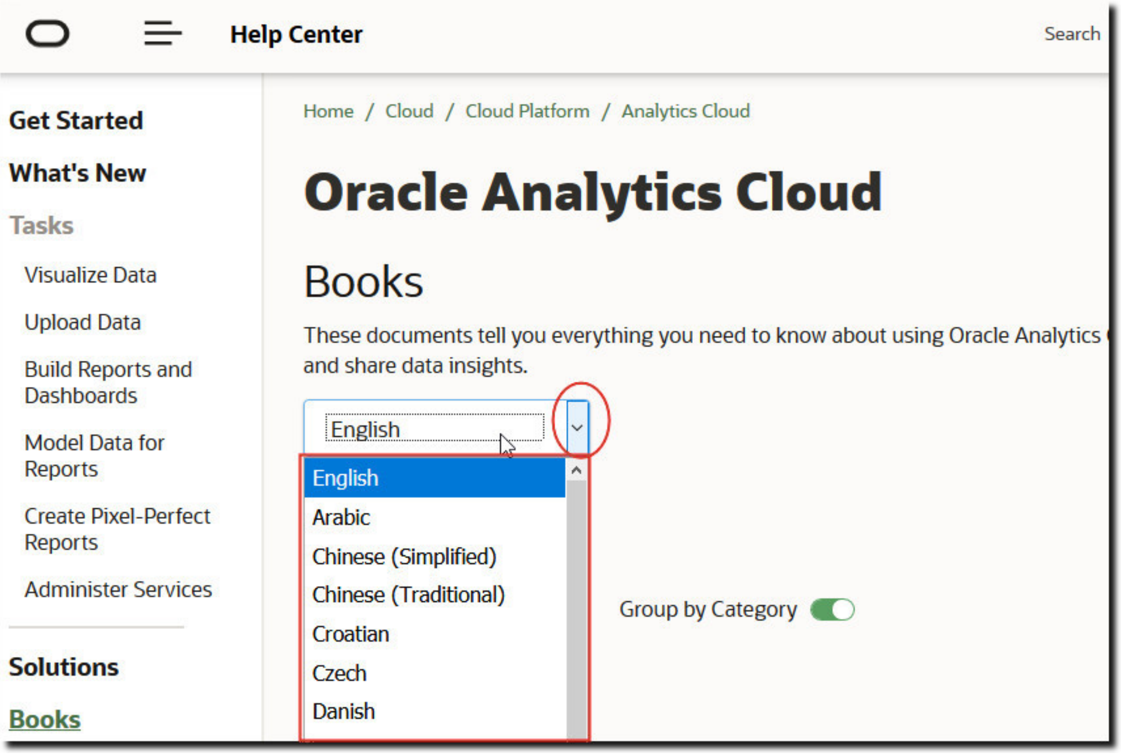
Preferences | Publisher Preferences | Mobile Preferences | Delivery Options | Application Roles

Starting Page	Default
Locale (location)	Default - English - United States
User Interface Language	Default - English
Time Zone	Default - (GMT-06:00) Central Time (US & Canada)
Currency	Default -
Subject Area Sort Order	Default - Sort in Saved Order
Prompts Auto-Complete	● Default ○ On ○ Off
Analysis Editor	Default - Start on Results tab when editing Analysis
Accessibility Mode	● Default ○ On ○ Off

How do I find documentation in my language?

In most cases, when you click Help in Oracle Analytics, user assistance is displayed in the same language as the user interface. For example, if you're working in French, the Help is displayed in French.

Several Oracle Analytics user guides are translated into the same 28 languages as the user interface. To find books translated in your language, navigate to your Oracle Analytics product on Oracle Help Center, select the Books tab and then select your language.



Oracle Analytics Cloud

Books

These documents tell you everything you need to know about using Oracle Analytics
and share data insights.

Configure Advanced System Settings

Administrators can set several advanced options using the **Advanced System Settings** page.

Topics:

- About System Settings
- Configure System Settings Using Console
- Manage System Settings Using REST APIs

About System Settings

Administrators can set a range of advanced, service-level options through the **Advanced System Settings** page. For example, you might want to change the default currency and time zone for analyses and dashboards to values that better suit your organization.

Topics:

- When Do System Settings Take Effect?
- Analytic Content Options
- Connection Options
- Email Delivered by Agents Options
- Format Options
- Other Options
- Performance and Compatibility Options
- Prompt Options
- Security Options
- Usage Tracking Options
- View Options

When Do System Settings Take Effect?

When you change your system settings, the time it takes for the changes to take effect can vary. Some setting changes happen right away, but most take a few minutes.

After the changes take effect, it's a good idea to refresh your browser to see any visible differences in the user interface. If other users are signed in when you make system setting changes, they must refresh their browsers too. In some cases, users might need to sign out and sign back in to see the changes.

To learn when a particular system setting is applied, refer to the table that lists the setting:

- Look for the **Change Effective** item in the **More Information** column. This tells you the time it takes to change the setting.
- See the **Apply Required** item to understand whether you need to click **Apply** to implement your changes.

Analytic Content Options

You use these options to set defaults and customizations for dashboards, analyses, and reports. For example, you can configure the analysis editor to open by default to the Criteria tab or the Results tab.

> **Note:**
>
> In Oracle BI Enterprise Edition 12c, you use configuration files to customize various settings. In Oracle Analytics Server, you can use the Console (System Settings page). If you upgrade from Oracle BI Enterprise Edition 12c to Oracle Analytics Server, your configuration file settings are preserved but no values are displayed on the System Settings page, that is, they appear blank. You don't have to re-enter values on the System Settings page but, for clarity, you might decide to do so.

System Setting	More Information
Analytics Publisher Reporting Toolbar Mode	Configures an alternate toolbar for pixel-perfect reports that are included in a dashboard. See Configure an Alternate Toolbar for Oracle Analytics Publisher • **1** — Doesn't display a toolbar for pixel-perfect reports. • **2** — Displays the URL to the report without the logo, toolbar, tabs, or navigation path. • **3** — Displays the URL to the report without the header or any parameter selections. Controls such as Template Selection, View, Export, and Send are still available. • **4** — Displays the URL to the report only. No other page information or options are displayed. • **6** — Displays parameter prompts for the report in a toolbar. **Valid Values**: 1,2,3,4,6 **Default:** 1 **Apply Required:** No **Change Effective:** After a few minutes **API Key:** AnalyticsPublisherReportingToolbarMode
Answers Editor Start Tab	Specifies whether the analysis editor opens by default to the Criteria tab or the Results tab. This setting applies when users click an **Edit** link for an analysis from a dashboard, the Home page, or the Catalog page. Users can override this default setting by specifying the **Full Editor** option in the My Account dialog. • **answerResults** — Opens the analysis editor by default to the Results tab. • **answerCriteria** — Opens the analysis editor by default to the Criteria tab. **Valid Values:** answerResults, answerCriteria **Default:** answerResults **Apply Required:** No **Change Effective:** After a few minutes **API Key:** AnswersEditorStartTab

System Setting	More Information
Answers Subject Area Sorting Order	Sets the default sort order for subject area content trees. Users can override this default setting in the My Account: Subject Area Sort Order dialog. • **asc** — Sorts A to Z. • **desc** — Sorts Z to A. • **rpd** — Uses the subject area sort order specified in the original analyses. **Valid Values:** asc, desc, rpd **Default:** rpd **Apply Required:** No **Change Effective:** After a few minutes **API Key:** AnalysisSubjectAreaSortingOrder
Custom Links XML	Specifies the XML code containing Classic Home page header customizations. You can use this XML code to customize the global header section of the Home page to better meet the needs of your users. For example, you can disable certain links or add custom ones. See Customize Links on the Classic Home Page. **Apply Required:** No **Change Effective:** After a few minutes **API Key:** CustomLinksXml
URL for Blocking Queries in Analyses	Specifies the URL for the JavaScript file to validate query criteria and block queries. See Validate and Block Queries in Analyses Using Custom JavaScript. **Apply Required:** No **Change Effective:** After a few minutes **API Key:** QueryBlockingScriptURL
Writeback Template XML	Defines the XML configuration for performing writeback on data elements. For example, you can use an XML template to enable users of a dashboard page or an analysis with the ability to modify, or write back, the data that they see in a table view. See Create Write-Back Template Files **Apply Required:** No **Change Effective:** After a few minutes **API Key:** WriteBackTemplateXML

Connection Options

Connection externalization isn't supported in Oracle Analytics Server.

System Setting	More Information
Connection Externalization Enabled	This option isn't used.

Email Delivered by Agents Options

You can use these options to customize the way agents deliver email.

System Setting	More Information
Maximum Email Size (KB)	Specifies the maximum size (KB) of a single email. If you set a maximum email size, you can avoid situations when SMTP servers reject emails that are too large, and in the event that an email exceeds the set limit, the email recipients receive an error message instead of the agent failing and just alerting the email author. **Valid Values:** 0-20480 **Default: 0** (unlimited email size) **Apply Required:** No **Change Effective:** Immediately **API Key:** EmailMaxEmailSizeKB
Maximum Number of Recipients per Email	Specifies the maximum number of recipients allowed in the To: or Bcc: line for a single email. You can set the maximum number of email recipients to avoid some SMTP servers from filtering out these emails as spam. If the recipient list exceeds the set limit, the list is split into smaller lists with the maximum number of allowed recipients in each list. **Valid Values:** 0-1024 **Default: 0** (unlimited number of email recipients) **Apply Required:** No **Change Effective:** Immediately **API Key:** EmailMaxRecipients
Safe Domains	If you want to restrict the email domain that Oracle Analytics can send emails to, enter the name of the domain. For example, `examplemaildomain.com`. Use a comma to separate multiple domain names. For example, `exampledomain1.com,exampledomain2.com`. By default, there are no restrictions. **Apply Required:** No **Change Effective:** Immediately **API Key:** EmailSafeDomains
Use BCC	Specifies whether to include the names of email recipients in the To: or Bcc: line. By default, email recipients are added to the Bcc: line. • **On** — Add email recipients to the Bcc: line. Names of email recipients are hidden. • **Off** — Add email recipients to the To: line. Everyone who receives the email sees the recipient list. **Default:** On **Apply Required:** No **Change Effective:** Immediately **API Key:** EmailUseBcc

System Setting	More Information
Use RFC 2231 Encoding	Specifies how to encode MIME email parameters. By default, RFC 2047 is used. • **On** — Use RFC 2231 to encode MIME email parameter values. RFC 2231 supports multi-byte languages. Select On if you deliver emails that contain multi-byte characters and use an email server that supports RFC 2231, such as Microsoft Outlook for Office 365 or Google Gmail. • **Off** — Use RFC 2047 to encode MIME email parameter values. **Default:** Off **Apply Required:** No **Change Effective:** Immediately **API Key:** EmailUseRFC2231

Format Options

You use these options to configure default currency and time zone settings for analyses and dashboards.

These options apply only to analyses and dashboards. They don't apply to data visualizations.

> **Note:**
>
> In Oracle BI Enterprise Edition 12c, you use configuration files to customize various settings. In Oracle Analytics Server, you can use the Console (System Settings page). If you upgrade from Oracle BI Enterprise Edition 12c to Oracle Analytics Server, your configuration file settings are preserved but no values are displayed on the System Settings page, that is, they appear blank. You don't have to re-enter values on the System Settings page but, for clarity, you might decide to do so.

System Setting	More Information
Currencies XML	Defines the default currency that's displayed for currency data in analyses and dashboards. For example, you can change from American dollars ($) to Euros (E). See Change the Default Currency for Analyses. **Apply Required:** No **Change Effective:** After a few minutes **API Key:** AnalysisCurrenciesXml

System Setting	More Information
Default Data Offset Time Zone	Specifies a time zone offset of the original data that users see in analyses and dashboards. Enter an offset value that indicates the number of hours away from Greenwich Mean Time (GMT) time. For example, to display values in United States Eastern Standard Time (EST), which is Greenwich Mean Time (GMT) - 5 hours, enter the value `GMT-05:00` or the equivalent value in minutes `-300`. If you don't set this option, no time zone conversion occurs because the value is "unknown". **Specifying a different offset value for each user** If you want to specify a different offset value where session variables can be used (for example, expressions, calculations), don't use the **Default Data Offset Time Zone** setting. Instead, set the system session variable DATA_TZ in the semantic model. See About Session Variables. **Apply Required:** No **Change Effective:** After a few minutes **API Key:** DefaultDataOffsetTimeZone
Default Time Zone for Date Calculations	Specifies the time zone used for evaluating date calculations such as getting the current date/time, truncating datetime values to a date, and extracting time fields from date/time expressions. If you leave this field blank, Oracle Analytics uses the operating system time zone when evaluating date calculations. **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** DefaultTimeZoneforDateCalculations
Default User Preferred Time Zone	Specifies a default preferred time zone that users see in analyses and dashboards before they select their own in the My Account Preferences dialog. If you don't set this option, Oracle Analytics uses the local time zone. **Specifying a different time zone for each user** If you want to specify a different offset value where session variables can be used (for example, expressions, calculations), don't use the **Default User Preferred Time Zone** setting. Instead, set the system session variable TIMEZONE in the semantic model. See About Session Variables. **Apply Required:** No **Change Effective:** After a few minutes **API Key:** DefaultUserPreferredTimeZone
User Currency Preferences XML	Determines whether users see a **Currency** option in their My Account preferences dialog and the list of currencies available to them. If you provide the **Currency** option, users can select in which currency they prefer to view columns of currency data in analyses and dashboards. See Define User-Preferred Currency Options. **Apply Required:** No **Change Effective:** After a few minutes **API Key:** UserCurrencyPreferencesXml

Other Options

These system setting options in the Console enable you to set the behavior for a variety of actions such as database queries, default URLs, display defaults, and sorting.

System Setting	More Information
Disable Right Trim for VARCHAR Data	Specifies whether the automatic removal of trailing spaces from varchar columns is enabled (Off) or disabled (On). For example, when this property is enabled (Off), when a user starts entering values in a field, the filter dialog automatically trims any trailing spaces. • **On** — Preserves trailing whitespaces in varchar columns. If you primarily use Oracle Database sources, you might want to keep the default Oracle Database behavior of preserving trailing whitespaces rather than removing them. When you toggle this property on, you avoid the overhead of trimming spaces, and this can improve performance. If you disable this property (set it to **On**) and you construct a filter such as PRODUCT_DESCRIPTION = 'My Product '), you must make sure the amount of trailing whitespace used exactly matches the varchar column value. If you don't, the filter won't correctly match the data values. • **Off** — Trims trailing whitespaces in varchar columns when processing queries. This is the default for Oracle Analytics. For example, if a user enters the text 'My Product ', it trims it to 'My Product'. **Default: Off** **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** DataQueryDisableRightTrimVARCHARData
Enable Subrequest Shipping	Specifies if sub-requests to source databases are executed separately as standalone queries or executed together. By default, sub-requests are shipped separately which can improve performance if you execute complex reports with a large group of sub-requests, that is, you prefer to ship the sub-requests separately in multiple simplified queries rather than ship a large single complicated query all at once. In Oracle BI Enterprise Edition, the default is set to `NO`. If you used Oracle BI Enterprise Edition and want to retain the previous default behavior, set this property to `NO` to continue executing database sub-requests together. • **Default** — Database sub-requests are shipped separately. This is the same as the value **YES**. • **YES** — Database sub-requests are shipped separately. • **NO** — Database sub-requests are shipped together, all at once. **Default: Default** **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** EnableSubrequestShipping

System Setting	More Information
Enforce Safe Domains in Actons	Determines whether action links that users add to analyses and dashboards can invoke any URL or only URLs that administrators specify in the safe domains list. • **On** — Don't allow actions to invoke any URL that's not in the safe domain list. • **Off** — Allow actions to invoke any URL, even if the URL isn't listed as a safe domain. **Default: On** for a brand new service and **Off** for an existing service. **Apply Required:** No **Change Effective:** After a few minutes **API Key:** EnforceSafeDomainsActions
Hide EPM Cloud Members with No Access	Specifies if users can view all EPM dimension members in a hierarchy prompt list of values or when adding the hierarchy to a canvas, even if they don't have data access to some of the members. • **On** —Show only those members of an EPM dimension that users have data access to. If this setting is **On**, users who don't have access to the root member of the dimension hierarchy won't see *any* EPM members in hierarchies or hierarchy prompts. • **Off** — Users can view all the members in an EPM dimension even if they don't have access to view data for some members. **Default: Off** **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** HideEPMCloudMembersWithNoAccess
Hide Loading Messages	Specifies if a detailed message is displayed during data load processing. • **On** — Detailed loading messages are hidden and a simplified message **Loading...** is displayed instead. • **Off** — Detailed loading messages are displayed. **Default: Off** **Apply Required:** No **Change Effective:** After a few minutes **API Key:** HideLoadingMessages
Locale	Applies to content migrated from Oracle BI Enterprise Edition. After you migrate content from your Oracle BI Enterprise Edition environment to Oracle Analytics, you may see a different language in messages, dates, or currencies within analyses. For example, if you look at a migrated analysis in Polish, the currencies or dates might display based on the Oracle Analytics default locale, not the original Oracle BI Enterprise Edition locale. To preserve the Oracle BI Enterprise Edition currencies and dates in Oracle Analytics, change this setting to **Polish**. **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** DataQueryLocale

System Setting	More Information
Portal Path	Specifies the path of the dashboard page that's displayed by default when users sign in to Oracle Analytics. For example, `/shared/<folder>/_portal/<name>`. You can specify a single path for all users and multiple paths by user role, for example `{"application role 1":"catalog dashboard path 1","application role 2":"catalog dashboard path 2","default":"catalog dashboard path 3"}`. This setting applies to all users, but users can override it after they've signed in. You can enter a maximum of 5,000 characters in this field. **Apply Required:** No **Change Effective:** After a few minutes **API Key:** PortalPath
Recursive Datetime Type Checking	Specifies whether to enforce strict recursive data type checking for comparisons between identical data types (for example, integer to integer) or non-compatible data types (for example, integer to short integer) on all data sources or with all datasets. • **On** — Enforces strict recursive checking for identical or non-compatible data types on all data sources or datasets. • **Off** — Relaxes strict recursive checking for date and time data types on all data sources or datasets. However, if there are too many data type inconsistencies, you may want to change the data types to be compatible or use constants of the correct data type when comparing a column to a value. For example, after you migrate content from Oracle BI Enterprise Edition to Oracle Analytics, you might start seeing this type of check error in your reports because eary versions of Oracle BI Enterprise Edition didn't enforce strict checks: `[nQSError: 22024] A comparison is being carried out between non-compatible types <type1> and <type2>.` **Default: On** **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** RecursiveDatetimeTypeChecking
Repeat Rows on Excel Exports for Tables and Pivots	Specifies whether cells that span rows and cells that span columns are repeated when exporting tables and pivot tables to Excel. • **On** — If switched on, cells that span rows and cells that span columns are repeated, regardless of the Value Suppression setting in the Analysis editor. • **Off** — If switched off, the Value Suppression setting in the Analysis editor is honored and cells that span rows and cells that span columns don't repeat when exporting tables and pivot tables to Excel. **Default: Off** **Apply Required:** No **Change Effective:** After a few minutes **API Key:** AnalysisRepeatRowsExcelExportsTablesPivots

System Setting	More Information
Sort Null Values First	Specifies whether to sort NULL values before other values (On) or after (Off). Select the value that matches your database. If this setting doesn't match your database setting, then the database setting takes precedence. • **On** — Sorts NULL values before other values. • **Off** — Sorts NULL values after other values. **Default: Off** **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** SortNullValuesFirst
Sort Order Locale	Applies to content migrated from Oracle BI Enterprise Edition. After you migrate content from your Oracle BI Enterprise Edition environment to Oracle Analytics, you may experience different sorting behaviors in analyses. For example, if you look at a migrated analysis in Polish, the upper case and lower case letters might sort based on the Oracle Analytics default locale, not the original Oracle BI Enterprise Edition locale. To preserve the Oracle BI Enterprise Edition sort behavior in Oracle Analytics, change this setting to **Polish**. **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** DataQuerySortOrderLocale

Performance and Compatibility Options

You use these options to configure performance and compatibility settings between Oracle BI Enterprise Edition and Oracle Analytics. For example, you can set the maximum temporary file size.

LiveLabs Sprint

> **Note:**
>
> In Oracle BI Enterprise Edition 12c, you use configuration files to customize various settings. In Oracle Analytics Server, you can use the Console (System Settings page). If you upgrade from Oracle BI Enterprise Edition 12c to Oracle Analytics Server, your configuration file settings are preserved but no values are displayed on the System Settings page, that is, they appear blank. You don't have to re-enter values on the System Settings page but, for clarity, you might decide to do so.

System Setting	More Information
Always Use DBMS_RANDOM in Oracle Data Sources	Specifies whether Contextual Insight algorithms use the `DBMS_RANDOM` package in Oracle data sources to generate random numbers. • **On** — Contextual Insights always use the `DBMS_RANDOM` package in Oracle data sources for random number generation. This is the default. • **Off** — Contextual Insights uses the value of the RAND_SUPPORTED property to determine whether or not to use the `DBMS_RANDOM` database feature. Semantic modelers can set the RAND_SUPPORTED property to ON or OFF in the query features selection pane for their Oracle data source. See Modify a Database's Data Source Properties and Supported Query Features. If RAND_SUPPORTED is OFF, the `ORA_HASH` function is used. **Default:** On **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** AlwaysUseDBMSRANDOMOracleDataSources
Brushing Enabled for Data Sets	Specifies whether brushing is enabled by default for workbooks that use data from datasets. • **On** — Brushing is on by default for workbooks that use dataset data. • **Off** — Brushing is off by default for workbooks that use dataset data. Users can override this setting in the workbook and canvas properties. **Default:** On **Apply Required:** No **Change Effective:** Immediately **API Key:** EnableBrushingSubjectAreas
Brushing Enabled for Subject Areas	Specifies whether brushing is enabled by default for workbooks that use data from subject areas. • **On** — Brushing is on by default for workbooks that use subject area data. • **Off** — Brushing is off by default for workbooks that use subject area data. Users can override this setting in the workbook and canvas properties. **Default:** On **Apply Required:** No **Change Effective:** Immediately **API Key:** EnableBrushingSubjectAreas
Cache Dashboard Listing Dropdown Menu	Specifies how often the Dashboards menu list on the Oracle Analytics Classic Home page is populated during a user session. • **On** — Dashboards menu lists are only populated once per user session. This improves performance but may result in stale lists until the user logs out and back in which refreshes the lists. • **Off** — Dashboards menu lists are populated every time they're opened. **Default: Off** **Apply Required:** No **Change Effective:** After a few minutes **API Key:** CacheDashboardListingDropdownMenu

System Setting	More Information
Cache Enable	Specifies whether data query caching is enabled or disabled. • **On** — Data caching is enabled. • **Off** — Caching is disabled. See Configure Query Caching. **Default:** On **Apply Required:** Yes **Change Effective:** Immediately **API Key:** EnableDataQueryCache
Default Limit Values By for Filter	Determines the default behavior of the workbook filter option **Limit Values By**. • **Auto** — By default, workbook filters use the **Auto** behavior, that is, allow other filters in the workbook (if any) to limit values. This setting can improve user experience. • **None** — By default, workbook filters aren't limited by any other filter (if any) . This setting can improve performance. Users can override the default that you select here, within their workbooks. **Default:** Auto **Apply Required:** No **Change Effective:** After a few minutes **API Key:** DefaultLimitValuesByForFilters
Enable Auto Insights on Datasets	Specifies whether the Auto Insights feature is available when datasets are created or modified. • **On** — The Enable Insights option is available in the Dataset Inspect dialog and insights are automatically generated and available for workbooks that use datasets with the Enable Insights option selected. • **Off** — Auto Insights and its related features are disabled. **Default:** On **Apply Required:** No **Change Effective:** After a few minutes **API Key:** EnableAutoInsightsDatasets
Enable Database Analytics Node in Data Flows	Specifies whether the Database Analytics node is displayed in data flows. • **On** — The Database Analytics node is available in data flows so that data flow designers can apply database analytics functions to the data. • **Off** — The Database Analytics node isn't available in data flows. This prevents data flow designers from generating a potentially high number of SQL statements and slowing database performance. **Default:** On **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** EnableDatabaseAnalyticsNodeDataFlows

System Setting	More Information
Enable Immediate Dashboard Rendering	Specifies whether to display available dashboard content immediately or wait until all the dashboard content is ready. • **On** — Display dashboard content immediately even if some content is unavailable. • **Off** — Wait for all the dashboard content to be ready before displaying content. **Default:** Off **Apply Required:** No **Change Effective:** After a few minutes **API Key:** EnableImmediateDashboardRendering
Evaluate Support Level	Specifies who can issue database functions: `EVALUATE`, `EVALUATE_ANALYTIC`, `EVALUATE_AGGR`, and `EVALUATE_PREDICATE`. By default (`0`), the `EVALUATE` database functions are disabled. See EVALUATE_SUPPORT_LEVEL • **1** — Service administrators only. Users with the BI Service Administrator application role can invoke `EVALUATE` database functions. • **2** — Anyone. Any user who signs in to Oracle Analytics can invoke `EVALUATE` database functions. • **0** (or any other value) — No one. All `EVALUATE` database functions are disabled in Oracle Analytics. **Valid Values:** 0, 1, 2 **Default:** 0 **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** EvaluateSupportLevel
Load Semantic Models Using Multiple Threads	Specifies if semantic models load using multiple threads. If you find that your large datasets load slowly and impact system processing times, enabling this option may improve performance. • **On** — Semantic models load in parallel. • **Off** — Semantic models don't load in parallel. **Default: Off** **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** LoadSemanticModelsWithMultipleThreads
Maximum Query Limit (seconds)	Specifies the maximum length of time a single query can run before it's canceled and users see a timeout message. The default is 660 seconds (11 minutes). **Valid Values:** 60-660 **Default:** 660 **Apply Required:** No **Change Effective:** After a few minutes **API Key:** MaximumQueryLimit

System Setting	More Information
Maximum Working File Percent Size	Specifies that the temporary file doesn't exceed a specified percentage of the global work directory size limit. See MAX_WORK_FILE_SIZE_PERCENT. The size limit defaults for temporary files is 5% (of 100 GB), equivalent to 5 GB. The file limit applies individually to each temporary file, while the size specified for the total global work directory applies collectively to all temporary files created. You can increase or decrease this value within the range of 5% to 50%. This enables temporary file sizes between 5GB and 50GB. Raising this setting above 50% limits concurrency for large operations. **Valid Values:** 5-50 **Default:** 5 **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** MaximumWorkingFilePercentSize
Mobile Data Watch Service Enabled	Specifies if the Data Watch service is active. • **On** — The Data Watch service is active and all mobile users can specify the threshold value for bring backs. • **Off** — The Data Watch service is inactive. **Default:** On **Apply Required:** No **Change Effective:** Immediately **API Key:** EnableMobileDataWatchService
Mobile Data Watch Service Frequency	Specifies the frequency at which the Data Watch service must scan the server for changes based on the frequency of changes in your data sources. The default is 240 (4 hours). You can disable this service by changing this setting frequency to 0 or by toggling the **Mobile Watch Service Enabled** setting to off. **Valid Values:** 0-10139 **Default:** 240 **Apply Required:** No **Change Effective:** Immediately **API Key:** MobileDataWatchServiceFrequency
OBIEE Compatibility Release	Specifies the on-premise Oracle BI Enterprise Edition version number for feature compatibility. This only applies if you upgrade from Oracle BI Enterprise Edition to Oracle Analytics, and you want to use a feature from a specific on-premises release in Oracle Analytics. **Valid Values:** 11.1.1.9, 11.1.1.10, 11.1.1.11, 12.2.1.0, 12.2.1.1, 12.2.1.3, 12.2.1.4, 12.2.2.0, 12.2.3.0, 12.2.4.0, 12.2.5.0 **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** OBIEECompatibilityRelease

System Setting	More Information
Override Database Features	Specifies if users can use request variables to override database features. • **1** — Only administrators can override database features. • **2** — Any user can override database features. • **0** — No user can override database features. **Valid Values:** 0, 1 , 2 **Default: 0** **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** OverrideDatabaseFeatures
Query Limit Extension	Determines whether the query limit can extend to 60 minutes to accommodate the occasional, longer running query. • **On** — The query limit can be extended to 60 minutes. • **Off** — The **Maximum Query Limit** setting on this page is used and never extends. **Default:** Off **Apply Required:** No **Change Effective:** After a few minutes **API Key:** QueryLimitExtension
Restrict Data Export and Delivery	Restricts the maximum number of rows users can export or deliver by email in formatted and unformatted content. Data export and delivery limits depend on the size of your Oracle Analytics service. **Valid Values:** Maximum - no restriction, 90% of Maximum, 80% of Maximum, 70% of Maximum, 60% of Maximum, 50% of Maximum, 40% of Maximum, 30% of Maximum, 20% of Maximum, 10% of Maximum, Minimum - 1000 rows **Default: Maximum - no restriction** **Apply Required:** No **Change Effective:** After a few minutes **API Key:** RestrictDataExportAndDelivery
Strong Datetime Type Checking	Specifies whether to enforce strict checking for date and time data types and whether to reject queries that contain incompatibilities in date and time data types. • **On** — Enforces strict checking for date and time data types. • **Off** — Relaxes strict checking for date and time data types. However, invalid queries or queries with severe date and time incompatibilities may still be still rejected. For example, date and time incompatibilities might be rejected if your relational database uses strict checking for those data types. See STRONG_DATETIME_TYPE_CHECKING. **Default:** On **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** StrongDatetimeTypeChecking

Prompt Options

You use these options to configure prompt behavior in analyses and dashboards. For example, you can enable search results to automatically display as highlighted when users enter search parameters, without the need to click **Search**.

These options apply only to analyses and dashboards. They don't apply to data visualizations.

See Configure for Prompts.

> **Note:**
>
> In Oracle BI Enterprise Edition 12c, you use configuration files to customize various settings. In Oracle Analytics Server, you can use the Console (System Settings page). If you upgrade from Oracle BI Enterprise Edition 12c to Oracle Analytics Server, your configuration file settings are preserved but no values are displayed on the System Settings page, that is, they appear blank. You don't have to re-enter values on the System Settings page but, for clarity, you might decide to do so.

System Setting	More Information
Auto Apply Dashboard Prompt Values	Enables the option to hide the **Apply** button so that prompt values can be applied without clicking any button. If this is property is **On**: • Displays the **Show Apply Button** and **Show Reset Button** fields in the Edit Page Settings dialog. • Displays the **Prompts Apply Buttons** and **Prompts Reset Buttons** fields in the Dashboard Properties dialog. • Displays the **Prompt Buttons on Current Page** option on the dashboard builder's Tools menu. **Default:** On **Apply Required:** No **Change Effective:** After a few minutes **API Key:** AutoApplyDashboardPromptValues
Auto Search on Prompt Value Search Dialog	Enables search results to automatically display and highlight when users enter search parameters, without the need to click **Search**. **Default:** On **Apply Required:** No **Change Effective:** After a few minutes **API Key:** EnableAnalysisAutoSearchPromptDialog
Case Insensitive Auto Complete	Specifies whether, when a user enters a prompt value in analyses and dashboards, the auto-complete functionality is case-insensitive. • **On** — Case isn't considered when a user enters a prompt value such as "Oracle" or "oracle." • **Off** — Case is considered when a user enters a prompt value, so the user must enter "Oracle" and not "oracle" to find the Oracle record. **Default:** On **Apply Required:** No **Change Effective:** After a few minutes **API Key:** AutoCompletePromptDropDownsCaseInsensitive

System Setting	More Information
Show Null Value When Column Is Nullable	Specifies whether to show the term "NULL" at runtime in the column prompt above the column separator in the drop-down list when the database allows null values. • **always** — Always shows the term "NULL" above the column separator in the drop-down list. • **never** — Never shows the term "NULL" in the drop-down list. • **asDataValue** — Displays the data value in the drop-down list, not the term "NULL" above the separator in the drop-down list. **Valid Values:** always, never, asDataValue **Default:** always **Apply Required:** No **Change Effective:** After a few minutes **API Key:** AnalysisPromptsShowNullValueWhenColumnIsNullable
Support Auto Complete	Enables or disables the auto-complete functionality available in prompts. • **On** — Enables auto-complete, which means that the **Prompts Auto-Complete** field is displayed and set to **On** in the My Account dialog and in the Dashboard Properties dialog. • **Off** — Disables auto-complete, which means that the auto-complete fields in the My Account and Dashboard Properties dialogs aren't available. **Default:** Off **Apply Required:** No **Change Effective:** After a few minutes **API Key:** EnableAnalysisAutoCompletePrompt

Security Options

Use Security options to control how users can perform specific actions in analyses and dashboards.

Most of these options apply only to analyses and dashboards. They don't apply to data visualizations or pixel-perfect reports. The exception is **User Inactivity Timeout**, which applies to analyses, dashboards, and data visualizations.

> **Note:**
>
> In Oracle BI Enterprise Edition 12c, you use configuration files to customize various settings. In Oracle Analytics Server, you can use the Console (System Settings page). If you upgrade from Oracle BI Enterprise Edition 12c to Oracle Analytics Server, your configuration file settings are preserved but no values are displayed on the System Settings page, that is, they appear blank. You don't have to re-enter values on the System Settings page but, for clarity, you might decide to do so.

System Setting	More Information
Allow HTML/JavaScript/CSS Content	Determines whether users can apply and save HTML, JavaScript, and CSS markup in various text fields for analyses and dashboards, and how any previously saved markup is used. • **Always** — Enables users to apply markup. Displays the **Contains HTML/JavaScript/CSS Markup** option in dialogs where additional formatting might be useful. For example: – For analyses: Various dialogs in the analysis editor, Analysis Properties dialog, Column Properties (Column Format) dialog, Edit Column Formula dialog, Narrative dialog, Ticker dialog, Static Text dialog, and New Calculated Measure dialog. – For dashboards: Various dialogs in the dashboard editor, Text Properties dialog, and Edit Header and Edit Footer dialogs (under Print and Export Options). • **Never** — Prevents users from applying markup. Hides the **Contains HTML/JavaScript/CSS Markup** option. Users can only enter plain text. Oracle Analytics ignores any markup that users previously entered and saved for their analyses and dashboards. • **HTML Only** — Enables users to apply HTML markup. Displays the **Contains HTML/JavaScript/CSS Markup** option in dialogs where additional formatting might be useful but only safe HTML is allowed (no JavaScript or CSS). When an analysis or dashboard opens, Oracle Analytics sanitizes any markup that users have entered and applies only the HTML markup. • **On Open** — Prevents users from applying additional markup (existing markup is retained). Hides the **Contains HTML/JavaScript/CSS Markup** option so users can enter only plain text. Any previously saved markup for analyses and dashboards continues to be applied. **Note:** The **On Open** option was previously named "Off". For security reasons, markup is not allowed when you upgrade from Oracle BI Enterprise Edition 12c to Oracle Analytics Server. To turn the feature on in Oracle Analytics Server, you must enable **Allow HTML/JavaScript/CSS Content**. **Default:** On Open **Apply Required:** No **Change Effective:** After a few minutes **API Key:** AllowHTMLJavaScriptCSSContent
Enable Push Notifications	Specifies if mobile application push notifications are enabled (on) or disabled (off). • **On** — Enables mobile application push notifications to receive alerts and messages. • **Off** — Disables mobile application push notifications to stop receiving alerts and messages. **Default:** On **Apply Required:** No **Change Effective:** Immediately **API Key:** EnableMobilePushNotifications

System Setting	More Information
Export Data to CSV and Tab-Delimited Files as Text	Specifies if leading apostrophes are added when data is exported to CSV or tab-delimited files, so all fields are treated as text. • **On** — Leading apostrophes are automatically added to CSV and tab-delimited files during exports. • **Off** — Data is exported to CSV files as-is. **Caution:** When **Off**, opening exported CSV files may invoke unwanted formulas. See Export the Results of Analyses. **Note:** This setting applies only to visualizations and analyses. It doesn't apply to pixel-perfect reports. **Default:** Off **Apply Required:** No **Change Effective:** After a few minutes **API Key:** ExportDataToCSVFilesAsText

System Setting	More Information
Save Workbook Thumbnails	To help people identify workbook content, Oracle Analytics can display thumbnail images for workbooks on the Home page. The information that's shown in these thumbnails is blurred to protect sensitive data from being exposed to users that don't have the same access as data authors. This setting overrides any **Save thumbnails** value set in the Workbook Properties dialog at the individual workbook level. **Note:** This setting doesn't apply to watchlists because they don't use thumbnails. Instead, watchlists display miniaturized visualizations that reload whenever you refresh the Home page. • **On** — Display blurred workbook thumbnails on the Home page. If this setting enabled (on), workbook owners can hide the thumbnail for individual workbooks if they need to. See Set Workbook Thumbnails. • **Off** — Don't display any workbook thumbnails on the Home page. Instead, show the standard icon for all workbooks. **Default: On** **Apply Required:** No **Change Effective:** After a few minutes **API Key:** SaveWorkbookThumbnail
Sign Out Inactive Users Automatically	Specifies whether to automatically sign out users after the inactivity timeout is reached. • **On** — Users are automatically signed out when the inactivity timeout is reached. • **Off** — Users remain signed in even if the inactivity timeout is reached. **Default:** Off **Apply Required:** No **Change Effective:** After a few minutes **API Key:** SignOutInactiveUsersAutomatically

System Setting	More Information
TLS Connection Mode	Specifies the security configuration that's used on external data source connections that use TLS. • **Enhanced** — Use the latest encryption and security configuration for external connections that use TLS. Before you switch to using the **Enhanced** mode, Oracle recommends that you test all your connections to database servers, SMTP servers, and other customer-managed servers that use TLS. To learn how, sign in to My Oracle Support (https://support.oracle.com) and read the article TLS Security Updates for Connections in Oracle Analytics (Doc 3029871.1). • **Legacy** — Use the existing legacy TLS connection mode. This setting applies to both public and private data source connections. **Default:** Legacy **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** TLSConnectionMode
URL for Browser Script Actions	Specifies the URL for the JavaScript file containing custom Browser Script Actions. **Apply Required:** No **Change Effective:** After a few minutes **API Key:** URLBrowserScriptActions
User Inactivity Timeout (minutes)	Specifies the number of minutes users are inactive before their browser or mobile connection must be re-authenticated. This system setting applies to analyses, dashboards, and data visualizations. **Valid Values:** 5 - 480 **Default:** 60 **Apply Required:** No **Change Effective:** After a few minutes **API Key:** UserInactivityTimeout

Usage Tracking Options

You use these options to specify how you want to monitor system usage. For example, you can set the number of rows that you want stored in the usage tracking tables.

> **Note:**
>
> In Oracle BI Enterprise Edition 12c, you set options in configuration files to configure usage tracking. If you upgrade from Oracle BI Enterprise Edition 12c to Oracle Analytics Server, usage tracking settings are preserved and usage tracking continues to work but no values are displayed on the System Settings page, that is, they appear blank. You don't have to re-enter your usage tracking details through System Settings but, for clarity, you might decide to do so.

See Set Usage Tracking Parameters.

System Setting	More Information
Enable Usage Tracking	Specifies whether usage tracking is enabled. You must enable this setting to activate all other settings in the Usage Tracking section of this page. • **On** — Any enabled settings in the Usage Tracking section of this page are activated. • **Off** — No settings in the Usage Tracking section of this page are activated, even if they are enabled. **Default:** On **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** EnableUsageTracking
Usage Tracking Connection Pool	Specifies the name of the connection pool you created for your usage tracking statistics database. For example, `<database name>.<connection pool name>`. **API Key:** UsageTrackingConnectionPool **Apply Required:** Yes **Change Effective:** After a few minutes
Usage Tracking Init Block Table	Specifies the name of the fully-qualified database table you use for inserting records that correspond to the initialization block statistics, as it appears in the physical layer of your semantic model. For example, `<database name>.<catalog name >.<schema name >.<table name> or <database name>.<schema name >.<table name>`. **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** UsageTrackingInitBlockTable
Usage Tracking Logical Query Logging Table	Specifies the name of the database table you want to use to store logical query details. For example, `<database name>.<catalog name >.<schema name >.<table name> or <database name>.<schema name >.<table name>`. **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** UsageTrackingLogicalQueryLoggingTable
Usage Tracking Maximum Rows	Indicates the number of rows allowed in usage tracking tables, with a value of **0** indicating an unlimited number of rows. **Valid Values:** Any positive number (up to 64-bit integer) **Default:** 0 **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** UsageTrackingMaximumRows
Usage Tracking Physical Query Logging Table	Specifies the name of the database table you want to use to store physical query details. For example, `<database name>.<catalog name >.<schema name >.<table name> or <database name>.<schema name >.<table name>`. **Apply Required:** Yes **Change Effective:** After a few minutes **API Key:** UsageTrackingPhysicalQueryLoggingTable

Note: See Recommended Configuration Settings for Usage Tracking using SQL Server.

View Options

You use these options to configure default search and viewing settings for users working with analyses and dashboards.

These options apply only to analyses and dashboards. They don't apply to data visualizations.

> **Note:**
>
> If you change a view setting, you must apply the change for the new value to take effect.

> **Note:**
>
> In Oracle BI Enterprise Edition 12c, you use configuration files to customize various settings. In Oracle Analytics Server, you can use the Console (System Settings page). If you upgrade from Oracle BI Enterprise Edition 12c to Oracle Analytics Server, your configuration file settings are preserved but no values are displayed on the System Settings page, that is, they appear blank. You don't have to re-enter values on the System Settings page but, for clarity, you might decide to do so.

See Configure and Manage Analyses and Dashboards.

System Setting	More Information
Default Scrolling Enabled	Specifies how data scrolls in tables, pivots, heat matrix, and simple and advanced trellis visualizations. • **On** — Data displays with a fixed header and content scrolling controls for users to browse the data. • **Off** — Data displays with content paging controls for users to browse the data. **Default:** On **Apply Required:** No **Change Effective:** After a few minutes **API Key:** `AnalysisDefaultScrollingEnabled`
Enable Enrichments in Workbooks	Specifies whether workbook editors can add dataset enrichments to a visualization directly from the Data Panel. This setting enables enrichments in workbooks for all users. Workbook editors who own a dataset or have editing privileges for it can enable or disable knowledge enrichments for that dataset using the **Enable Knowledge Enrichments** option. See Enable Knowledge Enrichments in the Workbook Editor. • **On** — Workbook editors can drag and drop enrichment based data elements to visualization canvases. • **Off** — Knowledge enrichments are not available for datasets. **Default:** On **Apply Required:** No **Change Effective:** After a few minutes **API Key:** `EnableEnrichmentsInWorkbook`

System Setting	More Information
Enable Personalization in Workbooks	Specifies whether users can personalize workbooks. • **On** — Content designers can enable or disable personalization options (**Filter** and **Parameter**) in their workbooks. • **Off** — Personalization options aren't available for workbooks. **Default:** On **Apply Required:** No **Change Effective:** After a few minutes **API Key:** `EnableUserPersonalizationInWorkbooks`
Prompt Auto Complete Matching Level	Specifies whether the auto-complete functionality uses matching to find the prompt value that the user enters into the prompt field. This setting doesn't apply if the user accesses the Search dialog to locate and specify a prompt value. • **StartsWith** — Searches for a match that begins with the text that the user types. For example, the user types `M` and the following stored values are displayed: MicroPod and MP3 Speakers System. • **WordStartsWith** — Searches for a match at the beginning of a word or group of words. For example, the user types `C` and the following values are displayed: ComCell, MPEG Camcorder, and 7 Megapixel Digital Camera. • **MatchAll** — Searches for any match within the word or words. **Valid Values:** StartsWith, WordStartsWith, MatchAll **Default:** MatchAll **Apply Required:** No **Change Effective:** After a few minutes **API Key:** `AnalysisPromptAutoCompleteMatchingLevel`
Table/Pivot View: Maximum Visible Rows	Specifies the maximum number of rows you want displayed for content paging in table and pivot table views in analyses and dashboards. The minimum number of rows you can specify to display is 100. **Valid Values:** 100-5000 **Default:** 5000 **Apply Required:** No **Change Effective:** After a few minutes **API Key:** `TablePivotViewMaximumVisibleRows`
View Interactions: Add/ Remove Values	Specifies whether the **Add/Remove Values** option is selected by default in the Analysis Properties dialog: Interactions tab. • **On** — The **Add/Remove Values** option is selected by default in the Analysis Properties dialog: Interactions tab. • **Off** — The **Add/Remove Values** option isn't selected by default in the Analysis Properties dialog: Interactions tab. **Default:** Off **Apply Required:** No **Change Effective:** After a few minutes **API Key:** `AnalysisViewInteractionsAddRemoveValues`

System Setting	More Information
View Interactions: Create/Edit/ Remove Calculated Items	Specifies whether the **Create/Edit/Remove Calculated Items** option is selected by default in the Analysis Properties dialog: Interactions tab. • **On** — The **Create/Edit/Remove Calculated Items** option is selected by default in the Analysis Properties dialog: Interactions tab. • **Off** — The **Create/Edit/Remove Calculated Items** option isn't selected by default in the Analysis Properties dialog: Interactions tab. **Default:** Off **Apply Required:** No **Change Effective:** After a few minutes **API Key:** `AnalysisViewInteractionsCreateEditRemoveCalculatedItems`
View Interactions: Create/Edit/ Remove Groups	Specifies whether the **Create/Edit/Remove Groups** option is selected by default in the Analysis Properties dialog: Interactions tab. • **On** — The **Create/Edit/Remove Groups** option is selected by default in the Analysis Properties dialog: Interactions tab. • **Off** — The **Create/Edit/Remove Groups** option isn't selected by default in the Analysis Properties dialog: Interactions tab. **Default:** Off **Apply Required:** No **Change Effective:** After a few minutes **API Key:** `AnalysisViewInteractionsCreateEditRemoveGroups`
View Interactions: Display/ Hide Running Sum	Specifies whether the **Display/Hide Running Sum** option is selected by default in the Analysis Properties dialog: Interactions tab. • **On** — The **Display/Hide Running Sum** option is selected by default in the Analysis Properties dialog: Interactions tab. • **Off** — The **Display/Hide Running Sum** option isn't selected by default in the Analysis Properties dialog: Interactions tab. **Default:** Off **Apply Required:** No **Change Effective:** After a few minutes **API Key:** `AnalysisViewInteractionsDisplayHideRunningSum`
View Interactions: Display/ Hide Sub-totals	Specifies whether the **Display/Hide Sub-totals** option is selected by default in the Analysis Properties dialog: Interactions tab. • **On** — The **Display/Hide Sub-totals** option is selected by default in the Analysis Properties dialog: Interactions tab. • **Off** — The **Display/Hide Sub-totals** option isn't selected by default in the Analysis Properties dialog: Interactions tab. **Default:** Off **Apply Required:** No **Change Effective:** After a few minutes **API Key:** `AnalysisViewInteractionsDisplayHideSubtotals`

System Setting	More Information
View Interactions: Drill	Specifies whether the **Drill** (when not a primary interaction) option is selected by default in the Analysis Properties dialog: Interactions tab. • **On** — The **Drill** (when not a primary interaction) option is selected by default in the Analysis Properties dialog: Interactions tab. • **Off** — The **Drill** (when not a primary interaction) option isn't selected by default in the Analysis Properties dialog: Interactions tab. **Default:** Off **Apply Required:** No **Change Effective:** After a few minutes **API Key:** `AnalysisViewInteractionsDrill`
View Interactions: Include/ Exclude Columns	Specifies whether the **Include/Exclude Columns** option is selected by default in the Analysis Properties dialog: Interactions tab. • **On** — The **Include/Exclude Columns** option is selected by default in the Analysis Properties dialog: Interactions tab. • **Off** — The **Include/Exclude Columns** option isn't selected by default in the Analysis Properties dialog: Interactions tab. **Default:** On **Apply Required:** No **Change Effective:** After a few minutes **API Key:** `AnalysisViewInteractionsIncludeExcludeColumns`
View Interactions: Move Columns	Specifies whether the **Move Columns** option is selected by default in the Analysis Properties dialog: Interactions tab. • **On** — The **Move Columns** option is selected by default in the Analysis Properties dialog: Interactions tab. • **Off** — The **Move Columns** option isn't selected by default in the Analysis Properties dialog: Interactions tab. **Default:** On **Apply Required:** No **Change Effective:** After a few minutes **API Key:** `AnalysisViewInteractionsMoveColumns`
View Interactions: Sort Columns	Specifies whether the **Sort Columns** option is selected by default in the Analysis Properties dialog: Interactions tab. • **On** — The **Sort Columns** option is selected by default in the Analysis Properties dialog: Interactions tab. • **Off** — The **Sort Columns** option isn't selected by default in the Analysis Properties dialog: Interactions tab. **Default:** On **Apply Required:** No **Change Effective:** After a few minutes **API Key:** `AnalysisViewInteractionsSortColumns`

Configure System Settings Using Console

Use Console to configure and customize system settings to suit your Oracle Analytics Server environment.

1. In the Oracle Analytics Home page, click the **Navigator**, and then click **Console**.
2. Click **Advanced System Settings**.

3. Update the property value.
4. Click **Apply** to save you changes and then click **OK** to confirm.

   Wait a few moments for the changes to refresh through the system.

Manage System Settings Using REST APIs

You can use Oracle Analytics Server REST APIs to programmatically view and manage system settings. For example, you might create a script to update usage tracking options.

- Typical Workflow for Using System Setting REST APIs
- REST API Key Values for System Settings
- Examples for System Setting REST APIs

Typical Workflow for Using System Setting REST APIs

Here are the common tasks to start using Oracle Analytics Server REST APIs to programmatically view and manage system settings. If you're using system setting REST APIs for the first time, follow these tasks as a guide.

Task	Description	REST API Documentation
Understand prerequisites	Understand and complete several prerequisite tasks. You must have administrator permissions in Oracle Analytics Server to manage system settings using REST APIs (**BI Service Administrator**).	Quick Start tasks
Understand the authentication mechanism allowed by your configuration.	Authentication in Oracle Analytics Server is managed using the same authentication mechanism as your Oracle Analytics Server environment.	Authenticate
Understand API keys for system settings	Each system setting has a public API key name that you can use in REST API operations.	REST API Key Values for System Settings
Get system setting details	Get details about specific system settings, all system settings, or system settings that haven't been applied yet.	Get system settings
Update system settings	Update one or more system setting.	Update system settings

REST API Key Values for System Settings

Each system setting has its own REST API key name. For example, the REST API key for the system setting **Cache Enable** is **EnableDataQueryCache**. You use this REST API key to identify the system setting in a REST API operation.

For example, if you want to disable the **Cache Enable** setting, you specify the name of the key (`EnableDataQueryCache`) with the value `false`.

```
{
     "items":[
       {
         "key": "EnableDataQueryCache",
         "value": "false"
       }
```

```
    ]
}
```

System Setting Display Name	REST API Key Value
Allow HTML/JavaScript/CSS Content	AllowHTMLJavaScriptCSSContent
Always Use DBMS_RANDOM in Oracle Data Sources	AlwaysUseDBMSRANDOMOracleDataSources
Analytics Publisher Reporting Toolbar Mode	AnalyticsPublisherReportingToolbarMode
Answers Editor Start Tab	AnswersEditorStartTab
Answers Subject Area Sorting Order	AnalysisSubjectAreaSortingOrder
Auto Apply Dashboard Prompt Values	AutoApplyDashboardPromptValues
Auto Search on Prompt Value Search Dialog	EnableAnalysisAutoSearchPromptDialog
Brushing Enabled for Data Sets	EnableBrushingDatasets
Brushing Enabled for Subject Areas	EnableBrushingSubjectAreas
Cache Dashboard Listing Dropdown Menu	CacheDashboardListingDropdownMenu
Cache Enable	EnableDataQueryCache
Case Insensitive Auto Complete	AutoCompletePromptDropDownsCaseInsensitive
Connection Externalization Enabled	EnableConnectionExternalization
Currencies XML	AnalysisCurrenciesXml
Custom Links XML	CustomLinksXml
Default Data Offset Time Zone	DefaultDataOffsetTimeZone
Default Limit Values By for Filters	DefaultLimitValuesByForFilters
Default Scrolling Enabled	AnalysisDefaultScrollingEnabled
Default Time Zone for Date Calculations	DefaultTimeZoneforDateCalculations
Default User Preferred Time Zone	DefaultUserPreferredTimeZone
Disable Right Trim for VARCHAR Data	DataQueryDisableRightTrimVARCHARData
Enable Auto Insights on Datasets	EnableAutoInsightsDatasets
Enable Database Analytics Node in Data Flows	EnableDatabaseAnalyticsNodeDataFlows
Enable Enrichments in Workbooks	EnableEnrichmentsInWorkbook
Enable Immediate Dashboard Rendering	EnableImmediateDashboardRendering
Enable Personalization in Workbooks	EnablePersonalizationInWorkbooks
Enable Push Notifications	EnableMobilePushNotifications
Enable Subrequest Shipping	EnableSubrequestShipping
Enable Usage Tracking	EnableUsageTracking
Enforce Safe Domains in Actions	EnforceSafeDomainsActions
Evaluate Support Level	EvaluateSupportLevel
Export Data to CSV and Tab-Delimited Files as Text	ExportDataToCSVFilesAsText
Hide EPM Cloud Members with No Access	HideEPMCloudMembersWithNoAccess
Hide loading messages	HideLoadingMessages
Load Semantic Models Using Multiple Threads	LoadSemanticModelsWithMultipleThreads
Locale	DataQueryLocale
Maximum Email Size (KB)	EmailMaxEmailSizeKB
Maximum Number of Recipients per Email	EmailMaxRecipients
Maximum Query Limit (seconds)	MaximumQueryLimit
Maximum Working File Percent Size	MaximumWorkingFilePercentSize
Mobile Data Watch Service Enabled	EnableMobileDataWatchService
Mobile Data Watch Service Frequency	MobileDataWatchServiceFrequency

System Setting Display Name	REST API Key Value
OBIEE Compatibility Release	OBIEECompatibilityRelease
Override Database Features	OverrideDatabaseFeatures
Portal Path	PortalPath
Prompt Auto Complete Matching Level	AnalysisPromptAutoCompleteMatchingLevel
Query Limit Extension	QueryLimitExtension
Recursive Datetime Type Checking	RecursiveDatetimeTypeChecking
Repeat Rows on Excel Exports for Tables and Pivots	AnalysisRepeatRowsExcelExportsTablesPivots
Restrict Data Export and Delivery	RestrictDataExportAndDelivery
Safe Domains	EmailSafeDomains
Save Workbook Thumbnail	SaveWorkbookThumbnail
Show Null Value when Column is Nullable	AnalysisPromptsShowNullValueWhenColumnIsNullable
Sign Out Inactive Users Automatically	SignOutInactiveUsersAutomatically
Sort Null Values First	SortNullValuesFirst
Sort Order Locale	DataQuerySortOrderLocale
Strong Datetime Type Checking	StrongDatetimeTypeChecking
Support Auto Complete	EnableAnalysisAutoCompletePrompt
Table/Pivot View: Maximum Visible Rows	TablePivotViewMaximumVisibleRows
TLS Connection Mode	TLSConnectionMode"
URL for Blocking Queries in Analyses	QueryBlockingScriptURL
URL for Browser Script Actions	URLBrowserScriptActions
Usage Tracking Connection Pool	UsageTrackingConnectionPool
Usage Tracking Init Block Table	UsageTrackingInitBlockTable
Usage Tracking Logical Query Logging Table	UsageTrackingLogicalQueryLoggingTable
Usage Tracking Maximum Rows	UsageTrackingMaximumRows
Usage Tracking Physical Query Logging Table	UsageTrackingPhysicalQueryLoggingTable
Use BCC	EmailUseBcc
Use RFC 2231 Encoding	EmailUseRFC2231
User Currency Preferences XML	UserCurrencyPreferencesXml
User Inactivity Timeout (minutes)	UserInactivityTimeout
View Interactions: Add/Remove Values	AnalysisViewInteractionsAddRemoveValues
View Interactions: Create/Edit/Remove Calculated Items	AnalysisViewInteractionsCreateEditRemoveCalculatedItems
View Interactions: Create/Edit/Remove Groups	AnalysisViewInteractionsCreateEditRemoveGroups
View Interactions: Display/Hide Running Sum	AnalysisViewInteractionsDisplayHideRunningSum
View Interactions: Display/Hide Sub-totals	AnalysisViewInteractionsDisplayHideSubtotals
View Interactions: Drill	AnalysisViewInteractionsDrill
View Interactions: Include/Exclude Columns	AnalysisViewInteractionsIncludeExcludeColumns
View Interactions: Move Columns	AnalysisViewInteractionsMoveColumns
View Interactions: Sort Columns	AnalysisViewInteractionsSortColumns
WriteBack Template XML	WriteBackTemplateXML

Examples for System Setting REST APIs

REST API for Oracle Analytics Server includes several examples that explain how to use the System Setting REST APIs.

Get system settings - Examples

- Example 1 - Get a list of all the system settings and their current values
- Example 2 - Get current values for a specific set of system settings
- Example 3 - Get a list of system settings that aren't applied yet

Update system settings - Examples

- Example 1 - Update system settings using a JSON file
- Example 2 - Update system settings directly

Manage Content

Administrators can manage Oracle Analytics content from the Console. For example, if an employee leaves an organization, you might assign ownership of their workbooks and machine learning models to a different employee.

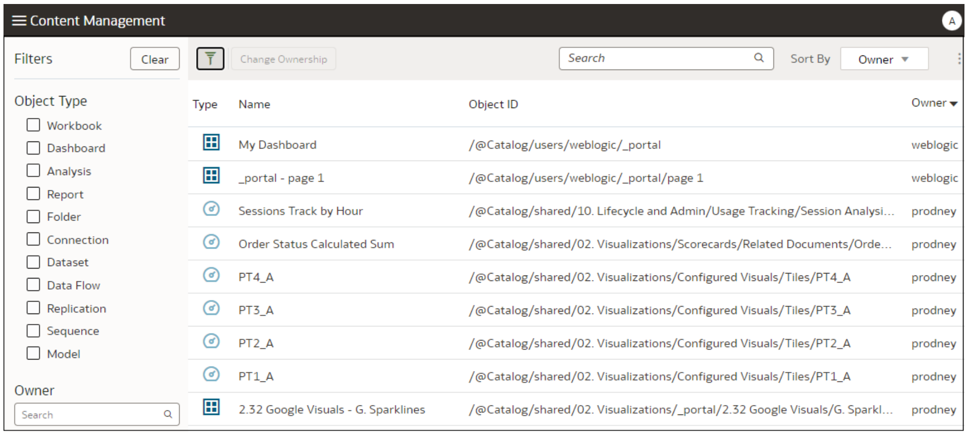
Topics

- Overview of Content Management
- Change Ownership of Content
- Change Ownership of Content in a User's Private Folder
- Frequently Asked Questions About Content Management

Overview of Content Management

Oracle Analytics enables you to view and manage Oracle Analytics content. For example, if an employee leaves an organization, you can reassign their workbooks and machine learning models to a different employee.

As an administrator, you can use the Content Management page to view, manage, and change ownership for all content types.



☐ Workbook
☐ Dashboard
☐ Analysis
☐ Report
☐ Folder
☐ Connection
☐ Dataset
☐ Data Flow
☐ Replication
☐ Sequence
☐ Model

Type	Name	Object ID	Owner ▾
⊞	My Dashboard	/@Catalog/users/weblogic/_portal	weblogic
⊞	_portal - page 1	/@Catalog/users/weblogic/_portal/page 1	weblogic
◎	Sessions Track by Hour	/@Catalog/shared/10. Lifecycle and Admin/Usage Tracking/Session Analysi...	prodney
◎	Order Status Calculated Sum	/@Catalog/shared/02. Visualizations/Scorecards/Related Documents/Orde...	prodney
◎	PT4_A	/@Catalog/shared/02. Visualizations/Configured Visuals/Tiles/PT4_A	prodney
◎	PT3_A	/@Catalog/shared/02. Visualizations/Configured Visuals/Tiles/PT3_A	prodney
◎	PT2_A	/@Catalog/shared/02. Visualizations/Configured Visuals/Tiles/PT2_A	prodney
◎	PT1_A	/@Catalog/shared/02. Visualizations/Configured Visuals/Tiles/PT1_A	prodney
⊞	2.32 Google Visuals - G. Sparklines	/@Catalog/shared/02. Visualizations/_portal/2.32 Google Visuals/G. Sparkl...	prodney

From the **Actions** menu for each item, you can also use the **Open in Classic Catalog** option to display the catalog folder where the item is stored so that you can make other configuration changes. For example, to change an item's properties or permissions, hover over over the item, click **Actions** at the far right-hand side, and click **Open in Classic Catalog**. **Note**: You need to own the item to see the **Open in Classic Catalog** option.

About Content Ownership

As an administrator, you can change ownership to:

- Yourself, as administrator.
- A different user.
- Every user with a specific application role (some restrictions apply, see Frequently Asked Questions About Content Management).

If you own content, you have these privileges:

- If you own an object with an Object ID prefixed with /@Catalog/, you can review the properties of that object and change permissions even if you have no other permissions on it.
- If you own an object with an Object ID prefixed with /@default/, you always have full permissions on that object.

Change Ownership of Content

You can change the ownership of Oracle Analytics content from the Console. For example, if an employee leaves your organization, you can reassign their workbooks and machine learning models to different employees so that they can use them.

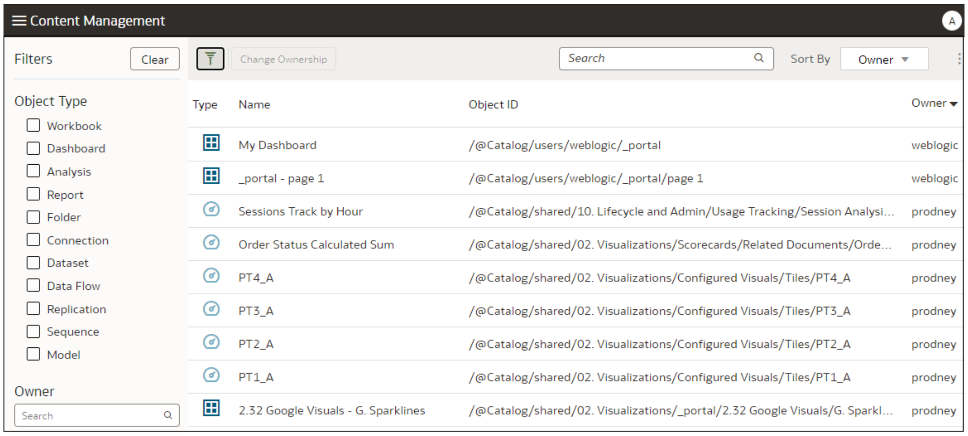
Changing ownership enables you to reuse analytics content if the original content author is no longer in your organization. You can also quickly provide analytics users with access to analytics content.

Depending on the object, you can assign ownership to yourself, another user, or a role:

- If you select an object with an object ID that starts with /@default/, you can assign it to another user.
- If you select an object with an object ID that starts with /@Catalog/, you can assign it to another user or to an application role.
- If you want to assign multiple objects to an application role, make sure you select only objects with object IDs starting with /@Catalog/.

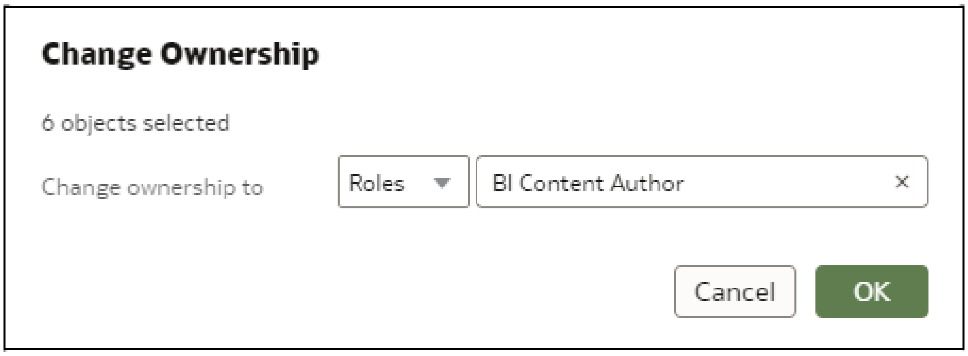
To change ownership of content in a user's private folder, see Change Ownership of Content in a User's Private Folder.

1. In the Oracle Analytics Home page, click the **Navigator**, and then click **Console**.
2. Click **Content Management**.



Type	Name	Object ID	Owner
▦	My Dashboard	/@Catalog/users/weblogic/_portal	weblogic
▦	_portal - page 1	/@Catalog/users/weblogic/_portal/page 1	weblogic
◎	Sessions Track by Hour	/@Catalog/shared/10. Lifecycle and Admin/Usage Tracking/Session Analysi...	prodney
◎	Order Status Calculated Sum	/@Catalog/shared/02. Visualizations/Scorecards/Related Documents/Orde...	prodney
◎	PT4_A	/@Catalog/shared/02. Visualizations/Configured Visuals/Tiles/PT4_A	prodney
◎	PT3_A	/@Catalog/shared/02. Visualizations/Configured Visuals/Tiles/PT3_A	prodney
◎	PT2_A	/@Catalog/shared/02. Visualizations/Configured Visuals/Tiles/PT2_A	prodney
◎	PT1_A	/@Catalog/shared/02. Visualizations/Configured Visuals/Tiles/PT1_A	prodney
▦	2.32 Google Visuals - G. Sparklines	/@Catalog/shared/02. Visualizations/_portal/2.32 Google Visuals/G. Sparkl...	prodney

3. Locate the items whose ownership you'd like to reassign:
   - To locate all objects belonging to a user, click **Filters**, then enter the user's username in the **Owner** field. You can further refine the selection using the **Object Type** options.
   - Use the **Object Type** options to restrict the list to specific types (click **Filters** to display).
   - Use the **Search** box to locate text in the **Name** field. For example, enter 'cluster' to display objects with cluster in the name.
4. Click to select an item or use Ctrl and click to select multiple items.
5. Click **Change Ownership**.



Change Ownership

6 objects selected

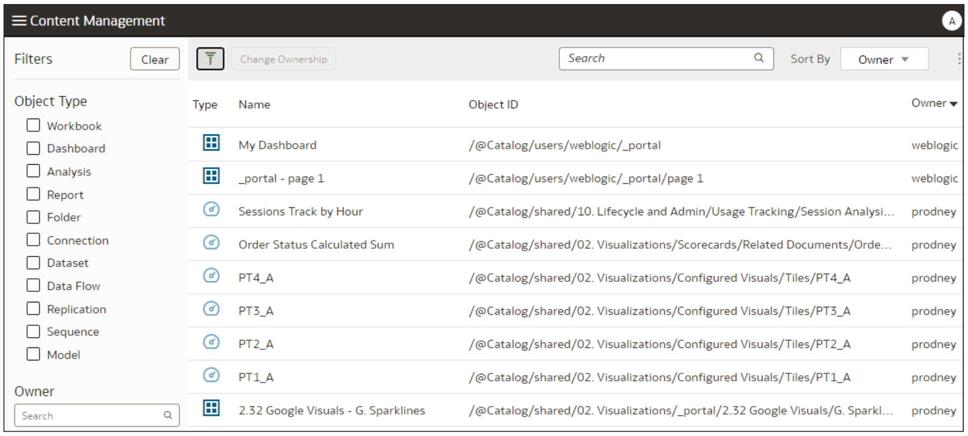
Change ownership to: Roles | BI Content Author

6. Use the **Change ownership to** options to specify a new owner (or owners) for the objects.
7. Click **OK**.

Change Ownership of Content in a User's Private Folder

You can transfer ownership of content that users save in private folders. For example, if an employee leaves your organization, you might move their private workbooks and machine learning models from the `\User Folders\<User>\` folder to a different folder so that other users can edit and deploy them.

1. In Console, change the ownership of the private objects to the administrator:
   a. In the Oracle Analytics Home page, click the **Navigator**, and then click **Console**.
   b. Click **Content Management**.



Type	Name	Object ID	Owner
⊞	My Dashboard	/@Catalog/users/weblogic/_portal	weblogic
⊞	_portal - page 1	/@Catalog/users/weblogic/_portal/page 1	weblogic
◎	Sessions Track by Hour	/@Catalog/shared/10. Lifecycle and Admin/Usage Tracking/Session Analysi...	prodney
◎	Order Status Calculated Sum	/@Catalog/shared/02. Visualizations/Scorecards/Related Documents/Orde...	prodney
◎	PT4_A	/@Catalog/shared/02. Visualizations/Configured Visuals/Tiles/PT4_A	prodney
◎	PT3_A	/@Catalog/shared/02. Visualizations/Configured Visuals/Tiles/PT3_A	prodney
◎	PT2_A	/@Catalog/shared/02. Visualizations/Configured Visuals/Tiles/PT2_A	prodney
◎	PT1_A	/@Catalog/shared/02. Visualizations/Configured Visuals/Tiles/PT1_A	prodney
⊞	2.32 Google Visuals - G. Sparklines	/@Catalog/shared/02. Visualizations/_portal/2.32 Google Visuals/G. Sparkl...	prodney

c. Click **Filters**, and then enter the name of the user in the **Owner** field.

   You see all content owned by that user. Private objects are prefixed with /@Catalog/users/<*username*>/ in the **Object ID**). For example, private content owned by someone with the username "john.smith" is prefixed with /@Catalog/users/john.smith/.

d. Select one or more private objects owned by the user.

e. Click **Change Ownership** to display the Change Ownership dialog.

f. Under **Change ownership to**, click **Users**, and enter your username or Admin, then click **OK**.

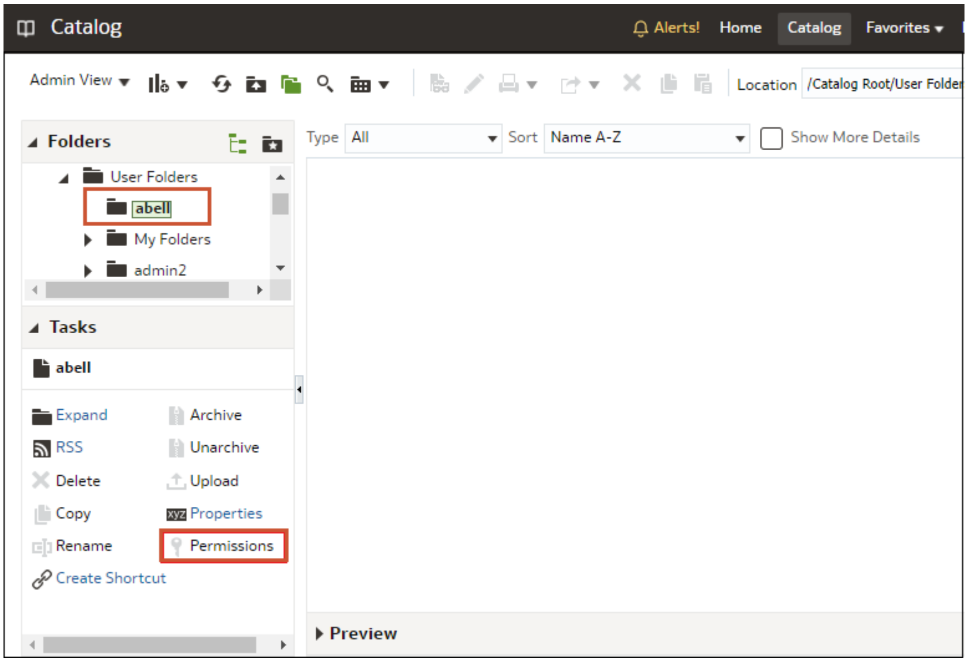
2. In the Catalog, change the permissions for the private objects and move them to a new folder:

   a. Click **Navigator**, then click **Home**, and from the **Page Menu** select **Open Classic Home**.

   b. Click **Catalog**, then click **Admin View** in the top left-hand corner.

   c. Under **User Folders**, click **My Folders**, then select the user's private folder.

   d. In the **Tasks** panel, click **Permissions**, and assign control of the folder and its contents to the a different user.



e. Move the content from the user's private folder to a different folder that other users can access.

   In the source folder, select the objects you want to move, then click **Copy**. Then, in the target folder, click **Paste**.

   For example, you might move workbooks and machine learning models from the \User Folders\USER1\ to \User Folders\USER2\, or to a shared folder that multiple users can access.

Frequently Asked Questions About Content Management

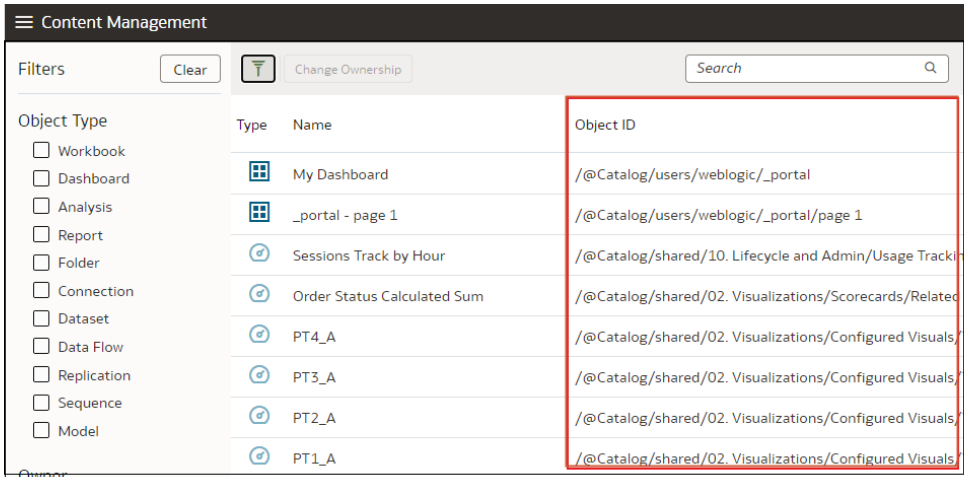
Find the answers to common questions about content management in Oracle Analytics.

What restrictions apply when reassigning ownership to roles?

- You can assign objects with an object ID prefixed with `/@Catalog/` to users or roles.
- You can assign objects with an object ID prefixed with `/@default/` to users only.

If you want to reassign multiple items to a role, then first deselect items with an object ID prefixed with `/@default/`).

To see how object IDs are prefixed, look at the **Object ID** column on the content management page.



Type	Name	Object ID
⊞	My Dashboard	/@Catalog/users/weblogic/_portal
⊞	_portal - page 1	/@Catalog/users/weblogic/_portal/page 1
◎	Sessions Track by Hour	/@Catalog/shared/10. Lifecycle and Admin/Usage Tracki
◎	Order Status Calculated Sum	/@Catalog/shared/02. Visualizations/Scorecards/Related
◎	PT4_A	/@Catalog/shared/02. Visualizations/Configured Visuals/
◎	PT3_A	/@Catalog/shared/02. Visualizations/Configured Visuals/
◎	PT2_A	/@Catalog/shared/02. Visualizations/Configured Visuals/
◎	PT1_A	/@Catalog/shared/02. Visualizations/Configured Visuals/

What does `@default` prefix or `@Catalog` prefix mean in an object ID?

A `@Catalog` prefix indicates a workbook, connection, dataset, data flow, replication, sequence, or model. A `@default` prefix indicates an analysis, dashboard, report, or folder.

Enable Users to Export to Excel

As an administrator, you configure the instance for users to export visualizations to Excel. See Export Formatted Data From a Visualization to Excel.

> **Note:**
>
> Any shared libraries that are required for the Chromium browser must be installed.

Topics:

- Certification - Systems for Export to Excel
- Install NodeJS and Modules
- Set Up Environment Variables for the Node Process
- Start and Stop the NodeJS Process

Certification - Systems for Export to Excel

Here're the operating systems certified for using export to Excel in Oracle Analytics Server.

Supported Processors

Processor	OS Version	OS Update Level	Oracle JDK Version*	Additional information
Linux x86-64	Oracle Linux 8	0+	1.8.0_341+	Oracle Fusion Middleware installer supports JDK 1.8.0_341+.
Microsoft Windows x64	2012, 2016, 2019	SP 0+	1.8.0_341+	-

* Oracle Analytics Server only supports Oracle JDK. Others, such as OpenJDK, aren't supported.

Install NodeJS and Modules

The first step in configuring for exporting to Excel is to install NodeJS and its modules.

Disclaimer: Because this solution requires the customer to download and install third party software (NodeJS and modules), any licence implications or security vulnerabilities are the responsibility of the customer.

> **Note:**

1. Download and install the latest LTS version of Node.js (version 18.18.0 or later) from https://nodejs.org/en/download for your operating system.
2. Create or modify `package.json` to include the below NPM modules as a dependency:

NPM Module	Version
express	^4.17
jquery-deferred	^0.3.1
log4js	6.4.1
html-to-xlsx	^2.2.0
puppeteer	20.7.4

   `Sample package.JSON`

   ```
   {
      "name": "exportexcel",
      "version": "1.0.0",
      "description": "",
      "dependencies": {
         "express": "^4.17",
         "jquery-deferred": "^0.3.1",
         "log4js": "6.4.1",
         "html-to-xlsx": "^2.2.0",
         "puppeteer": "20.7.4"
      }
   }
   ```

3. Configure the NPM registry and proxy, if required, using these commands:

   ```
   To set npm registry - npm config set registry <registry url>
             To set proxy - npm config set proxy http://
   <username><password>@proxy-server-url>:<port>
                      npm config set https-proxy http://
   <username><password>@proxy-server-url>:<port>
             Note: username and password are optional
   ```

4. Run `npm install` to install the NPM modules listed in the table.

Set Up Environment Variables for the Node Process

You must configure environment variables to point to the NodeJS installation on the virtual machine and for the Oracle Analytics Server Platform, before you can start the NodeJS process.

1. Set the environment variables for the node process.

Environment variable	Description	Required?
NODE_PATH	Environment variable that nodejs looks up for node packages. This is the path to the node_modules directory. For example, `<Directory where node_modules are installed>/node_modules`	Yes
PUPPETEER_EXECUTABLE_PATH	Environment variable that Puppeteer looks up for the chrome executable. This is the path to the chrome executable. The default location to download Chromium is: Linux: `$HOME/.cache/puppeteer` Windows: `[drive]:\users\<user>\.cache\puppeteer` The path to the chrome executable is:. Linux: `$HOME/.cache/puppeteer/chrome/<version>/chrome-linux64/chrome` Windows: `[drive]:\users\<user>\.cache\puppeteer\chrome\<version>\chrome-win64\chrome.exe` To change the default location where you download Chromium, check `https://pptr.dev/guides/configuration`.	Yes
TMP_DIR	Writable directory where the xlsx files are stored temporarily. If not set, the default is the operating system temp directory.	No
LOG_DIR	Writable directory where log files are written to. If not set, the default is the operating system temp directory.	No
EXCEL_NODE_PORT	Port where the export Excel service listens. If this environment variable isn't set, the server starts on port 3001.	No

2. Set the environment variable required by the Oracle Analytics Server Platform.

Environment variable	Description	Required?
OAS_FORMATTED_EXPORT_ENABLED	The environment variable looked up by the Oracle Analytics Server Platform to enable the export excel feature. When you set this, the NodeJS setup is completed and node process (export excel service) is running. Accepted values - true/false. **Note:** You must restart the system after you set the environment variable.	Yes

Start and Stop the NodeJS Process

Once you set the environment variables, as an administrator you can start and stop the NodeJS process. You must start the NodeJS process before users can begin exporting to Excel.

1. Start the NodeJS process using this sample script:

   On Linux:

```
LOG_DIR=<log directory> #Writable log directory

mkdir -p ${LOG_DIR}

"nohup" "node" "bi/modules/oracle.bi.tech/obitech-serverside-exportexcel-
bundle.js" >
${LOG_DIR}/biexportexcel.log &
i=0
max_attempts=60
excelAppPID=0
echo "Export excel app is starting..."
while [[ i < max_attempts ]];
do
   pgrep -f "bi/modules/oracle.bi.tech/obitech-serverside-exportexcel-
bundle.js"
   if [ $? != 0 ]; then
       i=$((i+1))
       sleep 1
   else
       echo "Export excel app is running"
       exit 0
   fi
done
echo "Export excel app failed to start after 60 seconds"
exit 1
```

2. Stop the NodeJS process using this sample script:

   On Linux:

```
excelAppPID=$(pgrep -f "bi/modules/oracle.bi.tech/obitech-serverside-
exportexcel-bundle.js")
```

```
if [ $? != 0 ]; then
excelAppPID=""
fi

if [ -n "${excelAppPID}" ]; then
echo "Stopping Export excel App with pid: ${excelAppPID}; killing
(SIGTERM) ..."
kill $excelAppPID
fi
```

Part III

Scale and Deploy for High Availability and Performance

Learn how to manage deployment, availability, and capacity.

This part includes the following chapters:

5 Scale Your Deployment

This chapter describes how to manage the capacity and availability of your deployment. By default, system components are installed in a cluster configuration and are scalable. User web requests can be directed to one of many Oracle BI Presentation Services components. In turn, each Presentation Services component can take advantage of the availability of multiple Oracle BI Servers.

You can expand or reduce the capacity of the system by adjusting the number of processes available to the cluster. Increasing or decreasing the capacity of a system by making effective use of resources is known as *scalability*. A scalable system can handle increasing numbers of requests without adversely affecting response time and throughput.

Topics:

About Scaling Oracle Analytics Server

Scaling is the process of increasing or decreasing the capacity of the system by changing the number of processes available to service requests from clients.

Scaling out a system provides additional capacity, while scaling in a system reduces capacity. Scaling is also a critical part of configuring a deployment for high availability. You can expand or reduce the capacity of the system by adjusting the number of processes that are available to the cluster. A cluster consists of multiple server instances that run simultaneously and work together to provide increased scalability and reliability.

Scaling the environment applies principally to resource-intensive system processes and Java components. When you deploy more processes, the system can handle more requests while staying responsive to requests.

Vertical scaling involves adding more components to the same computer, to make increased use of the hardware resources on that computer. For example, you can vertically scale Oracle Analytics Server by increasing the number of system components servicing requests on a given computer which results in increased use of the hardware resources on a given computer.

Horizontal scaling involves adding more computers to the environment. For example, you horizontally scale Oracle Analytics Server by distributing the processing of requests across multiple computers.

You can scale both Java components and system components. See About the Administration Server, Managed Servers, and System Components.

The system components that support both horizontal and vertical scale-out are Presentation Services, the Oracle BI Server, and the JavaHost.

Oracle BI Scheduler uses Presentation Services and Oracle BI Server processes to perform computationally intense work on its behalf, while the Cluster Controller only manages other components and doesn't itself do any computationally intense work. Because of this, there is no need to scale out either Oracle BI Scheduler or the Cluster Controller. You can distribute these two processes as needed for high availability deployments, but they don't need to be scaled for capacity.

How Do I Know When to Scale Out Processes?

Scale out system components and Managed Servers based on observed load. You can use the performance metrics that are provided in Fusion Middleware Control to monitor process state and to determine when you must increase capacity to improve performance. For example, you might want to add a computer to the deployment when CPU usage is over 50%, or when memory use is close to the system limit. See Monitor Service Levels for more information about viewing system metrics.

You also must scale out processes to achieve redundancy when you want to configure your system environment to be highly available. See Deploy for High Availability.

What Processes Should I Scale?

Oracle Analytics Server provides support for scale-out using a combination of the installer (for horizontal scale-out) and WebLogic Scripting Tool (WLST) to scale system components both vertically and horizontally.

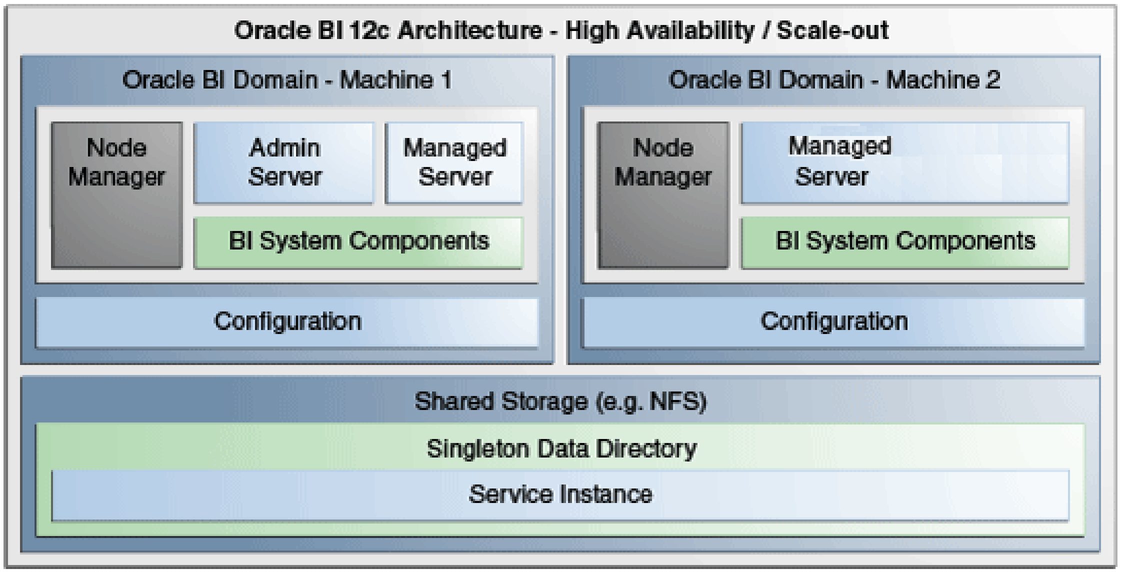
Follow these guidelines to scale Managed Servers and system components:

- Ensure that you run at least one Managed Server on each computer in the deployment. During installation the Configuration Assistant provisions one Managed Server. Don't disable or remove it.
- Don't remove individual Java components, because many perform essential services for the system. Keep a full set of Java components on each Managed Server. Any unused components likely don't have a significant performance impact.
- You can decide based on observed load which system components to run on each computer. You can have zero or more of each component type on a given computer in the deployment. For example, you can have three Oracle BI Servers, two JavaHosts, and four Presentation Services components. By default a symmetric set of components is created on the scaled out computer.
- You don't need to scale any configured HTTP servers along with either the Managed Servers or system components. HTTP server configuration is independent of the number of processes that you run.

Set Up Shared Files and Directories

When you've multiple instances of a component, files and directories including global cache and shared Oracle BI Scheduler scripts are located on a shared storage device (such as NAS or SAN).

Using shared files and directories simplifies management of your system including scale out of components.



This section contains the following topics:

- Change the Singleton Data Directory (SDD)
- Set Up the Global Cache

Change the Singleton Data Directory (SDD)

Metadata is stored in a singleton data directory (SDD).

The default location is set to:

```
DOMAIN_HOME/bidata
```

The SDD path is defined in the file bi-environment.xml, located in:

```
DOMAIN_HOME/config/fmwconfig/bienv/core/bi-environment.xml
```

See Key Directories.

1. Create a shared directory and make sure it's available on all hosts.

   For example, on Linux:

   ```
   ls /oraclehome/user_projects/domains/bi/bidata
   ```

2. Stop all processes by running the following command located in:

   ```
   DOMAIN_HOME/bitools/bin
   ```

   For example on Linux enter:

   ```
   ./stop.sh
   ```

3. Back up the file bi-environment.xml, and existing SDD if desired.
4. Open the file bi-environment.xml for editing, and specify the singleton path.

For example:

```
<bi:singleton-data-directory>/oraclehome/user_projects/domains/bi/bidata/</
bi:singleton-data-directory>
```

5. Save the file.
6. Copy the contents of the bidata directory to the shared directory previously created.

   For Publisher, copy the repository from the `DOMAIN_HOME/config/fmwconfig/biconfig/bipublisher/Admin` directory to the new `SDD/components/bipublisher/repository/Admin` directory.
7. Start all processes by running the following command located in:

   `*DOMAIN_HOME*/bitools/bin/`

   For example on Linux enter:

   ```
   ./start.sh
   ```

The SDD is now configured for all host computers.

Set Up the Global Cache

The global cache is a query cache that is shared by all Oracle BI Servers participating in a cluster.

See About the Global Cache.

It's recommended that you configure the global cache so that cache seeding and purging events can be shared by all Oracle BI Servers participating in a cluster.

- Use the Performance tab of the Oracle Analytics Server Instance Configuration page in Fusion Middleware Control to set the **Global cache path** and **Global cache size** options.

  See Use Fusion Middleware Control to Set Global Cache Parameters.

Manage Capacity (Vertically Scale)

You can change the number of system components to suit capacity requirements.

You should first configure shared files and directories for clustered components to use (see Set Up Shared Files and Directories).

You can change the number of system components to suit capacity requirements.

Only advanced users should use the commands described in this section.

- Add System Components
- Remove System Components

Add System Components

You can add system components to a computer when the system is stopped (offline).

> **Note:**
>
> If SSL is configured, see Configuring SSL in Oracle Business Intelligence in *Managing Security for Oracle Analytics Server*.

Assumptions:

- You must have appropriate file system permissions.
- Ports are allocated from a port range, unless otherwise specified.
- Supported system component types are OBIPS (BI Presentation Server), OBICCS (Cluster Controller), OBIJH (BI JavaHost), and OBISCH (BI Scheduler).

  OBIS isn't scaled out because OBIS instances are managed as part of service instances.

  See About System Components.
- You can only create two instances each of the component types OBICCS, OBISCH (one active, one passive). Therefore, if you must add an instance on another host, then you must first remove an existing instance.

1. Start the WebLogic Scripting Tool (WLST).

   ```
   ORACLE_HOME/oracle_common/common/bin/wlst.sh
   ```
2. Select your domain using the `readDomain(DOMAIN_HOME)` command.

   For example on Linux:

   ```
   readDomain('/oraclehome/user_projects/domains/bi')
   ```

   For example on Windows:

   ```
   readDomain('D:\\Oracle\\Middleware\\ORACLE_HOME\\user_projects\\domains\\bi')
   ```
3. Create the new Presentation Services system component(s) using an appropriate WLST command.

   All commands take a DOMAIN_HOME a machine name and an optional port specification.

   createOBIPSComponent(domainHome, machine, port=<portnumber>)

   Where machine is the WebLogic logical computer name (for example 'm1'). Use WLST or the WebLogic Admin Console (if running) to discover the logical machine name.

   For example on Linux:

   ```
   createOBIPSComponent('/u01/../../domains/bi','m1')
   ```

   For example on Windows:

   ```
   createOBIPSComponent('D:\\Oracle\\Middleware\\ORACLE_HOME\\user_projects\
   \domains\\bi,'m1')
   ```

   Use these component commands.

   - createOBICCSComponent(domainHome, machine, port=None, portMonitor=None) Use this command to create a new cluster component.

- createOBISCHComponent(domainHome, machine, port=None, portMonitor=None,portScript=None)
  Use this command to create a new scheduler component.
- createOBIPSComponent(domainHome, machine, port=None)

  Use this command to create a new BI Presentation Server component.
- createOBIJHComponent(domainHome, machine, port=None)

  Use this command to create a new JavaHost component
- listBISystemComponents(domainHome)
  Use this commend to return an object listing all of the system components in the domain.
- getBISystemComponents(domainHome, instanceId)

  Use this command to display system component details with a specified instanceID.

4. Once the command completes successfully you must commit the changes using the `updateDomain('DOMAIN_HOME')` command.

   The WLST command is the same for all environments.

   For example:

   `updateDomain()`

5. Close the domain using the `closeDomain('DOMAIN_HOME')` command.

   The WLST command is the same for all environments.

   For example:

   `closeDomain()`

6. Start the new component in:

   DOMAIN_HOME/bitools/bin/

   For example, enter:

   ```
   ./start.sh
   ```

   > **Note:**
   >
   > If you make a mistake or decide to abandon the changes, you should use the `closeDomain()` command without using the `updateDomain()` command.

   The new component is created, new ports are allocated, and the new component is started.

Remove System Components

You can remove an unwanted Presentation Services system component from a computer.

Assumptions:

- Run the command when the system is stopped (offline), as long as you have appropriate file system (offline) privileges.

- Supported system component types are OBIPS (BI Presentation Server), OBICCS (BI Cluster Controller), OBIJH (BI JavaHost), and OBISCH (BI Scheduler). See Oracle Analytics Server Components.

1. Stop the system using the stop script located in:

   DOMAIN_HOME/bitools/bin/

   For example on Linux enter:

   ```
   ./stop.sh
   ```

   See Stop Component Processes in a Domain.

2. Delete a system component by running the deleteBISystemComponent WLST command from *ORACLE_HOME*/oracle_common/common/bin/wlst.sh:

   deleteBISystemComponent(domainHome, instanceId)

   Where domainHome is the *DOMAIN_HOME* for the domain, and instanceID is the BI component ID (for example, obips1, obis4)

   For example:

   ```
   deleteBISystemComponent("/oraclehome/user_projects/domains/bi", "obips1")
   ```

   This removes the system component and deallocates the port.

   The name of the deleted system component is displayed.

3. Start the system by running the following command located in:

   DOMAIN_HOME/bitools/bin/

   For example on Linux enter:

   ```
   ./start.sh
   ```

   See Start Component Processes in a Domain.

Manage Availability (Horizontally Scale)

If you've multiple instances of a given component in the deployment, you should first configure shared files and directories for the clustered components to use.

See Set Up Shared Files and Directories.

After horizontally scaling out, you typically configure an HTTP server and load balancer to distribute requests across multiple Managed Servers. See Load Balancing in a Cluster in *Administering Clusters for Oracle WebLogic Server*.

Only advanced users whould use the commands described in this section. Note that availability commands automatically create a symmetric set of processes when configuring additional hosts.

> **Note:**
>
> To add a new node to a cluster when SSL has been configured, you must scale out to the new cluster (see Add New Computers), then ensure SSL is set up correctly (see Configuring SSL in Oracle Business Intelligence in *Managing Security for Oracle Analytics Server*).

- Add New Computers
- Remove Existing Computers

Add New Computers

You can add a new computer to extend the cluster across multiple computers, increasing availability and capacity.

Assumptions

- The new computer must meet the same prerequisites for installation (such as operating system and memory) as the primary computer.
- SDD must have been set up.
- *ORACLE_HOME* must be the same absolute path on both computers.
- You must use the same JDK version on the new computer as used on the existing computers.
- It's recommended, but not required, that the *DOMAIN_HOME* is the same on both computers.
- A symmetric set of active-active components is created on the new computer.
- The system must be stopped (offline).
- You must have appropriate file system (offline) or Oracle WebLogic Server Administrator (online) permissions.
- The same ports are allocated as on the primary computer.
- The Managed Server is added to the existing cluster.
- Cluster Controller, Scheduler, and BI Server control isn't changed.
- Optional base computer and server parameters are provided to support the case where the primary server (bi_server1) has been deleted.
- Unless provided, the name of the computer hosting Oracle WebLogic Server defaults to the listen address and must contain fewer than 32 characters.

Adding a new computer creates an additional Managed Server, Node Manager, system components, and services on the new computer.

1. Shut down all services on the primary computer before running the clone script:

   ```
   ORACLE_HOME/user_projects/domains/bi/bitools/bin/stop.sh
   ```

2. On the primary computer, run the clone script:

   The script creates the JAR file containing the domain directory and updates the config.xml file on the new computer with the new computer name, Managed Server and system components.

   ```
   DOMAIN_HOME/bitools/bin/clone_bi_machine.sh|cmd [-m <new computer name>]
   <listen address> <Jar file>
   ```

   `<new computer name>` is optional and defaults to the listen address. However, if the listen address is longer than 32 characters, then use the `-m` command and `<new computer name>` to manually set a value that's less than 32 characters.

   The SSL certificate step is done for you in the script.

   For example:

```
ORACLE_HOME/user_projects/domains/bi/bitools/bin/clone_bi_machine.sh
example.com /oracle/nfs_shared/clone/demobi2.jar
```

3. On the new computer, install Oracle WebLogic Server and Oracle Analytics Server.

   See About the Oracle Analytics Server Installation in *Installing and Configuring Oracle Analytics Server*.

4. Test connectivity between the two computers.

5. Copy the JAR file from the primary computer (with the Administration Server) to the new computer.

6. On the new computer, apply the JAR file by running the unpack command.

   ```
   ORACLE_HOME/oracle_common/common/bin/unpack.sh -template=[location of copied
   jar file from primary computer] -domain=DOMAIN_HOME -
   nodemanager_type=PerDomainNodeManager
   ```

   For example:

   ```
   /../../Oracle_Home/oracle_common/common/bin/unpack.sh -template=/refresh/home/
   oracle/nfs_shared/clone/demobi2.jar -domain=/../../Oracle_Home/user_projects/
   domains/bi -nodemanager_type=PerDomainNodeManager
   ```

7. Start the following components:

   a. On the new computer: Start Node Manager and Managed Server (this is to create the config.xml file).

      For example:

      ```
      /../../Oracle_Home/user_projects/domains/bi/bin/startNodeManager.sh
      ```

      The system components for the new computer should remain down for now.

   b. On the primary computer: Start Node Manager, Administration Server, Managed Server, and system components.

      For example:

      ```
      /../../Oracle_Home/user_projects/domains/bi/bitools/bin/start.sh -i
      AdminServer,bi_server1,obis1,obips1,obiccs1,obijh1,obisch1,bi_server2
      ```

   > **Note:**
   >
   > If Node Manager is started on the new computer, then you start all other components using the same script from the primary computer in this order:
   >
   > - On the new computer:
   >   Node Manager for the new computer.
   > - Node Manager for the primary computer.
   >
   >   Administration Server.
   >
   >   Managed Server for the primary computer.
   >
   >   System components for the primary computer.

8. Re-synchronize the data source on the new computer.

   On the new computer, run the following script:

   ```
   DOMAIN_HOME/bitools/bin/sync_midtier_db.sh
   ```

9. On the primary computer, start the new system components for the new computer:

   Change directory to *DOMAIN_HOME*/bitools/bin/.

   Enter:

   ```
   ./start.sh -i obis2,obips2,obiccs2,obijh2,obisch2
   ```

   The scaled-out process is complete and all components should now be running.

   The Managed Server, Node Manager, and system components are created. Service instances are registered on the new computer, and ports are allocated.

Remove Existing Computers

Remove a failed or redundant computer from a cluster. The computer, Managed Server, Node Manager, and system components are removed from the domain. Service instances are unregistered, and ports are deallocated.

Assumptions

- You must have appropriate file system (offline) permissions, or Oracle WebLogic Server Administrator (online) permissions.
- The system can be running (online) or stopped (offline).
- You can't remove the primary computer.
- You can add or remove service instance registrations.
- No binary files, configuration files, or state information, are deleted from the computer that you remove.
- There is no loss of service when you use the stop command.
- There is no loss of availability when you use the stop command, unless the user forces it.
- Control of the Cluster Controller, Scheduler, and BI Server is unchanged.

Prerequisites

If possible, you should stop active components on the target computer before removing it using the status.sh and stop.sh scripts. See Use Commands to Start, Stop, and View Status of Processes.

- Use the deleteBIMachine WLST command to remove components created by cloneBIMachine or clone_bi_machine.sh in:

  ORACLE_HOME/oracle_common/common/bin/wlst.sh

  For example:

  `deleteBIMachine`(*DOMAIN_HOME*, `machineName`)

  Or run the delete_bi_machine.sh script in:

  DOMAIN_HOME/bitools/bin/delete_bi_machine.sh|cmd machineName

  The command displays the removed computer name.

Validate that Your System Has Been Scaled Correctly

You need to validate components to ensure they're configured properly for your system’s size and scope.

You can use Fusion Middleware Control and the Oracle WebLogic Server Administration Console to verify the status of the scaled-out components.

Topics:

- Use Fusion Middleware Control to View System Component Availability
- Use the Administration Console to View Managed Server Availability

You can use the Command Line Interface to see the status of components in a Domain. See View the Status of Components in a Domain.

Use Fusion Middleware Control to View System Component Availability

You can use Fusion Middleware Control to view the status of all system components in your deployment.

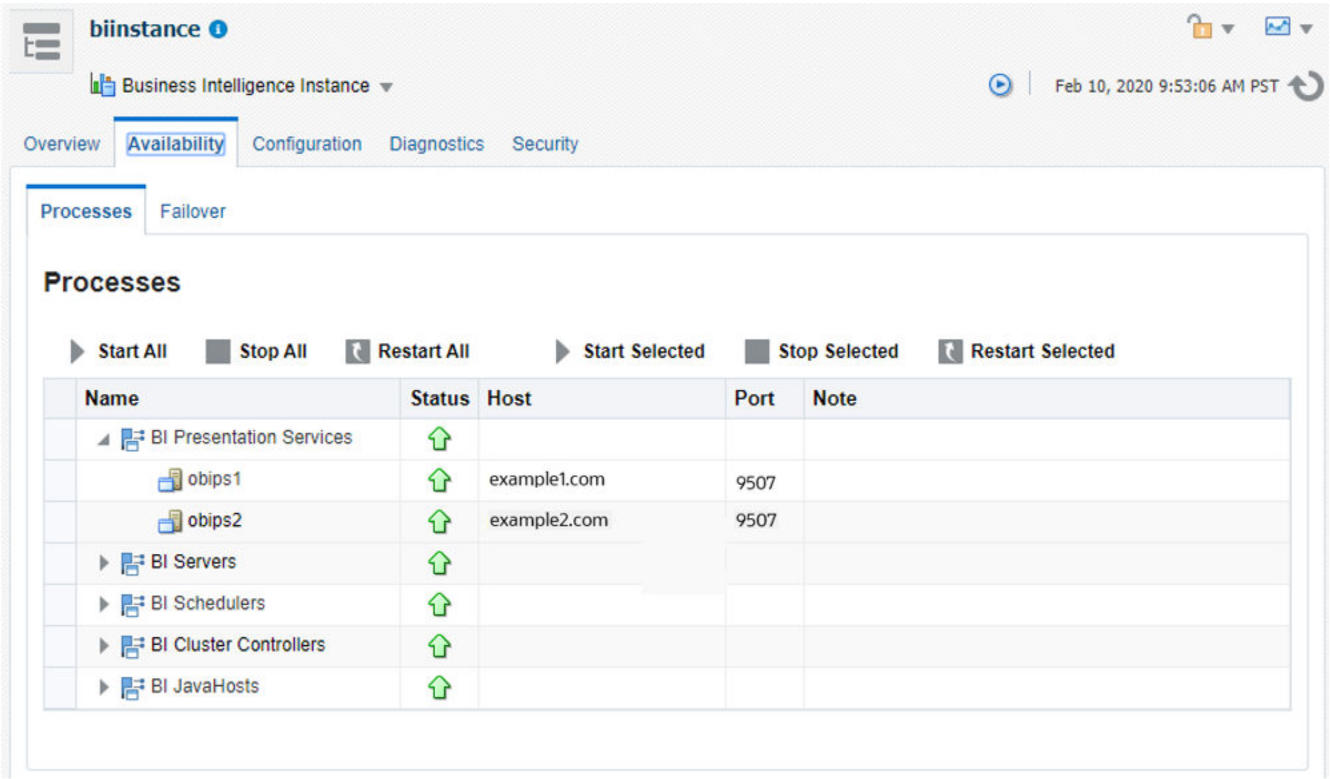
1. Login to Fusion Middleware Control.
2. Expand the **Business Intelligence** folder and select the **biinstance** node.

   Fusion Middleware Control displays the Overview page.
3. Display the Processes tab of the Availability page.

   On this page, you can:

   - View the status of all configured system components
   - View the host, and port of each system component running
   - Start, stop, or restart all processes
   - Start, stop, or restart selected system components

The following illustration shows the Processes tab of the Availability page.



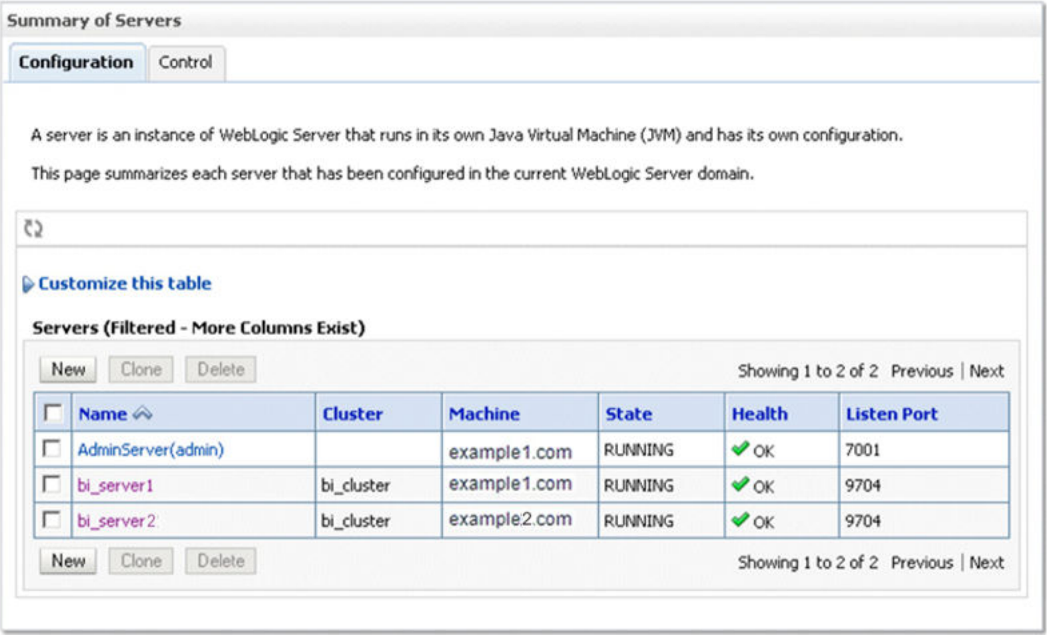
Name	Status	Host	Port	Note
BI Presentation Services	⇧			
obips1	⇧	example1.com	9507	
obips2	⇧	example2.com	9507	
BI Servers	⇧			
BI Schedulers	⇧			
BI Cluster Controllers	⇧			
BI JavaHosts	⇧			

Use the Administration Console to View Managed Server Availability

You can use the Administration Console to view the status of all Managed Servers in your deployment.

1. Log in to the Oracle WebLogic Server Administration Console.
2. Select **Environment**, then select **Servers** to go to the Summary of Servers page. On this page, you can see any Managed Servers that were added on new hosts in your deployment.

The next illustration shows the Summary of Servers page.



A server is an instance of WebLogic Server that runs in its own Java Virtual Machine (JVM) and has its own configuration.

This page summarizes each server that has been configured in the current WebLogic Server domain.

Servers (Filtered - More Columns Exist)

□	Name	Cluster	Machine	State	Health	Listen Port
□	AdminServer(admin)		example1.com	RUNNING	✓ OK	7001
□	bi_server1	bi_cluster	example1.com	RUNNING	✓ OK	9704
□	bi_server2	bi_cluster	example2.com	RUNNING	✓ OK	9704

6

Deploy for High Availability

This chapter describes how to configure components for high availability. It also describes the functionality available in Fusion Middleware Control to manage system availability, and provides information about using the Cluster Manager in the Model Administration Tool. This chapter doesn't provide information about setting up additional high availability configuration for other components in the stack, including database tier, web tier, Administration Server, and identity management availability.

For more information about these topics and how they relate to deployments, *Enterprise Deployment Guide for Oracle Analytics Server* explains how to deploy Oracle Analytics Server based on an architectural blueprint that follows Oracle recommended best practices for security and high availability, including web tier, database tier, Administration Server, and identity management availability.

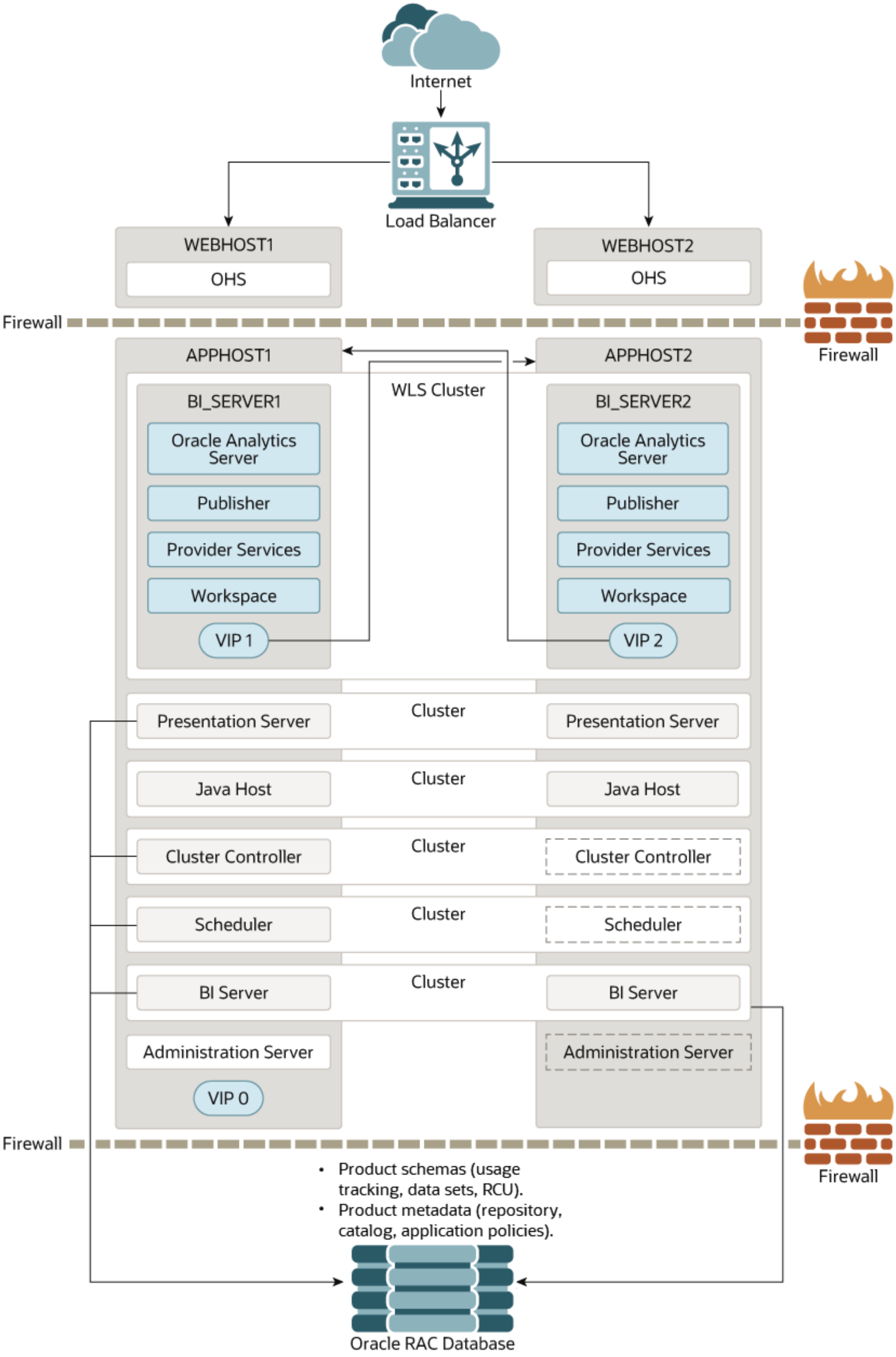
Topics:

- About Components in a Clustered Environment
- Configure Components for High Availability
- Optional Configuration for High Availability
- Use the Cluster Manager
- Troubleshoot a Fusion Middleware Control Clustered Environment

About Components in a Clustered Environment

The figure below shows the system components and Java components in a highly available deployment.

See About the Administration Server, Managed Servers, and System Components for more information about system components and Java components.


Internet
Load Balancer
WEBHOST1
OHS
WEBHOST2
OHS
Firewall
Firewall
APPHOST1
APPHOST2
WLS Cluster
BI_SERVER1
Oracle Analytics Server
Publisher
Provider Services
Workspace
VIP 1
BI_SERVER2
Oracle Analytics Server
Publisher
Provider Services
Workspace
VIP 2
Presentation Server
Cluster
Presentation Server
Java Host
Cluster
Java Host
Cluster Controller
Cluster
Cluster Controller
Scheduler
Cluster
Scheduler
BI Server
Cluster
BI Server
Administration Server
VIP 0
Administration Server
Firewall
Firewall
• Product schemas (usage tracking, data sets, RCU).
• Product metadata (repository, catalog, application policies).
Oracle RAC Database

In the figure above, the Java components are deployed on the BI_SERVER1 and BI_SERVER2 Managed Servers on APPHOST1 and APPHOST2. These Managed Servers are configured in an Oracle WebLogic cluster.

Presentation Services, JavaHost, Cluster Controller, Oracle BI Scheduler, and Presentation Services are system components installed on APPHOST1 and APPHOST2 and configured as a cluster. The Cluster Controller and Oracle BI Scheduler on APPHOST2 are passive (they're started but don't service requests) and are only made active if APPHOST1 components fail.

Customer metadata is stored in the database. You can export and import the metadata as BAR files using the Oracle Analytics Console or by using scripts. See About the Oracle Analytics Server Console and Take Snapshots and Restore .

Recommendations for Availability

In a production system, it's recommended that you deploy two or more instances of every component on two or more computers, so that each component type has an instance running on more than one computer for fault tolerance.

This configuration provides redundancy for Managed Servers and system components, an essential requirement for high availability and failover. You can see whether the system has any single points of failure by using the Failover tab of the Availability page in Fusion Middleware Control. See Use Fusion Middleware Control to Identify Single Points of Failure.

You can also ensure high availability by configuring redundancy in the database tier (Oracle RAC recommended), web tier, and for the Administration Server. See Create a High Availability Environment in *High Availability Guide*.

Note also the following requirements:

- All Oracle BI Servers participating in the cluster must be within the same domain and on the same LAN subnet. Geographically separated computers aren't supported.
- The clock on each server participating in a cluster must be kept in synchronization. Out-of-sync clocks can skew reporting.

Use Fusion Middleware Control to Identify Single Points of Failure

If there's a single point of failure in a process, you can use Fusion Middleware Control to find it.

Before you begin this procedure, ensure that you're familiar with the information in Use Fusion Middleware Control.

1. Go to the Overview page, as described in Display Oracle Analytics Server Pages in Fusion Middleware Control.
2. Display the Failover tab of the Availability page.

   On this page, you can view scaled out system components and whether to configure primary/secondary system components.

Achieve High Availability Using an Active-Passive Model

As an alternative to setting up the active-active configuration, you can set up an active-passive configuration using Oracle Fusion Middleware Cold Failover Cluster (Cold Failover Cluster).

In a Cold Failover Cluster configuration, you configure two or more application server instances to serve the same application workload, but only one is active at any particular time.

You can use a two-node Cold Failover Cluster to achieve active-passive availability. In a Cold Failover Cluster, one node is active while the other is passive, on standby. In the event that the active node fails, the standby node is activated, and Oracle Analytics Server continues servicing clients from that node. All Oracle Analytics Server components are failed over to the new active node. No Oracle Analytics Server components run on the failed node after the failover.

See Active-Passive High Availability Solutions in *High Availability Guide* for detailed information.

Configure Components for High Availability

To configure for high availability, you must ensure that the system has no single points of failure by scaling out the Oracle BI Server, Presentation Services, and the JavaHost so that you've at least two of each component type, distributed across at least two computers.

The table below lists the tasks that you must perform to configure high availability.

Task	Where to Go for More Information
Horizontally scale out the deployment so that it includes two computers with a full set of Java and system components on each host. This task includes running the installer, and configuration assistant, and scaling out system components.	Manage Availability (Horizontally Scale)
Verify that the new components are available.	Use Fusion Middleware Control to View System Component Availability

Optional Configuration for High Availability

The steps in this section describe how to perform optional configuration for high availability.

This section contains the following topics:

- Set Optional Cluster Controller Parameters
- Set Optional Presentation Services Parameters
- Set Optional Presentation Services Plug-in Parameters

Set Optional Cluster Controller Parameters

You can set optional parameters that are related to Cluster Controller heartbeat frequency in the `bi-cluster-config.xml` file.

1. Open the `bi-cluster-config.xml` file for editing at:

   BI_DOMAIN`/config/fmwconfig/biconfig/OBICCS`

2. The following table describes default values for the cluster communication parameters under the ClusterProperties element. Optionally, modify the parameter values as required for the deployment.

   Use these cluster communication parameters in `NQClusterConfig.INI`.

   - ServerPollSeconds
     The frequency in seconds of heartbeat messages between the Cluster Controller and the Oracle BI Server and Oracle BI Scheduler nodes in the cluster. The default value is 5.

- ControllerPollSeconds
  The frequency in seconds of heartbeat messages between the Cluster Controllers. The default value is 5.

3. Save and close the file.
4. Restart Oracle Analytics.

This code shows example parameters in the `bi-cluster-config.xml` file. Note that any additional elements that aren't shown in this example are centrally managed and can't be set manually.

```
<bi:ClusterProperties>
  <bi:ClusterEnabled>true</bi:ClusterEnabled>
  <bi:ServerPollSeconds>5</bi:ServerPollSeconds>
  <bi:ControllerPollSeconds>5</bi:ControllerPollSeconds>
</bi:ClusterProperties>
```

Set Optional Presentation Services Parameters

You can optionally configure certain parameters that control the communication between Presentation Services and the JavaHost component.

To configure Presentation Services, set parameters in the instanceconfig.xml file on each computer that hosts Presentation Services.

1. Open the configuration file instanceconfig.xml for editing at:

   BI_DOMAIN/config/fmwconfig/biconfig/OBIPS

2. Under the ServerInstance tag, the JavaHostProxy element has optional sub-elements. The following table describes these optional sub-elements.

Subelement	Attribute	Description
LoadBalancer/ Ping	keepAliveMaxFailures	Specifies the number of ping failures required before the host is declared nonfunctioning. The default value is 5.
LoadBalancer/ Ping	keepAliveFrequencySecs	Specifies the ping frequency in seconds. The default value is 20.

3. Save and close the file.
4. Restart Oracle Analytics Server.

Set Optional Presentation Services Plug-in Parameters

You can optionally configure the Presentation Services Plug-in to control session redirection behavior.

1. Open the bridgeconfig.properties file for editing, for example at:

   BI_DOMAIN/config/fmwconfig/biconfig/bridgconfig.properties

2. Optionally, you can include the parameter AlwaysKeepSessionAffiliation to control whether requests that belong to the same session can be redirected to another Presentation Services component if the current Presentation Services Plug-in component score is too low.

   The instance score is an internal score that the load balancing algorithm associates with each Presentation Services Plug-in instance in the cluster. It's based on various metrics that are collected by the load balancer.

Set this parameter to true to disallow request redirection, or false to allow requests to be redirected. For example:

```
oracle.bi.presentation.sawconnect.loadbalance.AlwaysKeepSessionAffiliation=true
```

3. Save and close the file.
4. Restart the **analytics** application from the Oracle WebLogic Server Administration Console. If Presentation Services Plug-in is using the Presentation Catalog, then you must reastart the **xmlpserver** application.

Use the Cluster Manager

The Cluster Manager in the Model Administration Tool was used in previous updates to monitor and manage Oracle BI Server, Oracle BI Scheduler, and Cluster Controller instances.

Although you use Fusion Middleware Control for most administrative tasks that relate to clustered components, the Cluster Manager provides a useful way to view the state of clustered components. For example, you can view the currently active Oracle BI Scheduler instance and see which Oracle BI Server is the primary BI Server. Fusion Middleware Control shows the current status of clustered components, but doesn't provide a way to view the current state.

The Cluster Manager enables you to monitor, analyze, and manage the operations of Oracle BI Server, Oracle BI Scheduler, and Cluster Controller instances in a cluster. It provides status, cache, and session information. The Cluster Manager is available only when the Model Administration Tool is connected to a clustered DSN.

If all Cluster Controllers or Oracle BI Servers in the cluster are currently stopped or offline, then you can't access the Cluster Manager to start them. You must manually start one Cluster Controller (generally, the primary) and one Oracle BI Server.

The Cluster Manager window has two panes: the Explorer pane on the left side and the Information pane on the right side. The Explorer pane displays hierarchical information about the servers, schedulers, and controllers that comprise a cluster. The Information pane shows detailed information about an item selected in the Explorer pane.

The Cluster Manager window refreshes every minute by default. You can change the interval.

1. In the Model Administration Tool, open a semantic model.
2. Select **Manage**, then **Clusters**.
3. Select **Refresh**, then **Every**, and select another value from the list.
4. To refresh the display at any time, ensure that the Cluster Manager is the active window and press F5, or select **Refresh**, then **Now**. This action retrieves the most current information for the cluster.

View and Manage Cluster Information

Cluster information provides an insight into the application.

The section describes how to view status, cache, and session information about a cluster, and the meaning of the information provided.

Status Information

The Status view is automatically displayed when you first open the Cluster Manager window.

You can also access the Status view by selecting **View**, then **Status** in the Cluster Manager window.

The categories of information that are displayed in the Information pane might vary depending on the server to which the Model Administration Tool is connected. The following table describes categories that might appear.

Column	Description
Last Reported Time	The time that the Cluster Controller or Oracle BI Server communicated with the Controlling Cluster Controller. If the server or controller is offline, then this field might be blank.
Name	The name of the computer that's hosting the Oracle BI Server or Cluster Controller.
Role	The role of the object in the cluster: • **Controlling.** A Cluster Controller that's currently assigned the responsibility for control of the cluster. • **Primary.** The Primary Cluster Controller. This role isn't displayed if the Primary Cluster Controller is currently the controlling Cluster Controller. • **Secondary.** The Secondary Cluster Controller. This role isn't displayed if the Secondary Cluster Controller is currently the controlling Cluster Controller. • **Clustered server.** An Oracle BI Server that's a member of the cluster. This role isn't displayed for the clustered server that's defined as the master server. • **Master.** The clustered server that the Model Administration Tool connects to for editing semantic models. • **Active.** The Oracle BI Scheduler is active.
Sessions	This field is available when either Servers or an individual server is selected in the Explorer pane. It shows the number of sessions that are currently logged on to a clustered server.
Start Time	The timestamp showing when the Cluster Controller or Oracle BI Server was last started. This field is blank if the Cluster Controller or clustered server is offline.

Column	Description
Status	The status of the object in the cluster: • **Online.** The Cluster Controller or Oracle BI Server is online. Online Cluster Controllers can accept session requests and assign them to available servers within the cluster. Online Oracle BI Servers can be assigned sessions by the Cluster Controller. • **Quiesce.** This status is applicable to clustered servers only. When a server is quiesced, any activity in progress on outstanding sessions can complete before the server transitions to Offline status. • **Offline.** The Cluster Controller or Oracle BI Server is offline. Offline Cluster Controllers can't accept session requests or assign sessions to available servers within the cluster. Offline Oracle BI Servers don't communicate with the controlling Cluster Controller and can't accept sessions assigned by the controlling Cluster Controller. If the server subsequently becomes available, then the server can participate in the cluster. To stop the Cluster Controller or clustered server after quiescing it, enter the **Stop** command. • **Forced Offline.** This status applies to clustered servers only. The Oracle BI Server has been stopped. This is identical to the offline status, except that if the Oracle BI Server comes back online, it's not assigned requests. The server remains in this state until the **Start** command is issued against this server from the Model Administration Tool Cluster Manager, or both Cluster Controllers are shut down and restarted. • **Online: Active.** The Oracle BI Scheduler instance is online, running, and the one to which Oracle BI Scheduler clients connect. This instance runs jobs. • **Online: Inactive.** The Oracle BI Scheduler is online but not running. This instance is ready to take over for the active instance if the active instance becomes unavailable. • **Online: Inactive Pending.** The Oracle BI Scheduler was active and is trying to go into an inactive state. This might take a few minutes (for example, if multiple jobs are running).
Type	When Clusters is selected in the Explorer pane, this field is available. There are three types: • **Controller.** The object is a Cluster Controller. • **Server.** The object is an Oracle BI Server. • **Scheduler.** The object is a Scheduler Server.

Cache Information

The Cache view is available in the Cluster Manager window if caching is enabled.

The categories of information and their display sequence are controlled by the Options settings. The table below describes categories that might appear.

Column	Description
Business Model	Name of the business model that's associated with the cache entry.
Column count	Number of columns in each row of this cache entry's result set.
Created	Time the result set of the cache entry was created.
Creation elapsed time	Time, in milliseconds, needed to create the result set for this cache entry.
Full size	Full size is the maximum size used, considering variable length columns, compression algorithm, and other factors. The actual size of the result set is smaller than Full size.

Column	Description
Last used	Last time the result set of the cache entry satisfied a query. (After an unexpected shutdown of an Oracle BI Server, the Last used time might temporarily have a stale value, that's older than the true value.)
Row count	Number of rows that are generated by the query.
Row size	Size of each row (in bytes) in this cache entry's result set.
SQL	Text of the SQL statement that generated the cache entry.
Use count	Number of times that this cache entry's result set has satisfied a query (since Oracle BI Server startup).
User	Name of the user who submitted the query that resulted in the cache entry.

To view cache information click an individual server in the Explorer pane, and then select **View**, then **Cache**.

Session Information

You can review Session information in two places.

The Session view is available for Oracle BI Servers. The information is arranged in two windows, described in the table.

- Session window: Appears on the top. Shows users currently logged on to the Oracle BI Server.
- Request window: Appears on the bottom. Shows active query requests for the user selected in the Session window.

The following table describes the information that's displayed in the Session window.

Column	Description
Catalog	Name of the Presentation Catalog to which the session is connected.
Client Type	Type of client session. The client type of Administration is reserved for the user who's logged in with the Administrator user ID.
Last Active Time	Timestamp of the last activity on the session or the query.
Logon Time	Timestamp when the session logged on to the Oracle BI Server.
Repository	Logical name of the repository to which the session is connected.
Session ID	Unique internal identifier that the Oracle BI Server assigns each session when the session is initiated.
User	Name of the user connected.

The following table describes the information that's displayed in the Request window.

Column	Description
Last Active Time	Timestamp of the last activity on the session or the query.
Request ID	Unique internal identifier that the Oracle BI Server assigns each query when the query is initiated.
Session ID	Unique internal identifier that the Oracle BI Server assigns each session when the session is initiated.

Column	Description
Start Time	Time of the initial query request.
Status	These are the possible values. Due to the speed at which some processes complete, not all values for any given request or session might appear. • **Idle.** There's presently no activity on the request or session. • **Fetching.** The request is being retrieved. • **Fetched.** The request has been retrieved. • **Preparing.** The request is being prepared for processing. • **Prepared.** The request has been prepared for processing and is ready for execution. • **Executing.** The request is currently running. To terminate a request, select it and click **Kill Request**. The user receives an informational message that indicates that the Administrator canceled the request. • **Executed.** The request has finished running. • **Succeeded.** The request ran to completion successfully. • **Canceled.** The request has been canceled. • **Failed.** An error was encountered during the processing or running of the request.

To view session information:

- Select a server in the Explorer pane, and select **View**, then **Sessions**.

  Session information for the server is displayed in the Information pane. It shows all users logged into the server and all current query requests for each user.

To disconnect a session:

- In the Session view, right-click the session in the Session window (top window) and click **Disconnect**.

  When you disconnect a session, the ODBC session is terminated. Users who were connected during this session receive error messages if they attempt to run queries. Users must log out, then log back in again to start a new session.

To terminate a query request:

- In the Session view, right-click the request in the Request window (bottom window) and click **Kill Request**.

  When you terminate a query request, the user who's initiating the query receives an error.

Server Information

Information about the cluster server is available from the application menu.

Selecting **Server info** from the **View** menu provides information about the cluster server, such as server version number.

Troubleshoot a Fusion Middleware Control Clustered Environment

Use Fusion Middleware Control and the Administration Console to check the status of system processes.

See Use Fusion Middleware Control to View System Component Availability and Use the Administration Console to View Managed Server Availability.

After you enable clustering, load balancing, and failover capabilities, you can troubleshoot issues that might occur in the deployment using the following:

- Messages and errors that are reported in Fusion Middleware Control
- Log files for Fusion Middleware Control components, also available through Fusion Middleware Control

Review the log files for every Fusion Middleware Control system component in the cluster. Log files record any client-side failures that might occur due to an incorrect configuration. Although some failover events aren't logged, the Cluster Controller log file records crashes of any Oracle BI Scheduler or Oracle BI Server component. You can also review the Event Viewer log on Windows and the syslog on Linux.

See Diagnose and Resolve Issues.

Avoid Errors with Network Appliance Devices When the Oracle BI Server Is Running on Linux

The following information applies to deployments with Oracle BI Server components on Linux platforms that access Oracle Analytics Server shared files and directories on a NAS device from Network Appliance.

For environments with Oracle BI Server components on Linux that use the NTFS security style, the recommended Network Appliance Data ONTAP storage operating system version is 6.3.1 or later.

Linux computers saving to an NTFS qtree in Data ONTAP versions 6.0.3 through 6.3 might see permission errors when trying to save designs. Use the following Data ONTAP setting to silently ignore attempts to set permissions on NTFS qtrees after the design file is saved:

```
options cifs.ntfs_ignore_unix_security_ops on
```

7

Manage Performance Tuning and Query Caching

This chapter describes ways to improve query performance, including a performance tuning overview and information about monitoring system metrics. It also describes how to manage and use the query cache, a feature that enables the Oracle BI Server to save the results of a query in cache files and then reuse those results later when a similar query is requested. Using cache, the cost of database processing must be paid only once for a query, not every time the query is run.
See also the following Oracle Fusion Middleware resources on performance tuning for your system:

Tuning Performance

Tuning Performance of Oracle WebLogic Server

Topics:

- Monitor Service Levels
- About Query Performance Tuning
- Set Performance Parameters in Fusion Middleware Control
- About the BI Server Query Cache
- Configure Query Caching
- Monitor and Manage the Cache
- Strategies for Using the Cache
- Cache Event Processing with an Event Polling Table
- Configure the Data Source Cache
- Manage Presentation Services Cache Settings
- Improve Web Client Performance
- Optimize Performance by Setting the JVM Heap Size
- Improve Performance when Downloading Large Reports
- Capture Metrics Using the Dynamic Monitoring Service

Monitor Service Levels

Understanding service levels typically involves monitoring process state and viewing system metrics.

Oracle Analytics Server automatically and continuously measures runtime performance in real time. The performance metrics are automatically enabled; you don't need to set options or perform any extra configuration to collect them.

System metrics are available in Fusion Middleware Control for system components within a given Oracle Analytics Server installation. If you encounter a problem, such as an application that's running slowly or is hanging, then you can view more detailed performance information to learn more information about the problem.

You can use WSLT commands to periodically save metric information to a file so that you have a record of past metric values. See DMS Custom WLST Commands in *WLST Command Reference for Infrastructure Components*.

You can also view metrics for Java components using the Oracle WebLogic Server Administration Console.

This section contains the following topics:

- Use Fusion Middleware Control to View All Metrics
- Use the Administration Console to View Metrics for Java Components

Use Fusion Middleware Control to View All Metrics

You can view and graph all the available metrics in Fusion Middleware Control.

The data is logged transiently (that is, logging starts when you go to the page and select a particular metric for display).

1. In the tree navigator, expand the **Business Intelligence** folder and right-click the **biinstance** node.
2. Select **Monitoring**, then select **Performance Summary**. The Performance Summary page appears, displaying a selection of metrics for this installation.
3. To customize the metrics that are displayed on the Performance Summary page, click **Show Metric Palette**. Then, expand the appropriate metric category and select or deselect individual metrics. The metrics that you select are displayed on the Performance Summary page.

Use the Administration Console to View Metrics for Java Components

Use the Administration Console to view metrics for Java components.

You can view metrics on the Monitoring tab for the selected Managed Server, or you can use the Metric Browser. If your deployment is based on the Simple Install type, use the Monitoring tab for the Administration Server.

1. Log in to the Administration Console.
2. Expand the **Environment** node in the Domain Structure window.
3. Click **Servers** to display the Summary of Servers page.
4. Click the server name (for example, **oracle_bi1** or **AdminServer(admin)**).
5. Click the Monitoring tab.
   Click **Help** for more information about the metrics displayed on this tab.

1. Log in to the Administration Console.
2. Click **Monitoring Dashboard** under Charts and Graphs.
3. Click the Metric Browser tab.

   Click **Help** for more information about using the Metric Browser.

About Query Performance Tuning

This section describes some important considerations for improving query performance with the Oracle BI Server.

The following list summarizes methods that you can use to improve query performance:

- **Tuning and indexing underlying databases:** For Oracle BI Server database queries to return quickly, the underlying databases *must* be configured, tuned, and indexed correctly. Note that different database products have different tuning considerations.

  If there are queries that return slowly from the underlying databases, then you can capture the SQL statements for the queries in the query log and provide them to the database administrator (DBA) for analysis. See Manage the Query Log.
- **Aggregate tables:** It's extremely important to use aggregate tables to improve query performance. Aggregate tables contain precalculated summarizations of data. It's much faster to retrieve an answer from an aggregate table than to recompute the answer from thousands of rows of detail.

  The Oracle BI Server uses aggregate tables automatically, if they've been properly specified in the semantic model. See *Managing Metadata Repositories for Oracle Analytics Server* for examples of setting up aggregate navigation.
- **Query caching:** The Oracle BI Server can store query results for reuse by subsequent queries. Query caching can dramatically improve the apparent performance of the system for users, particularly for commonly used dashboards, but it doesn't improve performance for most ad-hoc analysis. See About the BI Server Query Cache.
- **Setting parameters in Fusion Middleware Control:** You can set various performance configuration parameters using Fusion Middleware Control to improve system performance. See Set Performance Parameters in Fusion Middleware Control.
- **Setting parameters in NQSConfig.INI:** The `NQSConfig.INI` file contains additional configuration and tuning parameters for the Oracle BI Server, including parameters to configure disk space for temporary storage, set virtual table page sizes, and several other advanced configuration settings. See Configuration File Settings.

You can also improve the overall performance of the system by increasing throughput by scaling out system components. See Scale Your Deployment .

Set Performance Parameters in Fusion Middleware Control

There are several performance options that you can set in Fusion Middleware Control.

This section contains the following topics:

- Use Fusion Middleware Control to Disallow Semantic Model Updates
- Use Fusion Middleware Control to Set the User Session Log-Off Period
- Use Fusion Middleware Control to Set Configuration Options for Data in Tables and Pivot Tables
- Use Fusion Middleware Control to Set the Maximum Number of Rows Processed to Render a Table View

Use Fusion Middleware Control to Disallow Semantic Model Updates

You can use Fusion Middleware Control to allow or prevent updates to the default semantic model file.

Setting this parameter affects whether you can update the semantic model when the Model Administration Tool connects. It also affects whether you can perform other semantic model update operations using other utilities, such as biserverxmlcli. Note that the aggregate persistence feature isn't available when semantic model updates are prevented.

Preventing semantic model updates can improve Oracle BI Server performance, because in this mode the Oracle BI Server doesn't need to handle lock control.

If you choose to prevent semantic model updates, then when the Model Administration Tool opens a semantic model, a message informs the user that the semantic model is read-only.

Before you begin this procedure, ensure that you're familiar with the information in Use Fusion Middleware Control.

1. Go to the Business Intelligence Overview page.
2. Display the Performance tab of the Configuration page.
3. Click **Lock and Edit** to enable changes to be made.
4. Select **Disallow RPD Updates** to prevent updates to the semantic model file.
5. Click **Apply**, then click **Activate Changes**.
6. Go to the Processes tab of the Availability Page and click **Restart All**.

For information about corresponding configuration file elements, see Map User Interface Labels with Configuration File Elements.

Use Fusion Middleware Control to Set the User Session Log-Off Period

You can override the time to elapse, in minutes, before a user is automatically logged off.

Before you begin this procedure, ensure that you're familiar with the information in Use Fusion Middleware Control.

1. Go to the Business Intelligence Overview page.
2. Display the Performance tab of the Configuration page.
3. Click **Lock and Edit** to enable changes to be made.
4. Complete the **User Session Expiry** option.
5. Click **Apply**, then click **Activate Changes** to process your changes and release the lock to enable another system administrator to make changes.
6. Return to the Business Intelligence Overview page and click **Restart**.

For information about corresponding configuration file elements, see Map User Interface Labels with Configuration File Elements.

Use Fusion Middleware Control to Set Configuration Options for Data in Tables and Pivot Tables

This procedure describes basic configuration options for data in tables and pivot tables.

Advanced configuration settings are described in Configure for Display and to Process Data in Views.

Before you begin this procedure, ensure that you're familiar with the information in Use Fusion Middleware Control.

1. Go to the Business Intelligence Overview page.
2. Display the Performance tab of the Configuration page.
3. Click **Lock and Edit** to enable changes to be made.
4. Complete the following options:
   - **Maximum Number of Rows to Download** option
   - **Maximum Number of Rows Per Page to Include** option

> **Note:**
>
> In general, the 'Maximum Number of Rows to Download' option and the 'Maximum Number of Rows Processed when Rendering a Table View' option discussed in Use Fusion Middleware Control to Set the Maximum Number of Rows Processed to Render a Table View should be set to the same values to avoid getting 'Exceeded configured maximum number of allowed input records' errors.

5. Click **Apply**, then click **Activate Changes**.
6. Return to the Business Intelligence Overview page and click **Restart**.

For information about corresponding configuration file elements, see Map User Interface Labels with Configuration File Elements.

Use Fusion Middleware Control to Set the Maximum Number of Rows Processed to Render a Table View

You can override the maximum number of rows that can be processed from the BI Server for rendering a view.

Reducing the number of rows in a table can significantly improve performance by reducing the system resources that are consumed by a given user session.

Advanced configuration settings are described in Configure for Display and to Process Data in Views.

Before you begin this procedure, ensure that you're familiar with the information in Use Fusion Middleware Control.

1. Go to the Business Intelligence Overview page.
2. On the Configuration page, select the Performance tab.

3. Click **Lock and Edit** to enable changes to be made.
4. Complete the **Maximum Number of Rows Processed when Rendering A Table View** field using the description in the help topic for the page. Enter an integer value greater than 50.
5. Click **Apply**, then click **Activate Changes**.
6. Return to the Overview page and click **Restart**.

Note the following when setting this value:

- The default value is 65000. The minimum value is 50. If the user exceeds the maximum value, then the server returns an error message when the table view is rendered. The maximum value is at least 16 bits, which varies by platform. The system is likely to consume all its memory before approaching a number larger than this value.
- In general, the **Maximum Number of Rows Processed when Rendering a Table View** field and the **Maximum Number of Rows to Download** field discussed in Use Fusion Middleware Control to Set Configuration Options for Data in Tables and Pivot Tables should be set to the same value to avoid getting `'Exceeded configured maximum number of allowed input records'` errors.
- The **Maximum Number of Rows Processed when Rendering a Table View** field sets `ResultRowLimit` in `instanceconfig.xml` (see the Performance tab section in Map User Interface Labels with Configuration File Elements). Both `ResultRowLimit` and `CubeMaxRecords` limit the number of rows returned, and the limit is determined by the setting with the larger value (see Manually Configure Cube Settings for Pivot Tables and Graphs).

For information about corresponding configuration file elements, see Map User Interface Labels with Configuration File Elements.

About the BI Server Query Cache

You can configure the BI Server to maintain a local, disk-based cache of query result sets (query cache).

The query cache enables the BI Server to satisfy many subsequent query requests without accessing back-end data sources, thereby increasing query performance.

As updates occur on the back-end databases, the query cache entries can become stale. Therefore, you must periodically remove entries from the query cache using one of the following methods:

- **Manually.** In the Model Administration Tool, in the **Manage** menu, select **Cache** to open the Cache Manager. The Cache Manager provides the most flexibility in choosing which cache entries to purge and when to purge them, but it requires manual intervention. See Use the Cache Manager.
- **Automatically.** In the Model Administration Tool, you can disable cache for the system, set caching attributes for a specific physical table, and use BI Server event tables to purge cache automatically. See Monitor and Manage the Cache.
- **Programmatically.** The BI Server provides ODBC-extension functions for purging cache entries programmatically. These functions give you the choice and the timing flexibility of the Cache Manager with the automation of event tables. You can write your own scripts to call these functions at times that fit your needs. See Purge and Maintain Cache Using ODBC Procedures.

The parameters that control query caching are located in Fusion Middleware Control and in the NQSConfig.INI file, described in Configuration File Settings. See Use Agents to Seed the Oracle BI Server Query Cache.

This section contains the following topics:

- Query Cache Architecture
- Advantages of Caching
- Costs of Caching
- Cache Sharing Across Users
- About the Refresh Interval for XML Data Sources
- About the Global Cache

Query Cache Architecture

The query cache consists of cache storage space, cache metadata, and cache detection in query compilation.

The process of the BI Server accessing the cache metadata is very fast. If the metadata shows a cache hit, then the bulk of the query processing is eliminated, and the results are immediately returned to the user. The process of adding the new results to the cache is independent of the results being returned to the user; the only effect on the running query is the resources that are consumed in the process of writing the cached results.

Query cache entries are portable across different operating systems, such as Windows or Linux 64-bit architecture. Incompatible cache entries are automatically removed.

Note that query cache entries are *not* portable across different updates of Oracle Analytics Server.

Caching occurs by default at the subrequest level, which results in multiple cache entries for some SQL statements. Caching subrequests improves performance and the cache hit ratio, especially for queries that combine real-time and historical data. To disable subrequest caching, set the NQSConfig.INI file parameter `DISABLE_SUBREQUEST_CACHING` to `YES`. See Configuration File Settings.

Advantages of Caching

The fastest way to process a query is to skip the bulk of the processing and use a precomputed answer.

With query caching, Oracle BI Server stores the precomputed results of queries in a local cache. If another query can use those results, then all database processing for that query is eliminated. This can result in dramatic improvements in the average query response time.

In addition to improving performance, being able to answer a query from a local cache conserves network resources and processing time on the database server. Network resources are conserved because intermediate results aren't returned to Oracle BI Server . Not running the query on the database frees the database server to do other work. If the database uses a charge back system, then running less queries might also cut costs in the budget.

Another benefit of using the cache to answer a query is savings in processing time on Oracle BI Server , especially if the query results are retrieved from multiple databases. Depending on the query, there might be considerable join and sort processing in the server. If the query is already calculated, then this processing is avoided, freeing server resources for other tasks.

To summarize, query caching can dramatically improve query performance and reduce network traffic, database processing, and processing overhead.

Costs of Caching

Query caching has many obvious benefits, but also certain costs.

- Disk space for the cache
- Administrative costs of managing the cache
- Potential for cached results being stale
- CPU and disk I/O on server computer

With cache management, the benefits typically far outweigh the costs.

Disk Space

The query cache requires dedicated disk space.

How much space depends on the query volume, the size of the query result sets, and how much disk space that you choose to allocate to the cache. For performance purposes, use a disk exclusively for caching, and ensure that it is a high performance, high reliability type of disk system.

Administrative Tasks Associated with Caching

Some administrative tasks are associated with caching. You must set the cache persistence time for each physical table appropriately, knowing how often data in that table is updated.

When the frequency of the update varies, you must keep track of when changes occur and purge the cache manually when necessary.

You can also create a cache event polling table and modify applications to update the polling table when changes to the databases occur, making the system event-driven.

The Oracle BI Server also provides ODBC-extension functions for purging cache entries programmatically. You can write your own scripts to call these functions at the appropriate times.

Keep the Cache Up To Date

If the cache entries aren't purged when the data in the underlying databases changes, then queries can potentially return results that are out of date.

You must evaluate whether this is acceptable. It might be acceptable to allow the cache to contain some stale data. You must decide what level of stale data is acceptable and then configure (and follow) a set of rules to reflect those levels.

For example, suppose an application analyzes corporate data from a large conglomerate, and you're performing yearly summaries of the different divisions in the company. New data doesn't materially affect the queries because the new data affects only next year's summaries. In this case, the trade-offs for deciding whether to purge the cache might favor leaving the entries in the cache.

Suppose, however, that the databases are updated three times a day and you're performing queries on the current day's activities. In this case, you must purge the cache much more often, or perhaps consider not using the cache at all.

Another scenario is that you rebuild the dataset from the beginning at periodic intervals (for example, once per week). In this example, you can purge the entire cache as part of the process of rebuilding the dataset, ensuring that you never have stale data in the cache.

Whatever your situation, you must evaluate what is acceptable for noncurrent information returned to the users.

CPU Usage and Disk I/O

Although usually it is very minor, query caching does require a small amount of CPU time and adds to the disk I/O.

In most cases, the CPU usage and disk I/O is insignificant. The disk I/O might be noticeable only when queries return large datasets.

Cache Sharing Across Users

If shared logon is enabled for a particular connection pool, then the cache can be shared across users and doesn't need to be seeded for each user.

If shared logon isn't enabled and a user-specific database login is used, then each user generates their own cache entry.

See *Managing Metadata Repositories for Oracle Analytics Server* for information about enabling shared logon for connection pools.

About the Refresh Interval for XML Data Sources

Typically, XML data sources are updated frequently and in real time. Setting a refresh interval for XML data sources is analogous to setting cache persistence for database tables.

The refresh interval is a time interval after which the XML data sources are to be queried again directly, rather than using results in cache. This refresh interval is specified on the XML tab of the Connection Pool dialog.

The default interval setting is **Infinite**, meaning that the XML data source isn't automatically refreshed.

The refresh interval setting determines the time interval after which the Oracle BI Server XML Gateway connection is refreshed, as follows:

- For URLs that begin with `http://` or `https://`, the gateway is refreshed when it detects that the interval has expired.
- For URLs that reside on a local or network drive, the gateway is refreshed when the interval has expired and the system detects that the URLs have been modified.

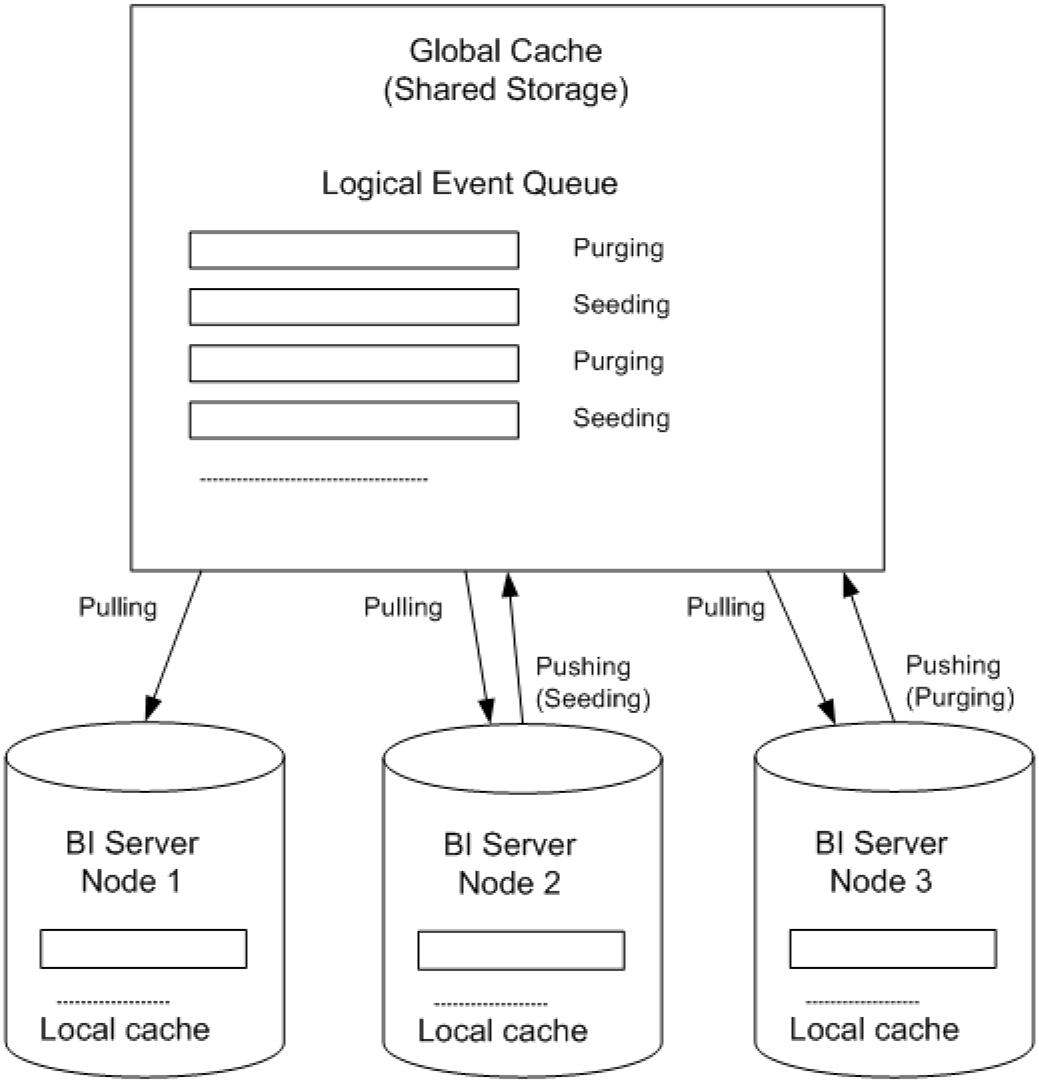
For more information about XML data sources, see *Managing Metadata Repositories for Oracle Analytics Server*.

About the Global Cache

In a clustered environment, Presentation Services can be configured to access a shared cache called the global cache.

This global cache resides on a shared file system storage device and stores purging events, seeding events (often generated by agents), and result sets that are associated with seeding events. The seeding and purging events are sorted by time and stored on the shared storage as a logical event queue. Individual Presentation Services nodes push to and pull from the logical event queue. Each Presentation Services still maintains its own local query cache for regular queries.

The figure below depicts global caching in a clustered environment. It shows three Presentation Services nodes sharing a global cache. The global cache stores seeding or purging events held in a logical event queue. The arrows from Node 2 and Node 3 to the shared cache show Presentation Services Node 2 pushing a seeding event to the queue and Presentation Services Node 3 pushing a purging event to the queue. The arrows from the shared storage to each Presentation Services node show each node pulling from the common location. This occurs on a periodic basis and enables participating Presentation Services nodes to obtain updates to the logical event queue made by other Presentation Services.



The Presentation Services node processes a seeding or purging event locally first in its caching system. It then pushes the event to the global cache on the shared storage. During the push event, the active Presentation Services node locks the logical event queue on the shared storage and then pushes in the seeding or purging event. If there's a conflict between seeding and purging (for example, one node wants to seed a query and another node wants to purge the same query), then the event that comes in last wins.

The logical event queue in the global cache on the shared storage is composed of seeding and purging events from individual Presentation Services nodes. The queue is sorted according to the timestamp of the events. Hence, clocks on all Presentation Services nodes participating in cluster must be synchronized.

Each Presentation Services node polls the global cache on a periodic basis for new cache entries. This polling frequency is configurable. A snapshot of the queued logical events on the shared storage is pulled back to the node and a local logical event queue is constructed and then processed.

> **Note:**
>
> The process of populating or purging seeded caches across all Presentation Services nodes that participate in the cluster doesn't occur in real time, and the elapse of the process is affected by multiple factors, such as the predefined polling interval, network bandwidth, and CPU loads.

Because the query cache result set tends to get large, network bandwidth might pose a constraint. Therefore, the following must be chosen carefully:

- The set of caches that qualify for seeded cache
- The time interval for Oracle Analytics nodes to pick up seeded caches from shared storage (to avoid network congestion)

The primary global cache parameters are configured in Fusion Middleware Control. Additional, optional parameters are configured in the NQSConfig.INI file for each Presentation Services node that participates in the cluster. See Use Fusion Middleware Control to Set Global Cache Parameters and Manually Edit Additional Global Cache Parameters.

A seeding or purging procedure is submitted to a specific Presentation Services node. If that Presentation Services is a node in a BI cluster and the global cache parameters have been defined in Presentation Services configuration files, then the seeding or purging events are propagated across all Presentation Services nodes that participate in the same clustered environment.

Configure Query Caching

You configure cache storage and other parameters in Fusion Middleware Control and in the NQSConfig.INI file, for both the query cache and the global cache.

You can also set in the Console, see Prompt Options in Configure Advanced System Settings in the Console.

You also must decide on a strategy for flushing outdated cache entries; see Monitor and Manage the Cache.

This section contains the following topics:

- Use Fusion Middleware Control to Enable and Disable Query Caching
- Use Fusion Middleware Control to Set Query Cache Parameters
- Manually Edit Additional Query Cache Parameters
- Use Fusion Middleware Control to Set Global Cache Parameters
- Manually Edit Additional Global Cache Parameters

Use Fusion Middleware Control to Enable and Disable Query Caching

You can use Fusion Middleware Control to enable or disable query caching.

The query cache is enabled by default.

Before you begin this procedure, ensure that you're familiar with the information in Use Fusion Middleware Control.

1. Go to the Business Intelligence Overview page, as described in Display Oracle Analytics Server Pages in Fusion Middleware Control.
2. Display the Performance tab of the Configuration page.
3. Click **Lock and Edit** to enable changes to be made.
4. To enable query caching, select **Cache enabled**. To disable query caching, deselect **Cache enabled**.
5. Click **Apply**, then click **Activate Changes**.
6. Return to the Business Intelligence Overview page and click **Restart**.

For information about corresponding configuration file elements, see Map User Interface Labels with Configuration File Elements.

Use Fusion Middleware Control to Set Query Cache Parameters

You can use Fusion Middleware Control to set the maximum number of cache entries in the query cache and the maximum size for a single cache entry.

Before you begin this procedure, ensure that you're familiar with the information in Use Fusion Middleware Control.

1. Go to the Business Intelligence Overview page, as described in Display Oracle Analytics Server Pages in Fusion Middleware Control.
2. Display the Performance tab of the Configuration page.
3. Click **Lock and Edit** to enable changes to be made.
4. Complete the elements using the descriptions in the help topic for the page.
   - **Maximum cache entry size**
   - **Maximum cache entries**
5. Click **Apply**, then click **Activate Changes**.
6. Return to the Business Intelligence Overview page and click **Restart**.

For information about corresponding configuration file elements, see Map User Interface Labels with Configuration File Elements.

Manually Edit Additional Query Cache Parameters

You can set additional query cache parameters in the NQSConfig.INI file.

Parameters include:

- The `DATA_STORAGE_PATHS` parameter specifies one or more directories for query cache storage, and the maximum size for each storage directory. These directories are used to store the cached query results and are accessed when a cache hit occurs. See About Cache Hits for more information about when cache is hit.

The cache storage directories reside on high performance storage devices, ideally devoted solely to cache storage. When the cache storage directories begin to fill up, the entries that are least recently used (LRU) are discarded to make space for new entries.

- The MAX_ROWS_PER_CACHE_ENTRY parameter controls the maximum number of rows for any cache entry. Limiting the number of rows is a useful way to avoid using up the cache space with runaway queries that return large numbers of rows. If the number of rows a query returns is greater than the value specified in the MAX_ROWS_PER_CACHE_ENTRY parameter, then the query isn't cached.
- Typically, if a query gets a cache hit from a previously run query, then the new query isn't added to the cache. The POPULATE_AGGREGATE_ROLLUP_HITS parameter overrides this default when the cache hit occurs by rolling up an aggregate from a previously run query.

See Configuration File Settings for more information about the additional query cache parameters.

Use Fusion Middleware Control to Set Global Cache Parameters

Setting global cache parameters ensures consistency across system cache configurations.

Before you begin this procedure, ensure that you're familiar with the information in Use Fusion Middleware Control.

1. Go to the Business Intelligence Overview page, as described in Display Oracle Analytics Server Pages in Fusion Middleware Control.
2. Display the Performance tab of the Configuration page.
3. Click **Lock and Edit** to enable changes to be made.
4. Complete the following options:
   - **Global cache path**
   - **Global cache size**
5. Click **Apply**, then click **Activate Changes**.
6. Return to the Business Intelligence Overview page and click **Restart**.

For information about corresponding configuration file elements, see Map User Interface Labels with Configuration File Elements.

Manually Edit Additional Global Cache Parameters

You can set additional global cache parameters in the NQSConfig.INI file.

Parameters include:

- The MAX_GLOBAL_CACHE_ENTRIES parameter controls the maximum number of entries that are allowed in the global cache store.
- The CACHE_POLL_SECONDS parameter specifies the interval in seconds at which the Oracle BI Server pulls from the logical event queue to synchronize with other server nodes in the cluster.
- The CLUSTER_AWARE_CACHE_LOGGING parameter controls whether logging is turned on for the global cache. Change this setting to YES only for debugging purposes.

  Log entries appear in nqquery.log. You can find this file at:

BI_DOMAIN/servers/obis*n*/logs

See Configuration File Settings.

Monitor and Manage the Cache

To manage the changes in the underlying databases and to monitor cache entries, you must develop a cache management strategy.

You need a process to invalidate cache entries when the data in the underlying tables that compose the cache entry changes, and a process to monitor, identify, and remove any undesirable cache entries.

This section contains the following topics:

- Choose a Cache Management Strategy
- Purge and Maintain Cache Using ODBC Procedures
- How Semantic Model Changes Affect the Query Cache

How Semantic Model Changes Affect the Query Cache

When you modify the semantic model (.rpd file), the changes can have implications for entries that are stored in the cache. For example, if you change the definition of a physical object or a dynamic variable, cache entries that reference that object or variable might no longer be valid. These changes might result in the need to purge the cache. There are two scenarios to be aware of: when you make changes, and when you're switching between semantic models.

Making Changes to the Semantic Model

When you modify a semantic model, any changes that you make that affect cache entries automatically result in a purge of all cache entries that reference the changed objects. The purge occurs when you check-in the changes. For example, if you delete a physical table from a semantic model, then all cache entries that reference that table are purged upon check in. Any changes made to a business model in the Business Model and Mapping or Logical layer purge all cache entries for that business model.

Switching Between Semantic Models

If you intend to remove a semantic model from the configuration of the BI Server, then ensure that you purge the cache of all cache entries that reference the semantic model. Failure to do so results in a corrupted cache.

Changes to Dynamic or Global Semantic Model Variables

The values of dynamic or global semantic model variables are refreshed by data that's returned from queries. When you define a dynamic or global semantic model variable, you create an initialization block or use a preexisting one that contains a SQL query. You also configure a schedule to run the query and periodically refresh the value of the variable.

If the value of a dynamic or global semantic modeler variable changes, then any cache entry which uses this variable in a column becomes stale, and a new cache entry is generated when data in that entry is needed again. The old cache entry isn't removed immediately, but remains until it is cleaned through the usual caching mechanism.

Choose a Cache Management Strategy

The choice of a cache management strategy depends on the volatility of the data in the underlying databases and the predictability of the changes that cause this volatility.

It also depends on the number and types of queries that comprise your cache and the usage those queries receive. This section provides an overview of the various approaches to cache management.

Disable Caching For the System

You can disable caching for the entire system to stop all new cache entries and stop any new queries from using the existing cache. Disabling caching lets you enable it at a later time without losing any entries that are stored in the cache.

Temporarily disabling caching is a useful strategy in situations where you might suspect having stale cache entries, but want to verify if they're actually stale before purging those entries or the entire cache. If you find that the data stored in the cache is still relevant, or after you have safely purged problem entries, then you can safely enable the cache. If necessary, purge the entire cache or the cache that's associated with a particular business model before enabling the cache again.

Cache and Cache Persistence Timing For Specified Physical Tables

You can set a cacheable attribute for each physical table, enabling you to specify whether queries for that table are added to the cache to answer future queries.

If you enable caching for a table, then any query involving the table is added to the cache. All tables are cacheable by default, but some tables mightn't be good candidates to include in the cache unless you set up suitable cache persistence settings. For example, suppose that you've a table that stores stock ticker data that's updated every minute. You can specify that you want to purge the entries for that table every 59 seconds.

You can also use cache persistence settings to specify how long the entries for this table are stored in the query cache. This is useful for data sources that are updated frequently.

1. In Model Administration Tool, in the Physical layer, double-click the physical table.
2. In the Physical Table properties dialog, in the General tab, make one of the following selections:
   - To enable caching, select **Cacheable**.
   - To prevent a table from being cached, deselect **Cacheable**.
3. To set a cache expiration time, specify a **Cache persistence time** and specify a unit of measure (days, hours, minutes, or seconds). If you don't want cache entries to automatically expire, select **Cache never expires**.
4. Click **OK**.

Configure BI Server Event Polling Tables

BI Server event polling tables store information about updates in the underlying databases.

You can configure an application (such as one that loads data into a data mart) to add rows to an event polling table each time a database table is updated. The BI Server polls this table at set intervals and invalidates any cache entries corresponding to the updated tables. Event polling tables can be the sole method of cache management, or they can be used with other cache management schemes. Event tables offer less flexibility about choice of cache entries and the timing of purges. See Set Up Event Polling Tables on the Physical Databases.

Purge and Maintain Cache Using ODBC Procedures

The Oracle BI Server provides ODBC-extension functions for purging cache entries.

Some of these functions are particularly useful for embedding in an Extract, Transform, and Load (ETL) task. For example, after a nightly ETL is performed, all Oracle BI Server cache entries can be purged. If only the fact table was modified, then only cache related to that table can be purged. In some cases, you might need to purge the cache entries associated with a specific database.

Only administrators have the right to purge cache. Therefore, scripts that call these ODBC-extension functions must run under credentials with administrator privileges.

The following ODBC functions affect cache entries that are associated with the semantic model specified by the ODBC connection:

- **SAPurgeCacheByQuery.** Purges cache entries that exactly match a specified query. For example, using the following query, you would have one or more query cache entries that retrieve the names of all employees earning more than $100,000:

  ```
  SELECT lastname, firstname FROM employee WHERE salary > 100000;
  ```

  The following call purges the cache entries that are associated with this query:

  ```
  Call SAPurgeCacheByQuery('SELECT lastname, firstname FROM employee WHERE
  salary > 100000' );
  ```

- **SAPurgeCacheByTable.** Purges all cache entries that are associated with a specified physical table name (fully qualified) for the semantic model to which the client has connected.

  This function takes up to four parameters that represent the four components (database, catalog, schema, and table name proper) of a fully qualified physical table name. For example, you might have a table with the fully qualified name of `DBName.CatName.SchName.TabName`. To purge the cache entries that are associated with this table in the physical layer of the semantic model, run the following call in a script:

  ```
  Call SAPurgeCacheByTable( 'DBName', 'CatName', 'SchName', 'TabName' );
  ```

  > **Note:**
  >
  > Wildcards aren't supported by the Oracle BI Server for this function. In addition, DBName and TabName can't be null. If either one is null, then an error message is displayed.

- **SAPurgeAllCache.** Purges all cache entries. This function takes one parameter that is a comma-separated list of the names of nodes. The following is an example of this call that purges cache entries on the obis1 and obis2 nodes:

  ```
  Call SAPurgeAllCache('obis1,obis2');
  ```

- **SAPurgeCacheByDatabase.** Purges all cache entries associated with a specific physical database name. A record is returned when any of the ODBC procedures to purge the cache are called. This function takes one parameter that represents the physical database name, and the parameter can't be null. The following shows the syntax of this call:

  ```
  Call SAPurgeCacheByDatabase( 'DBName' );
  ```

About ODBC Procedure Syntax

If there's a single quotation mark within the string argument of a procedure, then you must use another single quotation mark to escape it.

For example:

```
Call SAPurgeCacheByQuery('SELECT TOPN("- Currency"."Markdown %", 10) saw_0,
"XX Line"."Order No" saw_1, "- Bill-To Site"."Customer Name" saw_2, "-
Currency"."Net USD" saw_3, "- Currency"."Markdown USD" saw_4, "-
Currency"."Markdown %" saw_5 FROM "Apps 11i - XX Lines" WHERE
("XX Line"."Open Flag" = ''Y'') AND ("Operating Unit"."Group Name" =
''Group'')
AND ("- Currency"."Net USD" >= 10000) ORDER BY saw_0');
```

The line in bold highlights the extra single quotation marks that are used as escape characters for the items **`''Y''`** and **`''Group''`**.

About Sharing the Presentation Services Query Cache

When users access Answers to run queries, Presentation Services caches the results of the queries.

Presentation Services uses the request key and the logical SQL string to determine if subsequent queries can use cached results. If the cache can be shared, then subsequent queries aren't stored.

SAGetSharedRequestKey: An ODBC procedure that takes a logical SQL statement from Presentation Services and returns a request key value.

The following shows the syntax of this procedure:

```
SAGetSharedRequestKey('sql-string-literal')
```

The value of the request key is affected by the following factors:

- Whether the **Virtual Private Database** option is selected in the semantic model physical database object
- Whether any session variables are marked as **Security Sensitive** in the seamantic model

Presentation Services takes security sensitive variable values into consideration when computing the request key for logical requests against database objects marked as Virtual Private Databases.

See Manage Presentation Services Cache Settings.

About Result Records

A result record is returned after you issue a purge cache command.

The result record contains two columns. The first column is a result code and the second column is a short message that describes the result of the purge operation. The table below shows examples of result records.

Result Code	Result Message
1	SAPurgeCacheByDatabase returns successfully.
59115	Operation not performed because caching isn't enabled.
59116	The database specified doesn't exist.
59117	The table specified doesn't exist.

Strategies for Using the Cache

One of the main advantages of query caching is to improve apparent query performance.

Query caching might be valuable to seed the cache during off hours by running queries and caching their results. A good seeding strategy requires that you know when cache hits occur.

To seed the cache for all users, then you might seed the cache with the following query:

```
SELECT User, SRs
```

After seeding the cache using `SELECT User, SRs`, the following queries are cache hits:

```
SELECT User, SRs WHERE user = valueof(nq_SESSION.USER) (and the user was USER1)
SELECT User, SRs WHERE user = valueof(nq_SESSION.USER) (and the user was USER2)
SELECT User, SRs WHERE user = valueof(nq_SESSION.USER) (and the user was USER3)
```

This section contains the following topics:

- About Cache Hits
- Run a Suite of Queries to Populate the Cache
- Use Agents to Seed the Oracle BI Server Query Cache
- Use the Cache Manager

Run a Suite of Queries to Populate the Cache

To maximize potential cache hits, one strategy is to run a suite of queries to populate the cache.

The following are some recommendations for the types of queries to use when creating a suite of queries with which to seed the cache.

- **Common prebuilt queries.** Queries that are commonly run, particularly ones that are expensive to process, are excellent cache seeding queries. Queries whose results are embedded in dashboards are good examples of common queries.
- **SELECT lists with no expressions.** Eliminating expressions on `SELECT` list columns expands the possibility for cache hits. A cached column with an expression can only answer a new query with the same expression; a cached column with no expressions can answer a request for that column with any expression. For example, a cached request such as:

  ```
  SELECT QUANTITY, REVENUE...
  ```

  can answer a new query such as:

  ```
  SELECT QUANTITY/REVENUE...
  ```

  but not the reverse.
- **No WHERE clause.** If there is no `WHERE` clause in a cached result, then it can be used to answer queries that satisfy the cache hit rules for the select list with any `WHERE` clause that includes columns in the projection list.

In general, the best queries to seed cache with are queries that heavily consume database processing resources and that are likely to be reissued. Be careful not to seed the cache with simple queries that return many rows. These queries (for example, `SELECT * FROM PRODUCTS`, where `PRODUCTS` maps directly to a single database table) require very little database processing. Their expense is network and disk overhead, which are factors that caching doesn't alleviate.

When Oracle BI Server refreshes semantic model variables, it examines business models to determine if they reference those semantic model variables. If they do, Oracle BI Server purges all cache for those business models. See How Repository Changes Affect the Query Cache.

Use Agents to Seed the Oracle BI Server Query Cache

You can configure agents to seed the Oracle BI Server query cache.

Seeding the cache can improve response times for users when they run analyses or view analyses that are embedded on their dashboards. You can accomplish this by scheduling agents to run requests that refresh this data.

1. Log in to Oracle Analytics Server and select **New**, then select **Agent**.
2. On the General tab, select **Recipient** for the **Run As** option. Personalized cache seeding uses the data visibility of each recipient to customize agent delivery content for each recipient.
3. On the Schedule tab, specify when you want the cache to be seeded.
4. Optional: Select **Condition** and create or select a conditional request. For example, you might have a business model that determines when the ETL process is complete. You could use a report based on this business model to be the conditional trigger for the cache seed to begin.
5. On the Delivery Content tab, select an individual request or an entire dashboard page for which you want to seed the cache. Selecting a dashboard page can save time.
6. On the Recipients tab, select individual users or groups to be the recipients.
7. On the Destinations tab, clear all user destinations and select **Oracle Analytics Server Cache**.

8. Save the agent by selecting the **Save** button in the upper-right corner.

The only difference between cache seeding agents and other agents is that they clear the previous cache automatically and don't appear on the dashboard as alerts.

> **Note:**
>
> Cache seeding agents only purge exact match queries, so stale data might still exist. Ensure that the caching strategy always include cache purging, because agent queries don't address ad-hoc queries or drills.

About Cache Hits

When caching is enabled, each query is evaluated to determine whether it qualifies for a cache hit.

A cache hit means that Oracle BI Server was able to use cache to answer the query and didn't go to the database at all. Oracle BI Server can use the query cache to answer queries at the same or higher level of aggregation.

Many factors determine whether cache is hit. The table below describes these factors.

Factor or Rule	Description
A subset of columns in the `SELECT` list must match	All of the columns in the `SELECT` list of a new query have to exist in the cached query to qualify for a cache hit, or they must be able to be calculated from the columns in the query. This rule describes the minimum requirement to hit the cache, but meeting this rule doesn't guarantee a cache hit. The other rules listed in this table also apply.

Columns in the `SELECT` list can be composed of expressions on the columns of the cached queries	Oracle BI Server can calculate expressions on cached results to answer the new query, but all the columns must be in the cached result. For example, the query:
	`SELECT product, month, averageprice FROM sales WHERE year = 2000`
	hits cache on the query:
	`SELECT product, month, dollars, unitsales FROM sales WHERE year = 2000`
	because `averageprice` can be computed from `dollars` and `unitsales` (`averageprice = dollars/unitsales`).

Factor or Rule	Description

WHERE clause must be semantically the same or a logical subset	For the query to qualify as a cache hit, the WHERE clause constraints must be either equivalent to the cached results, or a subset of the cached results.
	A WHERE clause that's a logical subset of a cached query qualifies for a cache hit if the subset meets one of the following criterion:
	- A subset of IN list values. Queries requesting fewer elements of an IN list cached query qualify for a cache hit. For example, the following query:
	`SELECT employeename, region`
	`FROM employee, geography`
	`WHERE region in ('EAST', 'WEST')`
	qualifies as a hit on the following cached query:
	`SELECT employeename, region`
	`FROM employee, geography`
	`WHERE region in ('NORTH', 'SOUTH', 'EAST', 'WEST')`
	- It contains fewer (but identical) OR constraints than the cached result.
	- It contains a logical subset of a literal comparison. For example, the following predicate:
	`WHERE revenue < 1000`
	qualifies as a cache hit on a comparable query with the predicate:
	`WHERE revenue < 5000`
	- There is no WHERE clause. If a query with no WHERE clause is cached, then queries that satisfy all other cache hit rules qualify as cache hits regardless of their WHERE clause.
	In addition columns that are used on the WHERE clause must be on the projection list. For example, the following query:
	`SELECT employeename`
	`FROM employee, geography`
	`WHERE region in ('EAST', 'WEST')`
	Doesn't result in a cache hit for the seeding query in the previous list because REGION isn't on the projection list.

Dimension-only queries must be an exact match	If a query is dimension only, meaning that no fact or measure is included in the query, then only an exact match of the projection columns of the cached query hits the cache. This behavior prevents false positives when there are multiple logical sources for a dimension table.

Factor or Rule	Description
Queries with special functions must be an exact match	Other queries that contain special functions such as time series functions (`AGO`, `TODATE`, and `PERIODROLLING`), limit and offset functions (`OFFSET` and `FETCH`), relationship functions (`ISANCESTOR`, `ISLEAF`, `ISROOT`, and `ISSIBLING`), external aggregation functions, and generally filter metrics must also be an exact match with the projection columns in the cached query. In these cases, the filter must also be an exact match. For filter metrics, if the filter metric can be rewritten as a `WHERE` clause, then the subset cache might be leveraged.
Set of logical tables must match	To qualify as a cache hit, all incoming queries must have the same set of logical tables as the cache entry. This rule avoids false cache hits. For example, `SELECT * FROM product` doesn't match `SELECT * FROM product, sales`.
Session variable values must match, including security session variables	If the logical SQL or physical SQL statement refers to any session variable, then the session variable values must match. Otherwise, the cache isn't hit. In addition, the value of session variables that are security sensitive must match the security session variable values that are defined in the semantic model, even though the logical SQL statement itself doesn't reference session variables. See Ensure Correct Cache Results When Using Row-Level Database Security.
Equivalent join conditions	The resultant joined logical table of a new query request has to be the same as (or a subset of) the cached results to qualify for a cache hit.
`DISTINCT` attribute must be the same	If a cached query eliminates duplicate records with `DISTINCT` processing (for example, `SELECT DISTINCT`...), then requests for the cached columns must also include the `DISTINCT` processing; a request for the same column without the `DISTINCT` processing is a cache miss.

Queries must contain compatible aggregation levels	Queries that request an aggregated level of information can use cached results at a lower level of aggregation. For example, the following query requests the quantity sold at the supplier and region and city level:
	`SELECT supplier, region, city, qtysold`
	`FROM suppliercity`
	The following query requests the quantity sold at the city level:
	`SELECT city, qtysold`
	`FROM suppliercity`
	The second query results in a cache hit on the first query.

Limited additional aggregation	For example, if a query with the column `qtysold` is cached, then a request for `RANK(qtysold)` results in a cache miss. Additionally, a query that requests `qtysold` at the country level can get a cache hit from a query that requests `qtysold` at the country, region level.
`ORDER BY` clause must be comprised of columns in the select list	Queries that order by columns that aren't contained in the select list result in cache misses.
Avoiding cache misses using advanced hit detection	You can avoid some cache misses by setting the parameter `USE_ADVANCED_HIT_DETECTION` to `YES` in the NQSConfig.INI file. Advanced hit detection enables an expanded search of the cache for hits. See USE_ADVANCED_HIT_DETECTIONUSE_ADVANCED_HIT_DETECTION.

Factor or Rule	Description
Diagnosing cache hit behavior	To better assess cache hit behavior, set the ENABLE_CACHE_DIAGNOSTICS session variable to 4, as shown in the following example: `ENABLE_CACHE_DIAGNOSTICS=4`

Ensure Correct Cache Results When Using Row-Level Database Security

When using a row-level database security strategy, such as a Virtual Private Database (VPD), the returned data results are contingent on the authorization credentials of the user.

Because of this, Oracle BI Server must know whether a data source is using row-level database security and which variables are relevant to security.

To ensure that cache hits only occur on cache entries that include and match all security-sensitive variables, you must correctly configure the database object and session variable objects in the Model Administration Tool, as follows:

- **Database object.** In the Physical layer, in the General tab of the Database dialog, select **Virtual Private Database** to specify that the data source is using row-level database security.

  If you're using row-level database security with shared caching, then you *must* select this option to prevent the sharing of cache entries whose security-sensitive variables don't match.

- **Session Variable object.** For security-related variables, in the Session Variable dialog, select **Security Sensitive** to identify them as sensitive to security when using a row-level database security strategy. This option ensures that cache entries are marked with the security-sensitive variables, enabling security-sensitive variable matching on all incoming queries.

Refer to the following resources:

- Setting Up Row-Level Security in the Database in *Managing Metadata Repositories for Oracle Analytics Server*
- Managing Session Variables in *Managing Metadata Repositories for Oracle Analytics Server*
- *Managing Metadata Repositories for Oracle Analytics Server* for general information about database and session variable objects

Use the Cache Manager

The Cache Manager lets you view information about the entire query cache and information about individual entries in the query cache that are associated with the open semantic model.

You can also use the Cache Manager to select specific cache entries and perform various operations on those entries, such as viewing and saving the cached SQL statement, or purging them.

1. In the Model Administration Tool toolbar, select **Manage**, then select **Cache**.
2. Click the **Cache** tab to view the cache entries for the current semantic model, business models, and users.

The associated cache entries are reflected in the right pane, with the total number of entries shown in the view-only field at the top.

3. Select **Edit**, then select **Options** from the Cache Manager, or select **Tools**, **Options**, and then **Cache Manager** from the Model Administration Tool menu to control the cache entry information and its display sequence.

You can expand the semantic model tree to display all the business models with cache entries, and expand the business models to display all users with cache entries. The right pane displays only the cache entries associated with the selected item in the hierarchical tree.

About Cache Manager Options

The Cache Manager can display the options described here.

Option	Description
User	The ID of the user who submitted the query that resulted in the cache entry.
Created	The time the cache entry's result set was created.
Last used	The last time the cache entry's result set satisfied a query. (After an unexpected shutdown of the Oracle BI Server, the last used time might temporarily have a stale value—a value that's older than the true value.)
Creation elapsed time	The time, in seconds, that's needed to create the result set for this cache entry. The value that's stored in the cache object descriptor on disk is in units of milliseconds. The value is converted to seconds for display purposes.
Row count	The number of rows generated by the query.
Row size	The size of each row (in bytes) in this cache entry's result set.
Full size	Full size is the maximum size used, considering variable length columns, compression algorithm, and other factors. The actual size of the result set is smaller than Full size.
Column count	The number of columns in each row of this cache entry's result set.
Logical Request	The logical request that's associated with this cache entry. If subrequests are being cached, then this column shows the text of the subrequest.
Use count	The number of times that this cache entry's result set has satisfied a query since Oracle BI Server startup.
Business model	The name of the business model that's associated with the cache entry.
Repository	The name of the semantic model associated with this cache entry.
SQL	The SQL statement associated with this cache entry. If subrequests are being cached, then there might be multiple cache entries associated with a single SQL statement.
Query Server	The Oracle BI Server that serviced the query.
Fact Table Source	The fact table associated with the logical request for this cache entry.

Display Global Cache Information in the Cache Manager

Select **Action**, then select **Show Info** to display global cache information.

The table below describes the information that appears in the Global Cache Information window.

Column	Description
Amount of space still available for cache storage use	The amount of space, in megabytes, still available for cache storage.
Amount of space used on disks containing cache related files	The total amount of space, in megabytes, used on the disk that contains cache-related files (not just space used for the cache-related files).
Maximum allowable number of entries in cache	The maximum number of entries that can be in the cache, from the `MAX_CACHE_ENTRIES` parameter in the `NQSConfig.INI` file.
Maximum allowable number of rows per cache entry result set	The maximum number of rows that are allowed for each cache entry's result set, from the `MAX_ROWS_PER_CACHE_ENTRY` parameter in the `NQSConfig.INI` file.
Number of entries currently in cache	The current number of entries in the global cache. These entries might relate to multiple semantic models.
Number of queries not satisfied from cache since startup of Oracle BI Server	Cache misses, since the last time the Oracle BI Server was started.
Number of queries satisfied from cache since startup of Oracle BI Server	Cache hits, since the last time the Oracle BI Server was started.

With the Cache Manager as the active window, press **F5**, or select **Action**, then **Refresh** to refresh the display. This retrieves the current cache entries for the semantic model that you have open and the current global cache information. If the DSN is clustered, then information about all semantic models in the cluster is displayed.

Purge the Cache in the Model Administration Tool

Purging the cache deletes entries from the query cache.

You can purge cache entries in the following ways:

- Manually, using the Model Administration Tool Cache Manager facility.
- Automatically, by setting the **Cache Persistence Time** field in the Physical Table dialog for a particular table.
- Automatically, by setting up an Oracle BI Server event polling table.
- Automatically, as the cache storage space fills up.

> **Note:**
>
> You can also purge the cache programmatically using ODBC-extension functions. See Purge and Maintain Cache Using ODBC Procedures.
>
> In addition, cache can be purged when the value of dynamic or global semantic model variables changes. See Changes to Dynamic or Global Semantic Model Variables.

You can manually purge cache entries in the Cache Manager as follows:

1. Use the Model Administration Tool to open a semantic model.

2. Select **Manage**, then **Cache** to open the Cache Manager dialog.
3. Select Cache or Physical mode by selecting the appropriate tab in the left pane.
4. Browse the explorer tree to display the associated cache entries in the right pane.
5. Select the cache entries to purge, and then select **Edit**, then **Purge** to remove them. Or, right-click the selected entries and then select **Purge**.

   - In Cache mode, select the entries to purge from those displayed in the right pane.
   - In Physical mode, select the database, catalog, schema or tables to purge from the explorer tree in the left pane.

   In Cache mode, you can purge:

   - One or more selected cache entries that are associated with the open semantic model.
   - One or more selected cache entries that are associated with a specified business model.
   - One or more selected cache entries that are associated with a specified user within a business model.

   In Physical mode, you can purge:

   - All cache entries for all tables that are associated with one or more selected databases.
   - All cache entries for all tables that are associated with one or more selected catalogs.
   - All cache entries for all tables that are associated with one or more selected schemas.
   - All cache entries that are associated with one or more selected tables.

Purging deletes the selected cache entries and associated metadata. Select **Action**, then **Refresh** or press F5 to refresh the cache display.

Cache Event Processing with an Event Polling Table

You can use an Oracle BI Server event polling table (event table) as a way to notify the Oracle BI Server that one or more physical tables have been updated. Each row that's added to an event table describes a single update event, such as an update occurring to the Product table in the Production database.

The Oracle BI Server cache system reads rows from, or polls, the event table, extracts the physical table information from the rows, and purges stale cache entries that reference those physical tables.

The event table is a physical table that resides on a relational database accessible to the Oracle BI Server. Regardless of whether it resides in its own database, or in a database with other tables, it requires a fixed schema as described in Set Up Event Polling Tables on the Physical Databases. It is normally exposed only in the Physical layer of the Model Administration Tool, where it's identified in the Physical Table dialog as an Oracle BI Server event table.

Using event tables is one of the most accurate ways of invalidating stale cache entries, and is probably the most reliable method. It does, however, require the event table to be populated each time that a database table is updated. Also, because there is a polling interval in which the cache isn't completely up to date, there's always the potential for stale data in the cache. See Populate the Oracle BI Server Event Polling Table.

A typical method of updating the event table is to include SQL `INSERT` statements in the extraction and load scripts or programs that populate the databases. The `INSERT` statements add one row to the event table each time that a physical table is modified. After this process is in place and the event table is configured in the Oracle Analytics Server semantic model, cache invalidation occurs automatically. As long as the scripts that update the event table are accurately recording changes to the tables, stale cache entries are purged automatically at the specified polling intervals.

This section contains the following topics:

- Set Up Event Polling Tables on the Physical Databases
- Make the Event Polling Table Active
- Populate the Oracle BI Server Event Polling Table
- Troubleshoot Problems with Event Polling Tables

Set Up Event Polling Tables on the Physical Databases

You can configure a physical event polling table on each physical database to monitor changes in the database.

You can also configure the event table in its own database. The event table is updated every time a table in the database changes.

If the event polling table is on an Oracle Database, then configure the event table in its own database object in the Physical layer of the Model Administration Tool. Then, ensure that the feature `PERF_PREFER_IN_LISTS` isn't selected in the Features tab of the Database dialog for the event polling table. Following these guidelines avoids errors related to exceeding the maximum number of allowed expressions in a list.

To create an event polling table, run the Repository Creation Utility (RCU) to create the BIPLATFORM schemas in your physical database. RCU creates an event polling table called *S_NQ_EPT*. See *Installing and Configuring Oracle Analytics Server* for information about running the Repository Creation Utility.

Event tables must have the structure that's shown in the following table. Some columns can contain null values, depending on where the event table resides. The names for the columns must match the column names that are shown in the next table. Data Types shown are for an Oracle Database.

Event Table Column	Data Type	Description
CATALOG_NAME	VARCHAR2	The name of the catalog where the physical table that was updated resides. Populate the CATALOG_NAME column only if the event table doesn't reside in the same database as the physical tables that were updated. Otherwise, set it to the null value.

Event Table Column	Data Type	Description
DATABASE_NAME	VARCHAR2	The name of the database where the physical table that was updated resides. This is the name of the database as it's defined in the Physical layer of the Model Administration Tool. For example, if the physical database name is 11308Production, and the database name that represents it in the Model Administration Tool is SQL_Production, then the polled rows in the event table must contain SQL_Production as the database name. Populate the DATABASE_NAME column only if the event table doesn't reside in the same database as the physical tables that were updated. Otherwise, set it to the null value.
OTHER_RESERVED	VARCHAR2	Reserved for future enhancements. This column must be set to a null value.
SCHEMA_NAME	VARCHAR2	The name of the schema where the physical table that was updated resides. Populate the SCHEMA_NAME column only if the event table doesn't reside in the same database as the physical tables being updated. Otherwise, set it to a null value.
TABLE_NAME	VARCHAR2	The name of the physical table that was updated. The name must match the name that's defined for the table in the Physical layer of the Model Administration Tool. Values can't be null.
UPDATE_TS	DATE	The time when the update to the event table occurs. This must be a key (unique) value that increases for each row that's added to the event table. To ensure a unique and increasing value, specify the current timestamp as a default value for the column. For example, specify `DEFAULT CURRENT_TIMESTAMP` for Oracle Database. Values can't be null. Because this column must be a unique value that increases for each row that's added to the event table, you might need to set a very high precision if you require many inserts per second. Because of this, you might want to adjust the database feature `FRACTIONAL_SECOND_PRECISION` to enable fractional seconds to be used in the filters on the **UpdateTime** column. The Oracle BI Server truncates the timestamps to the number of digits that are defined by `FRACTIONAL_SECOND_PRECISION`. For example, for Oracle Database or Teradata, you might want to change `FRACTIONAL_SECOND PRECISION` from 0 to 6.
UPDATE_TYPE	NUMBER	Specify a value of 1 in the update script to indicate a standard update. You can't use null values.

The Oracle BI Server must have read and write permission on the event polling table. The server reads the event table at specified intervals to look for changed data. Applications add rows to the event table when database tables are modified (for example, during a load operation). When there are rows in the event table, there's changed data in the underlying databases. The server then invalidates any cache entries that correspond to the changed physical tables and periodically deletes obsolete rows from the event table. The next time it checks the event table, the process repeats.

> **Note:**
>
> In a clustered deployment, a single event polling table is shared by every Oracle BI Server node in the cluster. However, a single event polling table can't be shared by multiple Oracle BI Server clusters.

To enable the Oracle BI Server to have write access to the event polling table but not to any other tables in a database, perform the following tasks:

- Create a separate physical database in the Physical layer of the Model Administration Tool with a privileged connection pool.
- Assign a user to the connection pool that has delete privileges.
- Populate the privileged database with the event table.

The Oracle BI Server has write access to the event polling table, but not to any tables that are used to answer user queries.

Make the Event Polling Table Active

After the table is created on the physical database, you can make it active in the Oracle BI Server. To do this, you first import the physical table, and then you mark the table object as an event polling table.

To import the physical table:

1. In the Model Administration Tool, open the semantic model and import metadata from the physical database. To do this, select **File**, then select **Import Metadata**.
2. Follow the wizard steps. Be sure to select the **Tables** option in the Select Metadata Types screen to import the table metadata.

   See Import Metadata and Working with Data Sources in *Managing Metadata Repositories for Oracle Analytics Server* for detailed information about the Import Metadata wizard.
3. If you have multiple event polling tables, then repeat steps 1 and 2 for each event table. Be sure the data source that's specified for the event table has read and write access to the event table. The semantic model both reads the table and deletes rows from it, so it needs write permission. Event tables don't need to be exposed in the Business Model and Mapping or logical layer.

To mark the table object as an event polling table:

1. From the Tools menu, select **Utilities**.
2. Select the option **Oracle BI Event Tables** from the list of options.
3. Click **Execute**.
4. Select the table to register as an Event Table and click the **>>** button.
5. Specify the polling frequency in minutes, and click **OK**.

   The default value is 60 minutes. Don't set the polling frequency to less than 10 minutes. If you want a very short polling interval, then consider marking some or all of the tables noncacheable.

When a table has been registered as an Oracle BI Server event table, the table properties change. Registration as an event table removes the option to make the table cacheable, as there's no reason to cache results from an event polling table.

Populate the Oracle BI Server Event Polling Table

The Oracle BI Server doesn't populate the event polling table. The event table is populated by inserting rows into it each time a table is updated.

This process is normally configured by the database administrator, who typically modifies the load process to insert a row into the polling table each time a table is modified. This can be done from the load script, using database triggers (in databases that support triggers), from an application, or manually. If the process of populating the event table isn't done correctly, then the Oracle BI Server cache purging is affected, because the server assumes the information in the polling table is correct and up to date.

Troubleshoot Problems with Event Polling Tables

You can start troubleshooting event polling table issues in activity logs.

If you experience problems with cache polling, then you can search the Oracle BI Server activity logs for any entries regarding the server's interaction with the event table.

- The obis1-diagnostic.log file logs activity automatically about the Oracle BI Server. Log entries are self-explanatory and can be viewed in Fusion Middleware Control or in a text editor.
- When the Oracle BI Server polls the event table, it logs queries in the nqquery.log file using the administrator account (set upon installation) unless the logging level for the administrator account is set to 0. Set the logging level to 2 for the administrator account to provide the most useful level of information.

You can find the obis1-diagnostic.log and the nqquery.log in the following location:

```
BI_DOMAIN/servers/obisn/logs
```

Configure the Data Source Cache

To improve performance for data blending, you must configure a data source cache. You do this either using a database or on the file system.

- Configuring a data source cache stored on the file system
- Configuring a data source cache using a database
- Managing the data source cache

Create a Data Source Cache on the File System

You can store your data source cache data on the file system for direct access by BI Server. This is the default configuration. This may be adequate for small- to medium-sized data sources, such as spreadsheet files that you have uploaded. For larger data sources, configuring the data source cache on a database may provide better performance.

The file system-based data source cache works as follows:

1. BI Server receives logical SQL statement against the data source.

2. BI Server retrieves the metadata of the data source. Using the metadata's content, BI Server checks if there's any existing data source cache entry that can be used to answer the query instead of requesting data from the data source.
3. For any data source that doesn’t have an existing data source cache entry, BI Server starts a thread to load the data source’s data into the cache.
4. The logical SQL statement (and all other concurrent logical SQL statements that require data from the same data source) waits for the thread to finish.
5. Once the thread's execution is complete, any subrequest that depends on the newly-seeded data source cache entry uses the cache.

Configure the data source cache on the file system:

1. Make the following changes to the `NQSConfig.INI` file in the `XSA_CACHE` section:

   ```
   [XSA_CACHE]
   ENABLE = YES;

   # The schema and connection pool where the XSA data is cached.
   # Set PHYSICAL_SCHEMA to "".""" to use file-based XSA cache.
   # This indicates that a file-based XSA cache should be used by BI Server
   and the value of the CONNECTION_POOL parameter should be ignored.
   PHYSICAL_SCHEMA = "".""; # "<Database>"."<Schema>";
   CONNECTION_POOL = "".""; # "<Database>"."<Connection Pool>";

   # The path to the location where cache data files is persisted.
   # This is only used if file-based XSA cache is configured.
   # If a relative path is specified, it is relative to:
   # BI_DOMAIN/servers/obisn
   STORAGE_DIRECTORY = "storage"; # location where the cache data files are
   stored

   # The maximum space allocated in the schema for the cache data.
   MAX_TOTAL_SPACE = 5 GB;

   # The maximum size allowed for a single XSA cache entry.
   MAX_CACHE_ENTRY_SIZE = 200 MB;

   # The path to the location where descriptor files of the cache data is
   persisted.
   # If a relative path is specified, it is relative to:
   # BI_DOMAIN/servers/obisn
   DESCRIPTOR_STORAGE_PATH = "xsacache"; # location where the cache metadata
   files are stored

   # The number of threads available for seeding XSA cache entries.
   CACHE_SEED_THREAD_RANGE = 0-40;
   CACHE_SEED_THREAD_STACK_SIZE = 0; # default is 256 KB (32 BIT mode), 1 MB
   (64 BIT mode), 0 for default
   ```

2. Restart BI Server.
3.

Validate that the data source cache is configured correctly, check the obis1-diagnostic.log. Upon server startup, look for entries similar to the following:

```
[2017-01-13T14:41:30.715-07:00] [OracleBIServerComponent] [NOTIFICATION:1]
[] [] [ecid: ] [tid: 81c8] [101043] External Subject Area cache with
internal storage is started successfully using configuration from the
repository with the logical name Star.
[2011-01-13T14:41:30.716-07:00] [OracleBIServerComponent] [NOTIFICATION:1]
[] [] [ecid: ] [tid: 81c8] [101017] External Subject Area cache has been
initialized. Total number of entries: 0 Used space: 0 bytes Maximum space:
107374182400 bytes Remaining space: 107374182400 bytes. Cache table name
prefix is XC.
```

Once you enable the data source cache, it stores the data from the data source on the file system as defined in the STORAGE_DIRECTORY parameter in the XSA_CACHE section of NQSConfig.INI. It stores the metadata on the local file system in the directory specified by the DESCRIPTOR_STORAGE_PATH parameter under the XSA_CACHE section in NQSConfig.INI. To improve performance, put your data source data and data source cache metadata files on a RAM disk.

Create a Data Source Cache on a Database

For larger data sources, configuring the data source cache on a database may provide better performance.

The database-based data source cache works as follows:

1. BI Server receives a logical SQL statement against the data source.
2. BI Server retrieves the metadata of the data source. Using the metadata's content, BI Server checks if there's any existing data source cache entry that can be used to answer the query instead of requesting data from the data source. Any subrequest that can use the data source cache is rewritten to the native physical SQL of the cache database.
3. For any data source that doesn’t have an existing data source cache entry, BI Server starts a thread to load the data source’s data into the cache database. Each data source translates to a single database table containing all columns form the data source.
4. The logical SQL statement (and all other concurrent logical SQL statements that require data from the same data source) waits for the thread to finish.
5. Once the thread's execution is done, any subrequest that depends on the newly-seeded data source cache entry is rewritten to the native physical SQL of the cache database.

You can choose between the two following supported data source cache database setups:

- On a database that is different from the data warehouse.
- On the same database as the data warehouse, but on a different schema.

Configure the Data Source Cache on a Different Database than the Data Warehouse

1. Create the physical objects that represent the cache database in your semantic model. This includes a new database object, a new physical schema object, and a new connection pool. For this example, our database object is named XSACache, our connection pool is named CP, and a new physical schema object is named XSA_CACHE.
    - The user name used in the connection pool must be the same as the name of the schema.

- The user specified in the connection pool must have the necessary privileges to run DDLs and DMLs against tables in the schema. For example, on Oracle, the user must have at least the CREATE TABLE privilege.

2. Update parameters under the XSA_CACHE section in NQSConfig.INI to point to the right schema and connection pool.

```
[XSA_CACHE]
ENABLE = YES;

# The schema and connection pool where the XSA data is cached.
# Set PHYSICAL_SCHEMA to "".""  to use file-based XSA cache.
PHYSICAL_SCHEMA = "XSACache"."XSA_CACHE"; # "<Database>"."<Schema>";
CONNECTION_POOL = "XSACache"."CP"; # "<Database>"."<Connection Pool>";

# The path to the location where descriptor files of the cache data is
persisted.
# If a relative path is specified, it is relative to:
# BI_DOMAIN/servers/obisn
DESCRIPTOR_STORAGE_PATH = "xsacache"; # location where the cache metadata
files are stored
```

3. Restart BI Server.

4. To validate that the data source cache is configured correctly, check the obis1-diagnostic.log. Upon server startup, look for entries similar to the following:

```
[2017-01-13T14:41:30.715-07:00] [OracleBIServerComponent] [NOTIFICATION:1]
[] [] [ecid: ] [tid: 81c8] [101001] External Subject Area cache is started
successfully using configuration from the repository with the logical name
Star.
[2011-01-13T14:41:30.716-07:00] [OracleBIServerComponent] [NOTIFICATION:1]
[] [] [ecid: ] [tid: 81c8] [101017] External Subject Area cache has been
initialized. Total number of entries: 0 Used space: 0 bytes Maximum space:
107374182400 bytes Remaining space: 107374182400 bytes. Cache table name
prefix is XC2627531314.
```

Once you enable the data source cache, it stores the data for the data source in the database. However it stores the metadata on the local file system in the directory specified by the DESCRIPTOR_STORAGE_PATH parameter under the XSA_CACHE section in NQSConfig.INI. To improve performance, put your data source cache metadata files on a RAM disk.

Configuring the Data Source Cache on the Same Database as the Data Warehouse

This configuration improves query performance because joins can be pushed down to the data source instead of being run internally by BI Server.

1. Create the physical objects that represent the cache database in your semantic model under the data warehouse database object. This includes a new physical schema object and a new connection pool. For this example, the new connection pool is called XSA Cache Connection Pool, and the physical schema object is called XSA_CACHE.

   The approach is to have a dedicated connection pool for data source cache seeding and purging while allowing the data warehouse connection pool to query data source cache tables created in a separate schema. The data source cache schema should ideally have its own tablespace to simplify management.

- The user name used in the new connection pool must be the same as the name of the new schema.
- The user specified in the new connection pool must have the necessary privileges to run DDLs and DMLs against tables in the new schema. For example, on Oracle, the user must have at least the `CREATE TABLE` privilege.
- The user specified in the original data warehouse connection pool must have the necessary privileges to `SELECT` tables in the new schema. For example, on Oracle, the user in the existing data warehouse connection pool must have at least the `SELECT ANY TABLE` privilege.
- The data source cache connection pool must not be the primary connection pool under the database object. In other words, the data source cache connection pool must be ordered after the data warehouse connection pool that's used for data warehouse queries.

2. Update parameters under the `XSA_CACHE` section in `NQSConfig.INI` to point to the right schema and connection pool.

   ```
   [XSA_CACHE]
   ENABLE = YES;

   # The schema and connection pool where the XSA data is cached.
   # Set PHYSICAL_SCHEMA to ""."" to use file-based XSA cache.
   PHYSICAL_SCHEMA = "Oracle Data Warehouse"."Catalog"."XSA_CACHE"; #
   "<Database>"."<Schema>";
   CONNECTION_POOL = "Oracle Data Warehouse"."XSA Cache Connection Pool"; #
   "<Database>"."<Connection Pool>";

   # The path to the location where descriptor files of the cache data is
   persisted.
   # If a relative path is specified, it is relative to:
   # BI_DOMAIN/servers/obisn
   DESCRIPTOR_STORAGE_PATH = "xsacache"; # location where the cache metadata
   files are stored
   ```

3. Restart BI Server.
4. To validate that the data source cache is configured correctly, check the `obis1-diagnostic.log`. Upon server startup, look for entries similar to the following:

   ```
   [2017-01-13T14:41:30.715-07:00] [OracleBIServerComponent] [NOTIFICATION:1]
   [] [] [ecid: ] [tid: 81c8] [101001] External Subject Area cache is started
   successfully using configuration from the repository with the logical name
   Star.
   [2011-01-13T14:41:30.716-07:00] [OracleBIServerComponent] [NOTIFICATION:1]
   [] [] [ecid: ] [tid: 81c8] [101017] External Subject Area cache has been
   initialized. Total number of entries: 0 Used space: 0 bytes Maximum space:
   107374182400 bytes Remaining space: 107374182400 bytes. Cache table name
   prefix is XC2627531314.
   ```

Once you enable the data source cache, it stores the data for uploaded files in the database. However it store the metadata on the local file system in the directory specified by the `DESCRIPTOR_STORAGE_PATH` parameter under the `XSA_CACHE` section in `NQSConfig.INI`. To improve performance, Oracle recommends that you put your data source cache metadata files on RAMDisk.

Manage the Data Source Cache

Once you've enabled the data source cache, you may need to delete individual cache entries or the entire cache.

Removing Data Source Cache Entries

Oracle BI Server provides ODBC-extension functions for purging cache entries. The format of the stored procedure is as follows:

```
call SAPurgeXSACache('<XSA_PATH>', '<XSA_TABLE>');
```

`<XSA_PATH>` is the path to a data source definition. This is the value used within an `XSA()` clause in a logical SQL. This parameter also supports the wildcard character `%` that can be used to purge all data source cache entries in the system.

`<XSA_TABLE>` is a table name within a data source. This is the value that follows an `XSA()` clause in a logical SQL. This parameter also supports the wildcard character `%` that can be used to purge cache entries of all tables within a data source definition.

For example, assume a data source cache entry is generated using a query: `SELECT XSA('weblogic'.'Sample Order Lines')."Columns"."Product Category" FROM XSA('weblogic'.'Sample Order Lines').`

- To purge the data source cache entry: `call SAPurgeXSACache('''weblogic''.''Sample Order Lines''', 'Columns');`
- To purge all data source cache entries in the system: `call SAPurgeXSACache('%', '%');`

Manually Clean Up the Cache on the File System

You can remove data source cache entries manually using the following steps:

1. Stop BI Server.
2. Delete the files in the directories specified by the `DESCRIPTOR_STORAGE_PATH` and `STORAGE_DIRECTORY` parameters under the `XSA_CACHE` section in `NQSConfig.INI`.

Manually Cleaning Up the Cache in a Database

You may need to drop all tables under a data source cache schema for the following reasons:

- To clean-up an existing schema to be used by a new BI Server installation, replacing an old installation.
- To start the data source cache in a clean slate during maintenance down-time.

To clean the cache:

1. Ensure that there are no running BI Servers using this schema. If there are such servers running when the tables are removed, these servers are left in an inconsistent state and it results in data source queries that fail due to missing table errors until these servers are rebooted.
2. From an SQL client, log in to the database as the user of the data source cache schema.
3. Run the following PL/SQL:

```
BEGIN
FOR i IN (SELECT table_name FROM user_tables where table_name like 'XC%')
```

```
LOOP
EXECUTE IMMEDIATE('DROP TABLE ' || user || '.' || i.table_name || '
CASCADE CONSTRAINTS PURGE');
END LOOP;
END;
```

Note that data source cache table names are prefixed with XC.

4. Delete the files in the directory specified by the DESCRIPTOR_STORAGE_PATH parameter under the XSA_CACHE section in NQSConfig.INI.

Manage Presentation Services Cache Settings

When users run analyses, Presentation Services can cache the results of those analyses. Presentation Services determines if subsequent analyses can use cached results. If the cache can be shared, then subsequent analyses aren't stored.

The files for the Presentation Services cache are created by the ODBC driver but generally do correspond to ODBC requests that the Presentation Services cache keeps open. The files are stored in the following directory:

BI_DOMAIN/servers/obips/cache/*subFolderForEachCache*

The files for the cache are removed whenever Presentation Services shuts down cleanly. If Presentation Services shuts down unexpectedly, then various cache files might be left on disk. You can delete the files when Presentation Services isn't running.

The Presentation Services cache is different from the cache that's accessed by the Oracle BI Server. You can change the defaults for the Presentation Services cache by modifying the instanceconfig.xml file to include the cache entries.

The following procedure provides information about configuration changes with which you can manage the Presentation Services cache.

See About Sharing the Presentation Services Query Cache.

1. Open the instanceconfig.xml file for editing in:

   BI_DOMAIN/config/fmwconfig/biconfig/OBIPS

2. Locate the section in which you must add the elements described in the table below.

   > **Note:**
   >
   > Avoid specifying values of less than 3 minutes for the elements that affect minutes. At such a low amount of time, refreshes can occur frequently, which can negatively affect performance and cause flickering on the screen.

3. Include the elements and their ancestor elements as appropriate, as shown in the following example:

```
<ServerInstance>
  <Cache>
    <Query>
      <MaxEntries>1000</MaxEntries>
      <MaxExpireMinutes>60</MaxExpireMinutes>
```

```
            <MinExpireMinutes>10</MinExpireMinutes>
            <MinUserExpireMinutes>10</MinUserExpireMinutes>
            <RefreshIncludeBIServerCache>false</RefreshIncludeBIServerCache>
          </Query>
        </Cache>
      </ServerInstance>
```

4. Save your changes and close the file.
5. Restart Oracle Analytics Server.

About Elements to Configure the Cache

You can configure the cache elements described here.

Element	Description	Default Value
MaxEntries	Specifies the maximum number of open record sets that Presentation Services keeps open at any one time. The minimum value is 3. For systems under significant loads, you can increase this value to 700 or 1000.	1000
MaxExpireMinutes	Specifies the maximum amount of time, in minutes, that an entry in the cache can exist before it's removed. Depending on the number of analyses being run, an entry might be removed before the time limit expires.	60
MinExpireMinutes	Specifies the minimum amount of time, in minutes, that an entry in the cache can exist before it's removed. The setting for CacheMinUserExpireMinutes can force an entry for a particular user to exist for a longer time than that specified by the CacheMaxExpireMinutes element.	10
MinUserExpireMinutes	Specifies the minimum amount of time, in minutes, that an entry in the cache can exist after it's been viewed by a user. For example, if CacheMaxExpireMinutes is set to 60 minutes and a user views the entry during the 59th minute, the entry exists for that user for an additional 10 minutes. The user can continue paging through the data without requiring a new analysis to be run.	10
RefreshIncludeBIServerCache	Specifies if the Presentation Services cache is bypassed when a dashboard or analysis is refreshed. When you refresh an analysis or dashboard the cache is bypassed and the latest version of the data is used. Previously when you refreshed, the cache would be used. Set this parameter to *false* to revert to the previous behavior.	true

Improve Web Client Performance

You can improve the performance of the web client by configuring the web server to serve up all static files, as well as enabling compression for both static and dynamic resources.

By enabling caching and content expiration on the web server, web browsers can determine how often to reload the static files from the server.

Follow the instructions for the web server to set up static file caching and compression for the files located in this directory.

> **Note:**
>
> See the following documents for full information about how to configure Oracle WebLogic Server to work with web servers such as Apache HTTP Server, Microsoft Internet Information Server (Microsoft IIS), and Oracle HTTP Server:
>
> *Using Oracle WebLogic Server Proxy Plug-Ins*
>
> *Administering Oracle HTTP Server*

The following sections provide example configurations:

- Configure Apache HTTP Server for Static File Caching
- Configure Oracle HTTP Server for Static File Cache

Configure Apache HTTP Server for Static File Caching

This example configuration assumes that you have installed the web server plug-in that enables Apache HTTP Server to proxy requests to Oracle WebLogic Server.

Make sure that the `PLUGIN_HOME/lib` directory is added to LD_LIBRARY_PATH, or equivalent for your operating system.

The steps in this section show an example configuration only. You can adjust your configuration as needed. See *Using Oracle WebLogic Server Proxy Plug-Ins* for full information.

To add configuration directives for the plug-in:

1. Locate the `httpd.conf` file for your Apache HTTP Server.
2. Open the file for editing and add directives similar to the following:

```
LoadModule weblogic_module modules/mod_wl.so

<IfModule mod_weblogic.c>
   WebLogicPort 9704
   Debug OFF
   WebLogicHost localhost
   WLLogFile /tmp/wl-proxy.log
</IfModule>

<LocationMatch "/analytics/saw\.dll.*">
SetOutputFilter DEFLATE
SetHandler weblogic-handler
</LocationMatch>

<LocationMatch "/analytics/.*\.jsp.*">
SetOutputFilter DEFLATE
```

```
SetHandler weblogic-handler
</LocationMatch>
```

Note the following:

- Modify the LoadModule directive based on where and how you installed the plug-in.
- The IfModule directive enables the connection to Oracle WebLogic Server. See *Using Oracle WebLogic Server Proxy Plug-Ins* for more information about the connectivity options, including how to configure a cluster and SSL considerations.
- The LocationMatch directives are used to route all dynamic requests to Oracle WebLogic Server. Be sure to include the SetOutputFilter DEFLATE directive, which enables GZip compression for all dynamic requests.

3. Save and close the file.

You can add configuration directives for handling static files.

1. Locate the httpd.conf file for your Apache HTTP Server.
2. Open the file for editing and add directives similar to the following:

```
Alias /analytics ORACLE_HOME/bi/bifoundation/web/appv2
<Directory ORACLE_HOME/bi/bifoundation/web/appv2>
# Disable cross-server ETags
FileETag none
# Enable compression for all files
SetOutputFilter DEFLATE
# Don't compress images
SetEnvIfNoCase Request_URI \.(?:gif|jpe?g|png)$ no-gzip dont-vary
# Enable future expiry of static files
ExpiresActive on
ExpiresDefault "access plus 1 week"
Header set Cache-Control "max-age=604800"
DirectoryIndex default.jsp
</Directory>

# Restrict access to WEB-INF
<Location /analytics/WEB-INF>
Order Allow,Deny
Deny from all
</Location>
```

Note the following:

- You must ensure that Apache HTTP Server has access to the static files for the Oracle BI web client in `*ORACLE_HOME*/bi/bifoundation/web/appv2`. Ensure that the web server is running and has read access to this location.
- The Alias and Directory entries tell Apache HTTP Server to handle requests for static files rather than routing them to Oracle WebLogic Server. Note the following about the directives related to compression and static file expiry:
  - FileETag

    ```
    FileETag none
    ```

This directive tells the web server to disable generation of ETag headers in the response. Default ETag generation for Apache HTTP Server is tied to the file system for a single server, so generating ETags isn't recommended.

- Compression Related Directives

  ```
  SetOutputFilter DEFLATE
  # Don't compress images
  SetEnvIfNoCase Request_URI \.(?:gif|jpe?g|png)$ no-gzip dont-vary
  ```

  These directives ensure that Apache HTTP Server compresses all files except images. Typically, images are already compressed and don't benefit from additional compression.

- Control of Expires Header

  ```
  # Enable future expiry of static files
  ExpiresActive on
  ExpiresDefault "access plus 1 week"
  ```

  This fragment tells Apache HTTP Server to enable setting the Expires header. In this example, the default expiration is set to one week after the first time the file was accessed by the client. You can increase this time period, but ensure that static files are refreshed often enough to handle any patches or updates made on the static files.

- Control of the Cache-Control Header

  ```
  Header set Cache-Control "max-age=604800"
  ```

  This fragment tells Apache HTTP Server to set the Cache-Control header. In this example, the default is set to one week (in seconds) to match the Expires header. This value must always be kept in sync with the Expires header.

- Handling Default URLs

  ```
  DirectoryIndex default.jsp
  ```

  This directive provides a fallback handler when a user requests the /analytics URL without specifying any content under it. This URL is then routed to Oracle WebLogic Server for further processing.

- The final directive restricts access to the WEB-INF folder. This folder is part of the J2EE container's deployment descriptor and mustn't be exposed to web clients.

3. Save and close the file.

Configure Oracle HTTP Server for Static File Cache

Configuration for Oracle HTTP Server is similar to configuration for Apache HTTP Server, except that you don't need to download and install the plug-in because the mod_wl_ohs.so module is installed by default with Oracle HTTP Server.

Some configuration is performed in the mod_wl_ohs.so module directly, and some configuration is performed in httpd.conf. See *Administering Oracle HTTP Server*.

Optimize Performance by Setting the JVM Heap Size

You can improve memory performance by increasing the default JVM heap size for the Administration Server and Managed Servers in the startup script for Oracle WebLogic Server.

Newly deployed web application server instances use default memory heap settings, which are often too small to accommodate Oracle Analytics Server 64-bit requirements. You can use these steps to increase the JVM heap size SERVER_MEM_ARGS parameters using the values recommended here.

Make sure you have enough physical memory allocated available on the servers and then use the settings in the procedure as a starting point, and after load testing, adjust as needed. For a stable even performance over time, you must set the initial JVM heap size (-Xms) to the same value as the maximum JVM heap size (-Xmx). Making the initial value the same size as the maximum value means that the JVM won't spend time trying to work out whether it needs to increase the size of the heap.

This procedure sets the same parameter values for both the Administration Server and Managed Servers.

1. Stop the Administration Server and Managed Servers. See Start and Stop Your System .
2. Set the JVM heap size of the Managed Server:
   a. Browse to `<Oracle_Home>/user_projects/domains/bi/bin` and backup the file setStartupEnv.sh (Linux) | setStartupEnv.cmd (Windows).
   b. Edit setStartupEnv.sh (Linux) | setStartupEnv.cmd (Windows) and update the following in the section `"${STARTUP_GROUP}" = "BISUITE-MAN-SVR`":

```
# 64 bit JVM memory settings

        SERVER_MEM_ARGS_64="-Xms4g -Xmx4g"

        export SERVER_MEM_ARGS_64

        SERVER_MEM_ARGS_64HotSpot="-Xms4g -Xmx4g"

        export SERVER_MEM_ARGS_64HotSpot

        SERVER_MEM_ARGS_64JRockit="-Xms4g -Xmx4g"

        export SERVER_MEM_ARGS_64JRockit
```

   If Oracle Analytics Publisher receives substantial use and the machine has sufficient physical memory, then consider configuring the JVM memory of the Managed Server larger than 4GB.
3. Set the JVM heap size of the Administration Server:
   a. Browse to `<Oracle_Home>/user_projects/domains/bi/bin` and back up the file setStartupEnv.sh (Linux) | setStartupEnv.cmd (Windows).

b. Edit setStartupEnv.sh (Linux) | setStartupEnv.cmd (Windows) and update the following lines highlighted in bold inside the section `"${STARTUP_GROUP}" = "AdminServerStartupGroup"`:

```
# 64 bit JVM memory settings

        SERVER_MEM_ARGS_64="-Xms4g -Xmx4g"

        export SERVER_MEM_ARGS_64

        SERVER_MEM_ARGS_64HotSpot="-Xms4g -Xmx4g"

        export SERVER_MEM_ARGS_64HotSpot

        SERVER_MEM_ARGS_64JRockit="-Xms4g -Xmx4g"

        export SERVER_MEM_ARGS_64JRockit
```

4. After setting the parameters, save and close the file, then restart the Administration Server and Managed Servers for the changes to take effect.
5. In a scaled-out system, repeat these steps for each domain home.

   The settings are now copied over when you horizontally scale-out.

Improve Performance when Downloading Large Reports

Downloading large reports may require you to increase the timeout settings.

If downloading a report takes more than 6 minutes, increase the timeout setting as follows:

1. Open `<ORACLE HOME>/user_projects/domains/bi/config/fmwconfig/biconfig/bridgeconfig.properties`
2. Search for the following parameter: `oracle.bi.presentation.sawconnect.ConnectionPool.SocketReadTimeoutSec`. If it's not present in the file, add it.
3. The default value for this parameter is 360 seconds (6 minutes). Increase the parameter as needed and save the configuration file.

Use F5 BIG-IP for Load Balancing

If you're using F5 BIG-IP for load balancing and a large report takes more than 5 minutes to download, increase the timeout setting as follows:

1. Log into the BIG-IP Configuration Utility.
2. From the main tab of the navigation pane, expand **Local Traffic** and click **Profiles**, then **Protocol**, then **TCP**.
3. In your TCP profile, search for the **Idle Timeout** field. The default value for this parameter is 300 seconds (5 minutes). Increase the parameter as needed.

For information on the timeout setting, refer to the ASKF5 Knowledge Center article K7559: Overview of the TCP profile (9.x-10.x)

Use Oracle HTTP Server for Load Balancing

If you're using Oracle HTTP Server for load balancing and a large report takes more than 5 minutes to download, increase the timeout setting as follows:

1. Open `ORACLE_INSTANCE/config/OHS/component_name/mod_wl_ohs.conf`
2. Search for the `<IfModule weblogic_module>` section of the file, and add a parameter called `WLIOTimeoutSecs` within that section.
3. The default value for this parameter is 300 seconds (5 minutes). Increase the parameter as needed and save the configuration file. Restart Oracle HTTP Server.

See Parameters for Web Server Plug-Ins in *Using Oracle WebLogic Server Proxy Plug-Ins*.

Capture Metrics Using the Dynamic Monitoring Service

In addition to the Metrics Browser in Fusion Middleware Control, you can view metrics using the Dynamic Monitoring Service (DMS) and WLST commands.

This section describes how to use these methods.

Use the Dynamic Monitoring Service for Metrics

You can use the Dynamic Monitoring Service (DMS) to view metrics.

Access the service using the following URL:

`http://<host>:<AdminServer port>/dms`

Using the Metrics Tables list in the left pane, select **Non-J2EE Metrics** to view the list of metrics for Oracle Analytics Server. This is the same list that you see in the Metrics Browser of Fusion Middleware Control.

You can use the Dynamic Monitoring Service to quickly obtain a snapshot of metrics. You can use the URL for a given metric as the source for a web query in a Microsoft Excel spreadsheet that you combine with a macro that automatically copies values to an archive sheet and refreshes the query on a loop for a given period.

Suppose that you want to use the Dynamic Monitoring Service to see the details of the metric table called Oracle_BI_General. When you click the **Oracle_BI_General** link in the Metrics Tables list, the table is displayed on the right side. This metric table consists of several monitored values, such as Active_Execute_Requests and Total_Sessions. You can use the information from the tables that are displayed in this Metrics Browser as part of WLST commands.

For information on accessing DMS metrics using WLST commands, see Use WLST Commands for Metrics..

Use WLST Commands for Metrics

Using WLST, you can collect metrics about the system.

You can use WLST commands to capture metrics.

1. Navigate to the `ORACLE_HOME/oracle_common/common/bin` directory.
2. Run the WLST utility.

3. Connect to the system using the connect command, as shown in the following example:

   ```
   connect('user','password','t3://<host><port>)
   ```

4. View the following prompt:

   ```
   wls:/bi/serverConfig>
   ```

You can now interactively use the DMS custom WLST commands. For example, to list all the metric tables that start with "Oracle_BI", enter the following command:

```
wls:/bifoundation_domain/serverConfig> displayMetricTables('Oracle_BI*')
```

This command generates a long list of data for each of the Oracle BI metric tables. So it is more useful to focus on a given metric table, such as "Oracle_BI_General". The following command displays output such as that shown in this sample.

```
wls:/bifoundation_domain/serverConfig> displayMetricTables('Oracle_BI_General')


-----------------
Oracle_BI_General
-----------------
Active_Execute_Requests.value:  0
Active_Fetch_Requests.value:    0
Active_File_Handles.value:      1
Active_Initblock_Executions.value:      0
Active_Logins.value:    0
Active_Prepare_Requests.value:  0
Avg._Failed_Logins_Elapsed_Time.value:  0
Avg._Initblock_Executions_Elapsed_Time.value:   0
Avg._Succeeded_Logins_Elapsed_Time.value:       0
Avg._query_elapsed_time.value:  0
Busy_File_Handles.value:        0
File_Handle_Waiters.value:      0
Free_File_Handles.value:        502
Host:   oracle-bc5ac6af
Max._Initblock_Execution_Elapsed_Time.value:    0
Max_File_Handles.value: 503
Name:   Oracle BI General
New_Execute_Requests.value:     19
New_Fetch_Requests.value:       32
New_Initblock_Executions.value: 0
New_Logins.value:       7
New_Prepare_Requests.value:     19
New_Requests.value:     187
OBPERF_***.value:       7
Oracle_BI_Applications: Oracle BI Server
Parent: /Oracle BI Server
Process:        Oracle BI Server:4004:/instance1/coreapplication_obis1
Queries/sec.value:      0
ServerName:     /instance1/coreapplication_obis1
Succeeded_Initblock_Execution_Ratio_as_%.value: 0
Succeeded_Logins_Ratio_as_%.value:      7
Total_sessions.value:   0
```

Using the scripting capability of WLST, you can embed DMS commands into a Python script to store the required metric values in a file. The following is an example of such a script.

```
# Script to dump timestamp (in milliseconds) for a single Oracle BI metric
# to a file
#
from java.util import Date
from java.text import SimpleDateFormat
```

```
#
# Modify to connect to your server
connect('biadmin','password','t3://localhost:9500')
#
# This is the number of times to sample the metric
sample_length = 100
#
# This is where you define what metric table and metric to dump to file
metric_table = "Oracle_BI_General"
metric_of_interest = "Avg._query_elapsed_time.value"
#
# Some metrics have non-text characters in the name. Provide a reference here
# so it dumps to file without error in file name
output_file_metric_ref = "Avg_Qry_Elapse"
#
# This section defines the output file name with unique time
start_time = str(SimpleDateFormat("dd-MMM-yyyy_HH-mm-ss").format(Date()))
output_filename = start_time + "_" + output_file_metric_ref + "_dump.txt"
#
# Open the file and write summary of metric to be dumped
file = open(output_filename,'w')
print >>file, "Start Metric Dump of: " + str(metric_table) + " : " +
str(metric_of_interest) + " at " + str(SimpleDateFormat("dd-MMM-yyyy HH-mm-
ss").format(Date()))
#
#
# The following section forms a loop according to the sample length defined
# earlier. The 'displayMetricTables()' command returns the metric table in the
# form of a JMX composite data array. The code following this command access
# the metric data from this array. In this case, a particular metric of
# interest is tested for and only the value of that metric is output to file.
#
counter = 0
while counter <= sample_length:
    results = displayMetricTables(metric_table)
    for table in results:
        name = table.get('Table')
        rows = table.get('Rows')
    rowCollection = rows.values()
    iter = rowCollection.iterator()
    while iter.hasNext():
        row = iter.next()
    rowType = row.getCompositeType()
    keys = rowType.keySet()
    keyIter = keys.iterator()
    while keyIter.hasNext():
        columnName = keyIter.next()
        value = row.get(columnName)
        if columnName == metric_of_interest:
            print >>file, str(SimpleDateFormat("dd-MMM-yyyy HH-mm-ss-
SSS").format(Date())) + "," + str(value)
    counter = counter + 1
file.close()
disconnect()
```

Certain Oracle BI metric tables, such as "Oracle_BI_Thread_Pool", are in effect two-dimensional. With the "Oracle_BI_Thread_Pool" table, you can query the metric values for various "Names", such as "Server" or "Usage_Tracking". To export the required metric value to a file in this case, you must modify the logic that was used in looping in the previous example script to handle the two dimensions. The following example script provides one way to handle this case.

```
# Script to dump timestamp (in milliseconds) and a
#single Oracle BI metric to a file for metrics with multiple sections
#
from java.util import Date
from java.text import SimpleDateFormat
#
# Modify to connect to your server
connect('biadmin','password','t3://localhost:9500')
#
# This is the number of times to sample the metric
sample_length = 100
#
# This is where you define what metric table, category, and metric to
# dump to file
metric_table = "Oracle_BI_Thread_Pool"
category_of_interest = "Server"
metric_of_interest = "Avg._Request/sec.value"
#
# Some metrics have non-text characters - provide a reference here
# so it dumps to file without error
output_file_metric_ref = "Avg_Req_Sec"
#
# This section defines the output file name with unique time
start_time = str(SimpleDateFormat("dd-MMM-yyyy_HH-mm-ss").format(Date()))
output_filename = start_time + "_" + output_file_metric_ref + "_dump.txt"
#
# Open the file and write summary of metric to be dumped
file = open(output_filename,'w')
print >>file, "Start Metric Dump of: " + str(metric_table) + " : " +
str(metric_of_interest) + " for Category: " + str(category_of_interest) + " at " +
str(SimpleDateFormat("dd-MMM-yyyy HH-mm-ss").format(Date()))
#
#
counter = 0
while counter <= sample_length:
    results = displayMetricTables(metric_table)
    for table in results:
        name = table.get('Table')
        rows = table.get('Rows')
    rowCollection = rows.values()
    iter = rowCollection.iterator()
    while iter.hasNext():
        row = iter.next()
        if row.containsValue(category_of_interest):
            rowType = row.getCompositeType()
            keys = rowType.keySet()
            keyIter = keys.iterator()
            while keyIter.hasNext():
                columnName = keyIter.next()
                value = row.get(columnName)
                if columnName == metric_of_interest:
                    print >>file, str(SimpleDateFormat("dd-MMM-yyyy HH-mm-ss-
SSS").format(Date())) + "," + str(value)
    counter = counter + 1
file.close()
disconnect()
```

Part IV
Resolve Issues

Learn how to resolve issues in Oracle Analytics Server.

This part includes the following chapters:

- Diagnose and Resolve Issues
- Track Usage

8 Diagnose and Resolve Issues

This chapter describes how to diagnose and resolve issues using tools such as Fusion Middleware Control and log files.

Topics:

What Diagnostic Tools Are Available?

There are several diagnostic tools available to help you troubleshoot issues.

Various diagnostic tools are available to help you find the causes and solutions to issues, as described in the table.

Tool	Description	Reference
Diagnostic bundle	Enables collection of log and configuration information attachment for Oracle Support requests	Collect Diagnostic Bundles
Overview page in Fusion Middleware Control	Enables you to view recent issues with the system.	Display Oracle Analytics Server Pages in Fusion Middleware Control
Performance metrics	Enables you to view metrics that affect performance.	Monitor Service Levels
Diagnostic pages in Fusion Middleware Control	Enables you to drill into problems and view and configure log files.	Use Fusion Middleware Control to View Log Information, Error Messages, and Alerts Use Fusion Middleware Control to Configure Log File Rotation Policy and Specify Log Levels
Usage tracking	Enables you to generate usage tracking statistics that can be used in a variety of ways such as database optimization, aggregation strategies, or billing users or departments based on the resources that they consume.	Track Usage

Tool	Description	Reference
Consistency Check Manager	Enables you to check the validity of the semantic model.	Checking the Consistency of a Repository or a Business Model in *Managing Metadata Repositories for Oracle Analytics Server*
Model Check Manager	Enables you to check for modeling problems that might affect Summary Advisor and the aggregate persistence engine.	"Using Model Check Manager to Check for Modeling Problems" in *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*
ODBC/JDBC procedures	Enables you to obtain diagnostic information for the BI Server.	Use ODBC/JDBC Procedures to Obtain BI Server Diagnostics
Workbook performance information	Use the performance tools in Developer Options to review performance statistics such as query time, server time, and streaming time for visualization components in a workbook.	Developer Options in *Visualizing Data in Oracle Analytics Server*

Collect Diagnostic Bundles

This section explains how to collect diagnostic bundles needed by Oracle Support to help resolve issues.

If you encounter a problem, record any error messages in the user interface, bundle all your log files in a ZIP file, and then contact Oracle Support for assistance.

Assumptions:

- You must review the product FAQ.
- Installation and configuration was successful, specifically the Configuration Assistant has completed without error. If this isn't the case please see below for details on how to collect diagnostics prior to contacting Oracle Support.
- No security sensitive information is collected.

Pre-requisites:

You must have file system permissions.

1. Collect a diagnostic bundle by running the following command:

   ```
   DOMAINHOME/bitools/bin/diagnostic_dump.sh <zip file name>.
   ```

2. Provide the zip file to the Support or Development organization when requested.

View and Configure Diagnostic Log Files

Diagnostic log files can help you troubleshoot issues before and after they occur.

You can view diagnostic log files and configure settings that affect diagnostic log files and the information that they contain, as described in the following sections:

- Use Fusion Middleware Control to View Log Information, Error Messages, and Alerts
- Configure Log File Rotation Policy and Specify Log Levels

Use Fusion Middleware Control to View Log Information, Error Messages, and Alerts

You can search for and view the log entries for components using Fusion Middleware Control Log Viewer.

You can search the log files for log messages, and you can apply filters that can, for example, target a certain date range, user, user transaction, or level of message (error, warning, notification, and so on). You can also view log files in their entirety from the Fusion Middleware Control Log Viewer.

When log entries for error messages and warnings are generated across multiple log files, they can be difficult to trace. However, it's possible to view logging information for specific user transactions across multiple log files. Transaction level logging associates a unique transaction ID, which is called the Execution Context ID (ECID), with every log and error message that is generated in response to a user request. This logging enables rapid diagnosis of the cause of underlying issues. However, some messages in the log (for example system messages for server startup or shutdown) don't have a transactional attribute. All log messages that are related to client requests do have a transactional attribute.

Before you begin this procedure, ensure that you are familiar with the information in Use Fusion Middleware Control.

1. Go to the Overview page.
2. Display the Log Messages tab of the Diagnostics page.
3. View lists of the following:
   - Recent errors under the **Most Recent Errors** region
   - Recent warnings under the **Most Recent Warnings** region
4. Select a link under View Log Messages to display messages for all log files, or for the messages for the log files of a specified component:
   - **Search the log files using the Log Viewer**
   - **Presentation Services Log**
   - **Server Log**
   - **Scheduler Log**
   - **JavaHost Log**
   - **Cluster Controller Log**
   - **Action Services Log**
   - **Security Services Log**
   - **Administrator Services Log**

   Fusion Middleware Control displays messages in the Log Messages page that correspond to your selection.
5. Enter appropriate search criteria to display corresponding error messages.

   To view messages by ECID, click **View Related Messages** and select the **by ECID (Execution Context ID)** menu option.
6. Select one or more rows to view the log file entry details for the selected messages.

Configure Log File Rotation Policy and Specify Log Levels

You can configure criteria that determine when a new log file must be created, based on the size of the log file and the age of the log file.

You can also specify log levels to determine what level of message the log files contain.

This section contains the following topics:

- Use Fusion Middleware Control to Configure Log File Rotation Policy and Specify Log Levels
- Manually Change Additional Log File Settings

Use Fusion Middleware Control to Configure Log File Rotation Policy and Specify Log Levels

Configuring log file rotation policies and log levels ensures that the log files remain manageable while retaining sufficient data.

Before you begin this procedure, ensure that you're familiar with the information in Use Fusion Middleware Control.

1. Go to the Business Intelligence Overview page.
2. Display the Log Configuration tab of the Diagnostics page.
3. Click **Lock and Edit** to enable you to make changes.
4. Complete the elements using the descriptions in the help topic for the page. You can specify which log levels to use, and for some you can set their granularity.

   Log Configuration

   - **Maximum File Size** option
   - **Maximum Log Age** option

   Query Logs

   - **Maximum File Size** option
   - **Maximum Log Age** option

   Default Log Level

   Component Specific Log Levels
5. Click **Apply**, then click **Activate Changes**.
6. Return to the Business Intelligence Overview page and click **Restart**.

Manually Change Additional Log File Settings

In addition to the log file settings that you can change in Fusion Middleware Control, you can change other settings manually. Use various elements in the log configuration file for a component to change these settings.

1. Open the component log configuration file located in:

   ```
   BI_DOMAIN/config/fmwconfig/biconfig/
   ```

   See What Are Diagnostic Log Configuration Files and Where Are They Located?

2. Locate the section in which you must add the Format element, which specifies the log file format. The default is ODL-TEXT.

   To use the Fusion Middleware Control Log Viewer to view and search through the log files for Oracle Analytics Server, then the files must be in either ODL-Text or ODL-XML format.

3. Include the element and its ancestor elements as appropriate, as shown in the following example:

   ```
   <server>
      <ServerInstance>
         <Log>
            <Format>ODL-TEXT</Format>
         </Log>
      </ServerInstance>
   </server>
   ```

   For an example of a JavaHost Server diagnostic log configuration file, see What Are Diagnostic Log Configuration Files and Where Are They Located? .

4. Save your changes and close the file.

5. Restart Oracle Analytics Server.

Diagnose Issues Using the Log Viewer

You can use the Log Viewer in Fusion Middleware Control to find messages that can assist you in resolving issues with the system.

1. Display Fusion Middleware Control.

2. In the Navigator, select WebLogic Domain, right-click bi, and select **Logs**, then **View Log Messages**.

   The Log Messages page is displayed. The Log Viewer collects lines from all log files and displays them on this page. You can filter the lines to view the ones in which you're interested.

3. To start filtering the list, enter search criteria to locate the messages in which you're interested:

   - If you know that an error occurred around a certain date, then set the **Date Range** to **Time Interval**. Select the start and end dates for filtering.
   - If the error happens continually, then set the **Date Range** to **Most Recent**. Select **Days** and specify a number such as 1 or 3.
   - For Message Types, select the following: Incident Error, Error, Warning, and Notification. If the number of messages that's returned is too large, then deselect Notification to see only errors and warnings.

     The advantage of selecting Notification is that you can see what the Oracle Analytics Server system was doing, which can assist you in determining where something went wrong.

4. To filter for the messages:

   a. Click **Add Fields**, then select **Module**, and click **Add**.

   b. Ensure that **Module** is set to **contains**, then enter the following value:

      ```
      oracle.bi.management
      ```

That value specifies the name of the Java package from which all log entries for systems management originate.

5. Click **Search**.

   The page lists all log messages that meet the criteria, including the errors and warnings that lead up to the problem that you're diagnosing.

6. To save a copy of the log messages, click **Export Messages to File**, then **As Oracle Diagnostic Log Text (.txt)** or other format appropriate to your needs.

As you view the log messages, you can see that the Message column explains what operations happened at what times. You can learn important information such as when servers were restarted or a configuration change occurred. You can use the values in the **Log File** column to learn which files were written to, which gives a clue as to what the system was doing. For example, a value of obis1-diagnostic.log indicates an interaction with the Oracle BI Server and a value of sawlog5.log indicates an interaction with Presentation Services.

You can view the log messages to see what might have contributed to a particular situation. For example, suppose that you make changes in Fusion Middleware Control to specify a different semantic model, but you can't see the semantic model in Presentation Services. When you view the log messages, you find an error message that indicates that the computer that hosts the Managed Server and to which the new semantic model was copied has run out of memory. An earlier error message indicates that the Administration Server had reported the change to the semantic model and had tried to synchronize the change to the Managed Server.

Understand Diagnostic Log Files and Log Configuration Files

Diagnostic log files and log configuration files provide a means for troubleshooting and researching system functions.

This section discusses diagnostic log files and diagnostic log configuration files, and contains the following topics:

- What Are Diagnostic Log Files and Where Are They Located?
- What Are Diagnostic Log Configuration Files and Where Are They Located?
- What Are Log File Message Categories and Levels?
- What Is Log File Rotation?
- What Messages Are Included in the System Log?

What Are Diagnostic Log Files and Where Are They Located?

Diagnostic log files store message information that's generated by the Oracle Analytics servers.

These log files are stored in the following location:

`*BI_DOMAIN*/servers/*INSTANCE_KEY*/logs` (for system components)

`*BI_DOMAIN*/servers/*WLS_SERVER_NAME*/logs` (for JEE components).

For example:

`oraclehome/user_projects/domains/bi/servers/obis1/logs` (for BI Server)

`oraclehome/user_projects/domains/bi/servers/AdminServer/logs` (for JEE Administration server)

The following diagnostic log files are used:

- Presentation Services
  - sawn.log — The Presentation Services log file that represents the latest part of diagnostic log output and is searchable in the Fusion Middleware Control Log Viewer.

  For more information specifically about Presentation Services logging, see Log Information in Presentation Services.
- Oracle BI Server
  - obis<*n*>_query.log — The Oracle BI Server query log, which isn't searchable in the Fusion Middleware Control Log Viewer.

    For example, where <*n*> = -1, -2.

    Note: The date and timestamp is in the log file.
  - obis1-diagnostic<n>.log — The Oracle BI Server main diagnostic log, which is searchable in the Fusion Middleware Control Log Viewer.

    For example, where <*n*> = 1, 2

    Note: The date and timestamp is in the log file.
  - nqsadmintool.log — The log for Model Administration Tool.
  - Oracle BI Server utilities — For example, biserverxmlexec and equalizerpds, also generate their own logs when they run.
- JavaHost
  - jh.log — The JavaHost Server main diagnostic log, which is searchable in the Fusion Middleware Control Log Viewer.

    Note: The date and timestamp is in the log file.
- Oracle BI Scheduler
  - nqscheduler.log — The Oracle BI Scheduler log file, which is searchable in the Fusion Middleware Control Log Viewer.

    Note: The date and timestamp is in the log file.
- Cluster Controller
  - nqcluster.log — The Cluster Controller diagnostic file, which is searchable in the Fusion Middleware Control Log Viewer.

    Note: The date and timestamp is in the log file.
- BI JEE log (Action Services and Security Services), both of the following log files are searchable in the Fusion Middleware Control Log Viewer:
  - AdminServer-diagnostic.log
  - bi_server1-diagnostic.log

> **Note:**
>
> For the following log files, you can't set the time zone in which messages are logged in the files: nqcluster.log, nqscheduler.log, and obis1-diagnostic<n>.log. The messages are logged in the files in Greenwich Mean Time (GMT). When you view the messages in the Fusion Middleware Control Log Viewer, you see the messages in the local time zone.

What Are Diagnostic Log Configuration Files and Where Are They Located?

Diagnostic log configuration files control output to diagnostic log files for Oracle Analytics Server.

> **Note:**
>
> Editing a diagnostic log configuration file for a single component isn't advised, because changes might subsequently be overwritten.

Log configuration files are stored in the following locations:

BI_DOMAIN/config/fmwconfig/biconfig/*BI_COMPONENT_NAME*

For example:

oraclehome/user_projects/domains/bi/config/fmwconfig/biconfig

- ./OBICCS/ccslogconfig.xml
- ./OBIJH/logging-config.xml
- ./OBIPS/instanceconfig.xml
- ./OBSCH/schedulerconfig.xml
- ./OBIS/logconfig.xml

About Formats in Diagnostic Log Configuration Files

Diagnostic log configuration files conform to the Oracle Diagnostic Log (ODL) standard, although they can differ slightly in appearance.

Example of BI Server Diagnostic Log Configuration File Format

```
<server>
   <ServerInstance>
      <Log>
         <MaximumFileSizeKb>10000</MaximumFileSizeKb>
         <MaximumLogAgeDay>60</MaximumLogAgeDay>
         <Format>ODL-TEXT</Format>
            <Level>
               <IncidentError>1</IncidentError>
               <Error>1</Error>
               <Warning>16</Warning>
               <Notification>1</Notification>
               <Trace>16</Trace>
            </Level>
      </Log>
      <UserLog>
         <MaximumFileSizeKb>10000</MaximumFileSizeKb>
         <MaximumLogAgeDay>10</MaximumLogAgeDay>
         <Format>ODL-TEXT</Format>
      </UserLog>
   </ServerInstance>
</server>
```

Example of JavaHost Server Diagnostic Log Configuration File Format

```
<?xml version = '1.0' encoding = 'utf-8'?>
<logging_configuration>
   <log_handlers>
      <log_handler name="odl-handler" class="oracle.core.ojdl.logging.ODLHandlerFactory">
         <property name="path" value="${oracle.component.logpath}/jh.log"/>
         <property name="maxFileSize" value="10485760"/>
         <property name="maxLogSize"  value="50000000"/>
      </log_handler>
   </log_handlers>

   <loggers>
      <logger name="saw" level="NOTIFICATION:1" useParentHandlers="false">
         <handler name="odl-handler"/>
      </logger>
      <logger name="oracle.bi" level="NOTIFICATION:1" useParentHandlers="false">
         <handler name="odl-handler"/>
      </logger>
   </loggers>
</logging_configuration>
```

Oracle Analytics Server components control their diagnostic log files by using server-specific settings in their log configuration files, for example:

- Presentation Services log configuration file:

  - `writerClassId` settings configure messages that the system writes to the sawjson.log file.

- Oracle BI Server log configuration file:

  - `Log` settings configure messages that the system writes to the obis1-diagnostic.log file. See What Messages Are Included in the System Log?

  - `UserLog` settings configure messages that the system writes to the nqquery.log file. See Manage the Query Log.

- Oracle BI Scheduler log configuration file:

  - `Log` settings configure messages that the system writes to the nqscheduler.log file.

- JavaHost Server log configuration file:

  - `log_handlers` elements and subelements enable configuration of the log file rotation policy and the specification of the log file name and its location.

  - `loggers` elements and subelements enable appropriate handling of Java component (JavaHost Server) log levels, by mapping the JavaHost Server log levels to the standard Oracle Diagnostic Log (ODL) log levels.

What Are Log File Message Categories and Levels?

Categories and levels for log file messages define the detail and level of importance with which the system writes messages to a log file.

Fusion Middleware Control enables you to control these settings in the logconfig.xml file.

Each message category in a log file is set to a specific default value between 1 and 32, and only messages with a level less than or equal to the log level is logged.

Log file message categories are described in the table.

Category:Level	Description
IncidentError:1	A serious problem caused by unknown reasons has occurred. You can fix the problem only by contacting Oracle Support Services. No performance impact.
Error:1	A problem that requires attention from the system administrator has occurred. No performance impact.
Warning:1	An action occurred or a condition was discovered that must be reviewed and might require action before an error occurs. No performance impact.
Notification:1	A report of a normal action or event has occurred. This could be a user operation, such as "login completed" or an automatic operation such as a log file rotation. No performance impact.
Notification:16	A configuration-related message or problem has occurred. Low performance impact. You can enable this level broadly in a production environment without having a significant performance impact in the software.
Trace:1	A trace or debug message that is used for debugging or performance monitoring has been written. Typically this message contains detailed event data that is clear enough to be understood by someone who does not know internal implementation details. Small performance impact. This level might be enabled broadly occasionally on a production environment to debug issues with the software. Enabling logging at this level might have a small performance impact, but not to the point of making the software unusable.
Trace:16	A fairly detailed trace or debug message has been written. The message is clear enough to be understood by Oracle Support Services engineers who have a deep knowledge of the product but might not know full details of the internal implementation. High performance impact. This level mustn't be enabled on a production environment, except on special situations to debug issues with the software.
Trace:32	A highly detailed trace or debug message has been written. The message is intended for an Oracle developer working on the software who knows enough details about the implementation of the subsystem that generates the message. Very high performance impact. This level isn't expected to be enabled in a production environment and developers use it only to debug the software on a test or development environment.

In the following log configuration file example, in the Notification message category, only level 1 messages are logged. If the log level is set to 0, then nothing is logged for that message category.

```
<Level>
   <IncidentError>1</IncidentError>
   <Error>1</Error>
   <Warning>1</Warning>
   <Notification>1</Notification>
   <Trace>1</Trace>
 </Level>
```

Avoid manually changing the default settings in the log file. Use Fusion Middleware Control to make changes. See Use Fusion Middleware Control to Configure Log File Rotation Policy and Specify Log Levels.

What Is Log File Rotation?

Log file rotation is the creation of new log files, when the file exceeds a specified threshold or date.

Take the MaximumFileSizeKb setting for the component log configuration file for the Oracle BI Scheduler as an example. Whenever a log file exceeds the size that's specified by this setting, then the existing Scheduler log file is renamed, and a new log file is created. Additionally, a log file date that's older than the MaximumLogAgeDay setting is deleted.

Different components have different log file names and different settings within their log configuration files. For example, the file naming convention for the Scheduler is as follows:

- nqscheduler.log — The latest log file.
- nqscheduler-<*n*>.log — The renamed previous log file.

  where <*n*> = date and timestamp, for example nqscheduler-20100909-2135.log

See Use Fusion Middleware Control to Configure Log File Rotation Policy and Specify Log Levels.

What Messages Are Included in the System Log?

The Oracle BI Server writes messages to the obis1-diagnostic.log file, based on configuration settings.

In addition to writing messages to this log file, the Oracle BI Server writes certain severe messages to the system log file for Linux systems. The following list includes the kinds of messages that the Oracle BI Server writes to the system log file:

- When the Oracle BI Server can't start (for example, because another server has previously started), then the system log file includes a message such as the following one:

  ```
  Another server is already running on : @1%ls and port: @2%ls.
  ```

- When memory problems occur, the system log file includes a message such as the following one:

  ```
  Could not enable the Low-Fragmentation Heap.
  ```

- When the hard disk on the computer is full, the system log file includes a message such as the following one:

  ```
  Out of disk space.
  ```

Manage the Query Log

The Oracle BI Server can log query activity at the individual user level. Use logging for quality assurance testing, debugging, and troubleshooting by Oracle Support Services. In production mode, query logging is typically disabled.

The query log file is named nqquery.log, and is located in:

```
BI_DOMAIN/servers/obisn/logs
```

Oracle BI Server query logging is tracked at a user level. Tracking queries for the entire user community is a resource-intensive process.

> **Note:**
>
> For production systems, it's recommended that query logging be enabled only for a very targeted user community. In production systems, you can use usage tracking as the production-level logging facility. See Track Usage.

It's recommended that you test users only when the user name clearly indicates it's a test user and have verified that query logging is enabled. If logging is enabled for such users, then it's recommended that they be given names such as sales_admin_with_logging, sales_dev_with_logging, or sales_test_with_logging, so that you can readily identify them. Even production administrator logins shouldn't have query logging enabled, because it strains the available resources.

Also disable query logging for the following:

- The SQL statement in the initialization string. The **Initialization string** field is in the Initialization Block dialog, in the General tab.

  The LOGGING column references stored values for the log level.
- Set the logging level to 0 (zero) for each production user. The **Logging level** field is in the User dialog, in the User tab. In the Model Administration Tool, select **Identity** from the Manage option on the main toolbar. In the Identity Manager dialog, double-click a user and select the User tab.

This section contains the following topics:

- Configure Query Logging
- Use the Log Viewer

Configure Query Logging

This section includes information about setting the size of the query log, choosing a logging level, and enabling query logging for a user.

Because query logging can produce very large log files, the logging system is turned off by default. You can enable logging to test that the semantic model is configured properly, to monitor activity on the system, to help solve performance problems, or to assist Oracle Support Services. You must enable query logging on the system for each user whose queries you want logged. You do this using the Model Administration Tool.

Set the Query Logging Level

You can configure the amount of data query logs collect per user.

You can enable query logging levels for individual users, as described in Set the Query Log Level for a User. You can't configure a logging level for a group.

A session variable overrides the logging level for a particular user. For example, if the administrator has a logging level of 4 and the session variable logging level is defined as the default 0 (zero) in the semantic model, then the logging level for the administrator is 0.

Set the logging level based on the amount of logging that's appropriate for your organization. In normal operations, logging is generally disabled (that is, the logging level is set to 0). If you decide to enable logging, then select a logging level of 1 or 2. These two levels are designed for use by administrators.

You might want to diagnose performance or data issues by setting a temporary log level for a query. You can enable query logging for a select statement by adding a prefix clause in the Advanced SQL Clauses section of the Advanced tab in Oracle BI Presentation Services. For example, for the select statement:

```
SELECT year, product, sum(revenue) FROM time, products, facts;
```

You can specify the logging level of 5 in the **Prefix** field as follows:

```
Set Variable LOGLEVEL=5;
```

For this query, the logging level of 5 is used regardless of the value of the underlying `LOGLEVEL` variable.

> **Note:**
>
> Use logging levels greater than 2 only with the assistance of Oracle Support Services.

The query logging levels are described in the following table.

Logging Level	Information That Is Logged
Level 0	No logging.
Level 1	Logs the SQL statement issued from the client application. Also logs the following: • Physical query response time — The time for a query to be processed in the back-end database. • Number of physical queries — The number of queries that are processed by the back-end database. • Cumulative time — The sum of time for all physical queries for a request (that is, the sum of all back-end database processing times and DB-connect times). • DB-Connect time — The time taken to connect to the back-end database. • Query cache processing — The time taken to process the logical query from the cache. • Elapsed time — The time that has elapsed from when the logical query is presented to the Presentation Services until the result is returned to the user. Elapsed time can never be less than response time, because elapsed time takes into account the small extra time between the logical query being presented to the Presentation Services to the start of preparation of the query. In cases where this difference in time is negligible, the elapsed time equals the response time. • Response time — The time taken for the logical query to prepare, run, and fetch the last record. This matches the TOTAL_TIME_SEC that is logged in usage tracking, as described in Understand the Usage Tracking Tables. • Compilation time — The time taken to compile the logical query. • For each query, logs the query status (success, failure, termination, or timeout), and the user ID, session ID, and request ID. • Total Time in BI Server — the time spent in the BI Server for query execution only (that is, not compilation time).
Level 2	Logs everything logged in Level 1. Additionally, for each query, logs the semantic model name, business model name, subject area name, SQL statement issued against the physical database, queries issued against the cache, number of rows returned from each query against a physical database and from queries issued against the cache, and the number of rows returned to the client application.

Logging Level	Information That Is Logged
Level 3	Logs everything logged in Level 2. Additionally, adds a log entry for the logical query plan, when a query that was supposed to seed the cache was not inserted into the cache, when existing cache entries are purged to make room for the current query, and when the attempt to update the exact match hit detector fails. Don't select this level without the assistance of Oracle Support Services.
Level 4	Logs everything logged in Level 3. Additionally, logs the query execution plan. Don't select this level without the assistance of Oracle Support Services.
Level 5	Logs everything logged in Level 4. Additionally, logs intermediate row counts at various points in the execution plan. Don't select this level without the assistance of Oracle Support Services.
Level 6 and 7	Not used.

Set the Query Log Level for a User

You can configure the amount of query data to log per user.

1. In the Model Administration Tool, select **Manage**, then **Identity**.

   The Identity Manager dialog is displayed.

2. Double-click the name of the user for which you want to set the query logging level.

   The User dialog is displayed.

3. Set the logging level by clicking the **Up** or **Down** arrows next to the **Logging Level** field.

   To disable query logging for a user, set the logging level to 0.

4. Click **OK**.

Use the Log Viewer

Use the Log Viewer utility (or a text editor) to view the query log.

Each entry in the query log is tagged with the name of the user who issued the query, the session ID of the session in which the query was initiated, and the request ID of the individual query.

Run the Log Viewer Utility

The log viewer utility allows you to search for and review specific log files.

To run the Log Viewer utility (located on Linux in `ORACLE_HOME/bi/bifoundation/server/bin`), open a command prompt, and enter `nqlogviewer` with any combination of its arguments. The syntax is as follows:

```
nqlogviewer [-u user_name] [-f log_input_filename]
          [-o output_result_filename]
          [-s session_ID] [-r request_ID]
```

In this syntax:

- *`user_name`* is the name of a user in the semantic model. This parameter limits the scope to entries for a particular user. If not specified, all users for whom query logging is enabled are displayed.
- *`log_input_filename`* is the name of an existing log file from where the content is taken. This parameter is required.
- *`output_result_filename`* is the name of a file in which to store the output of the log. If the file exists, then the results are appended to the file. If the file doesn't exist, then a new file is created. If this argument isn't specified, then output is sent to the monitor screen.
- *`session_ID`* is the session ID of the user session. The BI Server assigns each session a unique ID when the session is initiated. This parameter limits the scope of the log entries to the specified session ID. If not specified, then all session IDs are displayed.
- *`request_ID`* is the request ID of an individual query. The BI Server assigns each query a unique ID when the query is initiated. This parameter limits the scope of the log entries to the specified request ID. If not specified, then all request IDs are displayed.

  The request ID is unique among the active requests, but not necessarily unique during the session. Request IDs are generated in a circular manner, and if a request is closed or if the session is long enough, then a request ID is reused.

You can also locate user names, session IDs, and request IDs through the Session Manager.

Administrators can view the query log using the **Manage Sessions** option in the Presentation Services Administration page.

Interpret the Log Records

After you have logged some query information and started the log viewer, you can analyze the log. Log entries for levels 1 and 2 are generally self-explanatory.

The log entries can provide insights to help database administrators (DBAs) in charge of the underlying databases tune them for optimum query performance. The query log can also help you check the accuracy of applications that use the BI Server.

The log is divided into the following sections:

- **SQL Request** — This section lists the SQL statement that is issued from the client application. You can use this information to rerun the query from the same application, or from a different application.
- **General Query Information** — This section lists the semantic model, the business model, and the subject area from which the query was run. You can use this information to provide statistics on query usage that you can use to set priorities for future application development and system management.
- **Database Query** — This section begins with an entry that reads "Sending query to the database named <*data_source_name*>," where <*data_source_name*> is the name of the data source to which the BI Server is connecting. Multiple database queries can be sent to one or more data sources. Each query has an entry in the log.

  The database query section has several uses, such as recording the SQL statement that was sent to the underlying databases. You can use this logged SQL statement to run queries directly against the database for performance tuning, results verification, or other testing purposes. You can also use this information to examine the tables that are being queried to verify that aggregate navigation is working as you expect. If you understand the structure of the underlying database, then it might also provide some insights into potential performance improvements, such as useful aggregate tables or indexes to build.

- **Query Status** — The query success entry in the log indicates whether the query completed successfully, or failed. You can search through the log for failed queries to determine why they failed. For example, all the queries during a particular time period might have failed due to database downtime.

Log Information in Presentation Services

This section describes logging specifically in Presentation Services.

Topics include:

- Use the Presentation Services Logging Facility
- Structure for the Presentation Services Configuration File
- Examples of the Formats of Logged Messages
- Presentation Services Message Structure
- Presentation Services Log Filters
- Diagnose Issues with Agents

See Understand Diagnostic Log Files and Log Configuration Files.

Use the Presentation Services Logging Facility

You can troubleshoot issues and errors using the Presentation Services logs.

By default, Presentation Services is configured to log all error events and informational and warning events of sufficient importance. An example of an important informational event is a server starting up or a server shutting down. Log files are named sawjson*xx*.log, where the *xx* is replaced by an incremented number.

To debug specific issues that a user might be encountering, the logging level can be increased to log more information than the default configuration. For example, while debugging a particular Presentation Services connectivity issue, you can increase the maximum logging on the saw.odbc log source only. This adds detailed logging for that component, without cluttering the log with detailed logging from other events. All Presentation Services configuration information is loaded from the instanceconfig.xml file.

> **Note:**
>
> Because logging affects performance, don't increase the logging on a production implementation, except to diagnose specific issues.

Structure for the Presentation Services Configuration File

The structure of the Presentation Services configuration file allows system output to be presented properly.

The structure of the configuration file is shown in the following example. The cardinality of each node is shown in brackets.

Example of the Logging Section in instanceconfig.xml File

```
Logging [1..1]
Writers [0..1]
```

```
Writer [0..1]
WriterClassGroups [0..1]
Filters [0..1]
FilterRecord [0..n]
```

An example of an instanceconfig.xml file that has four writers is shown below.

Example of the instanceconfig.xml File with Log Writers

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Oracle Business Intelligence Presentation Services Configuration File -->
<WebConfig xmlns="oracle.bi.presentation.services/config/v1.1">
   <ServerInstance>
      ...
      ...
      <Logging xmlns="oracle.bi.presentation.services/config/v1.1">
         <Writers>
            <Writer implementation="FileLogWriter" name="Global File Logger" 
writerClassId="1" dir="{%ORACLE_BIPS_INSTANCE_LOGDIR%}" filePrefix="sawjsonlog" 
rotateBy="size" rotationIntervalHours="24" maxFileSizeKb="10240" filesN="10" 
fmtName="JSON"/>
            <Writer implementation="CoutWriter" name="Console Logger" writerClassId="2"/>
            <Writer implementation="FileLogWriter" name="Audit Logger" writerClassId="3" 
dir="{%ORACLE_BIPS_INSTANCE_LOGDIR%}" filePrefix="auditjson" rotateBy="size" 
maxFileSizeKb="10240" filesN="10" fmtName="Devops-ServiceLog"/>
         </Writers>
         <WriterClassGroups>
            <WriterClassGroup name="All">1,2,3</WriterClassGroup>
            <WriterClassGroup name="File">1</WriterClassGroup>
            <WriterClassGroup name="Console">2</WriterClassGroup>
            <WriterClassGroup name="AuditLog">3</WriterClassGroup>
         </WriterClassGroups>
         <Filters>
            <FilterRecord writerClassGroup="File" path="saw" information="1" 
warning="31" error="31" trace="0" incident_error="32"/>
            <FilterRecord writerClassGroup="AuditLog" path="oa.devops.servicelog.audit" 
information="32" trace="0" warning="0" error="0" incident_error="0"/>
         </Filters>
      </Logging>
      ...
      ...
    </ServerInstance>
</WebConfig>
```

The following table contains a description of each node in the configuration hierarchy.

Element	Attribute	Description
Writers	None	Contains writers configuration. This configuration is loaded on startup.
Writer	None	Configures a writer.
Writer	implementation	The following implementations are defined: • **FileLogWriter.** Writes to a disk file. • **CoutWriter.** Writes to standard output.
Writer	name	Unique name for the writer.

Element	Attribute	Description
Writer	writerClassId	Specifies an integer number in the range 1 through 10. This number is used by filters to allow or prohibit logging. Each distinct writer must have a unique value, which is used later for filter configuration. Different writers might have the same class ID, but if they do, those writers can't be distinguished by filters.
Writer	fmtName	Specifies the format of logged messages. Valid values are: • **JSON**. Formats messages with identifying headings. • **ODL-Text**. Formats messages in Oracle Diagnostic Text format. • **ODL-XML**. Formats messages in Oracle Diagnostic XML format. See Examples of the Formats of Logged Messages for examples.
Writer (FileLogWriter specific attribute)	dir	Specifies the directory where log files are created.
Writer (FileLogWriter specific attribute)	maxFileSizeKb	Specifies the maximum size of the logging file in kilobytes. When the file size limit is reached, the file is closed and a new logging file is created.
Writer (FileLogWriter specific attribute)	filePrefix	Specifies the prefix for log files.
Writer (FileLogWriter specific attribute)	filesN	Specifies the maximum number of logging files. When this number is exceeded, the first file is deleted and re-created again. Then the logger starts to write to the beginning of the first file.
WriterClassGroups	None	Contains the definition for writer classes. A writer class is a group of writer class IDs.
WriterClassGroup (Contains [as child text] a comma-delimited list of class IDs.)	name	Specifies the name of the WriterClassGroup.
Filters	None	Contains filter configuration.
FilterRecord	writerClassGroup	Specifies the group of writers to which this record is applied. WriterClassGroup is likely defined previously in the WriterClassGroups section.
FilterRecord	path	Specifies the log source path. To enable the logging of SOAP information, enter the following value: `saw.httpserver.request.soaprequest` The current filter record is applied to the software component that's identified by that path and all its subcomponents.
FilterRecord	information	Contains an integer that specifies the severity of the corresponding message type. Only messages with a severity index less than the provided number are logged.

Element	Attribute	Description
FilterRecord	warning	Contains an integer that specifies the severity of the corresponding message type. Only messages with a severity index less than the provided number are logged.
FilterRecord	error	Contains an integer that specifies the severity of the corresponding message type. Only messages with a severity index less than the provided number are logged.
FilterRecord	trace	Contains an integer that specifies the severity of the corresponding message type. Only messages with a severity index less than the provided number are logged.
FilterRecord	incident_error	Contains an integer that specifies the severity of the corresponding message type. Only messages with a severity index less than the provided number are logged.

Examples of the Formats of Logged Messages

The fmtName attribute of the Writer element formats logged messages in one of three formats: default (JSON), ODL-Text, and ODL-XML.

The following example shows the default JSON format.

Example of the Default Format (JSON)

The default format generates messages with identifying headings, such as:

```
{"ts":"1722816936932","time":"1722816936.932","datetime":"2024-08-05T00:15:36Z
","app":"OBIPS","level":"NOTIFICATION","severity":"1","category":"saw.sawserve
r","ecid":"","rid":"","thread":"","sik":"","user":"","msg":"Oracle Analytics
Presentation Services have started
successfully.","file":"sawserver.cpp","line":"871","log_sources":"saw.sawserve
r","stacktrace":"","properties":{}}
```

The following example shows the ODL-Text format.

Example of ODL-Text Format

The short format generates messages in a shortened form without identifying headings, such as:

```
[timestamp] [component id] [messagetype:level] [message-id] [module id]
([field-name: field-value])* message-text [[
supplemental-detail
]]

[2024-09-20T19:20:26Z] [OBIPS] [NOTIFICATION:1] [] [saw.sawserver] [ecid: ]
[tid: ] [SI-Key: ] [userId: ] Oracle Analytics Presentation Services have
started successfully.[[
File:sawserver.cpp
Line:871
Location:
        saw.sawserver
]]
```

The following example shows the ODL-XML format.

Example of ODL-XML Format

The xml format generates messages in XML format, such as:

```
<msg time="2024-09-20T19:25:41Z" comp_id="OBIPS" module="saw.sawserver"
msg_id="" type="NOTIFICATION" level="1" ecid="" tid="
">SI-Key="" userId=""
 <txt>Oracle Analytics Presentation Services have started successfully.</txt>
   <suppl_detail>
   <File>sawserver.cpp</File>
   <Line>871</Line>
   <Location>
      <Logsource>saw.sawserver</Logsource>
   </Location>
   <Properties>
   </Properties>
 </suppl_detail>
</msg>
```

Presentation Services Message Structure

Each message that is logged by Presentation Services has several components, as described below.

Message Component	Description
Message Text	The text of the log message to the user.
Message Type	One of five types: information, warning, error, incident_error or trace. See What Are Log File Message Categories and Levels?
Severity	The severity is represented as a positive integer. The lower the value, the more important the message. A message with severity of 0 is the most important type of message, whereas a message with a severity of 32 isn't important.
Message Properties	Properties indicate other kinds of information. The kind varies among messages and might include user name, the IP address of the client browser, the thread ID, and so on.

Presentation Services Log Filters

FilterRecords customize logging details. Use FilterRecords to specify the implementation (output type) and logging levels for categories of web logs: Incident Error, Error, Trace, Warnings, and Information.

In the following example, the first two FilterRecords contain the following string:

```
path="saw"
```

This string logs the informational events at level 1, the error messages at level 31, and so on:

```
<FilterRecord writerClassGroup="Console" path="saw" information="1"
warning="31" error="31" trace="0" incident_error="32" />
<FilterRecord writerClassGroup="File" path="saw" information="1" warning="31"
error="31" trace="0" incident_error="32" />
```

This high-level path applies to every event.

You can customize FilterRecords by adding new FilterRecords, such as the third one shown in the preceding example, with finer-grain specification of log levels for events of various types. In this example, information is being logged to a disk file from saw.mktgsqlsubsystem.log, which generates Marketing job events.

You can disable logging of job details by changing the information level from 1 to 0, as shown in the following example, or by commenting out the lines:

```
<FilterRecord writerClassGroup="Console" path="saw" information="1"
warning="31" error="31" trace="0" incident_error="32" />
<FilterRecord writerClassGroup="File" path="saw" information="1" warning="31"
error="31" trace="0" incident_error="32" />
```

Diagnose Issues with Agents

This section contains the following topics:

- Debug Agents Using Fusion Middleware Control
- Debug Agents Manually

Debug Agents Using Fusion Middleware Control

Agent error and debug log entries are written to the main scheduler log file `nqscheduler.log`, and are visible in Log Viewer using Fusion Middleware Control.

See Use Fusion Middleware Control to View Log Information, Error Messages, and Alerts.

An agent log entry includes key agent events, and provides information on a single trace line. For example:

```
Agent Chain Completed. Status: Completed, Agent ID: /users/weblogic/ChainedAgent,
UserID: weblogic, OBIPS: example.com:0:9710, Job/Instance ID: 123/4567.
```

The table below details agent event logging, and associated trace types and levels.

Event	State	Message Type:Level
Agent Chain Started	Running	NOTIFICATION:1
Agent Chain Started	ReRunning	NOTIFICATION:1
Agent Chain Complete	Failed	ERROR:1
Agent Chain Complete	Timed Out	ERROR:1
Agent Chain Complete	Timed Out	WARNING:1
Agent Chain Complete	Warning	WARNING:1
Agent Chain Complete	Cancelled	NOTIFICATION:1
Agent Chain Complete	Try Again	NOTIFICATION:1
Agent Chain Complete	Completed	NOTIFICATION:1
Agent Started	N/A	TRACE:1
Agent Finished	N/A	TRACE:1

You can determine the log output detail written to the nqscheduler.log file by setting the level in Fusion Middleware Control. See Configure Log File Rotation Policy and Specify Log Levels. You can also filter log entries using ECID to find information specific to a particular agent chain.

Debug Agents Manually

If an agent fails to run fully or if debugging is turned on in Oracle BI Scheduler, then a log file is generated for the agent.

You manually enable debugging by setting the `Trace` element to the value of `32` in the Oracle BI Scheduler schedulerconfig.xml file, or by assigning the Fusion Middleware Control setting, LogLevel to 'Finest'.

The schedulerconfig.xml file is located at:

```
<domain_home>/config/fmwconfig/biconfig/OBISCH
```

> **Note:**
>
> You don't need to restart the component. Set the value back to the default after debugging completes.

See What Are Diagnostic Log Configuration Files and Where Are They Located?

The location for agent log files is also specified in the instanceconfig.xml file for the Oracle BI Scheduler (see Agent Scheduler Configuration Settings.) The default location for log files is the Log directory in the installation directory on the computer where the Oracle BI Scheduler is installed.

The log file name has the following format:

Agent-*JobID*-*InstanceID*.*xxx*

In this file name:

- Agent is the prefix for all agent log files.
- *JobID* is the Oracle BI Scheduler job identifier for the agent.

- *InstanceID* is the Oracle BI Scheduler instance identifier for the agent.
- *xxx* is the file extension:
  - .err for agent error log files.
  - .log for debug log files.

The agent error and debug log files are written as separate files for each agent instance that fails to run. You can use a text editor to view the files. Entries are generally self-explanatory.

The presence of an error log doesn't necessarily mean that an agent failed completely. For example, suppose an agent delivers content to multiple email addresses. If some addresses are invalid or the mail server is down, then an error log is generated for the agent.

Use ODBC/JDBC Procedures to Obtain BI Server Diagnostics

This section describes how to use ODBC/JDBC procedures to obtain diagnostic information for the BI Server.

This section contains the following topics:

- About the BI Server ODBC/JDBC Procedures
- Obtain a List of Available Diagnostic Categories
- Run Specific Diagnostics
- About Parameters for ODBC/JDBC Procedures

About the BI Server ODBC/JDBC Procedures

You can use ODBC/JDBC procedures to obtain diagnostic information for the BI Server.

These procedures are especially useful on non-Windows platforms where you can't run the Model Administration Tool.

Use the nqcmd utility to run the procedures using ODBC. See Using nqcmd to Test and Refine the Repository in *Managing Metadata Repositories for Oracle Analytics Server*.

You can also run the procedures using JDBC.

You can also query data models remotely. For information, see "About Integrating with the Oracle BI Server as a Data Source" in *Integrator's Guide for Oracle Business Intelligence Enterprise Edition*.

Obtain a List of Available Diagnostic Categories

You can first run `OBISAvailableDiagnostics()` to get a list and description of the diagnostic categories that are available.

See the following example:

```
call OBISAvailableDiagnostics()
```

The results appear similar to the following:

Category	Description
General	General overview of the OBIS instance you are connected to.

Category	Description
DBInstance:DBNAME1	All of the statistics related to the DB instance named in DBNAME1
DBInstance:DBNAMEn	All of the statistics related to the DB instance named in DBNAMEn
LDAP:Instance1	All of the statistics related to the LDAP instance named in Instance1
LDAP:Instancen	All of the statistics related to the LDAP instance named in Instancen
DBConnectionPool:Instance1	All of the statistics related to the DB connection pool named in Instance1
DBConnectionPool:Instancen	All of the statistics related to the DB connection pool named in Instancen
ThreadPool:Instance1	All of the statistics related to the Thread pool named in Instance1
ThreadPool:Instancen	All of the statistics related to the Thread pool named in Instancen
Cache:Instance1	All of the statistics related to the Cache named in Instance1
Cache:Instancen	All of the statistics related to the Cache named in Instancen

All categories, except for the General category, are Instance categories. Instance categories are statistics related to a particular instance object (like a specific physical database). If multiple instances of an object are initialized, separate categories exist for each instance, in the format *category_name*:*instance_name*. See the preceding table for examples.

Note the following about the ODBC/JDBC categories:

- The ThreadPool category only displays statistics from threads created and managed by the DbConnection PoolMgr.
- The Cache category displays statistics from the Compiler Cache and the LDAP Internal Cache.

Run Specific Diagnostics

Running diagnostics for specific categories helps troubleshoot issues and ensures the system is optimized.

After you obtain the available diagnostic categories, you can call `OBISDiagnostics(`*`string`*`)` to obtain diagnostics for individual categories, where *`string`* is a category name. For example:

```
call OBISDiagnostics('ThreadPool:orcldb_pool')
```

The results appear similar to the following:

Parameter Name	Value
CAPACITY	1000
THREAD COUNT	20
BUSY THREAD COUNT	15
ACCUMULATED REQUESTS	5

Parameter Name	Value
MAX STACK SIZE	100

The spelling of the category must be correct, or no rows are returned.

Another example might be:

```
call OBISDiagnostics('General')
```

The results appear similar to the following:

Parameter Name	Value
TOTAL SESSIONS	10
QUERIES PER SEC	5
NEW LOGINS	10
ACTIVE LOGINS	7
NEW REQUESTS	30
DATA CACHE HIT PER SEC	5
NEW INIT BLOCKS	10

About Parameters for ODBC/JDBC Procedures

The following tables provide parameter reference information for each category type.

Parameter Name	Description
TOTAL SESSIONS	The total number of sessions connecting clients to the Oracle BI Server.
QUERIES PER SEC	The number of queries completed each second by the Oracle BI Server.
NEW LOGINS	The total number of new login requests received by the Oracle BI Server.
ACTIVE LOGINS	The total number of active logins within the Oracle BI Server.
NEW REQUESTS	The number of new process requests received by the Oracle BI Server.
DATA CACHE HIT PER SEC	The percentage of data cache hits for each second.
NEW INIT BLOCKS	The total number of new initialization block requests received by the Oracle BI Server.

Parameter Name	Description
QUERIES PER SEC	The number of queries completed each second by the back-end database.
FAILED QUERIES PER SEC	The number of queries that failed each second in the back-end database.
NEW PREPARES	The number of prepares sent to the back-end database.
ROWS PER SEC	The number of rows retrieved each second from the back-end database.

Parameter Name	Description
KB PER SEC	The number of kilobytes retrieved each second from the back-end database.

Parameter Name	Description
NEW REQUESTS	The total number of new LDAP authentication requests received.
NEW IMPERSONATED REQUESTS	The total number of new impersonated LDAP authentication requests received.
ACTIVE REQUESTS	The number of LDAP authentication requests active within the Oracle BI Server.

Parameter Name	Description
CAPACITY	The maximum number of connections that the database connection pool allows.
CONNECTION COUNT	The current number of open connections in the thread pool.
BUSY CONNECTION COUNT	The number of connections that have been assigned to process a query, or that are currently processing a query, in the database connection pool.
AVG REQUESTS PER SEC	The average number of requests each second that have been submitted to the database connection pool.
AVG OPEN REQUESTS PER SEC	The average number of connections that are opened each second. Connections might be opened for new connections, because other connections timed out, or because of problems with a connection.

Parameter Name	Description
CAPACITY	The maximum number of threads allowed by the thread pool.
THREAD COUNT	The current number of threads in the thread pool.
BUSY THREAD COUNT	The current number of threads that have been assigned work. The thread might be blocked waiting for a resource or data, or it could be actively running on a CPU.
ACCUMULATED REQUESTS	The total number of requests that have been submitted to the thread pool.
MAX STACK SIZE	The maximum number of stack bytes consumed for all threads in the thread pool.

Parameter Name	Description
CAPACITY	The total capacity of the specified cache object.
TOTAL REQUESTS	The total number of requests each second against the specified cache object.
AVG REQUESTS	The average number of requests each second against the specified cache object.
AVG HITS	The average number of hits each second for the specified cache object.

Parameter Name	Description
AVG MISS	The average number of misses each second for the specified cache object.

9 Track Usage

Usage tracking enables administrators to track user-level queries to content.

Tracking usage is helpful way to determine which user queries are creating performance bottlenecks, based on query frequency and response time. Administrators set up the criteria to track user queries and generate usage reports that can be used in a variety of ways such as database optimization, aggregation strategies, or billing users or departments based on the resources that they consume.

Topics:

About Tracking Usage

You can configure usage tracking in services that offer enterprise modeling features. Usage information is tracked at the detailed user query level so you can answer questions such as:

- How are users engaging with Oracle Analytics Server?
- Where are they spending or not spending their time?
- How long do users spend in each session, between sessions, and between queries?
- How are queries within sessions, across sessions, and across users related to each other?
- Are users drilling up and down in analyses?
- What queries are running when issues are reported?

The usage statistics that you gather can help you to monitor system usage and performance so you can better understand and predict user behavior. You can increase your efficiency and reduce errors if you know in advance how your system is likely to be used.

When you enable usage tracking, the system collects data records for every query that is run and writes them all to database tables. Both logical and physical queries are tracked and logged in separate tables, along with various performance measures such as the time taken to run the query and number of rows searched while processing a user query.

Prerequisites for Usage Tracking

If you want to track usage, verify you meet the following prerequisites:

- You currently use Semantic Modeler or Model Administration Tool to manage your semantic model.
  To configure usage tracking, you must add the usage tracking database details to your semantic model using either Semantic Modeler or Model Administration Tool.
- You have appropriate access permissions on the database where you want to store usage information.
  You must have the credentials for a user who has permissions to create the usage tracking tables on the database schema and write usage data to the tables.
- The database supports usage tracking: Oracle Database or Oracle Autonomous Data Warehouse
- You have created a data connection to your usage tracking database with the following settings. See Connect to Data.
  - **System Connection** - Select the **System Connection** check box.
    When you select the **System Connection** check box, the connection becomes available in Semantic Modeler. Similarly in Model Administration Tool, the **System Connection** option enables you to select **Use Data Connection** and enter the connection's **Object ID** instead of manually entering the connection details in the **Data Source Name** field. See Specify the Usage Tracking Database.
  - **User Name** and **Password** - The **User Name** must match the name of the schema in the database that you want to use for usage tracking. For example, if the schema you want to use is called UT_Schema the **User Name** must be UT_Schema.

> **Note:**
>
> If you use Model Administration Tool, you can also define database connections for semantic models and the usage tracking database using the Console. If you use the Console, you can select **Use Console Connection** and enter the connection's **Name** while specifying the usage tracking database in Model Administration Tool, instead of entering the connection details in the **Data Source Name** field.

If you want to use Oracle Autonomous Data Warehouse as the usage tracking database, complete these additional tasks before you specify the usage tracking database in your semantic model:

- Download the Oracle Autonomous Data Warehouse wallet. See Download Client Credentials (Wallets) in *Using Oracle Autonomous Database Serverless*.
- Upload the Oracle Autonomous Data Warehouse wallet to Oracle Analytics Server.
- Create a self-service connection to Oracle Autonomous Data Warehouse and ensure that you select the **System Connection** check box. See Connect to Oracle Autonomous Data Warehouse.

About the Usage Tracking Database

The system stores usage tracking details in a database that you specify. The database can be an Oracle Database or . You specify the database and connection pool details in your semantic model using Semantic Modeler or Model Administration Tool.

See Specify the Usage Tracking Database.

About Usage Tracking Parameters

After specifying the database where you want to store usage tracking information, you must set various usage tracking parameters through the Console (**Advanced System Settings** page).

Parameters required to configure usage tracking:

- Enable usage tracking
- Connection pool name
- Physical and logical query logging table names
- Maximum number of query rows in the usage tracking tables

After you set these parameters and apply the changes, Oracle Analytics:

- Creates the physical and logical query logging tables in the database specified in the semantic model. The table names are based on the names that you provide in the physical and logical query logging table name parameters.
- Starts to log usage tracking data in these tables.

See Set Usage Tracking Parameters.

About Analyzing Usage Data

You can use the system to create useful usage reports from the tracking data added to the physical and logical query logging tables.

You can connect to the database, create a dataset from the tables, and create reports and visualizations to help you understand your users' queries and take appropriate action to improve performance.

Understand the Usage Tracking Tables

The system stores usage tracking data in three database tables.

The usage tracking process creates these tables with table names that you specify through settings in the Systems Settings page.

- Usage Tracking Logical Query Logging Table
- Usage Tracking Physical Query Logging Table
- Usage Tracking Initialization Block Table

See Set Usage Tracking Parameters.

Usage Tracking Logical Query Logging Table

The following table describes each column in the database table that tracks logical queries. Where appropriate, the data type such as variable character field (varchar and varchar2) and length is specified. As you review the descriptions in this table, you might assume that certain time-related columns can be added or subtracted to equal exact values. For example, you might assume that `TOTAL_TIME_SEC` is equal to `END_TS` minus `START_TS`. The columns don't provide such exact values because:

- Various processes run in parallel and their speed depends on the load and on database performance. Server-based operations might be either light or intensive.
- If all connections are full, then the query enters a queue and waits to be processed. The timing depends on the load and the configuration.

User, Session, and ID-related Columns

Column	Description
ID	In the Logical Query table, this column indicates the unique row identifier. In the Physical Query table, this column is denoted by the name `LOGICAL_QUERY_ID`.
NODE_ID	Contains `<hostname>:<component_name>` where <component_name> can be overridden by the environment variable `COMPONENT_NAME`. For example, `examplehost:obis1` (for a single instance). Default value of `COMPONENT_NAME` is obis1.
PRESENTATION_NAME	Indicates the name of the Catalog. Default is Null and data type is Varchar(128).
IMPERSONATOR_USER_NAME	Specifies the user name of the impersonated user. If the request isn't run as an impersonated user, then the value is None. Default is None and the data type is Varchar(128).
USER_NAME	Specifies the name of the user who submitted the query.
ECID	Indicates the system-generated execution context ID. Data type is Varchar2(1024).
TENANT_ID	Specifies the name of the tenant of the user who ran the initialization block. Data type is Varchar2(128).
SERVICE_NAME	Specifies the name of the service. Data type is Varchar2(128).
SESSION_ID	Indicates the ID of the session. Data type is Number(10).
HASH_ID	Indicates the `HASH` value for the logical query. Data type is Varchar2(128).

Query Origin-related Columns

Column	Description
QUERY_SRC_CD	The source of the request. Note that the requestor can set QUERY_SRC_CD to any string value to identify itself. Possible values include: • Report - If the source is an analysis or any export operation. • Drill - If the source is a change in dimension caused by drilling up or down. • ValuePrompt - If the source is the Value drop-down list in a filter dialog or a dashboard prompt. • VisualAnalyzer - If the source is a workbook to visualize data. • DisplayValueMap or MemberBrowserDisplayValues or MemberBrowserPath - If the source is a value related to the display of an analysis. • SOAP - If the source is a call from web services such as DataSetSvc. • Seed - If the source is an agent that seeds the cache of the analytics server. • Null - If the source is the Admininistration Tool physical table or column row count, or view data.
SAW_DASHBOARD	Indicates the path name of the dashboard. If the query wasn't submitted through a dashboard, then the value is NULL.
SAW_DASHBOARD_PG	Indicates the page name in the dashboard. If the request isn't a dashboard request, then the value is NULL. Default is Null and the data type is Varchar(150).
SAW_SRC_PATH	Specifies the path name in the Catalog for the analysis.

Query Details-related Columns

Column	Description
ERROR_TEXT	Contains the error message from the back-end database. This column is applicable only if the SUCCESS_FLAG is set to a value other than 0 (zero). Multiple messages are concatenated and aren't parsed by the system. Default is Null and data type is Varchar(250).
QUERY_BLOB	Contains the entire logical SQL statement without any truncation. The QUERY_BLOB column is a character string of type Long.
QUERY_KEY	Contains an MD5 hash key generated by the system from the logical SQL statement. Default is Null and the data type is Varchar(128).

Column	Description
QUERY_TEXT	Indicates the SQL statement that was submitted for the query. The data type is Varchar(1024). You can change the length of this column (using the ALTER TABLE command), but note that the text written into this column is always truncated to the size that is defined in the physical layer. The semantic model administrator mustn't set the length of this column to a value greater than the maximum query length that's supported by the back-end physical database. For example, Oracle Databases enable a maximum Varchar of 4000, but Oracle Databases truncate to 4000 bytes, not 4000 characters. If you use a multibyte character set, the actual maximum string size has a varying number of characters, depending on the character set and characters used.
REPOSITORY_NAME	Specifies the name of the semantic model that the query accesses.
SUBJECT_AREA_NAME	Contains the name of the business model being accessed.
SUCCESS_FLG	Indicates the completion status of the query, as defined in the following list: • 0 - The query completed successfully with no errors. • 1 - The query timed out. • 2 - The query failed because row limits were exceeded. • 3 - The query failed due to some other reason.

Execution Timing-related Columns

Column	Description
COMPILE_TIME_SEC	Contains the time in seconds required to compile the query. The number for COMPILE_TIME_SEC is included in TOTAL_TIME_SEC.
END_DT	Indicates the date the logical query completed.
END_HOUR_MIN	Indicates the hour and minute the logical query completed.
END_TS	Indicates the date and time the logical query completed. The start and end timestamps also reflect any time that the query spent waiting for resources to become available. If the user submitting the query navigates away from the page before the query finishes, then the final fetch never happens and a timeout value of 3600 is recorded. However, if the user navigates back to the page before the timeout, then the fetch completes at that time, which is recorded as the end_ts time.
START_DT	Indicates the date that the logical query was submitted.
START_HOUR_MIN	Indicates the hour and minute that the logical query was submitted.

Column	Description
START_TS	Indicates the date and time that the logical query was submitted.
TOTAL_TIME_SEC	Indicates the time in seconds that the system spent working on the query while the client waited for responses to its analyses. TOTAL_TIME_SEC includes the time for COMPILE_TIME_SEC.
RESP_TIME_SEC	Indicates the time taken for query response. Data type is Number(10).

Execution Details-related Columns

Column	Description
CUM_DB_TIME_SEC	Contains the cumulative time of all queries sent to the database. Queries run in parallel, so the cumulative query time is equal to or greater than the total time connected to the database. For example, suppose a logical request spawns 4 physical SQL statements sent to the database, and the query time for 3 of the queries is 10 seconds, and for one query is 15 seconds, CUM_DB_TIME_SEC displays 45 seconds because the queries run in parallel.
CUM_NUM_DB_ROW	Contains the total number of rows returned by the back-end databases.
NUM_DB_QUERY	Indicates the number of queries that were submitted to the back-end databases to satisfy the logical query request. For successful queries (SuccessFlag = 0), this number is 1 or greater.
ROW_COUNT	Indicates the number of rows returned to the query client. When a large amount of data is returned from a query, this column isn't populated until the user displays all the data.
TOTAL_TEMP_KB	Specifies the total KB received for a query. Data type is Number(10).

Cache-related Columns

Column	Description
CACHE_IND_FLG	Holds Y to indicate a cache hit for the query; N to indicate a cache miss. Default is N.
NUM_CACHE_HITS	Indicates the number of times that the cache result returned for the query. NUM_CACHE_HITS is a 32-bit integer (or a 10-digit integer). Default is Null.
NUM_CACHE_INSERTED	Indicates the number of times that the query generated a cache entry. Default is Null. NUM_CACHE_INSERTED is a 32-bit integer (or a 10-digit integer).

Usage Tracking Physical Query Logging Table

The following table describes the database table that tracks physical queries. This database table records the physical SQL information for the logical queries stored in the logical query logging table. The physical query table has a foreign key relationship to the logical query table.

User, Session, and ID-related Columns

Column	Description
ID	Specifies the unique row identifier.
LOGICAL_QUERY_ID	Refers to the logical query in the logical query logging table. Data type is Varchar2(50).
HASH_ID	Indicates the HASH value for the logical query. Data type is Varchar2(128).
PHYSICAL_HASH_ID	Indicates the HASH value for the physical query. Data type is Varchar2(128).

Query Details-related Columns

Column	Description
QUERY_BLOB	Contains the entire physical SQL statement without any truncation. The QUERY_BLOB column is a character string of type long.
QUERY_TEXT	Contains the SQL statement submitted for the query. Data type is Varchar(1024).

Execution Timing-related Columns

Column	Description
END_DT	Indicates the date the physical query completed.
END_HOUR_MIN	Indicates the hour and minute the physical query completed.
END_TS	Indicates the date and time the physical query completed. The start and end timestamps also reflect any time that the query spent waiting for resources to become available.
TIME_SEC	Indicates the physical query execution time.
START_DT	Indicates the date the physical query was submitted.
START_HOUR_MIN	Indicates the hour and minute the physical query was submitted.
START_TS	Indicates the date and time the physical query was submitted.

Execution Details-related Columns

Column	Description
ROW_COUNT	Contains the number of rows returned to the query client.

Usage Tracking Initialization Block Table

The following table describes the database table that tracks information about the initialization blocks.

> **Note:**
>
> Currently the initialization block usage tracking tables include only session initialization blocks and don't include the semantic model initialization blocks.

User, Session, and ID-related Columns

Column	Description
`USER_NAME`	The name of the user who ran the initialization block. The data type is Varchar2(128).
`TENANT_ID`	The name of the tenant of the user who ran the initialization block. The data type is Varchar2(128).
`SERVICE_NAME`	The name of the service. The data type is Varchar2(128).
`ECID`	The system-generated execution context ID. The data type is Varchar2(1024).
`SESSION_ID`	The ID of the session. The data type is Number(10).

Query Details-related Columns

Column	Description
`REPOSITORY_NAME`	The name of the semantic model that the query accesses. The data type is Varchar2(128).
`BLOCK_NAME`	The name of the initialization block that was run. The data type is Varchar2(128).

Execution Timing-related Columns

Column	Description
`START_TS`	The date and time that the initialization block started.
`END_TS`	The date and time that the initialization block finished. The start and end timestamps also reflect the time that the query spent waiting for resources to become available.
`DURATION`	The length of time it took to run the initialization block. The data type is Number(13,3).

Execution Details-related Columns

Column	Description
`NOTES`	Notes about the initialization block and its running. The data type is Varchar2(1024).

Typical Workflow for Tracking Usage

Here are the tasks to track the user-level queries to Oracle Analytics Server.

Task	Description	More Information
Decide where to store your usage tracking data	Understand which database types you can use for usage tracking.	About the Usage Tracking Database
Set up a connection to the usage tracking database	Create a data connection (or a Console connection) to the database where you want to store usage tracking information.	Prerequisites for Usage Tracking
Specify the usage tracking database	Define the usage tracking database in your semantic model.	Specify the Usage Tracking Database
Specify usage tracking parameters	Enable usage tracking for your system, and then specify connection details and table names for the usage tracking database.	Set Usage Tracking Parameters
Analyze the usage tracking data	Create usage reports from usage tracking data.	Analyze Usage Tracking Data

Set Up Usage Tracking

Specify the database, schema, and connection pool details for usage tracking.

To set up usage tracking:

1. In the Oracle database, create a schema for usage tracking, and assign all privileges to that schema.
2. Add the Oracle database, connection pool, and the schema to be used for usage tracking to the semantic model.
3. Upload the modified semantic model file.
4. Restart the BI Server.
5. In Answers, select subject area, then and run queries.

This creates the usage tracking tables for logical and physical queries, and adds the entries for usage tracking.

Specify the Usage Tracking Database

Before you can track usage of reports, dashboards, and data visualization workbooks on your system, you must specify the database where you want to store the usage tracking data in your semantic model.

The database you specify must have at least one schema defined. The system creates usage tracking tables in the schema whose name matches the user name you specify in the database connection details. For example, if the name of a schema in the usage tracking database is “UT_Schema”, you must specify "UT_Schema" in the **User Name** field for the connection. The usage tracking-tables are created in the schema named “UT_Schema”.

You must configure the database and connection pool details in the physical layer of your semantic model. Use Semantic Modeler or the Model Administration Tool to configure the usage tracking database.

- Specify the Usage Tracking Database Using Semantic Modeler
- Specify the Usage Tracking Database Using Model Administration Tool

Specify the Usage Tracking Database Using Semantic Modeler

Use Semantic Modeler to configure your usage tracking database if you currently use Semantic Modeler to develop semantic models.

1. If you haven't done so already, create a data connection to your usage tracking database with the **System Connection** option selected.

   The database type must be either Oracle Database or Oracle Autonomous Data Warehouse and the **User Name** used to connect to the database must match the name of the schema where you want the user tracking tables to be stored. See Prerequisites for Usage Tracking.
2. On the Home page, click **Navigator** and then click **Semantic Models**. In the Semantic Models page, click a semantic model to open it.
3. Create a database object for the usage tracking database.
   a. Click **Physical Layer**.
   b. In the Physical Layer pane, click **Create** and then click **Create Database**.
   c. In **Name**, enter a name for the database of your semantic model (for example, UsageTracking) and click **OK**.
4. Add a connection pool to connect to the usage tracking database.
   a. In the database tab, click **Connection Pools**.
   b. Click **Add Source**.
   c. Double-click the **Name** field, and enter a name for the connection pool. For example, UTConnectionPool.
   d. Double-click the **Connection** field, and select the data connection you want to use from the list. For example MyUTDatabase.

   > **Note:**
   >
   > - **System connection** - Semantic models can only use data connections with the **System connection** option selected. See About Semantic Model Data Source Connections.
   > - **User Name** and **Password** - The **User Name** specified in the data connection must match the name of a schema in the database that you want to use for usage tracking. For example, if the schema you want to use is called UT_Schema, the **User Name** must be UT_Schema. See Prerequisites for Usage Tracking.

   e. Click **Open Detail**. In the Connection Pool pane, verify that the **Require fully qualified table names** check box isn't selected.

5. Validate your changes. See Run the Advanced Consistency Check Before Deploying a Semantic Model.
6. Save your changes.

Specify the Usage Tracking Database Using Model Administration Tool

Use Model Administration Tool to configure your usage tracking database if you currently use Model Administration Tool to develop semantic models.

You don't need to make any updates to your semantic model if you want to track usage in an existing database or connection pool. You can skip these steps. You can use the existing database, connection pool, and tables as part of the usage tracking system configuration. Usage tracking won't delete the existing tables and create new tables with the same name if the table schema matches between the old and new tables.

1. In Model Administration Tool, open the semantic model .

   From the **File** menu, select **Open**.

2. Specify the usage tracking database:

   a. In the Physical layer of the semantic model, right-click and select **New Database**.

   b. In the Database dialog, provide a name for the database of your semantic model; for example `SQLDB_UsageTracking`, specify the database type, for example `Oracle 12c`, and click **OK**.

   c. Right-click the newly created database, select **New Object**, and then select **Connection Pool**.

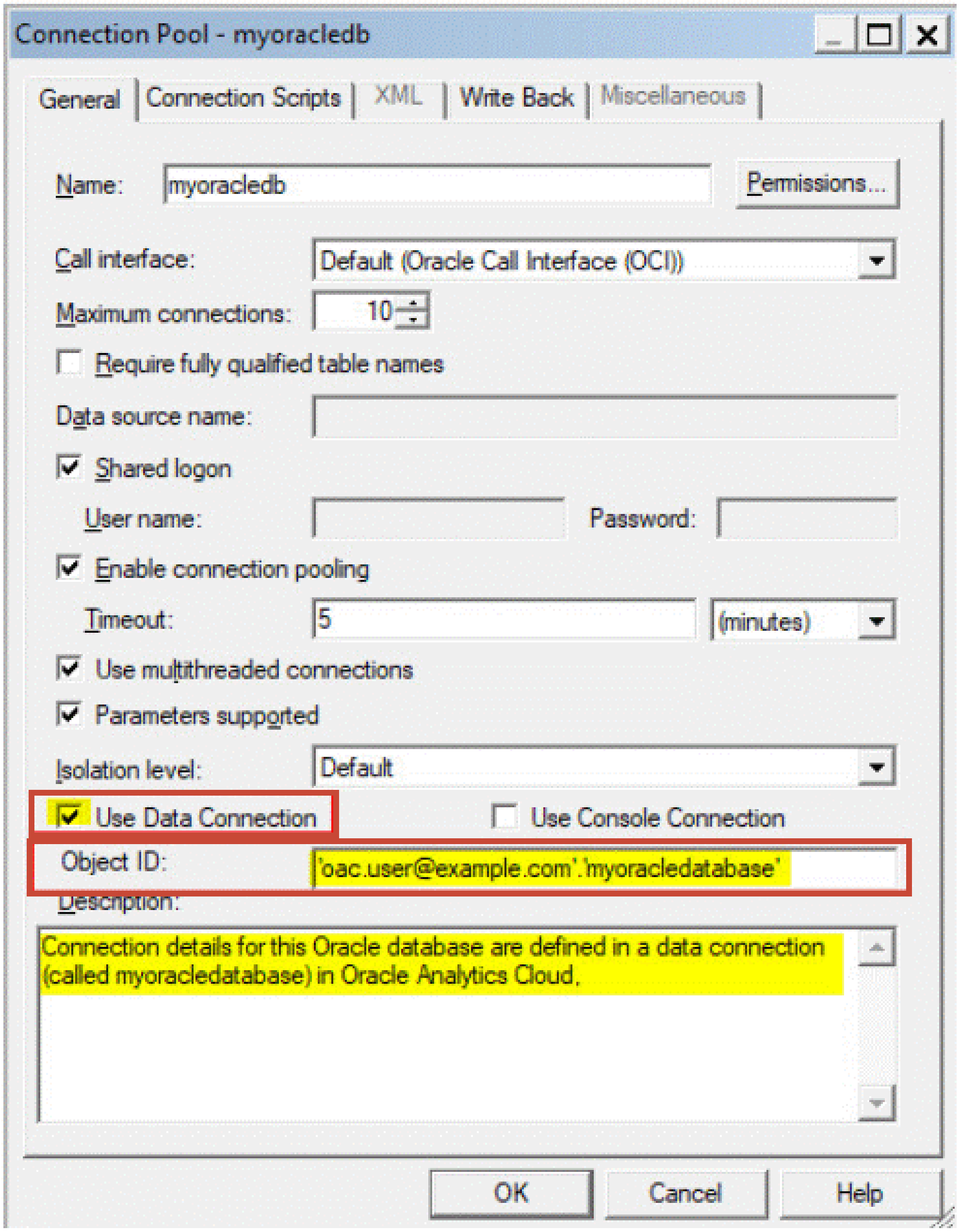
   d. In the Connection Pool dialog, enter connection pool details and specify values for:

      - **Call interface**: Select Default (Oracle Call Interface (OCI)).
      - **Require fully qualified table names**: Ensure that this check box isn't selected.
      - **Data Source Name****: Specify the data source to which you want this connection pool to connect and send physical queries. For example: `(DESCRIPTION =(ADDRESS = (PROTOCOL = TCP)(HOST = <DB Host>)(PORT = <DB port>))(CONNECT_DATA =(SERVER = DEDICATED)(SERVICE_NAME = <Servicename>)))`
      - **User name and Password**: Enter a user name that *matches the name of a schema* available in the usage tracking database.

      As an alternative to providing the **Data Source Name, you can refer to an existing database connection “by name” in the Connection Pool dialog.

      - Data connections - To use the connection details for a database defined through the Data tab as your usage tracking database, select **Use Data Connection** and enter the connection's **Object ID** instead of manually entering the connection details in the **Data Source Name** field. Ensure that the data connection you want to use was created with the **System Connection** option selected.
      - Console connections - If you use Model Administration Tool, you might define database connections for semantic models using the Console. To use the connection details for a database that you defined through the Console as your usage tracking database, select the **Use Console Connection** check box and enter the name of the database connection in the **Connection Name** field.

      For example:



Name: myoracledb

Call interface: Default (Oracle Call Interface (OCI))

Maximum connections: 10

□ Require fully qualified table names

Data source name:

☑ Shared logon

User name: Password:

☑ Enable connection pooling

Timeout: 5 (minutes)

☑ Use multithreaded connections

☑ Parameters supported

Isolation level: Default

☑ Use Data Connection □ Use Console Connection

Object ID: 'oac.user@example.com'.'myoracledatabase'

Description:

Connection details for this Oracle database are defined in a data connection (called myoracledatabase) in Oracle Analytics Cloud.

3. Validate your changes by clicking **Tools**, **Show Consistency Checker**, and then **Check All Objects**.

4. Optional: Save changes locally by clicking **File**, and then **Save**.

Set Usage Tracking Parameters

To start recording usage information, you must specify connection details for the database you want to use and names for the database tables used to track usage. You set these parameters through the Console (**Advanced System Settings** page).

> **Note:**
>
> In Oracle BI Enterprise Edition 12c, you set options in configuration files to configure usage tracking. If you upgrade from Oracle BI Enterprise Edition 12c to Oracle Analytics Server, usage tracking settings are preserved and usage tracking continues to work but no values are displayed on the System Settings page, that is, they appear blank. You don't have to re-enter your usage tracking details through System Settings but, for clarity, you might decide to do so.

1. Click **Console**.
2. Click **Advanced System Settings**.
3. Click **Usage Tracking**.
4. Enable usage tracking for your system. Ensure **Enable Usage Tracking** is switched on.
5. Set the following properties:
   - **Usage Tracking Connection Pool**
     Name of the connection pool that you created for your usage tracking database in the format, `<database name>.<connection pool name>`. For example, `UsageTracking.UTConnectionPool`.
   - **Usage Tracking Initialization Block Table**
     Name of the database table you want to use to store information about initialization blocks in the format, `<database name>.<catalog name>.<schema name>.<table name>` or `<database name>.<schema name>.<table name>`. For example, `UsageTracking.UT_Schema.InitBlockInfo`.
   - **Usage Tracking Physical Query Logging Table**
     Name of the database table you want to use to store physical query details in the format, `<database name>.<catalog name>.<schema name>.<table name>` or `<database name>.<schema name>.<table name>`. For example, `UsageTracking.UT_Schema.PhysicalQueries`.
   - **Usage Tracking Logical Query Logging Table**
     Name of the database table you want to use to store logical query details in the format, `<database name>.<catalog name>.<schema name>.<table name>` or `<database name>.<schema name>.<table name>`. For example, `UsageTracking.UT_Schema.LogicalQueries`.
   - **Usage Tracking Max Rows**
     Maximum number of rows that you want in the usage tracking tables. Minimum value is 1, maximum is 100,000, and 0 means unlimited. If the row count exceeds the maximum number of rows, then the usage tracking process deletes the excess rows based on the oldest timestamp.
6. Click **Apply**.

Oracle Analytics creates the usage tracking tables and starts to log user queries.

Recommended Configuration Settings for Usage Tracking using SQL Server

Use these configuration settings to optimize usage tracking using a SQL Server database for RCU.

Configuration Settings to optimize usage tracking using SQL Server

Where?	Configuration Settings	More Information
`odbc.ini`	In the section for SQL Server [sql_server_db], add the following two settings: • `EnableQuotedIdentifiers=1` • `WorkArounds2=2`	After setting these parameters, stop and start managed BI Server nodes (for example, to stop and start OBIS1, issue stop.sh -i obis1, and start.sh -i obis1).
`NQSConfig.ini`	Add the following parameters using these examples substituted with your own connection pool name and SQL Server database name. `PHYSICAL_TABLE_NAME="U_T"."OAS_UT_DB"."DEV_BIPLATFORM"."S_NQ_DB_ACCT" ; CONNECTION_POOL="U_T"."U_T" ; PHYSICAL_QUERY_TABLE_NAME="U_T"."OAS_UT_DB"."DEV_BIPLATFORM"."S_NQ_ACCT" ; INIT_BLOCK_TABLE_NAME="U_T"."OAS_UT_DB"."DEV_BIPLATFORM"."S_NQ_INITBLOCK" ; INIT_BLOCK_CONNECTION_POOL="U_T"."U_T" ;`	After setting these parameters, stop and start managed BI Server nodes (for example, to stop and start OBIS1, issue stop.sh -i obis1, and start.sh -i obis1).
In the Console	Set the following parameters using these examples substituted with your own connection pool name and SQL Server database name. `Usage Tracking Connection Pool = U_T.U_T Usage Tracking Initialization Block Table = U_T.OAS_UT_DB.DEV_BIPLATFORM.S_NQ_INITBLOCK Usage Tracking Physical Query Logging Table = U_T.OAS_UT_DB.DEV_BIPLATFORM.S_NQ_DB_ACCT Usage Tracking Logical Query Logging Table = U_T.OAS_UT_DB.DEV_BIPLATFORM.S_NQ_ACCT`	After setting these parameters, restart the whole OBI stack.

Analyze Usage Tracking Data

Create usage reports to understand the user queries and take appropriate action.

Follow these examples:

- Analyze Usage Tracking Data by Creating a Dataset
- Analyze Usage Tracking Data Using a Subject Area in the Semantic Model

Analyze Usage Tracking Data by Creating a Dataset

Create usage reports by creating datasets with data from the physical and logical query logging tables to understand the user queries.

1. On the Home page, click the **Page Menu** and select **Open Classic Home**. Create and run an analysis.

   The system populates the query in the usage tracking tables in the usage tracking database.

2. On the Home page, click **Create**, and click **Dataset**.
3. In Create Dataset, click the connection to the usage tracking database, and select the schema specified in the Physical Query and Logical Query Logging table names in System Settings. For example, the schema name provided in `<database name>.<schema name>.<table name>` for the Physical Query and Logical Query Logging table names.

   This is the database connection you created to set up usage tracking. See Prerequisites for Usage Tracking.

4. In Add Dataset, search for the usage tracking physical query logging table, add all the columns, name the dataset (for example, Physical Queries), and then click **Add**. Similarly, search for the usage tracking logical query logging table, add all the columns, name the dataset (for example, Logical Queries), and then click **Add**.
5. On the dataset Results page, click **Create Workbook**. Add both the datasets to the workbook: for example, the Physical Queries and Logical Queries datasets. Name the workbook (for example, Usage Tracking).
6. In the Prepare tab of the workbook, click **Data Diagram**, and create joins between the datasets using a column such as the ID column.
7. In Visualize, drag data to create visualizations based on your requirement.

   Refer to the usage tracking table descriptions in "Understand Usage Tracking Tables" to select applicable columns. For example, you can create a visualization to show how many queries took how much time.

Analyze Usage Tracking Data Using a Subject Area in the Semantic Model

Create usage reports using a subject area in the semantic model to understand the user queries.

You must import metadata to ensure that physical and metadata are synchronized. Don't customize by adding new columns in the usage tracking tables to avoid schema mismatch issues.

1. On the Home page, click the **Page Menu** and select **Open Classic Home**. Create and run an analysis.

The system populates the query in the usage tracking tables in the usage tracking database.

2. Import the semantic model that has the Usage Tracking tables updated with the query results. See Import the Deployed Model to Create a Semantic Model.
3. On the Home page, click **Data**, and then under **Datasets**, select the subject area that corresponds to the usage tracking tables to create a workbook.
4. On the New Workbook page, in Visualize, drag data to create visualizations based on your requirement.

   Refer to the usage tracking table descriptions in "Understand Usage Tracking Tables" to select applicable columns. For example, you can create a visualization to show how many queries took how much time.

Manually Set Up Direct Insertion to Collect Information for Usage Tracking

Direct insertion is the recommended method for setting up usage tracking.

This section describes how to manually set up direct insertion, and contains the following topics:

- Set Up the Usage Tracking Statistics Database
- Set Direct Insertion Parameters
- Set Optional Direct Insert Parameters

Set Up the Usage Tracking Statistics Database

Before you can use direct insertion usage tracking, you must set up a database to store the usage tracking statistics.

You must run the Repository Creation Utility (RCU) on the target database to create the required statistics schemas.

Typically, you use the database you installed for use with Oracle Analytics Server as the statistics database because this database already has the RCU-created schemas. The RCU-created table names for usage tracking are `S_NQ_ACCT`, `S_NQ_DB_ACCT`, and `S_NQ_INITBLOCK`. See Understand the Usage Tracking Tables.

You also need to import the database into the Physical layer of the semantic model.

1. Run the Repository Creation Utility on an external database of your choice. You can skip this step if you choose to use the database you installed for use with Oracle Analytics Server for usage tracking statistics, because this database has the RCU-created tables already.
2. Open the Model Administration Tool and import the database into the Physical layer. See *Managing Metadata Repositories for Oracle Analytics Server*.
3. Save and close the semantic model.

Set Direct Insertion Parameters

You can set specific parameters for direct insertion on any new installation.

To set up direct insertion for new (non-upgraded) installations, use a text editor.

To set up direct insertion usage tracking use the steps below.

1. On the BI Server computer, open the NQSConfig.INI file in a text editor. You can find this file at:

   BI_DOMAIN/config/fmwconfig/biconfig/OBIS

   Make a backup copy of the file before editing.

2. In the [USAGE_TRACKING] section, update the following parameters:

   - Set `ENABLE` to `YES`.
   - Set `DIRECT_INSERT` to `YES`.
   - Set `PHYSICAL_TABLE_NAME` to the name of the fully-qualified database table for collecting query statistic information, as it appears in the Physical layer of the semantic model. For example:

     ```
     PHYSICAL_TABLE_NAME = "My_DB"."DEV_BIPLATFORM"."S_NQ_ACCT";
     ```

   - Set `CONNECTION_POOL` to the name of the fully-qualified connection pool for the query statistics database, as it appears in the Physical layer of the semantic model. For example:

     ```
     CONNECTION_POOL = "My_DB"."Usage Connection Pool";
     ```

   - Set `INIT_BLOCK_TABLE_NAME` to the name of the fully-qualified database table for inserting records that correspond to the initialization block statistics, as it appears in the Physical layer of the semantic model. For example:

     ```
     INIT_BLOCK_TABLE_NAME = "My_DB"."DEV_BIPLATFORM"."S_NQ_INITBLOCK";
     ```

   - Set `INIT_BLOCK_CONNECTION_POOL` to the name of the fully-qualified connection pool for the table for inserting records that correspond to the initialization block statistics, as it appears in the Physical layer of the semantic model. For example:

     ```
     INIT_BLOCK_CONNECTION_POOL = "My_DB"."Usage Connection Pool";
     ```

   > **Note:**
   >
   > For Usage Tracking insertions to succeed, the connection pool must be configured with a user ID that has write access to the back-end database. Also, it's recommended that the connectivity type supports international data.

3. Save and close the file.
4. Restart the BI Server.

Set Optional Direct Insert Parameters

The Usage Tracking section of the NQSConfig.INI file has several parameters.

In addition to the setup parameters described previously, you can also update the following optional parameters in the Usage Tracking section of the NQSConfig.INI file:

- **BUFFER_SIZE.** This parameter indicates how much memory the BI Server allocates for buffering the insert statements. Such a buffer lets the BI Server submit multiple insert statements as part of a single transaction, improving Usage Tracking insert throughput. It also means that ordinary analyses don't have to wait on Usage Tracking insertions, which improves average query response time. You might want to adjust this value based on available memory and memory utilization on the server computer.

- **BUFFER_TIME_LIMIT_SECONDS.** This parameter indicates the maximum amount of time that an insert statement remains in the buffer before the Usage Tracking subsystem attempts to issue it. This time limit ensures that the BI Server issues the insert statements quickly, even during periods of extended quiescence.
- **NUM_INSERT_THREADS.** This parameter indicates the number of threads that remove insert statements from the buffer and issue them to the Usage Tracking database. Assuming separate connection pools for readers and inserters, the number of insert threads typically equals the Maximum Connections setting in the connection pool.
- **MAX_INSERTS_PER_TRANSACTION.** This parameter indicates the maximum number of insert statements that the Usage Tracking subsystem attempts to issue as part of a single transaction. The larger this number, the greater potential throughput for UsageMarathon Tracking inserts. However, a larger number also increases the likelihood of transactions failing due to deadlocks. A small value for `BUFFER_TIME_LIMIT_SECONDS` can limit the number of inserts per transaction.

See NQSConfig.INI File Configuration Settings.

Part V

Configure Oracle Analytics Server

Although the installer installs Oracle Analytics Server, some functionality requires additional configuration changes (for example, the specification of connection details to external systems and email systems).

You can also modify default configuration settings to adapt Oracle Analytics Server to your environment and user needs.

This part includes the following chapters:

10

Use Tools to Manage and Configure the System

This chapter introduces the tools to manage and configure the system.

Topics:

Why Use Oracle Analytics Server Console?

You can use the Oracle Analytics Server Console to manage parts of the system.

The Console enables you to perform some common administration tasks as well as configure more advanced system settings. For example, you can register a safe domain or enable the cache. See Get Started to Manage the System and Perform Common Administration Tasks.

On the Home page click the **Navigator** and select **Console**.

Why Use Fusion Middleware Control and WebLogic Server Administration Console?

You can use Fusion Middleware Control and WebLogic Server Administration Console to manage the system.

These Web-based tools support the most common system administration tasks. See Get Started to Manage the System.

Fusion Middleware Control enables you to manage system components by performing tasks such as monitoring status, starting and stopping processes, resolving issues, and configuring components. You can also manage some aspects of Java components. For example, you can monitor their status and start and stop them.

WebLogic Server Administration Console enables you to monitor status and configure security for Java components. See Introduction to System Administration in Oracle Analytics Server.

Locking Mechanism Enables Multiple Concurrent Administrators

With large deployments, you might have multiple administrators accessing the system concurrently to view the state of the system while other administrators might want to make configuration changes. Fusion Middleware Control and Oracle WebLogic Server prevent multiple administrators making concurrent updates of the same configuration settings by using a locking mechanism that enables only one administrator to make changes at any one time.

> **Note:**
>
> Multiple administrators using the same administrator account could unknowingly make concurrent updates of the same configuration settings. It's therefore recommended that multiple administrator users don't share the same administrator account.

Manage System Components Using Fusion Middleware Control

You can use Fusion Middleware Control to manage, monitor, and configure system components (for example, the BI Server, Presentation Services, and Oracle BI Scheduler). You can also use Fusion Middleware Control to manage the Administration Server and Managed Servers.

This section contains the following topics:

- Log into Fusion Middleware Control
- Display Oracle Analytics Server Pages in Fusion Middleware Control
- Use the Navigation Tree in Fusion Middleware Control
- Tips for Using Fusion Middleware Control

Log into Fusion Middleware Control

To log in to Fusion Middleware Control, open a web browser and enter the Fusion Middleware Control URL.

Enter the URL in the following format:

```
http://hostname.domain:port/em
```

The port number is the number of the Administration Server, and the default port number is 9500.

Fusion Middleware Control is available only if the Administration Server is running, as described in Conditions to Start the System.

1. Enter the URL in a web browser. For example:

   ```
   http://host1.example.com:9500/em
   ```

2. Enter the system administrator user name and password and click **Sign in**.

   This systemwide administration user name and password was specified during the installation process, and you can use it to log in to WebLogic Server Administration Console, Fusion Middleware Control, and Oracle Analytics Server.

   Alternatively, enter any other user name and password that has been granted the Oracle BI Administrator application role. Fusion Middleware Control is displayed.

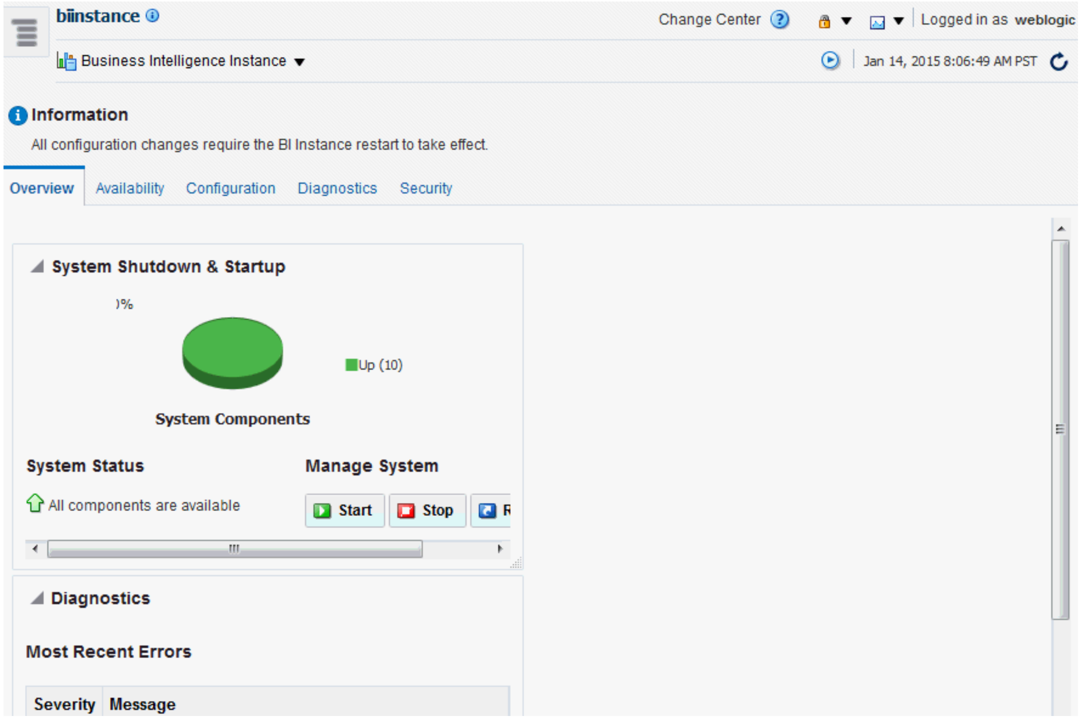
> **Note:**
>
> If you have the browser configured to send HTTP requests to a proxy server, then you might have to configure the browser to not send Administration Server HTTP requests to the proxy server. If the Administration Server is on the same computer as the browser, then ensure that requests that are sent to localhost or 127.0.0.1 are not sent to the proxy server.

Display Oracle Analytics Server Pages in Fusion Middleware Control

Use this topic to display Oracle Analytics Server pages that enable you to manage system components.

1. Log in to Fusion Middleware Control.
2. Expand the **Business Intelligence** folder and select the **biinstance** node.

   Fusion Middleware Control displays the Overview page, as shown below.



Information

All configuration changes require the BI Instance restart to take effect.

All components are available

> **Note:**
>
> If the Business Intelligence folder isn't visible or there is no **biinstance** node under it, then Oracle Analytics Server system components haven't been installed. See *Installing and Configuring Oracle Analytics Server*.

The Overview page displays the current system status, by providing information about availability, and issues identified within the BI domain (see What Is the Logical System

Architecture?). The Overview page also enables you to start and stop Oracle Analytics Server.

3. From the Overview page, select an appropriate tab to perform management tasks.

Use the Navigation Tree in Fusion Middleware Control

The navigation tree enables you to navigate and select nodes within the BI domain that can be managed by Fusion Middleware Control.

Depending on the choices made during installation, the following domain components can be displayed as nodes in the navigation tree:

- **Application Deployments**

  The Application Deployments node shows all the applications that are deployed into the BI domain (for example, analytics, Oracle Analytics Server for Microsoft Office, Publisher).

- **WebLogic Domain**

  These nodes display summary information for the WebLogic server. Select a node and click the **Oracle WebLogic Server Administration Console** menu option to display the WebLogic Server Administration Console, where you can administer Oracle WebLogic Server.

  - **bidomain**

    This node represents the WebLogic server domain for Oracle Analytics Server with an AdminServer node that contains the Administration Server and a bi_cluster node that contains Managed Servers (a single node cluster by default, for example, bi_server1). See About the Administration Server, Managed Servers, and System Components.

    - **AdminServer**
    - **bi_cluster**

- **Business Intelligence**

  - **biinstance**

    This node represents the Oracle Analytics Server system components that can be managed using Fusion Middleware Control.

    Select this node to display the Overview page and manage the system components.

- **Semantic Models**

  This node represents the Metadata Services (MDS) schema semantic models that can be managed using Fusion Middleware Control.

Tips for Using Fusion Middleware Control

There are several considerations to keep in mind when using Fusion Middleware Control with Oracle Analytics Server.

Keep the following tips in mind:

- See Getting Started Managing Oracle Fusion Middleware in *Administering Oracle Fusion Middleware*.
- You might want to have a user who can view information about Oracle Analytics Server within Fusion Middleware Control but not make any changes. You can configure such a user by making them a member of the Monitors group. See *Securing Resources Using Roles and Policies for Oracle WebLogic Server* .

Methods for Configuring System Settings

You can use various methods to configure system settings by changing values that are stored in domain-specific locations.

The system settings are related to either functional behavior (for example, cache, thresholds), or environmental settings (for example, host names, ports, files or metadata locations).

Use the following methods:

- Use the Oracle Analytics Server Console
- Use Fusion Middleware Control
- Use a Text Editor
- About the Scripts for Managing Service Instances

The table below shows which method to use when configuring system settings. Each method updates settings in specific configuration files.

What Do You Want to Do?	What Methods Can You Use?	How Are Updates Made?
Change common configuration settings in an easy to use user interface.	Oracle Analytics Server Console See Use the Oracle Analytics Server Console. Oracle recommends that you use this method. However, if a setting isn't available, you can use Fusion Middleware Control, see Use Fusion Middleware Control, or a text editor.	Change values in specific Oracle Analytics Server configuration pages in the Oracle Analytics Server Console . For example, to enable the cache, you click **Cache Enable** in the System Settings option in the Console.
Change configuration settings by manually editing a file.	Text editor See Use a Text Editor. Oracle recommends that you use this method when a setting isn't available in Fusion Middleware Control Oracle Analytics Server pages.	Change values in a configuration text file using a text editor.
Make more complex configuration changes using scripts.	Scripts See About the Scripts for Managing Service Instances. Oracle recommends that you use this method when instructed by the documentation.	Make configuration changes by running commands using scripts.

Use the Oracle Analytics Server Console

Use Console to set advanced system settings for Oracle Analytics Server.

1. Click **Console**.
2. Click **System Settings**.
3. Update the property value.
4. If required, click **Restart** and then click **OK** to confirm.

   Wait a few moments for the changes to refresh through the system.

Use Fusion Middleware Control

You can use Fusion Middleware Control to update specific configuration settings.

Configuration settings you can change include performance settings, dashboard and analysis default presentation settings, and mail server settings used by agents.

If a configuration setting isn't available in Fusion Middleware Control, you can use a text editor to update the setting in a configuration file. See Methods for Configuring System Settings.

1. Go to the Overview page, as described in Display Oracle Analytics Server Pages in Fusion Middleware Control.
2. Select the appropriate page and tabs to display the settings to change.
3. Click **Lock and Edit** to enable changes to be made.

> **Caution:**
>
> Oracle recommends that multiple administrators don't share the same administrator account. They can unknowingly make concurrent updates to the same configuration settings.

4. Make the appropriate changes on each page.
5. Click **Apply** on each page after you have made your changes.
6. When you have finished making your changes, do one of the following:
   - Click **Activate Changes** to process your changes and release the lock to enable another system administrator to make changes.
   - Click **Release Configuration** to undo all changes you made since clicking **Lock and Edit** and release the lock to enable another system administrator to make changes.
7. After you have activated your changes, go to the Overview page and click **Restart**.

Use a Text Editor

You can use a text editor to add or change a setting in a configuration file.

You would use text editor for the system configuration settings that are not available in Fusion Middleware Control.

Use the following procedure to update configuration files using a text editor.

1. Make a backup copy of the files that you plan to edit.
2. Open the appropriate configuration file in a text editor.

   See Configuration Files
3. In the configuration file, locate the element or create a new element if you must add a setting to the file.
4. Enter the appropriate changes.
5. Save your changes and close the configuration file.
6. Restart the system in Fusion Middleware Control, go to the Process tab in the Availability page, and restart all components.

See About Managing Processes.

Use Scripts

Oracle provides scripts to perform Oracle Analytics Server configuration tasks. For example, to create a domain during installation, or to add a machine for high availability.

To run scripts see Use Scripts to Manage Oracle Analytics Server Instances:

Update the Java Development Kit (JDK)

After you install and configure Oracle Analytics Server, you might need to update the JDK for the instance; for example, if an update is required per the policy of your organization.

Before deciding to update the JDK, ensure that you consider an appropriate version.

1. Stop all services for Oracle Analytics Server.
2. Download the appropriate JDK version from the Oracle Java web site and copy it to the *ORACLE_HOME* directory.
3. Rename the existing jdk directory to jdk.OLD.
4. Run the JDK Installer, which unzips the distribution into the jdk*version-num* directory.
5. Rename the directory from jdk*version-num* to jdk, to ensure that all existing configuration references remain valid.
6. Restart the services for Oracle Analytics Server.

For information on installing with a specific JDK, see *Installing and Configuring Oracle Analytics Server*.

Manage Java Components in the Oracle WebLogic Server Administration Console

You use the Oracle WebLogic Server Administration Console to manage Oracle Analytics Server Java components.

You display Oracle WebLogic Server Administration Console, using the following methods:

- Clicking a link on the WebLogic Domain menu in Fusion Middleware Control
- Entering a URL into a web browser window

The Oracle WebLogic Server Administration Console is available only if the Administration Server for WebLogic Server is running. See About Managing Processes.

1. If the Administration Server for WebLogic Server isn't running, start it.
2. Display the Oracle WebLogic Server Administration Console using the following methods:

   Clicking a link on the Overview page in Fusion Middleware Control:

   a. Display Fusion Middleware Control.
   b. Click the Oracle WebLogic Server Administration Console link in the WebLogic Domain menu.

      The Oracle WebLogic Server Administration Console login page is displayed.

   Using a URL in a web browser window:

   a. Start a web browser.
   b. Enter the following URL into the browser:

      `http://<hostname>:<port>/console/`

      For example, `http://example.com:9500/console/`

      where *hostname* is the DNS name or IP address of the Administration Server and *port* is the listen port on which the Administration Server is listening for requests (port 9500 by default). If you've configured a domain-wide Administration port, then use that port number. If you configured the Administration Server to use Secure Sockets Layer (SSL), then you must add the letter 's' after http as follows:

      `https://<hostname>:9500/console/`

      The preceding URL example uses SSL.

      The Oracle WebLogic Server Administration Console login page is displayed.

3. Enter the system administrator user name and password and click **Login**.

   This systemwide administration user name and password was specified during the installation process, and you can use it to log in to WebLogic Server Administration Console, Fusion Middleware Control, and Oracle Analytics Server. Alternatively, enter a user name that belongs to one of the following security groups:

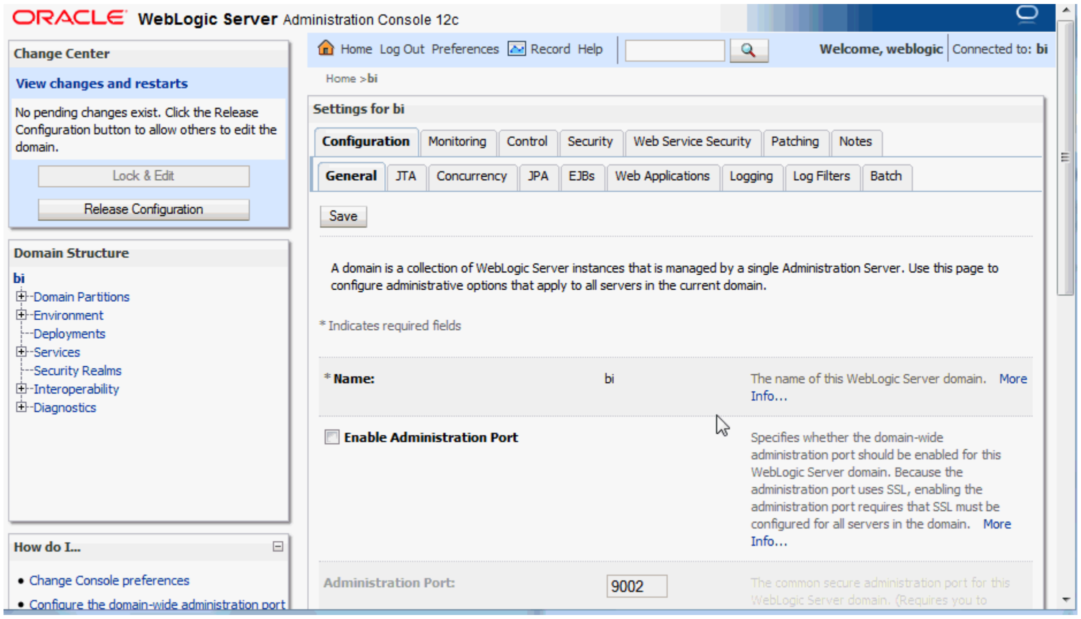
   - Administrators
   - Operators
   - Deployers
   - Monitors

   These groups provide various levels of access to system administration functions in the Oracle WebLogic Server Administration Console.

   Using the security system, you can add to or delete users from one of these groups to provide controlled access to the Console.

   If the browser is configured to send HTTP requests to a proxy server, then you might have to configure the browser to not send Administration Server HTTP requests to the proxy. If the Administration Server is on the same computer as the browser, then ensure that requests sent to localhost or 127.0.0.1 aren't sent to the proxy.

   In Oracle WebLogic Server Administration Console you select the bi domain, as shown in the figure below.


ORACLE WebLogic Server Administration Console 12c
Change Center
View changes and restarts
Lock & Edit
Release Configuration
Home Log Out Preferences Record Help
Welcome, weblogic Connected to: bi
Home >bi
Settings for bi
Configuration Monitoring Control Security Web Service Security Patching Notes
General JTA Concurrency JPA EJBs Web Applications Logging Log Filters Batch
Save
Domain Structure
bi
Domain Partitions
Environment
Deployments
Services
Security Realms
Interoperability
Diagnostics
How do I...
Change Console preferences
Configure the domain-wide administration port

No pending changes exist. Click the Release Configuration button to allow others to edit the domain.

A domain is a collection of WebLogic Server instances that is managed by a single Administration Server. Use this page to configure administrative options that apply to all servers in the current domain.

* Indicates required fields

* **Name:**	bi	The name of this WebLogic Server domain. More Info...
□ **Enable Administration Port**		Specifies whether the domain-wide administration port should be enabled for this WebLogic Server domain. Because the administration port uses SSL, enabling the administration port requires that SSL must be configured for all servers in the domain. More Info...
Administration Port:	9002	The common secure administration port for this WebLogic Server domain. (Requires you to

You can monitor and manage Oracle Analytics Server Java components from this page.

> **Note:**
>
> For more information on using the Oracle WebLogic Server Administration Console, see the *Oracle WebLogic Server Administration Console Online Help* system. That Help system describes how to use the console to override the context root for a deployed web application. Changing any context root for Oracle Analytics Server isn't supported, because many context roots are used for internal links and end-user end points.

11

Use Scripts to Manage Oracle Analytics Server Instances

This chapter describes the scripts that you can use to manage Oracle Analytics Server Instances.
Topics:

- About the Scripts for Managing Service Instances

About the Scripts for Managing Service Instances

You can use the scripts listed here to manage a service instance in a domain.

A service instance contains all of the Oracle Analytics Server metadata (that is, semantic model data, presentation catalog, security policy), and includes the customizations that you make to the metadata.

Use the following scripts located in `user_projects/domains/bi/bitools/bin`.

Command	Description
datamodel.sh|cmd	Use this script to download the semantic model from a BAR archive file for a service instance.
diagnostic_dump.sh|cmd	Use this script to collect diagnostic bundles needed by Oracle Support to help resolve issues.
exportarchive.sh|cmd	Use this script to export the metadata of your environment in the form of a BAR archive file.
grantPermissionSetsToBIRole.sh|cmd and revokePermissionSetsFromBIRole.sh|cmd	Use these scripts to make any adjustments to permissions for application roles.
importarchive.sh|cmd	Use this script to import an exported BAR archive file to replace metadata in the target environment.
runcat.sh|cmd	Use this script to start the catalog manager command line interface.
jobmanager.sh|cmd	Use this script to start the job manager command line interface.
start.sh|cmd	Use this script to start all component processes in a domain.
status.sh|cmd	Use this script to view the status of component processes in a domain.
stop.sh|cmd	Use this script to stop all component processes in a domain.
sync_midtier_db.sh|cmd	Use this script to re-synchronize connection details to the mid-tier database when connection details are changed. On Linux you perform this on the primary host. On Windows you perform this on every host.

Assumptions for all scripts against service instances and BAR files:

- You must have file system (offline) permissions.
- You run the scripts offline.
- You must start your system after making changes to service instances through scripts.

12 Configure Connections to External Systems

This chapter describes how to configure connections to systems that are external to Oracle Analytics Server.

Topics:

- Configure Email and Agents
- Configure for Actions with the Action Framework
- Configure Connections to Data Sources

Configure Email and Agents

You can use the **Mail Settings** page in the Console to configure a SMTP mail server so analysts can email their reports and data visualizations directly from Oracle Analytics. You can also use the **System Settings** page in the Console to customize Email Delivered by Agents Options.

Alternatively, you can use Fusion Middleware Control to configure a SMTP mail server and common email settings that are used only by agents. Advanced configuration settings for agents are described in Configure and Manage Agents.

Use Console to Configure a SMTP Mail Server to Deliver Reports

Connect to your organization’s mail server, so analysts can email their reports and data visualizations directly from Oracle Analytics.

1. In the Oracle Analytics Home page, click the **Navigator**, and then click **Console**.
2. Click **Mail Settings**.
3. Enter the name of the **SMTP Server** you want to use to deliver emails.

   For example, `mymail.example.com`.

   If your email server has a public IP address, you can enter the public IP address here instead of the server name.
4. Enter the **Port** number.

   Common SMTP ports include:

   - `25` (**Connection Security** = None)
   - `465` (**Connection Security** = SSL/TLS)
   - `587` (**Connection Security** = STARTTLS)
5. Enter the name and email address that you want to see in the “From” field of emails delivering reports (**Display name of sender** and **E-mail address of sender**).

   For example, `Joe Brown` and `joseph.brown@example.com`.
6. Click **Test** to verify the connection.

If you want to test the connection you must do so before you configure any security settings.

> **Note:**
>
> You can click **Delete** at any time to clear all the mail server settings and start again.

7. Optional: If the mail server requires authentication:
   a. Select **Authenticated**.
   b. Enter the **Username** and **Password** for a user with access to the mail server.

      The password must be between 8 and 255 characters.
8. Optional: To set up a secure mail server:
   a. Click **Connection Security**, and select the appropriate security protocol for your mail server.
      - **SSL/TLS**: Select if your mail server uses SSL or TLS. The port value defaults to 465.
      - **STARTTLS**: STARTTLS is a way to take an existing insecure connection and upgrade it to a secure connection using SSL or TLS. The port value defaults to 587.

      In **TLS Certificate**, the **Default Certificate** is selected for you. The default certificate allows encrypted mail server communication. In most cases, you don’t need to provide a compatible certificate as most mail servers can use the default certificate, including Office 365.
   b. Optional: Upload a custom TLS certificate. In **TLS Certificate**, select **Custom Certificate** , and then click **Select** to navigate to the certificate file (.pem).

      If you haven’t configured a virus scanner, you're prompted to configure one now or proceed without a virus scanner.
9. Click **Save**.

   Allow some time for your changes to refresh through your system and Email menu options to display.

Control Who Can Deliver Content (or Links to Content) by Email

By default, everyone can deliver and receive content by email after you set up the email server. If you want to restrict email delivery, you can grant the **Deliver Content in Email** privilege only to specific users and application roles or you can disable the feature for everyone.

The option to deliver *links* to content in email isn't enabled by default. So if you want to allow users to email content links, you must specifically grant the **Deliver Content Link in Email** privilege to one or more users or application roles.

> **Note:**
>
> The **Deliver Content in Email** and **Deliver Content Link in Email** privileges apply only to content you create from the Classic home page, that is, analyses, dashboards, pixel-perfect reports, briefing books, and so on.

1. On the Classic Home page, click the user profile icon, and then click **Administration**.
2. Click **Manage Privileges**, and navigate to the **Delivers** section.
3. To restrict who can send and receive content by email, configure the **Deliver Content in Email** privilege.
   - Select specific users and application roles, and set the permission to **Granted** (or **Denied**).
   - **Authenticated User -** Set to **Denied** to prevent access to everyone.

   Only users granted the **Deliver Content in Email** privilege see the **Email** option.
4. To allow users to send content links by email, configure the **Deliver Content Link in Email** privilege.
   - Select specific users and application roles, and set the permission to **Granted** (or **Denied**).
   - **Authenticated User -** Set to **Granted** to allow everyone to deliver content links.

   Only users granted the **Deliver Content Link in Email** privilege see the **Deliver link to results** option when configuring delivery agents.

Use Fusion Middleware Control to Configure Scheduler Email Settings that Affect Agents

Configuring email settings that affect agents ensures users and other notification recipients receive messages appropriately.

Before you begin this procedure, ensure that you're familiar with Fusion Middleware Control.

1. Go to the Overview page.
2. Display the Mail tab of the Configuration page as appropriate.
3. Click **Lock and Edit** to enable changes to be made.
4. Complete the following options:
   - **SMTP Server**
   - **Port**
   - **Display name of sender**

     This option is used in the SMTP From field as a meaningful substitution for the sender's address.
   - **Email address of sender**

     This option specifies the email address on the SMTP Server that's used as the sender's reply-to address for all mail sent from the Scheduler. The initial value is defaultuser@defaultmailserver.com, which you must change to reflect a valid email address. Note that if you want to indicate that email recipients needn't reply, add `no_reply@mycompany.com` or `do_not_reply@mycompany.com` to this field.

- **Username**
- **Password**
- **Confirm password**
- **Number of retries upon failure**
- **Maximum recipients**
- **Addressing method** To, Blind Copy Recipient (Bcc)
- **Connection Security**
- **Specify CA certificate source**
- **CA certificate directory**
- **CA certificate file**

  You can use the default CA certificate provided with the Oracle Analytics Server installation, to configure SMTP through SSL. The CA certificate is located in `ORACLE_HOME/bi/modules/oracle.bi.publictrust/openssl/cacerts.crt`.
- **SSL certificate verification depth**
- **SSL cipher list**

5. Click **Apply**, then click **Activate Changes**.
6. Return to the Overview page and click **Restart**.

For information about advanced configuring settings for agents, see Configure and Manage Agents.

For information about corresponding configuration file elements, see Map User Interface Labels with Configuration File Elements.

Configure for Actions with the Action Framework

You can run scripts for the configuration of actions in your organization.

Users can create actions in the Presentation Services user interface. An action is an operation or process invoked explicitly by a user clicking a link, or invoked automatically, as the final step of an agent. See Make Analyses Interactive and Create Agents to Deliver Content.

For a comprehensive discussion of how to use the Action Framework to enable actions for external systems, including descriptions of configuration settings and detailed examples, see "Using Actions to Integrate Oracle BI EE with External Systems" and "Sample Files" in *Integrator's Guide for Oracle Business Intelligence Enterprise Edition*.

Note:

If you use the *Integrator's Guide for Oracle Business Intelligence Enterprise Edition* when you configure the Action Framework, you must take into account the following differences between Oracle Analytics Server and Oracle BI EE.

- In Oracle Analytics Server, you use scripts to configure the Action Framework. The scripts extract, create, or update the Action Framework configuration, and load the `ActionFrameworkConfig.xml` and policy file to a specified location.
- In Oracle BI EE you don't use scripts to configure the Action Framework, instead you manually configure the `ActionFrameworkConfig.xml` file and policy file, and save them to a specific location, for example, `<Oracle Middleware Home>\user_projects\domains\bifoundation_domain\config\fmwconfig\biinstances\coreapplication`.

Topics:

- Modify the Actions Registry Configuration
- Modify the Actions Policy Configuration
- About Scripts to Retrieve, Create, or Update the Actions Framework Configuration

Modify the Actions Registry Configuration

Run scripts to update the actions registry configuration.

See Script to Extract the Current Actions Registry Configuration and Script to Update the Current Actions Registry Configuration File.

1. Run the script `extract_actions_registry_config.sh|.cmd` located in `/Oracle/Middleware/Oracle_Home/user_projects/domains/bi/bitools/bin` to extract the current `ActionFrameworkConfig.xml` file.

   The actions registry configuration is extracted to the location that you specify.
2. Make your changes to the extracted `ActionFrameworkConfig.xml` file.
3. Save your changes.
4. Run the script `update_actions_registry_config.sh|.cmd` located in `/Oracle/Middleware/Oracle_Home/user_projects/domains/bi/bitools/bin` to upload your changes to the service instance.
5. Restart Oracle Analytics Server.

   See Start and Stop Your System .

Modify the Actions Policy Configuration

Run scripts to update the actions policy configuration.

See Script to Extract the Actions Policy Configuration File and Script to Create or Update the Actions Policy Configuration File.

1. Run the script `extract_actions_policyRefFile.sh|.cmd` located in `/Oracle/Middleware/Oracle_Home/user_projects/domains/bi/bitools/bin` to extract the current `ActionsPolicyFile.xml` file.

The actions registry configuration is extracted to the location that you specify.

2. Make your changes to the extracted `ActionsPolicyFile.xml` file.
3. Save your changes.
4. Run the script `create_or_update_actions_policyRefFile.sh|.cmd` located in `/Oracle/Middleware/Oracle_Home/user_projects/domains/bi/bitools/bin` to upload your changes to the service instance.
5. Restart Oracle Analytics Server.

   See Start and Stop Your System .

About Scripts to Retrieve, Create, or Update the Actions Framework Configuration

You can use scripts to retrieve, create, or update the actions framework registry and policy configuration.

Topics:

- Script to Extract the Current Actions Registry Configuration
- Script to Update the Current Actions Registry Configuration File
- Script to Extract the Actions Policy Configuration File
- Script to Create or Update the Actions Policy Configuration File

Script to Extract the Current Actions Registry Configuration

Use this script to extract the current actions registry configuration on the service instance. If the registry isn't configured, the script doesn't return any content.

extract_actions_registry_config.sh|.cmd

Usage: `/Oracle/Middleware/Oracle_Home/user_projects/domains/bi/bitools/bin/extract_actions_registry_config.sh|cmd [-h domainHome] -a actionFrameworkRegistryConfigDestinationFilePath -k serviceInstanceKey`

`-h`: Use to specify the domain home including the target `domainName` directory, which in this case is `/bi`. The default is `DOMAIN_HOME` if set, otherwise it's `/Oracle/Middleware/Oracle_Home/user_projects/domains/bi`

`-a`: Use to specify the destination file path of the actions framework registry configuration.

`-k`: Use to specify the service instance key.

The script extracts the existing actions framework registry to a specified actions framework registry configuration destination file.

Example: `/Oracle/Middleware/Oracle_Home/user_projects/domains/bi/bitools/bin/extract_actions_registry_config.sh|.cmd -a /MyUserName/ActionsFrameworkRegistry.xml -k myService1`

When this script successfully executes, you can find the current configured actions registry in the `/MyUserName/ActionsFrameworkRegistry.xml` file.

Script to Update the Current Actions Registry Configuration File

Use this script to update the current actions registry configuration file with content that you specify in a source actions registry file.

If the actions registry configuration file isn't available in the service instance, the script creates the registry file with the source actions registry file content.

update_actions_registry_config.sh|.cmd

Usage: `/Oracle/Middleware/Oracle_Home/user_projects/domains/bi/bitools/bin/update_actions_registry_config.sh|.cmd [-h domainHome] -a actionFrameworkRegistryConfigDestinationFilePath -k serviceInstanceKey`

`-h`: Use to specify the domain home including the target `domainName` directory, which in this case is `/bi`. The default is `DOMAIN_HOME` if set, otherwise it's `/Oracle/Middleware/Oracle_Home/user_projects/domains/bi`

`-a`: Use to specify the destination file path of the actions framework registry configuration.

`-k`: Use to specify the service instance key.

The script updates the existing actions framework registry with the content of specified actions framework registry configuration source file.

Example: `/Oracle/Middleware/Oracle_Home/user_projects/domains/bi/bitools/bin/update_actions_registry_config.sh|.cmd -a /MyUserName/ActionsFrameworkRegistry.xml -k myService1`

When this script successfully executes, content from the `/MyUserName/ActionsFrameworkRegistry.xml` file is pushed or updated to the service instance as an actions registry configuration.

Script to Extract the Actions Policy Configuration File

Use this script to extract the actions policy configuration file if it's available in the service instance. If the policy file isn't available the script doesn't return any content.

extract_actions_policyRefFile.sh|.cmd

Usage: `/Oracle/Middleware/Oracle_Home/user_projects/domains/bi/bitools/bin/extract_actions_policyRefFile.sh|.cmd [-h domainHome] -a actionsPolicyRefFileDestinationPath -p policyFileRefName -k serviceInstanceKey`

`-h`: Use to specify the domain home including the target `domainName` directory, which in this case is `/bi`. The default is `DOMAIN_HOME` if set, otherwise it's `/Oracle/Middleware/Oracle_Home/user_projects/domains/bi`

`-a`: Use to specify the destination file path of the actions framework policy configuration.

`-p`: Use to specify the policy reference file name.

`-k`: Use to specify the service instance key.

The script extracts the existing actions policy reference file to a specified `actionsPolicyFileDestinationPath`.

Example: `/Oracle/Middleware/Oracle_Home/user_projects/domains/bi/bitools/bin/extract_actions_policyRefFile.sh|.cmd -a /MyUserName/ActionsPolicyFile.xml -p ActionsPolicyFile.xml -k myService1`

When this script successfully executes, you can find the policy file content for the specified policy file name in the `/MyUserName/ActionsPolicyFile.xml` file.

Script to Create or Update the Actions Policy Configuration File

Use this script to create or update the existing actions policy file content with content that you specify in a source actions policy file.

If the actions policy file isn't available as a key in the service instance, the script creates the actions policy file with the source actions policy file content.

create_or_update_actions_policyRefFile.sh|.cmd

Usage: `/Oracle/Middleware/Oracle_Home/user_projects/domains/bi/bitools/bin/create_or_update_actions_policyRefFile.sh|.cmd [-h domainHome] -a actionsPolicyRefFileSourcePath -p policyFileRefName -k serviceInstanceKey`

`-h`: Use to specify the domain home including the target `domainName` directory, which in this case is `/bi`. The default is `DOMAIN_HOME` if set, otherwise it's `/Oracle/Middleware/Oracle_Home/user_projects/domains/bi`

`-a`: Use to specify the destination file path of the actions policy reference file source path.

`-p`: Use to specify the policy reference file name.

`-k`: Use to specify the service instance key.

The script extracts the existing actions policy reference file to a specified `actionsPolicyFileDestinationPath`.

Example: `/Oracle/Middleware/Oracle_Home/user_projects/domains/bi/bitools/bin/create_or_update_actions_policyRefFile.sh|.cmd -a /MyUserName/ActionsPolicyFile.xml -p ActionsPolicyFile.xml -k myService1`

When this script successfully executes, the actions policy file content from the `/MyUserName/ActionsPolicyFile.xml` file is updated to the service instance as a value for the key `ActionsPolicyFile.xml` file.

Configure Connections to Data Sources

How you create and configure data source connections depends on the data modeling tool you're using to create the semantic model.

In Model Administration Tool, you define data source connections by importing metadata and configuring connection pools. See Import Metadata and Work with Data Sources.

For Model Administration Tool, you might need to update connection pool information in the semantic model during migrations to production and other environments. You can use the Oracle BI Server XML API to automate these connection pool changes. See Moving from Test to Production Environments.

In Semantic Modeler, you define data source connections before you import metadata into the model. See About Connections for Semantic Models.

13

Configure Presentation Setting Defaults

This chapter describes how to change default presentation settings in Oracle Analytics Server that administrators commonly change using Fusion Middleware Control.
Advanced configuration settings are described in Manually Change Presentation Settings.

Use Fusion Middleware Control to Change Presentation Setting Defaults

Changing presentation setting defaults ensures that users begin with the same look and feel when customizing their own presentations.

Before you begin this procedure, ensure that you are familiar with the information in Use Fusion Middleware Control.

1. Go to the Overview page.
2. Display the Presentation tab of the Configuration page.
3. Click **Lock and Edit** to enable changes to be made.
4. Complete the following options:
   - **Show page tabs** option
   - **Show section headings** option
   - **Allow dashboard sections to be collapsible** option
   - **Pivot Tables show auto-preview** option
5. Click **Apply**, then click **Activate Changes**.
6. Return to the Overview page and click **Restart**.

For information about advanced configuring settings for analyses and dashboards, see Configure and Manage Analyses and Dashboards.

For information about corresponding configuration file elements, see Map User Interface Labels with Configuration File Elements.

14

Configure Map and Spatial Information

This chapter describes how to configure map and spatial information for map views in Model Administration Tool.
When you install Model Administration Tool, you install functionality for users to see maps that display data. Before users can see maps in analyses and dashboards, you must understand the system requirements, specify layers and maps, and configure the metadata, as described in this chapter.

Topics:

- What Are the System Requirements for Map Views?
- Hardware Sizing and Deployment Strategy for Maps
- Manage Map Information for Visualizations
- Manage Map Information for Analyses

See Configure Advanced Options for Maps and Spatial Information for information about advanced configuration options for maps.

What Are the System Requirements for Map Views?

Several components need to be configured so that map views can be added to dashboards.

To include map views on dashboards, the system must include the following components:

- Oracle MapViewer, which is a J2EE service that serves as an engine for rendering maps using spatial data managed by Oracle Spatial. MapViewer is closely integrated with Oracle Analytics Server. MapViewer is installed as part of Oracle Analytics Server and deployed in the same domain as Oracle Analytics Server on the web application server.

  MapViewer provides services and tools that hide the complexity of spatial data queries and cartographic rendering, while providing customizable options for more advanced users. MapViewer is designed to integrate with Location-Based services and applications.

  See Configure MapViewer to Support Map Views.

- Spatial boundary data. NAVTEQ is one provider of this data to Oracle customers, which can be downloaded from the Oracle Technology Network. This spatial data and any other spatial metadata, including themes and styles, must be stored in an Oracle Database to be accessed by Oracle MapViewer for display in map views.

- Hosted maps. In Oracle Analytics Server, users can access hosted maps from the Oracle eLocation service. Terms and conditions of use are located at the following URL:

  http://elocation.oracle.com/elocation/legal.html

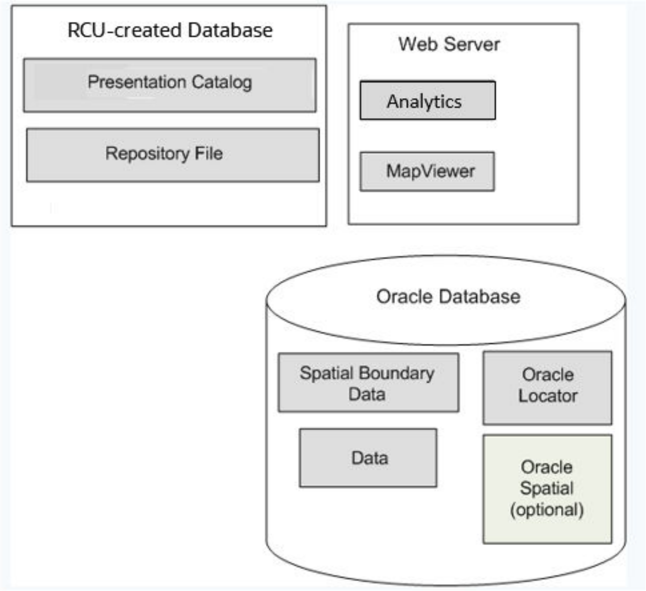
- Oracle Database, version 10*g* or later, to store the spatial data.

  Oracle Locator, which is a feature of Oracle Database (all editions) that provides core location functionality needed by most customer applications.

  If you use an Oracle Database as the Repository Creation Utility (RCU) database, then you can use that same Oracle Database for spatial data also. See *Installing and Configuring Oracle Analytics Server* for information.

- (Optional) Oracle Spatial is an option for Oracle Database Enterprise Edition that provides advanced spatial features to support high-end geographic information systems (GIS) and location-based services (LBS) solutions. The Spatial option is required only if you plan to make use of maps or features that require advanced spatial capabilities that are not available with the Locator option. Additional Oracle Spatial features include raster and 3D data management, spatial web services, topology, network data modeling, resource description framework (RDF), and semantic web capabilities.
- Metadata of the mapping between Oracle Analytics Server data and spatial data in the Presentation Catalog.

The illustration shows the default architecture for map views when Oracle Analytics Server is installed. You can store the data either in an Oracle Database or in other databases that Oracle Analytics Server supports. See Configure MapViewer to Support Map Views for a diagram of the preferred architecture for map views.



When these pieces are in place, you administer the map using the Oracle BI Presentation Services Administration page, as described in Manage Map Information for Analyses.

Hardware Sizing and Deployment Strategy for Maps

Rendering map views is computationally more intensive than rendering tabular views.

Map rendering entails the following, which explains why it requires greater computational resources:

- Query spatial data.
- Create the polygons and shapes that correspond to geographical entities such as countries and states.
- Place the polygons and shapes on a background map.
- Provide end-user interactivity such as the ability to pan and zoom, to adjust color thresholds, and to show or hide formats.

Assess the extent of expected usage of map views at your organization including the number of users that are expected to use map views, the amount of data to be displayed on map views, and the amount of spatial data that is expected to be displayed (such as only city boundaries or street level details). Based on this assessment, decide on an appropriate hardware sizing and deployment strategy. Also review the available documentation on best practices for achieving optimal performance and scalability with your Oracle MapViewer deployment.

Manage Map Information for Visualizations

You can use the Console to make maps and map backgrounds available to users, and add and update custom map layers so that users can visualize and interact with data through maps.

See the following in *Visualizing Data in Oracle Analytics Server*:

- Make Maps Available to Users
- Make Map Backgrounds Available to Users
- Add Custom Map Layers
- Update Custom Map Layers

Manage Map Information For Analyses

This chapter describes how you set up map information for dashboards and analyses, so that users can visualize and interact with data through maps.

Topics:

- Set Up Maps For Dashboards and Analyses
- Edit Background Maps For Dashboards and Analyses

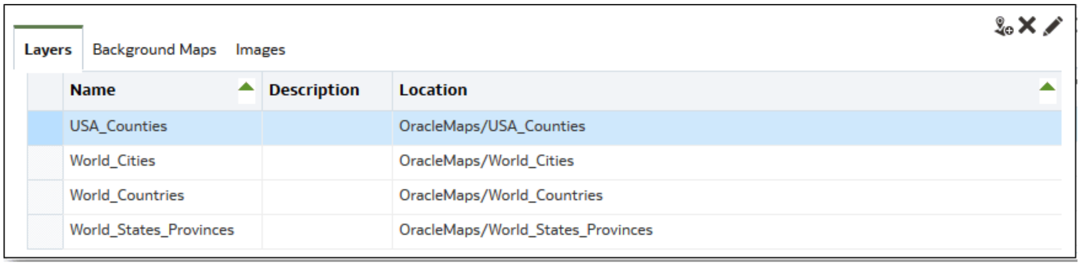
Set Up Maps For Dashboards and Analyses

As the administrator, you define how data columns that you've modeled are displayed on maps. You configure the map data, then users can analyze the data in map views.

Map views allow users to display data on maps in different formats and to interact with data. As the administrator, you must configure the metadata that defines the mapping between business intelligence data and spatial data.

Spatial features such as shape definitions are managed by database administrators for your instance. If a shape geometry definition doesn’t exist for a particular column value, then the shape can't be shown on the map and might affect user interactions on the map.

1. On the Classic Home page, click the user profile icon, **Administration**, and then click **Manage Map Data**.
2. On the **Layers** tab, click **Import Layers** from the toolbar.



Name	Description	Location
USA_Counties		OracleMaps/USA_Counties
World_Cities		OracleMaps/World_Cities
World_Countries		OracleMaps/World_Countries
World_States_Provinces		OracleMaps/World_States_Provinces

3. In the Import Layers dialog, select the layers you want to use and click **OK**.
4. Back on the Layers tab, select a layer and click the **Edit Layers** button.
5. In the Edit Layer dialog, associate layers with columns so that users can display data in the map view.
   a. In **Name**, specify the layer name to display to users who work with map views.
   b. In **Location**, specify which background map the layer originates from. Click **Location** to select a different layer.
   c. In **Description**, specify information to help users when they hover over the layer name in the Map Formats area.
   d. In **Layer Key**, specifiy the column of spatial data that you can associate with data. Each column value corresponds to a "shape" that originates from the background map. For example, a MY_CITIES layer might have a layer key called CITY. The default value is a "best guess". Select the appropriate column from the list.

      There are various reasons why a country such as Mexico might be drawn as a white area on a map:

      - The column has a null value for the country of Mexico, but a shape exists for Mexico in the spatial column.
      - The column has a value for the country of Mexico, but no shape exists for Mexico in the spatial column.
      - The column has a value for the country of Mexico and the shape exists for Mexico in the spatial column, but the names are mismatched. The data columns might have the value MEX and the spatial column might have MXC.

   e. In **BI Key Delimiter**, Review the single ASCII character (such as a comma or underscore) to function as a delimiter for combining the data columns that form a key. This value is available only when multiple columns are specified for one key.
   f. In **Geometry Type**, specify whether the layer is a polygon, point, or line geometry layer. The type that you select affects the formatting that users can apply to the layer.
   g. In **BI Key Columns Area**, secifies the columns of data that you want to associate with the layer. You can have multiple columns associated with a single layer. You can select multiple columns from one subject area or from multiple subject areas. The columns and delimiter that you select must exactly match the name of the **Layer Key** value. Suppose the Layer Key value is STATE_CITY. You must select the STATE and CITY BI data columns and specify the underscore character in the **BI Key Delimiter** field.

      Use the various options in this area:

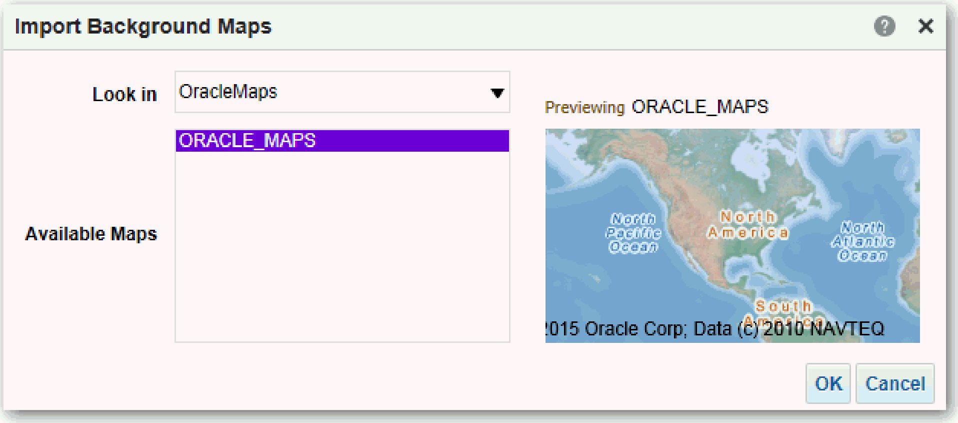
      - **Add** — Displays the list of available subject areas. Select a subject area and select all the data columns that you want to associate with the layer.
      - **Delete** — Deletes the selected key column.
      - **Edit** — Lets you edit the data columns associated with a layer.

When a content designer creates a map view, a default main map is selected as the basis for that map view. If at least one data column from the analysis is associated with a layer that's associated with a main map, then that main map is selected by default.

h. In **Show Qualified Names**, specifies whether to display the fully qualified name of the column in the BI Key Columns Area or simply the column name.

6. Click **OK** to close the dialog.
7. Click the Background Maps tab, then click the **Import Background Maps** button.
8. In the Import Background Maps dialog, select the connection in the **Look in** field and the main maps to use, then click **OK**.

   The connection that you select for the main map can be different from the connection for the layers or images.



Import Background Maps

Look in: OracleMaps

Available Maps: ORACLE_MAPS

Previewing ORACLE_MAPS

9. See Editing Background Maps for the steps required to prepare the background maps.

After you've added background maps and map layers, you can use the information to create a static image for a map. The static image is displayed to content designers and users who work with map views.

Edit Background Maps For Dashboards and Analyses

You edit background maps to ensure that users have a seamless experience with map views in dashboards and analyses.

A background map is a non-interactive map that serves as a base for the map view. It might display a satellite image or a map with roads. The background map specifies the order of layers on the map view.

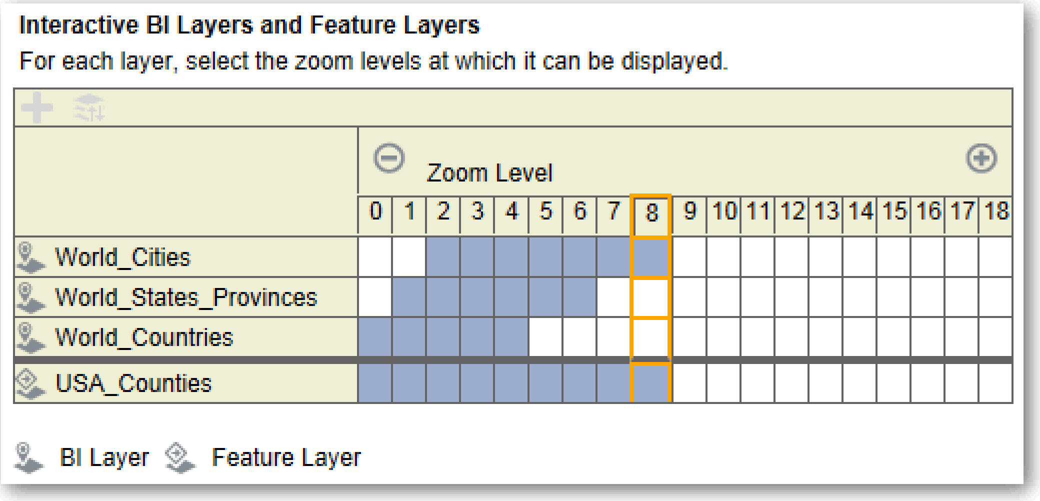
The ordering of map layers is very important. You must pay close attention to ensure that users have a seamless experience while navigating on the map (that is, drilling and zooming). In the Edit Background Map dialog, you assign each layer a minimum and maximum zoom range. Given that the map zoom slider can slide only from bottom to top vertically, the layers with lower minimum zoom levels are placed at the bottom of the slider. Ensure that the layer grid on the Interactive BI Layers section of the dialog follows a similar pattern, so that you place layers with lower minimum zoom levels at the bottom of the list.

Layer ordering becomes irrelevant when the zoom ranges of layers don’t intersect on the scale. Ordering becomes very important when layers have a common minimum and maximum zoom range. Use care to ensure that detailed layers aren’t hidden by the aggregated layers during drilling or zooming operations.

1. On the Classic Home page, click the user profile icon, **Administration**, and then click **Manage Map Data**.
2. Click the **Background Maps** tab, select a map, then click the **Edit Background Map** button to display the Edit Background Map dialog.
3. Specify the name and description of the map, which is displayed as a tooltip for the map when selecting a map from the list, when editing the map view.
4. The Location field displays the location of the background map in the data source. Click the **Location** button to change to a different map. If you select a background map that includes a different number of zoom levels, then the zoom levels are automatically adjusted for the layers that are associated with the map by scaling their ranges.
5. Click the **Add Layers** button to display a list of the layers that have been imported on the Layers tab, then select the layers to add to the map. This button is unavailable when all layers from the Layers tab have been added to the background map.

   When you add a layer that's part of the map definition, the layer displays at its default zoom levels. If the layer isn't part of the map definition, then specify the zoom levels yourself.

   The layers are listed from bottom to top, in terms of how they're applied to the map. A sample order is Countries, States, Cities. The lower level layers generally have the lower zoom levels. For example, if you have a States layer and a Cities layer, then include lower zoom levels for State than City.



Interactive BI Layers and Feature Layers

For each layer, select the zoom levels at which it can be displayed.

	Zoom Level																		
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
World_Cities			■	■	■	■	■	■	■										
World_States_Provinces		■	■	■	■	■	■												
World_Countries	■	■	■	■	■														
USA_Counties	■	■	■	■	■	■	■	■	■										

BI Layer Feature Layer

6. Click the **Sort Layers By Zoom Level** button to list the layers in ascending or descending order based on visibility on the map. This button is unavailable when layers are listed in the proper order.

   The sort order that's specified here doesn't affect the order in which layers are applied on the map. Instead, the sorting order affects the zoom levels. For example, the States layer might have zoom levels 1 through 3 and the Cities layer has zoom levels 4 through 9. The lower layers have the lower zoom level numbers. The zoom levels that you specify correspond to the tick marks on the zoom slider on the map.

   You can include both layers that have been associated with a column by using the Edit Layer dialog and layers that haven't been associated. Ensure that BI layers are ordered higher than non-BI layers. If a non-BI layer is ordered higher than any BI layers, then the non-BI layer is displayed on top of the lower BI layers on the map, which prevents the BI layers from being interactive.

7. Click the **Turn On Layer Visibility** or **Turn Off Layer Visibility** button to control the visibility of layers on the map. Use the buttons to indicate whether the layer is visible in the Preview map in this dialog only. The layer is still visible on a map view. You can modify the zoom levels for a layer with a visibility turned off.

8. Click a cell under a zoom level for a layer to affect the zoom level:

   - If you click a blue cell that's between other blue cells, then you see a popup menu with **Clear Before** and **Clear After** buttons, which allow you to change the zoom level in either direction. For example, if you click the cell for zoom level 4 and click the eraser on the right, then all cells to the right are cleared for that zoom level.

   - If you click a blue cell that at the end of a line of blue cells, then the cell turns white to indicate that it's no longer part of that zoom level.

   - If you click a white cell, then you increase the zoom level on either side of the existing blue cells. For example, suppose cells 4 through 6 are colored blue to reflect the zoom level. If you click in cell 2, then the zoom level becomes 2 through 6.

   If you don't set any zoom levels for a layer, then that layer doesn't display on the map.

9. Click the action icon beside the layer name to display a menu from which you can make various selections:

   - **Delete** — Removes the layer from this background map. The layer continues to be available on the Layers tab and can be added to this area again.

   - **Move Up** or **Move Down** — Moves the layer up or down so you can specify the order in which layers are applied to the map.

   - **Reset to Default Visibility** — Resets the current visibility range for this layer as defined in the underlying map definition. If this layer isn't natively associated with the map, then this option is disabled for that layer.

10. Use the yellow border that surrounds the column of boxes for a zoom level to determine which zoom level is currently displayed in the map area.

11. Use the panning and zooming controls to specify how the map is displayed to users. If you hover over the zoom slider, then you see tooltips that specify the names of the layers that are currently associated with that zoom level.

12. Click **OK**.

Changes to Spatial Metadata Require Restart

An administrator can edit the spatial metadata that is stored in the Oracle Database and accessed by MapViewer.

For example, an administrator can add a new layer. These edits are not visible on the Oracle BI Server Administration pages for managing maps until MapViewer is restarted and so refreshed with the latest updates.

15 Configure Time Zones

This chapter describes how to configure time zones.

Topics:

Why and Where Are Time Zones Used?

Time zones are used for a variety of purposes.

A time stamp can indicate when an object was changed, and users can specify a time for an agent to run. Users often are most comfortable working in their local time zones. As the administrator, you can configure the preferred time zones for users for various components.

Before you begin to set preferred time zones, see the next table for information about where time zones are used.

Type	Description
Oracle BI Server	If you have users in time zones that are different from the zone for Presentation Services, then you as the administrator can specify the time stamps that those users see in Oracle Analytics Server. For example, suppose the server is located in the Pacific time zone in the United States. You can specify that users on the east coast of the United States see time stamps that are displayed in Eastern Standard Time. If you make no time zone settings and if a user doesn't specify a preferred time zone using the My Account dialog, then that user sees time displayed according to the local time zone for Presentation Services. For information about how users specify their preferred time zones, see *Visualizing Data in Oracle Analytics Server*.
Data from the database	The administrator specifies the time zone for the data that's retrieved from the database. If you make no time zone settings, then users see the time stamp data in the time zone of the original data as set by the administrator.

Type	Description
Content that is displayed in Oracle Analytics Server	Users who create analyses can specify the time zone that's displayed in their analyses and dashboard prompts. This specification overrides those made by you as the administrator and by end users if they have previously used the column in their queries and have set the time zone. If the specified display time zone supports daylight saving time, then the timestamp values that are displayed are automatically adjusted for daylight saving time.
General time stamps that indicate when events happen	End users can specify the time zone for many general stamps including the following ones: • The scheduled time of agents. • The generated time of alerts or analyses. • The time on which objects in the Presentation Catalog are created, modified, and accessed.
Log files	Log files contain time stamps for various activities.

Set Time Zones

You can override the user’s time zone selection with one that's consistent for all users.

Use the following procedure to set time zones for users.

You can also set in the Console, see Format Options.

1. Determine the time zone that's set for the server on which Presentation Services is running.
2. Use elements in the Presentation Services configuration file (instanceconfig.xml) or use session variables.

   The instanceconfig.xml file is located in: `BI_DOMAIN/config/fmwconfig/biconfig/OBIPS`

   Consult the following:

   - See What Is the Precedence Order for Time Zones? for information about the precedence order for time zones.
   - See Description of Time Zone Settings for descriptions of the session variables and elements.
   - See Example: Configuration File Settings for Specifying the Time Zone.
   - See *Visualizing Data in Oracle Analytics Server* for complete information about session variables.
3. Encourage end users to specify their preferred time zones using the My Account dialog.
4. Encourage users who create analyses to do the following to set the time stamps for their analyses:
   a. Use the Data Format tab of the Column Properties dialog to specify the time zone that's displayed in the columns of their analyses.
   b. Use the Time Zone dialog to set the time zone that's displayed in dashboard prompts.

What Is the Precedence Order for Time Zones?

Time zones are presented in a specific order.

The actual time zone in which various types of content are displayed follows a precedence order described in the table below. An item with a lower number overrides one with a higher number. For example, Item 1 takes precedence over Item 2.

Time Zone For	Determined By
Data	1. The setting of the DATA_TZ session variable. 2. The setting of the DefaultDataOffset element in the instanceconfig.xml file. 3. The time zone of the original data as set by the administrator (because the time zone is unknown for Presentation Services).
Data display	1. The setting that a content designer makes. 2. The setting of the *DATA_DISPLAY_TZ* session variable. 3. The setting of the DefaultDataDisplay element in the instanceconfig.xml file. 4. User-Preferred Time Zone
General time stamps (not including column data and log files)	1. User-Preferred Time Zone 2. The time zone for Oracle BI Presentation Services.
Log file information	1. The setting of the Logging element in the instanceconfig.xml file. 2. The time zone for Presentation Services.

User-Preferred Time Zone

Users can select the time zone they want presented in the interface.

The user-preferred time zone is determined by the following:

- The specification that a user makes in the My Account dialog. Users can select a time zone in which they prefer to view content.

  For information about setting the time zone preference on the My Account dialog: Preferences tab, see *Visualizing Data in Oracle Analytics Server*.
- The setting of the *TIMEZONE* session variable.
- The setting of the *DefaultUserPreferred* element in the instanceconfig.xml file.

Where Are Time Zone Specifications Stored?

Time zone specifications are stored in their own file.

Whenever a time zone specification is displayed in a list, or as the value of a session variable or element in the instanceconfig.xml file, the specification originates from the TimeZones.xml file, located in:

```
orahome/bi/bifoundation/timezone
```

The TimeZones.xml file contains nearly all time zones from around the world. You need not add zones to this file, but you can edit this file if you care to. You can delete those zones that users in your organization don't use.

Specifying Time Zone Values

Various editors show the ampersand that appears in time zone values in one of two ways: either the ampersand character itself or its escape sequence. Use care when entering a time zone value, as follows:

- When you use the ampersand in the value of a session variable, include the ampersand character (&) in the value, such as `Pacific Time (US & Canada); Tijuana`.
- When you use the ampersand in the value of an element in the Oracle BI Presentation Services configuration file (instanceconfig.xml), include the escape sequence for the ampersand in the value, such as `Pacific Time (US &amp; Canada); Tijuana`.

Description of Time Zone Settings

The table below describes the session variables and the elements in the `instanceconfig.xml` file with which you set time zones.

When you include elements in the `instanceconfig.xml` file, you specify the time zone that all users see. When you use session variables, you can specify a different time zone for each user. If you use session variables and you specify values for the appropriate elements in the `instanceconfig.xml` file, then the values of the session variables override the settings in the `instanceconfig.xml` file.

> **Note:**
>
> Certain system session variables such as, USER or ROLES, can't be overridden by request variables. Other system session variables, such as DATA_TZ and DATA_DISPLAY_TZ (Timezone), can be overridden if configured in the Developer Client Tool.
>
> See Working with Repository Variables in *Managing Metadata Repositories for Oracle Analytics Server*.

Element	Session Variable	Description	Value
DefaultDataOffset	DATA_TZ	The time zone offset of the original data. To enable the time zone to be converted so that users see the appropriate zone, you must set the value of this element or variable. If you don't set this option, then no time zone conversion occurs because the value is "unknown". For example, suppose you want to convert to Eastern Standard Time (EST), which is Greenwich Mean Time (GMT) - 5. You must specify this value to enable the conversion to EST.	An offset that indicates the number of hours away from GMT time. For example: "GMT-05:00" or "-300", which means minus 5 hours.
DefaultDataDisplay	DATA_DISPLAY_TZ	Specifies the time zone to use for displaying data. If you don't set this option, then the value is the User-Preferred Time Zone.	One of the time zones that are specified in the TimeZones.xml file. See Specifying Time Zone Values
DefaultUserPreferred	TIMEZONE	Specifies the users' default preferred time zone before they select their own in the My Account dialog. If you don't set this option, then the value is the local time zone from Presentation Services.	One of the time zones that are specified in the TimeZones.xml file. See Specifying Time Zone Values
Logging	Not applicable	The time zone of the time stamps that appear in log files that are generated by Presentation Services. If you don't set this option, then the value is the local time zone from Presentation Services	One of the time zones that are specified in the TimeZones.xml file. See Specifying Time Zone Values
TimeZone	Not applicable	The parent element for the elements that modify the preferred time zone. A child of the ServerInstance element.	Not applicable

Example: Configuration File Settings for Specifying the Time Zone

The following shows a sample section of the instanceconfig.xml file in which the TimeZone element has been added.

```
<TimeZone>
     <DefaultDataOffset>0</DefaultDataOffset>
     <Logging>(GMT-08:00) Pacific Time (US &amp; Canada); Tijuana</Logging>
     <DefaultUserPreferred>(GMT-08:00) Pacific Time (US &amp; Canada);
Tijuana</DefaultUserPreferred>
```

```
     <DefaultDataDisplay>(GMT-06:00) Central Time (US &amp; Canada); Tijuana</
DefaultDataDisplay>


</TimeZone>
```

16

Localize Oracle Analytics Server

This chapter describes how to configure Oracle Analytics Server for deployment in one or more language environments other than English. Users can easily and dynamically change their language and locale preferences. When users select a language, they see many elements in that language. These elements include user interface components, metadata, messages, and help files.

Topics:

What Is Localization?

Localization refers to the process of adapting your deployment to a particular language.

If your users speak languages other than English, then use the information about localization to adapt your deployment to support multiple languages.

What Components Are Translated?

Several areas of the interface are translated into a user’s native language.

The following list outlines which components of Oracle Analytics Server are translated into languages other than English:

- Installer
- Web user interface
- Presentation Services messages:
  - Error
  - Warning
  - Information
- Oracle BI Server functions:
  - DayName
  - MonthName

> **Note:**
>
> If a query uses the DayName or MonthName function, the day or month name displays in the localized language but the column name remains in English (unless affected by other localization controls). For example, if the LOCALE parameter is set for German, the MonthName query returns the string "Mai" but the column header remains "Name of Month."

- Oracle BI Server and Oracle BI Scheduler messages:
  - Error
  - Warning
  - Information
- Log files:
  - obis1-diagnostic.log for Oracle BI Server
  - nqquery.log for Oracle BI Server
  - If Clustering is enabled, nQCluster.log for Oracle BI Server Cluster
- Metadata:
  - Dashboards and analyses (Presentation Catalog)
  - Presentation table and column names (.rpd file)
- Model Administration Tool interface
- ODBC setup

The following list outlines which components of Oracle Analytics Server are not localized:

- ODBC client tools:
  - nqcmd (Linux)
  - nqcmd.exe (Windows)
  - nqclient.exe (Windows)

Numerous Oracle Fusion Middleware components, such as Oracle WebLogic Server Administration Console and Fusion Middleware Control, are translated. See Oracle Fusion Middleware documentation for information.

Tasks to Localize Components

Localizing the system components for international use requires several steps.

As the administrator, you perform various tasks to localize components, as described in the following sections:

- Localize Presentation Services
- Set the Current Locale in the Oracle BI Server
- Localize Metadata Names in the Semantic Model
- Support Multilingual Data

Localize Presentation Services

As the administrator, you perform various tasks to localize Presentation Services.

Topics:

- Localize the User Interface for Presentation Services
- Localize Catalog Captions
- Tip for Arabic and Hebrew in Mozilla Firefox Browsers

Localize Catalog Captions

You can localize the names of classic reporting objects that users create in the catalog. Classic object names are also known as *captions*. Workbook custom captions aren't changed when you localize classic object names.

To localize captions for classic content, you export the captions from the catalog to a file, translate the captions, and then upload the localized captions back into the catalog. You must upload your translations to the *same* Oracle Analytics environment that you exported the captions from.

For example, your company's browser setting use Argentinian Spanish rather than Spain Spanish, you can set the language to Argentinian Spanish to override the current language setting.

If you want to migrate caption localizations to a *different* Oracle Analytics environment, you can export your catalog to a snapshot and then import the snapshot on the target environment. Caption translations are included in the snapshot.

Export Captions from the Catalog

The following procedure describes how to export text strings in the catalog.

1. In the Classic Home page, click the user profile icon and then click **Administration**.
2. In the **Manage Catalog Captions** area, click **Export Captions**.
3. Click **Browse** to display the Catalog browser, select the folder that contains the files you want to localize, and then click **OK**.

   For example, you might select `\Shared Folders\Sample Report`.
4. In the **Export Captions** dialog, click **OK** to download and save the XML file in a local area.

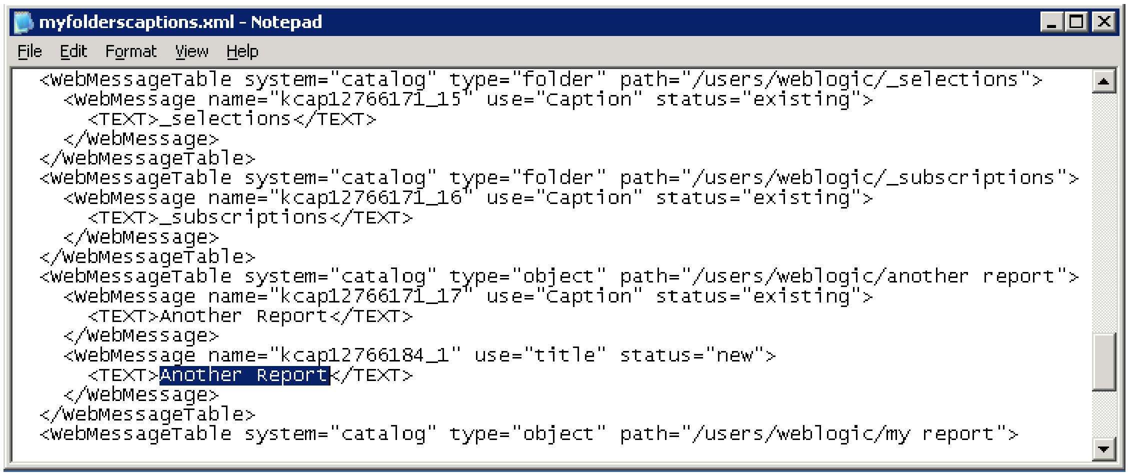
   For example, if you select the file `\Shared Folders\Sample Report`, you'll save a file locally named `_shared_Sample Report_captions.xml`.

Localize Your Captions

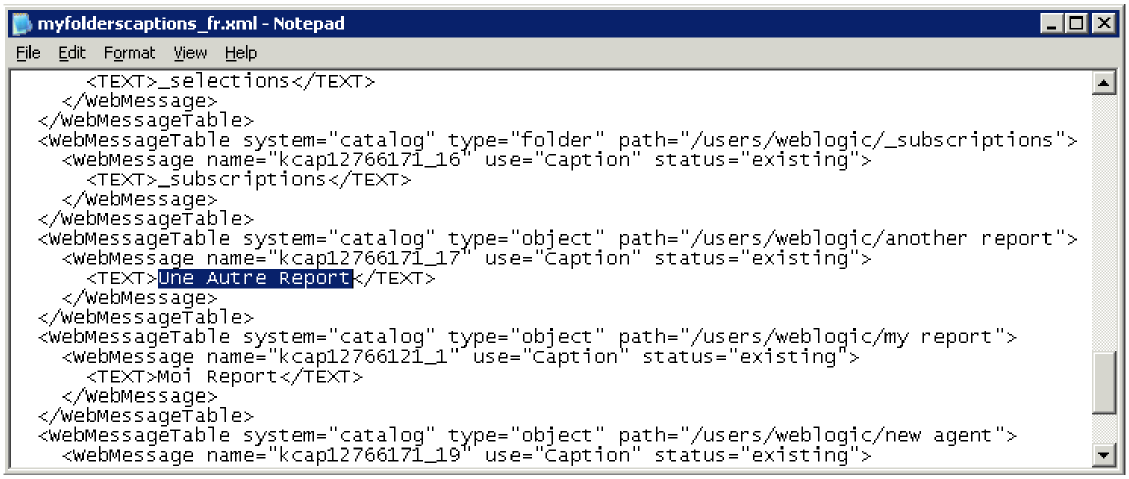
After you have exported your captions in an XML file, deliver the XML file to the localization team. For example, if you selected the Custom folder to download, you'll deliver a file named `_shared_Custom_captions.xml`.

You and the localization team are responsible for resolving any errors in the translated text strings. Consider that the contents of the catalog are updated whenever objects are added, deleted, or modified.

The first illustration shows an extract from an exported caption XML file before translation. The file is named `myfoldercaptions.xml`. The second illustration shows an extract from the file after translation. The file is named `myfoldercaptions_fr.xml`.



```
<webMessageTable system="catalog" type="folder" path="/users/weblogic/_selections">
  <webMessage name="kcap12766171_15" use="Caption" status="existing">
    <TEXT>_selections</TEXT>
  </webMessage>
</webMessageTable>
<webMessageTable system="catalog" type="folder" path="/users/weblogic/_subscriptions">
  <webMessage name="kcap12766171_16" use="Caption" status="existing">
    <TEXT>_subscriptions</TEXT>
  </webMessage>
</webMessageTable>
<webMessageTable system="catalog" type="object" path="/users/weblogic/another report">
  <webMessage name="kcap12766171_17" use="Caption" status="existing">
    <TEXT>Another Report</TEXT>
  </webMessage>
  <webMessage name="kcap12766184_1" use="title" status="new">
    <TEXT>Another Report</TEXT>
  </webMessage>
</webMessageTable>
<webMessageTable system="catalog" type="object" path="/users/weblogic/my report">
```


myfolderscaptions_fr.xml - Notepad
File Edit Format View Help

```
    <TEXT>_selections</TEXT>
  </webMessage>
</webMessageTable>
<webMessageTable system="catalog" type="folder" path="/users/weblogic/_subscriptions">
  <webMessage name="kcap12766171_16" use="Caption" status="existing">
    <TEXT>_subscriptions</TEXT>
  </webMessage>
</webMessageTable>
<webMessageTable system="catalog" type="object" path="/users/weblogic/another report">
  <webMessage name="kcap12766171_17" use="Caption" status="existing">
    <TEXT>Une Autre Report</TEXT>
  </webMessage>
</webMessageTable>
<webMessageTable system="catalog" type="object" path="/users/weblogic/my report">
  <webMessage name="kcap12766121_1" use="Caption" status="existing">
    <TEXT>Moi Report</TEXT>
  </webMessage>
</webMessageTable>
<webMessageTable system="catalog" type="object" path="/users/weblogic/new agent">
  <webMessage name="kcap12766171_19" use="Caption" status="existing">
```

Upload Localized Captions to the Catalog

After you've localized your captions in the required languages, you deploy the languages by uploading the translated XML files to the *same* Oracle Analytics environment that you exported the captions from. Use this procedure for each language.

1. In the Classic Home page, click the user profile icon and then click **Administration**.
2. In the **Manage Catalog Captions** area, click **Import Captions**.
3. Click **Browse**, then navigate to and select the localized XML file, and click **OK**.
4. Use the **Select language** option to select the language to which you've localized, then click **OK**.

   Imported XML files are copied to the `MsgDb` folder under the selected language.

Tip for Arabic and Hebrew in Mozilla Firefox Browsers

Right-to-left languages are displayed slightly differently in Mozilla Firefox browsers.

By default, scroll bars are displayed on the right side of the Mozilla Firefox browser. If you're using the Arabic or Hebrew languages, then it's not appropriate to have the scroll bars on the right side. You can change the browser settings in Firefox such that the scroll bars are displayed on the left side.

For information about changing the layout.scrollbar.side setting, see the Firefox documentation.

Localize the User Interface for Presentation Services

You can localize the user interface for Presentation Services, if your users speak languages other than English.

Users can select a language on the sign-in page for Oracle Analytics, and many elements of the interface are automatically displayed in the appropriate language. After signing in, users can change the language setting on the Preferences tab of the My Account dialog.

The user's setting is stored in the WEBLANGUAGE session variable. For the Presentation Services user interface, WEBLANGUAGE is set when a user selects a language on the sign-in page.

> **Note:**
>
> For Oracle Applications, WEBLANGUAGE is set to the language of the user's browser when a user logs in for the first time. For example, if a user with a browser language set to French logs in to Oracle Applications for the first time, then the value for WEBLANGUAGE is French, and the metadata is translated to French.

As the administrator, you perform various tasks to localize other elements of the user interface for Presentation Services, as described in the following sections:

- Understand the Directory Structure to Localize Presentation Services
- Localize Messages for Users' Preferred Currency
- Specify the Default Language for the Sign-In Page
- Specify the Language in the URL

Understand the Directory Structure to Localize Presentation Services

Oracle Analytics Server is installed with many files that control elements in the user interface and messages.

These files are installed in the messages and pages subdirectories of the `ORACLE_HOME/bi/bifoundation/web/msgdb` directory. To localize these elements and messages, you copy those files to the l_*xx* subdirectories in `SDD/components/OBIPS/custommessages/l_xx`

where SDD is the Singleton Data Directory, and _xx indicates the language extension. See Key Directories. After you have copied the files, you can modify their contents as appropriate for the language that corresponds to the subdirectory in which you have copied them.

Localize Messages for Users' Preferred Currency

You can localize the messages that are associated with a preferred currency.

See Define User-Preferred Currency Options Using a Static Mapping for information on working with users' preferred currencies.

1. Go to the `SDD/components/OBIPS/custommessages/l_xx` directory, where *xx* is the language extension for the language in which you are specifying preferred currencies.

   Where SDD is the Singleton Data Directory for example, `DOMAIN_HOME/bidata`. See Key Directories.

2. In the directory, create a subdirectory called custommessages.
3. In the directory, create a file in XML format, with the name of usercurrencymessages.xml.
4. Add entries such as the following one to this file for the language that corresponds to the directory in which you are working. The following example includes two messages: kmsgMyCurrency1 and kmsgMyCurrency2

   ```
   <?xml version="1.0" encoding="utf-8"?>
   <WebMessageTables xmlns:sawm="com.siebel.analytics.web/message/v1">
   <WebMessageTable system="CurrencyDisplay" table="Messages" code="false">
      <WebMessage name="kmsgMyCurrency1"><TEXT>My Currency Text 1</TEXT></
   WebMessage>
      <WebMessage name="kmsgMyCurrency2"><TEXT>My Currency Text 2</TEXT></
   WebMessage>
   </WebMessageTable>
   </WebMessageTables>
   ```

5. Edit the file to specify displayMessage="kmsgMyCurrency1" to use this message.
6. Repeat Steps 1 through 5 for each language for which you must localize these messages.
7. Restart the service for Oracle BI Presentation Services.

In Oracle Analytics Server, the appropriate localized text is displayed to the user. In this example, the text is My Currency Text 1.

Specify the Default Language for the Sign-In Page

You can change the language the user sees by overriding the language specified by their browser.

The default language in which the Presentation Services sign-in page is displayed is obtained from the user's client browser settings.

Update your browser's Internet Options settings to change the language specified by your browser.

Specify to Scale Numbers in Performance Tiles

In Presentation Services, you can use performance tiles to focus attention on a single piece of high-level aggregate data.

The tile can include a number such 1,000,000 or you can specify to "compress" or "scale" the value with an indicator such as 1M, for example.

To scale the number, Presentation Services searches for the scaling factors that the current locale allows, processes the number, and appends the indicator value. If no scaling factors are defined for the current locale, then no scaling is applied. Because the scaling of numbers differs by language, you can manually edit the localedefinitions.xml file to control the scaling, as described in the following procedure.

1. In a text editor, open the localedefinitions.xml file in the directory:

   ORACLE_HOME/bi/bifoundation/web/display

2. In the file, add a new property for "scaleFactors" and enter values appropriate to your locale.

   The following shows values for an English locale:

   ```
   <localeDefinition name="en">
       <!-- existing data -->
           <property name="scaleFactors">
               <property name="thousand">K</property>
               <property name="million">M</property>
               <property name="billion">B</property>
               <property name="trillion">T</property>
               <property name="quadrillion">Q</property>
               <property name="quintillion">Qu</property>
               <property name="sextillion">S</property>
               <property name="septillion">Sp</property>
               <property name="octillion">O</property>
               <property name="nonillion">N</property>
               <property name="decillion">D</property>
               <property name="undecillion">UD</property>
               <property name="duodecillion">DD</property>
           </property>
   </localeDefinition>
   ```

   The following shows values for an Indian locale:

   ```
   <localeDefinition name="en-in" basedOn="en">
       <!-- english - india -->
       <!-- existing data -->
           <property name="scaleFactors">
               <property name="thousand">K</property>
               <property name="hundred-thousand">L</property>
               <property name="ten-million">Cr</property>
           </property>
   </localeDefinition>
   ```

3. Save your changes and close the file.

4. Restart Presentation Services.

Specify the Language in the URL

When users start Oracle Analytics Server by displaying the sign-in page, they can select the language as part of the sign-in process.

Users can also select a language on the Preferences tab of the My Account dialog.

If you provide users with a URL with which they can display a dashboard or other page of the application, then you can define a URL parameter as a profile attribute. Doing so dynamically sets the language of the dashboards and analyses to be consistent with the application's language setting.

For operational applications, symbolic URLs embed dashboards and analyses in the integrated environment. For Oracle BI Presentation Services, the URL parameter Lang designates the language that the web page renders.

The Lang parameter can be included in the symbolic URL that is defined in the operational application to connect to Oracle Analytics Server. The Lang parameter is defined as a profile attribute, but when the symbolic URL is constructed at runtime, the value is set as the profile attribute LanguageCode. The next table provides examples of the parameter settings in the Symbolic URL parameters applet, including Lang.

For example, the following URL displays the sign-in page in French.

```
http://Server_Name:port_number/analytics/saw.dll?Dashboard&Lang=fr
```

Name	Type	Path Argument Value	Append	Sequence #
Cmd	Constant	Go	Y	1
Path	Constant	/shared/Sales/Pipeline/Overview/Top 10 Deals	Y	2
nQUser	Command	UseLoginId	Y	3
nQPassword	Command	UseLoginPassword	Y	4
PostRequest	Command	PostRequest	Y	5
Lang	Profile Attribute	LanguageCode	Y	6

Localize Data Visualization

As the administrator, you perform various tasks to localize Data Visualization.

Topics:

- Localize the User Interface for Data Visualization
- Localize Data Visualization Workbook Captions

Localize Data Visualization Workbook Captions

You can localize the names of custom Data Visualization workbook captions. For example, you might localize a customized workbook name into Spanish and French.

To localize the names of Data Visualization workbook captions, you export captions for the Data Visualization workbook to a file, translate the captions, and then upload the translated captions back into the workbook. You must upload your translations to the same Oracle Analytics environment that you exported the captions from.

If you want to migrate caption localizations to a *different* Oracle Analytics environment, you can export your workbook captions to a snapshot and then import the snapshot on the target environment. Caption translations are included in the snapshot.

Topics:

- Export Workbook Captions
- Localize Workbook Captions
- Import Localized Workbook Captions

Export Workbook Captions

You can export workbook captions so that they can be translated.

1. On the Home page, click the **Navigator**, and then click **Console**.
2. Click **Workbook Translations**.
3. Click the **Export** tab.
4. Expand Shared Folders, select the folder containing the visualization workbook caption files to localize, for example, `\Shared Folders\OAC_DV_SampleWorkbook`.
5. Click **Export** to download and save the exported `captions.zip` file, containing the .JS files that you want to localize, to the browser's download folder.

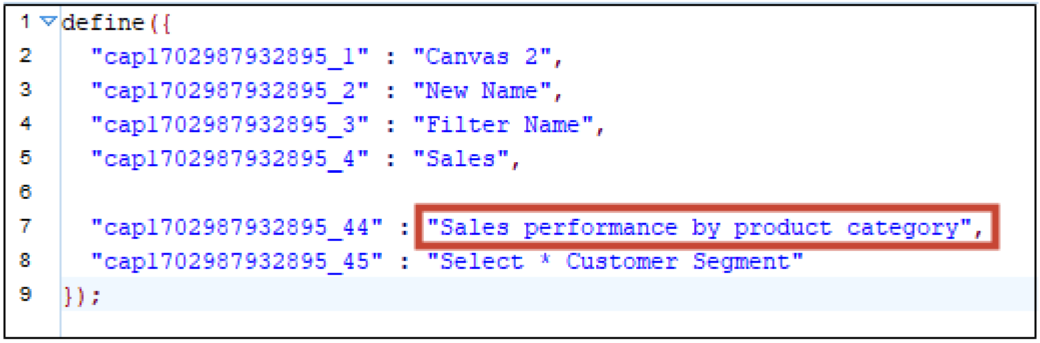
Localize Workbook Captions

After you have exported your Data Visualization workbook captions, you deliver the captions.zip file containing language-specific JS caption files for each supported language, to the localization team. For example, if you're localizing the French captions file, the file that you update might be named `@/Shared/DataVizWorkbookFolderNameExample/WorkbookNameExample/NLS/fr/captions.js`.

You and the localization team are responsible for resolving any errors in the translated text strings. Consider that the contents of the workbook are updated whenever objects are added, deleted, or modified.

1. Browse to the workbook captions ZIP file that you exported, and extract the language-specific JS file that you want to update.
2. Open the extracted language-specific JS file for editing.
3. Enter translated names into the appropriate caption elements to replace the existing text strings.

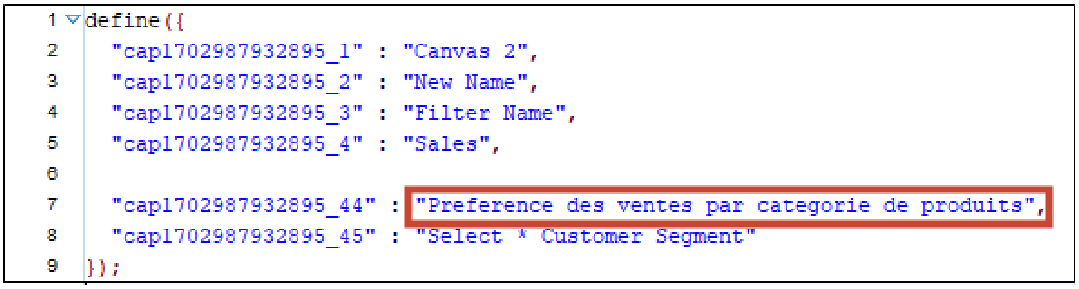
   For example, if you created a visualization title caption in `Canvas 2` named `Sales performance by product category`, you edit and replace the English text with the French translation which is `Performance des ventes par categorie de produits`.
   The French `captions.js` file before translation:

```
define({
  "cap1702987932895_1" : "Canvas 2",
  "cap1702987932895_2" : "New Name",
  "cap1702987932895_3" : "Filter Name",
  "cap1702987932895_4" : "Sales",

  "cap1702987932895_44" : "Sales performance by product category",
  "cap1702987932895_45" : "Select * Customer Segment"
});
```

   The French `captions.js` file after translation:



```
define({
  "cap1702987932895_1" : "Canvas 2",
  "cap1702987932895_2" : "New Name",
  "cap1702987932895_3" : "Filter Name",
  "cap1702987932895_4" : "Sales",

  "cap1702987932895_44" : "Preference des ventes par categorie de produits",
  "cap1702987932895_45" : "Select * Customer Segment"
});
```

4. Save the updated language-specific JS file and then add it to the exported translated captions ZIP file.
5. Optional: You can also use this method to import localized Classic catalog caption .XML files. You can add translated .XML files under the top level directory of the exported translated captions ZIP file. and zip them up together for import.

   For example:

   - `ar/_shared_Common_captions.xml`
   - `cs/_shared_Common_captions.xml`
   - ...
   - `zh-TW/_shared_Common_captions.xml`

Import Localized Workbook Captions

After you've localized your visualization workbook captions in the required language, you deploy the language by uploading the translated ZIP file to the *same* Oracle Analytics environment that you exported the workbook captions from.

1. In the Oracle Analytics Home page, click the **Navigator**, and then click **Console**.
2. Click **Workbook Translations** and click the **Import** tab.
3. Click **Select a file or drop it here**, and browse for, or drag and drop the ZIP file containing the translated JS file that you want to import.
4. Click **Import**.

Oracle Analytics displays translated language-specific text strings in a browser that's suitably configured to use the correct captions file for the required language.

Localize the User Interface for Data Visualization

You can localize the user interface display language and regional data formats for Data Visualization.

The order of precedence for language and locale settings apply applies as follows:

- Browser language preference (browser settings).
- User setting for language or locale overrides the browser language preference.
- URL query parameter for language or locale overrides the user setting.
- Embedding parameter for language or locale overrides the URL query parameter.

When you localize the user interface display language or local-based regional data formats for Data Visualization it doesn't include workbook custom captions. You localize workbook custom captions separately. See Localize Data Visualization Workbook Captions.

Topics:

- Localize Data Visualization User Interface Display Language
- Localize Data Visualization Regional Data Formats
- Workbook Data Format Changes When You Select a Different Locale

Localize Data Visualization User Interface Display Language

You can change the language for displaying Data Visualization user interface strings.

1. From the Home page, click the user profile icon.
2. Click **Profile**, and click the **My Profile** tab.
3. Click **Language** and select the language to use for the user interface.

   The language that you select takes precedence over the browser language.
4. Sign out of Oracle Analytics Server.
5. Sign in to Oracle Analytics Server using your credentials, and select **Use Profile Setting for Language** before you click **Sign In**, to display the language that you selected.

Localize Data Visualization Regional Data Formats

You can select a locale to display region-specific date and number formatting in Data Visualization workbooks.

1. From the Home page, click the user profile icon.
2. Click **Profile**, and click the **My Profile** tab.
3. Click **Locale** and select a locale.

   The locale that you select takes precedence over the browser locale.
4. Sign out of Oracle Analytics Server.
5. Sign in to Oracle Analytics Server using your credentials, and select **Use Profile Setting for Language** before you click **Sign In**, to display the language that you selected.

Workbook Data Format Changes When You Select a Different Locale

When you select a different locale, data formatting changes can occur in various workbook areas.

- **General workbook areas affected:**
  - date or time formats (timestamp uses a combination of date or time formatting)
    For example, mm/dd/yy (USA) versus dd/mm/yy (EU regions).
  - number formats (variations in the decimal and thousands separator)
    For example, 15.000.00, or 15,000.00
- **Workbook presentation mode areas affected:**
  - visualizations (data display, tooltips, titles)
  - filter controls (data display and data entry)
  - parameter controls (data display and data entry)
- **Workbook edit mode areas affected:**

  - parameter dialog value display or entry
  - conditional format dialog
  - visualization properties
  - any other workbook edit surfaces that expose dates, time, number

Set the Current Locale in the Oracle BI Server

Learn more about the various ways to customize localizations in the application.

The following sections provide information about setting the locale in Oracle BI Server:

- Set Locale Parameters on the Oracle BI Server
- Understand How the Error Message Language Is Determined
- Set the Language for Components of the Oracle BI Server
- Modify the Language of the User Interface for the Model Administration Tool
- Troubleshoot the Current Locale in the Oracle BI Server
- Ensure That Text for Oracle BI Server Utilities Is Displayed in the Correct Language
- Modify the Semantic Model When the Underlying Oracle Database NLS_CHARACTERSET Is Non-Unicode

Set Locale Parameters on the Oracle BI Server

To support multiple languages, the Oracle BI Server must be configured properly.

The General section of the NQSConfig.INI file contains parameters that are required for localization and internationalization. It also contains default parameters that determine how data is sent from the Oracle BI Server to a client application. See Configuration File Settings.

The following parameters in the NQSConfig.INI file affect localization:

- LOCALE
- SORT_ORDER_LOCALE
- SORT_TYPE
- CASE_SENSITIVE_CHARACTER_COMPARISON

To successfully run Oracle Analytics, ensure that you configure the appropriate locales on your operating system for the language in which users run the applications. Some locale- and language-related settings are interrelated and help determine how the Oracle BI Server sorts data.

Set the Locale on Linux Systems

The value to use for the C-runtime locale during server startup is specified in the SORT_ORDER_LOCALE parameter in the NQSConfig.INI file. This parameter is set normally by the Oracle BI Server.

The locale is used for functions such as displaying dates and currencies and sorting data.

If you must adjust the setting, then in the General section of the `NQSConfig.INI` file, set the LOCALE and SORT_ORDER_LOCALE parameters, entering a platform-independent name.

The following table shows language mappings from the platform-independent name to the specific name for each of the supported platforms. For example, Chinese uses the setting zh_CN.utf8 on Linux operating systems.

Name strings such as zh_CN.utf8 and fr-FR-UTF-8 are platform-specific names of the locale components, which must be installed by a system administrator. The NQSConfig.INI file uses the platform-*independent* names, such as Chinese or French (the names are case-insensitive).

Locale (Platform-Independent Name)	Name on Solaris	Name on AIX	Name on HP-UX/Linux
Arabic	ar_SA.UTF-8	AR_AA.UTF-8	ar_SA.utf8
Chinese	zh_CN.UTF-8	ZH_CN.UTF-8	zh_CN.utf8
Chinese-traditional	zh_TW.UTF-8	ZH_TW.UTF-8	zh_TW.utf8
Croatian	hr_HR.UTF-8	HR_HR.UTF-8	hr_HR.utf8
Czech	cs_CZ.UTF-8	CS_CZ.UTF-8	cs_CZ.utf8
Danish	da_DK.UTF-8	DA_DK.UTF-8	da_DK.utf8
Dutch	nl_NL.UTF-8	NL_NL.UTF-8	nl_NL.utf8
English-USA	en_US.UTF-8	EN_US.UTF-8	en_US.utf8
Finnish	fi_FI.UTF-8	FI_FI.UTF-8	fi_FI.utf8
French	fr_FR.UTF-8	FR_FR.UTF-8	fr_FR.utf8
German	de_DE.UTF-8	DE_DE.UTF-8	de_DE.utf8
Greek	el_GR.UTF-8	EL_GR.UTF-8	el_GR.utf8
Hebrew	he_IL.UTF-8	HE_IL.UTF-8	iw_IL.utf8
Hungarian	hu_HU.UTF-8	HU_HU.UTF-8	hu_HU.utf8
Italian	it_IT.UTF-8	IT_IT.UTF-8	it_IT.utf8
Japanese	ja_JP.UTF-8	JA_JP.UTF-8	ja_JP.utf8
Korean	ko_KR.UTF-8	KO_KR.UTF-8	ko_KR.utf8
Norwegian	no_NO.UTF-8	NO_NO.UTF-8	no_NO.utf8
Polish	pl_PL.UTF-8	PL_PL.UTF-8	pl_PL.utf8
Portuguese	pt_PT.UTF-8	PT_PT.UTF-8	pt_PT.utf8
Portuguese-Brazilian	pt_BR.UTF-8	PT_BR.UTF-8	pt_BR.utf8
Romanian	ro_RO.UTF-8	RO_RO.UTF-8	ro_RO.utf8
Russian	ru_RU.UTF-8	RU_RU.UTF-8	ru_RU.utf8
Slovak	sk_SK.UTF-8	SK_SK.UTF-8	sk_SK.utf8
Spanish	es_ES.UTF-8	ES_ES.UTF-8	es_ES.utf8
Swedish	sv_SE.UTF-8	SV_SE.UTF-8	sv_SE.utf8
Thai	th_TH.UTF-8	TH_TH.UTF-8	th_TH.utf8
Turkish	tr_TR.UTF-8	TR_TR.UTF-8	tr_TR.utf8

Understand How the Error Message Language Is Determined

For Presentation Services, the error message language is set based on the NQ_SESSION.WEBLANGUAGE session variable.

Presentation Services provides a default value for this variable upon installation. The value is updated when a user selects a language on the Oracle Analytics sign-in page.

For other clients, including third-party clients, the error message language is determined by the following precedence model:

- The error message language is set based on the WEBLANGUAGE session variable.
- If the WEBLANGUAGE session variable isn't set, then the error message language is based on the error language that is specified in the ODBC Data Source Name (DSN) that's used to access Presentation Services.
- If an error message language hasn't been set in the ODBC DSN, then the language that's specified in the ORACLE_BI_LANG environment variable is used for error messages.

  To change the value of ORACLE_BI_LANG, update the character code for this variable in NQSConfig.INI. You can view the character codes for supported languages in the `ORACLE_HOME/bi/bifoundation/server/locale` directory (for example, "en" for English, or "pt-BR" for Portuguese/Brazilian).
- If the ORACLE_BI_LANG environment variable isn't set, then error messages are displayed in English.

Note that the client for the Model Model Administration Tool doesn't set the WEBLANGUAGE session variable. Therefore, the client follows the precedence model starting with the ODBC DSN error message setting.

Set the Language for Components of the Oracle BI Server

The ORACLE_BI_LANG variable controls which language is used to present components of the application to the user.

To display the correct language for components in the Oracle BI Server, including the contents of the obis1-diagnostic.log file, you must set the ORACLE_BI_LANG variable, as described in the following procedure.

1. Open the NQSConfig.INI file for editing at:

   ```
   BI_DOMAIN/config/fmwconfig/biconfig/OBIS
   ```
2. Insert a line to set the ORACLE_BI_LANG environment variable. The following example shows the language being set to Japanese:

   ```
   <variable id="ORACLE_BI_LANG" value="ja"/>
   ```
3. Save your changes and close the file.
4. Restart Oracle Analytics Server system components and the BI Server.

   See Start and Stop Your System .

Modify the Language of the User Interface for the Model Administration Tool

The user interface of the Model Administration Tool inherits the language that is specified for the operating system.

For example, if the operating system is set to use the French language, then all user interface elements such as menus and buttons are displayed in French in all applications including the Model Administration Tool. The locale affects items such as currency, dates and times, units displayed, and keyboard layout, which differ from user interface elements such as menus and buttons.

The recommended approach is to allow the Model Administration Tool to inherit the language from the operating system. If you must change the language for the user interface of the Model Administration Tool without changing the operating system language, then you can use the ORACLE_BI_LANG environment variable for this purpose. For information on setting that variable, see Understand How the Error Message Language Is Determined.

You can also localize the names of subject areas, tables, hierarchies, columns, and their descriptions in the Presentation layer, as described in Localize Metadata Names in the Semantic Model.

Troubleshoot the Current Locale in the Oracle BI Server

Some locations require specific troubleshooting procedures.

This section provides the following information about troubleshooting the current locale in the Oracle BI Server:

- Handle the NLS Locale Not Supported Error Message
- Set the Japanese Locale on AIX Systems

Handle the NLS Locale Not Supported Error Message

If you don't have the appropriate locale installed, then the Oracle BI Server doesn't start, and the obis1-query.log file contains the following error:

NLS locale *xxx* isn't supported by the operating system.

where *xxx* is the locale that is specified in the NQSConfig.INI file for the SORT_ORDER_LOCALE parameter. Take the following actions to resolve this error:

- **Linux**. Install the locale that's indicated in the table displayed in Set the Locale on Linux Systems for the requested language.
- **Windows**. Add the corresponding language pack using the Regional Settings dialog.

Set the Japanese Locale on AIX Systems

AIX systems don't always interpret Japanese localization properly.

When using a Japanese localization on an AIX platform, you might discover that the Oracle BI Server doesn't start. If you encounter this issue, then use the following procedure.

1. Ensure that the JA_JP.UTF-8 locale is installed. If it's not, then install it.
2. Open the NQSConfig.INI file for editing at:

   ```
   BI_DOMAIN/config/fmwconfig/biconfig/OBIS
   ```

3. In the General section, set the following parameters, being aware that the settings are case-sensitive:
   - LOCALE = "Japanese";
   - SORT_ORDER_LOCALE = "Japanese";
4. Save and close the NQSConfig.INI file.

Ensure That Text for Oracle BI Server Utilities Is Displayed in the Correct Language

Ensure that usage information and error messages for the Oracle BI Server utilities are displayed in the correct language.

- For Linux environments, set the terminal encoding to UTF-8 to display multibyte characters. To do this, select the **Terminal** menu, then select **Set Character encoding**, then select **utf8**.
- For native Windows environments, set the font of the console to Lucida Console. If this option isn't displayed in the list, then first change the code page to 65001, which supports UTF-8, using the command `chcp 65001`.

Modify the Semantic Model When the Underlying Oracle Database NLS_CHARACTERSET Is Non-Unicode

When the NLS_CHARACTERSET in the Oracle Database is set to a non-unicode derived code page, you must configure an additional semantic model setting to make character sorting consistent between the Oracle Database and the Oracle BI Server.

1. In the Model Administration Tool, open the semantic model.
2. Expand the database object in the physical layer.
3. Open the Connection Pool dialog Connection Scripts tab, and add the following new connection pool script:

   ```
   alter session set NLS_SORT = unicode_binary
   ```
4. Save the semantic model, and restart the Oracle BI Server.

Localize Metadata Names in the Semantic Model

You can use the Externalize Strings utility in the Model Administration Tool to localize the names of subject areas, tables, hierarchies, columns, and their descriptions in the Presentation layer.

You can save these text strings to external files with ANSI, Unicode, and UTF-8 encoding options.

1. Open the semantic model in the Model Administration Tool.
2. Right-click any Presentation layer object, such as a subject area, presentation table, or presentation column, and select either **Externalize Display Names** then **Generate Custom Names**, or **Externalize Descriptions** then **Generate Custom Descriptions** to externalize strings. Note that when you select **Generate Custom Names** and then run the Externalize Strings utility, the translation key also appears in the Externalize Strings dialog.

   Selecting one of these right-click externalization options automatically selects the **Custom display name** or **Custom description** options in the Properties dialog for the selected object and all of its child objects.

   For example, if you right-click a subject area and select one of the externalization options, then the externalization flag is set on all presentation tables, columns, hierarchies, and levels within that subject area.

3. Select **Tools**, then select **Utilities**.
4. Select **Externalize Strings** and click **Execute**.
5. In the Externalize Strings dialog, select one or more subject areas in the left pane.

   In the right pane, the translated values and the original strings (names and descriptions) are displayed. These are placed in session variables for use by Presentation Services.

   Only those objects with the externalization flag set in the Presentation layer are displayed in the right pane.
6. Click **Save**.
7. In the Save As dialog, do one of the following:
   - If you selected a single subject area, then select a type of file and an encoding value and click **Save**.
   - If you selected multiple subject areas and want each one externalized in its own XML file, then select a directory name and press SHIFT while clicking **Save**. Each subject area is saved to an XML file with Unicode encryption.
8. In the Externalized Strings dialog, click **Close**.
9. (Optional) To disable externalization, right-click a Presentation layer object and select **Externalize Display Names**, then **Disable Externalization**, or **Externalize Descriptions** then **Disable Externalization**.

   Selecting one of these options automatically deselects the **Custom display name** or **Custom description** options in the Properties dialog for the selected object and all of its child objects.

When you've created the string files using the Externalize Strings utility, you can use the files to translate the strings for the metadata objects, as described in the following procedure.

1. Open each string file and examine the columns:
   - The first column contains the actual semantic model object names, which have a prefix of their type.
   - The second column contains the session variables that correspond to the name of each object or description, with a default prefix of CN_ for custom names and CD_ for custom descriptions.
   - The third column contains the translation keys that correspond to the name of each object.
2. Add a fourth column called Language. In this column, specify the code for the language in which the name was translated, such as de.
3. Load each string file into a database table.
4. In the Model Administration Tool, import the table into the physical layer.
5. Load the translated strings using row-wise initialization blocks. Ensure that you set the target of the initialization block to **Row-wise initialization** and that the execution precedence is set correctly.

   For example, you could do the following:

   a. Create a session initialization block that has the data source from a database, using a SQL statement such as the following one:

   ```
   SELECT 'VALUEOF(NQ_SESSION.WEBLANGUAGE)' FROM DUAL
   ```

b. In the Session Variable Initialization Block dialog for SET Language, specify the LOCALE session variable for the Variable Target.

   This specification ensures that whenever a user signs in, the WEBLANGUAGE session variable is set. Then this variable sets the LOCALE variable using the initialization block.

c. Create another session initialization block that creates a set of session variables using a database-specific SQL statement such as the following one in the Session Variable Initialization Block Data Source dialog:

```
select SESSION_VARIABLE, TRANSLATION from external where LANGUAGE =
'VALUEOF(NQ_SESSION.LOCALE)'
```

   This block creates all the variables whose language matches the language that the user specified during sign-in.

d. In the Session Variable Initialization Block Variable Target dialog, set the target of the initialization block to **Row-wise initialization**.

e. In the Execution Precedence area of the Session Variable Initialization Block dialog, specify the previously created initialization block, so that the first block that you created earlier is run first.

6. Save your changes.

> **Tips:**
>
> For information on the language for the Model Administration Tool, see Modify the Language of the User Interface for the Model Administration Tool.
>
> If you have an Oracle Application Development Framework data source, then you can propagate labels and tooltips from that data source, instead of using the Externalize Strings utility. See *Managing Metadata Repositories for Oracle Analytics Server*.

Support Multilingual Data

Oracle Analytics supports several language translations.

This section describes how you configure field information to display in multiple languages, and contains the following topics:

- What Is Multilingual Data Support?
- What Is Lookup?
- What Is Double Column Support?
- Design Translation Lookup Tables in a Multilingual Schema
- Create Logical Lookup Tables and Logical Lookup Columns
- Create Physical Lookup Tables and Physical Lookup Columns
- Support Multilingual Data in Essbase Through Alias Tables
- Enable Lexicographical Sorting

For information about using the Model Administration Tool, see *Managing Metadata Repositories for Oracle Analytics Server*.

What Is Multilingual Data Support?

Multilingual data support is the ability to display data from database schemas in multiple languages.

Oracle Analytics supports multilingual schemas by simplifying the administration and improving query performance for translations. Multilingual schemas typically store translated fields in separate tables called lookup tables. Lookup tables contain translations for descriptor columns in several languages, while the base tables contain the data in the base language. Descriptor columns provide a textual description for a key column where there is a logical one-to-one relationship between the descriptor column and the key column. An example of a descriptor column might be Product_Name, which provides textual descriptions for a Product_Key column.

What Is Lookup?

Lookup is when a query joins the base table and lookup table to obtain the translated values for each row in the base table.

Lookup tables might be dense and sparse in nature. A dense lookup table contains translations in all languages for every record in the base table. A sparse lookup table contains translations for only for some records in the base tables. Sometimes it is also possible that lookup tables are both dense and sparse. For example, a lookup table might contain complete translation for the Product Description field but only partial translation for the Product Category field. Dense and Sparse are types of lookup operation rather than being a table property. You configure lookup tables using the Model Administration Tool.

What Is Double Column Support?

Double column support is the ability to associate two columns (a descriptor ID column and a descriptor column) in the logical layer, and can help you to define language independent filters.

When the user creates a filter based on a descriptor column, the query tool displays a list of values to the user that are selected from the descriptor column.

This descriptor column technique is also useful when dealing with queries that involve LOB data types such as CLOBs and BLOBs and aggregate functions such as `COUNT` or `SUM`. Some data sources don't allow LOB columns to be used in the `GROUP BY` clause. So, instead of adding the LOB column to the `GROUP BY`, it's necessary to group by some other column that has a one-to-one relationship with the LOB column and then in join the LOB column after the aggregates have been computed.

Design Translation Lookup Tables in a Multilingual Schema

There are two common techniques of designing translation lookup tables in a multilingual schema as follows:

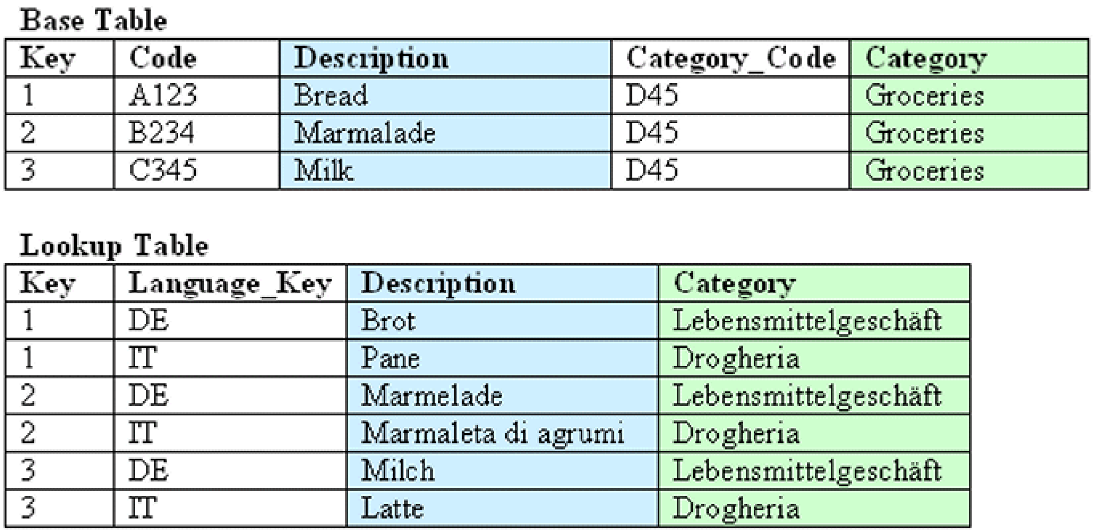
- A Lookup Table for Each Base Table
- A Lookup Table for Each Translated Field

A Lookup Table for Each Base Table

There is often a separate lookup table for each base table. The lookup table contains a foreign key reference to records in the base table, and contains the values for each translated field in a separate column.

Assuming a completely dense lookup table, the number of rows in the lookup table for a particular language equals the number of rows in the base table.

The example in the figure below shows each record in the lookup table matching only one row in the base table.



Base Table

Key	Code	Description	Category_Code	Category
1	A123	Bread	D45	Groceries
2	B234	Marmalade	D45	Groceries
3	C345	Milk	D45	Groceries

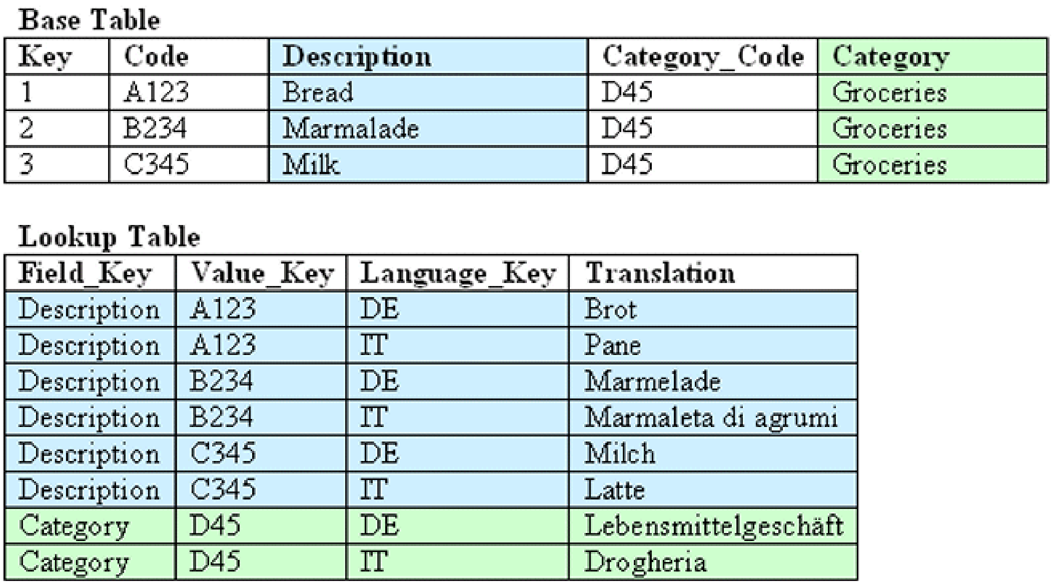
Lookup Table

Key	Language_Key	Description	Category
1	DE	Brot	Lebensmittelgeschäft
1	IT	Pane	Drogheria
2	DE	Marmelade	Lebensmittelgeschäft
2	IT	Marmaleta di agrumi	Drogheria
3	DE	Milch	Lebensmittelgeschäft
3	IT	Latte	Drogheria

A Lookup Table for Each Translated Field

The alternative approach to having one lookup table for each base table involves a separate lookup table for each translated field, as shown in the figure below.

Getting the translated value of each field requires a separate join to a lookup table. In practice there is often just one physical table that contains translations for multiple fields. When a single table contains translations for multiple fields, you must place a filter on the lookup table to restrict the data to only those values that are relevant to a particular column in the base table.



Base Table

Key	Code	Description	Category_Code	Category
1	A123	Bread	D45	Groceries
2	B234	Marmalade	D45	Groceries
3	C345	Milk	D45	Groceries

Lookup Table

Field_Key	Value_Key	Language_Key	Translation
Description	A123	DE	Brot
Description	A123	IT	Pane
Description	B234	DE	Marmelade
Description	B234	IT	Marmaleta di agrumi
Description	C345	DE	Milch
Description	C345	IT	Latte
Category	D45	DE	Lebensmittelgeschäft
Category	D45	IT	Drogheria

Create Logical Lookup Tables and Logical Lookup Columns

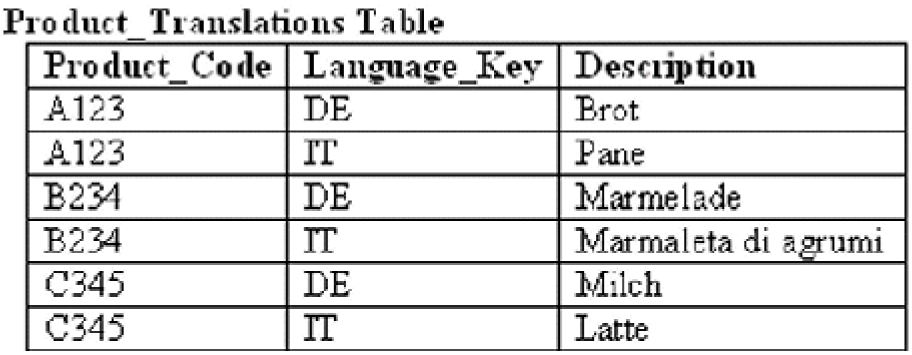
This section describes creating logical lookup tables and lookup columns and contains the following topics:

- Create Logical Lookup Tables
- Designate a Logical Table as a Lookup Table
- About the LOOKUP Function Syntax
- Create Logical Lookup Columns

Create Logical Lookup Tables

You create a logical lookup table object in the business model to define the necessary metadata for a translation lookup table.

A lookup table is a logical table with a property that designates it as a lookup table, as described in Designate a Logical Table as a Lookup Table. The figure below provides an example of a lookup table.



Product_Translations Table

Product_Code	Language_Key	Description
A123	DE	Brot
A123	IT	Pane
B234	DE	Marmelade
B234	IT	Marmaleta di agrumi
C345	DE	Milch
C345	IT	Latte

- Each of the lookup table's primary keys are considered together as a Lookup Key and perform the lookup. The lookup can be performed only when the values for all lookup key columns are provided. For example, in the figure above, the combined Product_Code and Language_Key form the primary key of this lookup table.
- A lookup key is different from a logical table key because lookup key columns are order sensitive. For example, Product_Code and Language_Key are considered a different lookup key to Language_Key and Product_Code. You can specify the order of lookup key columns in the Model Administration Tool. All columns of the lookup key must be joined in the lookup function.
- A lookup table has only one lookup key.
- A lookup table has at least one value column. In the figure above, the value column is Description, and it contains the translated value for the product description.
- There must be a functional dependency from a lookup key to each value column. In other words, the lookup key can identify the value column. The lookup key and value column should both belong to the same physical table.
- A lookup table is standalone without joining to any other logical tables.

  Consistency checking rules are relaxed for lookup tables, such that if a table is designated as a lookup table, it need not be joined with any other table in the subject area (logical tables would normally be joined to at least one table in the subject area).
- The aggregation results when using lookup columns should match the results from the base column. For example, the following code

  ```
  SELECT product.productname_trans, sales.revenue FROM snowflakesales;
  ```

  should return the same results as

  ```
  SELECT product.productname, sales.revenue FROM snowflakesales;
  ```

  If the lookup table productname_trans in this example uses the lookup key ProductID and LANGUAGE, then both queries return the same aggregation results.

  If the lookup key contains a column with a different aggregation level to productname, then the query grain changes and this affects the aggregation.

Designate a Logical Table as a Lookup Table

A logical table must be designated as a lookup table (using the Model Administration Tool) before you can use it as a lookup table.

To designate a logical table as a lookup table, you must first import the lookup table into the physical layer and drop it into the Business Model and Mapping layer using the Model Administration Tool. Then, for each logical lookup table, you must select the **Lookup table** option in the Logical Table dialog.

The order in which the columns are specified in the lookup table primary key determines the order of the corresponding arguments in the `LOOKUP` function.

For example, if the lookup table primary key consists of the RegionKey, CityKey, and LanguageKey columns, then the matching arguments in the `LOOKUP` function must be specified in the same order. You use the Model Administration Tool to change the order of primary key columns.

About the LOOKUP Function Syntax

A LOOKUP function is typically used in the Business Model and Mapping layer, as an expression in a translated logical table column.

The syntax of the LOOKUP function is as follows:

```
Lookup ::= LookUp([DENSE] value_column, expression_list ) | LookUp(SPARSE
value_
column, base_column, expression_list )

expression_list ::= expr {, expression_list }

expr ::= logical_column | session_variable | literal
```

For example:

```
LOOKUP( SPARSE SnowflakeSales.ProductName_TRANS.ProductName,
SnowflakeSales.Product.ProductName, SnowflakeSales.Product.ProductID,
VALUEOF(NQ_SESSION."LANGUAGE"))

LOOKUP( DENSE SnowflakeSales.ProductName_TRANS.ProductName,
SnowflakeSales.Product.ProductID, VALUEOF(NQ_SESSION."LANGUAGE"))
```

Note the following:

- A LOOKUP function is either dense or sparse, and is specified using the keyword DENSE or SPARSE. The default behavior is dense lookup, if neither DENSE or SPARSE is specified. For DENSE lookup, the translation table is joined to the base table through an inner join, while for SPARSE lookup, a left outer join is performed.
- The first parameter (the parameter after the DENSE or SPARSE keyword) must be a valid value column from a valid lookup table that's defined in the logical layer.
- If the SPARSE keyword is given, then the second parameter must be a column that provides the base value of the value_column. For DENSE lookup, this base column isn't required.
- The number of expressions in the expression_list must be equal to the number of the lookup key columns that are defined in the lookup table, which is defined by the value_column. The expression that's specified in the expression list must also match the lookup key columns one by one in order.

  For example:

  - The lookup key for lookup table ProductName_TRANS is both Product_code and Language_Key
  - The expressions in expression_list are SnowflakeSales.Product.ProductID and VALUEOF(NQ_SESSION."LANGUAGE")
  - The meaning of the lookup is:

    return the translated value of ProductName from the translation table with the condition of Product_code = SnowflakeSales.Product.ProductID and Language_Key = VALUEOF(NQ_SESSION."LANGUAGE")

Create Logical Lookup Columns

You use the Expression Builder in the Model Administration Tool to create a logical column that includes the lookup function.

The value of the logical column depends on the language that is associated with the current user.

You create a new logical column using a derived column expression in the Column Source tab, for example to get the translated product name:

`LAN_INT` is a session variable that's populated by the session initialization block MLS and represents either the base language or other languages:

- 0 for base language (for example, en - English)
- 1 for other language codes (for example, fr - French, or cn - Chinese)

`WEBLANGUAGE` is a session variable that is initialized automatically, based on the language selected when a user logs in.

The `INDEXCOL` function helps to select the appropriate column. In the preceding example, the expression returns the value of the base column (ProductName) only if the user language is the base language (that is, when the value of session variable `LAN_INT` is 0). If the user language isn't the base language (when the value of the session variable `LAN_INT` is 1), then the expression returns the value from the lookup table of the language that's passed in the `WEBLANGUAGE` session variable.

When you use the `DENSE` function (shown in the previous example), if there's no value for a column in the translated language, then the lookup function displays a blank entry.

When you use the `SPARSE` function (shown in the following example), and there is no value for a column in the translated language, then the lookup function displays a corresponding value in the base language.

```
INDEXCOL( VALUEOF(NQ_SESSION."LAN_INT"), "Translated Lookup Tables"."Product".
"ProductName", LOOKUP( SPARSE "Translated Lookup Tables"."Product
Translations".
"ProductName", "Translated Lookup Tables"."Product"."ProductName",
"Translated
Lookup Tables"."Product"."ProductID", VALUEOF(NQ_SESSION."WEBLANGUAGE")))
```

Tips for Using Derived Logical Columns

Using derived logical columns requires planning to reduce errors.

When working with logical lookup columns, keep the following tips in mind:

- You can't use a derived logical column that's the result of a `LOOKUP` function as part of a primary logical level key. This limitation exists because the `LOOKUP` operation is applied after aggregates are computed, but level key columns must be available before the aggregates are computed because they define the granularity at which the aggregates are calculated.

  You can use a derived logical column that's the result of a `LOOKUP` function as a secondary logical level key.
- For a derived logical column that has lookup functions in its derived expression:

  - The logical columns used in the lookup function shouldn't have their associated levels below the level of the derived logical column itself.
  - Configuring a descriptor ID column is the recommended approach.

Handle Non-ISO Type Language Codes

If the data has non-ISO type language codes in the tables, then there should be a table that maps ISO language codes to non-ISO language codes.

You can use the preexisting `WEBLANGUAGE` variable that sets the ISO language code when a user logs in. You define a separate `LANGUAGE` variable whose initialization block runs a query against a mapping table to fetch the non-ISO language code filtered by the value from the `WEBLANGUAGE` variable. The table below provides a mapping table for non-ISO language codes. `LANGUAGE` is a non-ISO language code.

WEBLANGUAGE	LANGUAGE	LAN_INT
en	ENG	0
cn	CHI	1
fr	FRA	1

Create Physical Lookup Tables and Physical Lookup Columns

You can create physical lookup table objects in the business model to define the necessary metadata for translation lookup tables. Physical lookup tables are similar to logical lookup tables in both semantics and usage.

Physical lookup tables address the following scenarios that logical lookup tables can't handle:

- The lookup table source is fragmented. In this case, use multiple physical lookup tables to hold the values. For example, translation values for fragmented product name data can be distributed in two physical lookup tables called productname_trans_AtoM and productname_trans_NtoZ.
- Different levels of translation tables are available. For example, translations are available in both an Essbase data source and a relational data source. It's preferable to use the same source as the base query.

Unlike logical lookup tables, which you designate by selecting an option in the Logical Table dialog, you configure physical lookup tables by constructing lookup functions in the logical table source mapping.

For example, suppose that you've the following physical tables:

- A base table called Categories, with columns such as categoryid and categoryname.
- A translation table called Categories_Trans, with columns such as categoryid, language_key, and categoryname. The translated value of categoryname is determined through a combination of the categoryid and language_key columns.

Suppose that you've a logical table called Categories. In that table, you add a new logical column called categoryname_p, which is a translation column that depends on the current language. The column isn't derived from any other logical column (unlike logical lookup columns).

The following procedure explains how to configure a physical lookup translation column using the previous example.

1. Open the semantic model in the Model Administration Tool.
2. In the Business Model and Mapping layer, create a new logical column by right-clicking the appropriate logical table (for example, Categories) and selecting **New Object**, then **Logical Column**.
3. Provide a name for the logical column (for example, categoryname_p).
4. Select the Column Source tab.
5. In the **Logical Table Source** box under **Derived from physical mappings**, double-click the logical table source object that contains the base table column. The Column Mapping tab of the Logical Table Source dialog is displayed.
6. Ensure that **Show unmapped columns** is selected.
7. In the **Expression** column for the new logical column (for example, categoryname_p), enter an expression such as the following:

   ```
   INDEXCOL(VALUEOF(NQ_SESSION."LAN_INT"),
   "DB_Name"."My_Category"."My_Schema"."Categories"."CategoryName",
   LOOKUP(SPARSE
   "DB_Name"."My_Category"."My_Schema"."CATEGORIES_TRANS"."CATEGORYNAME",
   "DB_Name"."My_Category"."My_Schema"."Categories"."CategoryName",
   "DB_Name"."My_Category"."My_Schema"."Categories"."CategoryID",
   VALUEOF(NQ_SESSION."LANGUAGE")))
   ```

   You can also use Expression Builder to create the expression.
8. Click **OK** in the Logical Table Source dialog.
9. Click **OK** in the Logical Column dialog.
10. Save your changes.

The Categories_trans physical translation table doesn't need to be incorporated into the logical table source. The `INDEXCOL` function checks that if the `LAN_INT` session variable is 0, then the categoryname column is fetched from the base table. Note the following about the `LOOKUP` function:

- The physical `LOOKUP` function works the same as a logical `LOOKUP` function. The only difference is that all the references to logical tables and columns are replaced by physical tables and columns.
- The first column of the `LOOKUP` function is a value column, which is a translation value column from a translation table. The second column is the base value column, if a sparse lookup exists. The remaining columns are columns or values to be joined to the physical translation table, which is the table that's implied by the value column of the `LOOKUP` function.

Because you can't use a dialog to configure a physical lookup table, you must ensure that the order of the join columns and values is compatible with the column sequence that's displayed in the Physical Table dialog for the physical translation table. For example, on the Keys tab of the Physical Table dialog for the Categories_trans table, the primary key is composed of the CategoryID and Language_Key columns.

The columns that are specified in the `LOOKUP` function correspond to these columns:

- The following line:

  ```
  "DB_Name"."My_Category"."My_Schema"."Categories"."CategoryID"
  ```

corresponds to the Categories_trans.CategoryID column.

- The following line:

  ```
  valueof(NQ_SESSION."LANGUAGE")
  ```

  corresponds to the Categories_trans.Language_key column.

See Create Logical Lookup Tables and Logical Lookup Columns for information about lookup concepts like the `LAN_INT` and `LANGUAGE` session variables and full syntax information for the `LOOKUP` function.

Support Multilingual Data in Essbase Through Alias Tables

Often, members in Essbase cubes have separate aliases for each user language to enable users to view member names in their own language.

Typically, you define a session variable to dynamically select the appropriate alias upon user login. See *Managing Metadata Repositories for Oracle Analytics Server* for information about Essbase alias tables and how to use them with session variables.

Enable Lexicographical Sorting

Lexicographical sorting is the ability to sort data in alphabetical order.

Most data sources support lexicographical sorting. However, if you notice that lexicographical sorting isn't working properly for a particular data source, then you can configure the Oracle BI Server to perform the sort rather than the back-end data source. To perform this configuration, ensure that ORDERBY_SUPPORTED isn't selected in the Features tab of the Database dialog in the Model Administration Tool. See *Managing Metadata Repositories for Oracle Analytics Server* for information about specifying database features.

Note that disabling ORDERBY_SUPPORTED in the data source can have a very large performance impact, because consequently, many joins aren't pushed down to the data source. In many cases, the performance impact is significant enough that ORDERBY_SUPPORTED can still be enabled in the data source, regardless of the impact on the lexicographical sorting functionality.

17
Configure Currency Options

This chapter describes how to configure currency options. When content designers create analyses, they often include data that shows currency, such as American dollars. As the administrator, you can perform various tasks that affect currency options that are available to users.

Topics:

- Change the Default Currency for Analyses
- Define User-Preferred Currency Options

Change the Default Currency for Analyses

You can change the default currency that is displayed, for example, from French Francs to Euros.

You can also set in the Console, see Format Options.

For information about using formatting functions in Answers, see *Visualizing Data in Oracle Analytics Server*.

1. Open the `currencies.xml` file in the directory `*ORACLE_HOME*/bi/bifoundation/web/display`.
2. Search for the currency to make the default, for example, USD, CAD, PEN, or MAD.
3. Copy the entire currency element.

   For example, copy the currency tag for the Euro:

   ```
   - <Currency tag="int:euro-l" type="international" symbol="_"
   displayMessage="kmsgCurrencyEuroLeft" digits="2" format="$ #">
   <negative tag="minus" format="-$ #" />
   </Currency>
   ```

4. Search for the text string int:wrhs, located near the top of the file.
5. Select the entire element and replace it by pasting the copied element over it.
6. Replace the tag attribute so it reads int:wrhs.

   For example, replace `tag="int:euro-l"` with `tag="int:wrhs"`.

7. Restart the service for Oracle BI Server.

> **Caution:**
>
> The currencies.xml file is overwritten when applying an update, a patchset, a bundle patch, or a one-off patch. If you have made changes to this file, reenter your changes in the new currencies.xml.

Specify the currency for a column in a customized subject area.

1. In Answers, modify the analysis that uses the subject area.
2. In the Analysis editor: Criteria tab, click the **Options** button for the currency column and select **Column Properties** to display the Column Properties dialog.
3. Click the Data Format tab and select the **Override Default Data Format** box.
4. In the **Treat Numbers As** box, select **Currency**.
5. In the **Currency Symbol** list, select **User's Preferred Currency**.
6. Complete the other options on the tab as appropriate.
7. If desired, save this setting as a systemwide default.
8. Click **OK** when you have finished, and repeat the preceding steps for any other columns to change.

Considerations for defining user-preferred currency:

- To create a default for all users, set the CURRENCYTAG element using an INIT BLOCK.
- If you configure a user-preferred currency, set the PREFERRED_CURRENCY element using an INIT BLOCK.

  See Define User-Preferred Currency Options.

  For more information about INIT BLOCKs, see Working with Initialization Blocks in *Managing Metadata Repositories for Oracle Analytics Server*.

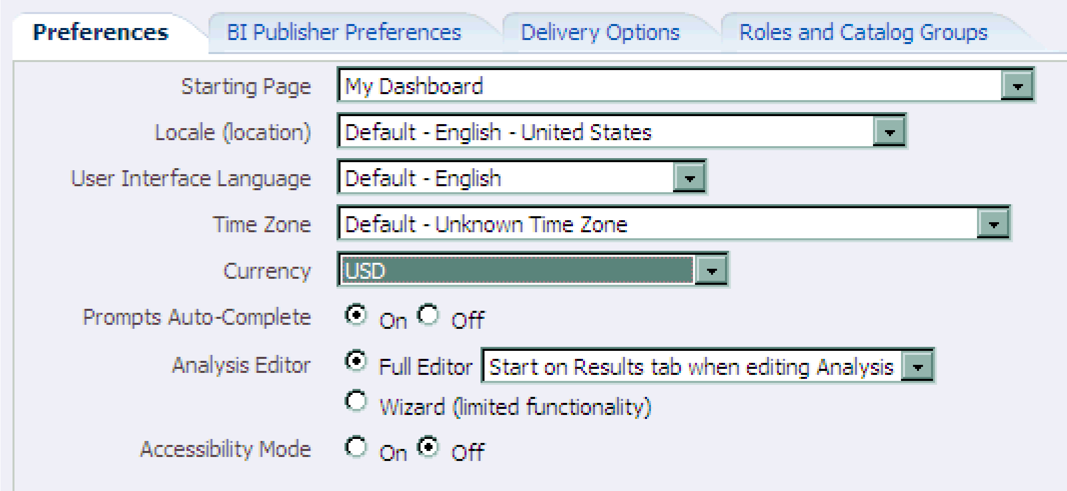
Define User-Preferred Currency Options

You can select the type of currency you prefer to view currency columns in analyses and dashboards.

You can also set in the Console, see Format Options.

Select the currency type you want in one of two ways:

- In the **Currency** box on the My Account dialog: Preferences tab
- In currency prompts



Starting Page	My Dashboard
Locale (location)	Default - English - United States
User Interface Language	Default - English
Time Zone	Default - Unknown Time Zone
Currency	USD
Prompts Auto-Complete	● On ○ Off
Analysis Editor	● Full Editor Start on Results tab when editing Analysis ○ Wizard (limited functionality)
Accessibility Mode	○ On ● Off

For information about setting the currency preference on the My Account dialog: Preferences tab or about currency prompts, see *Visualizing Data in Oracle Analytics Server*.

You define the currency options that are to be displayed in the **Currency** box and in a currency prompt in the userpref_currencies.xml file. (These currency options must be for currencies to which your installation can convert columns.) Defining the currency options also controls whether the **Currency** box is available on the My Account dialog: Preferences tab and whether the **Currency Prompt** option is available on the Definition pane of the Prompt editor.

When you define these currency options, you can use one of two types of mappings:

- **Static** — Produces a static list of currency options that all users see.
- **Dynamic** — Produces a list of currency options that changes dynamically based on a logical SQL statement that you specify. This is useful, for example, to dynamically change the currency options so that each user sees a list specific to his or her account. Each user sees a subset of the total options available. For example, one group of users might want to select from only Global Currency 1 or Global Currency 2 while another group might want to select from different options.

> **Note:**
>
> For the user-preferred currency options to take effect, the following configuration also must be done in the semantic model:
>
> - Creation of the PREFERRED_CURRENCY session variable.
> - Conversion setup of logical currency columns in the Business Model and Mapping layer
> - Creation of the userCurrencyPreference table using the currency information from your installation that enables you to dynamically change the currency options based on a logical SQL statement that you specify (required only if you use a dynamic mapping)
>
> See Configuring Logical Columns for Multicurrency Support in *Managing Metadata Repositories for Oracle Analytics Server*.

You can set the contents of the user's preferred currency by using the PREFERRED_CURRENCY session variable. You define the UserCurrencyPreferences element in any one of the following files:

- instanceconfig.xml file, or in userpref_currencies.xml files located in:

  `BI_DOMAIN/config/fmwconfig/biconfig/OBIPS`

- *yourfilename*, where the instanceconfig.xml file contains an entry pointing to your file. For example,

  ```
  <UserprefCurrenciesConfigFile>yourpath</UserprefCurrenciesConfigFile>
  ```

  where, *yourpath* is the location of the file.

Define User-Preferred Currency Options Using a Static Mapping

You can use a mapping to define a static list of options that all users see for selecting currency.

1. Use a text editor to open the `userpref_currencies.xml` file located in the following directory:

   `BI_DOMAIN/config/fmwconfig/biconfig/OBIPS`

2. Add a UserCurrencyPreferences element as follows:

   ```
   <UserCurrencyPreferences currencyTagMappingType="static">

   </UserCurrencyPreferences>
   ```

3. For each currency option to be displayed in the **Currency** box or in currency prompts, add a UserCurrencyPreference element between the <UserCurrencyPreferences> tags using this format:

   ```
   <UserCurrencyPreference sessionVarValue="sessionVarValuevalue"
   displayMessage="displayMessagevalue" displayText="displayTextvalue"
   currencyTag="currencyTagvalue"/>
   ```

   In this format:

   - `sessionVarValue="sessionVarValue "` sets the session variable PREFERRED_CURRENCY. For its value, specify a string that uniquely identifies the currency, for example, `gc1`.
   - (optional) `displayMessage="displayMessagevalue"` sets the presentation variable currency.userPreference to a localized value. To specify a localized value, you first must create the localized message for the currency in the usercurrencymessages.xml file. See Localize Messages for Users' Preferred Currency. Then, for the value of displayMessage, specify the WebMessage name that's identified in the usercurrencymessages.xml file for the currency. For example, if you created this English language entry in the usercurrencymessages.xml file:

     ```
     <WebMessage name="kmsgMyCurrency1"><TEXT>My Currency 1</TEXT></
     WebMessage>
     ```

     Then you would specify kmsgMyCurrency1 as the value of displayMessage.
   - (optional) `displayText="displayTextvalue"` sets the presentation variable currency.userPreference to a value that isn't localized. For its value, specify a string that identifies the currency, such as Global Currency 2.

     For more information about the currency.userPreference variable, see *Visualizing Data in Oracle Analytics Server*
   - `currencyTag="currencyTagvalue"` identifies the Currency Tag in the currencies.xml file whose displayMessage value is to be used to populate the Currency box on the My Account dialog: Preferences tab and currency prompts. (The currencies.xml file, which is located in *ORACLE_HOME*/bi/bifoundation/web/display, provides currency formats.)

Note:

The value of the currency.userPreference variable is obtained from the displayMessage and displayText attributes of the UserCurrencyPreference element using the following order of precedence:

a. displayText

b. displayMessage

If no values exist for displayText and displayMessage, then the value of the displayMessage attribute for the corresponding currency tag in the currencies.xml file is used.

4. Save and close the userpref_currencies.xml file.

5. Restart Oracle Analytics Server.

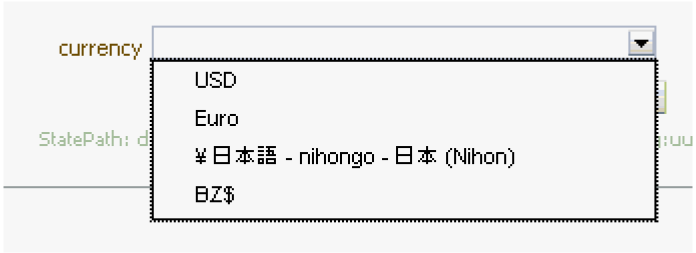
Example: Static Mapping to Define User-Preferred Currency Options

The following example shows a userpref_currencies.xml file that uses a static mapping to define user-preferred currency options.

You can also set in the Console, see Format Options.

```
<UserCurrencyPreferences currencyTagMappingType="static">
   <UserCurrencyPreference sessionVarValue="gc1" displayText="Global Currency
1" currencyTag="int:USD" />
   <UserCurrencyPreference sessionVarValue="gc2" displayText="Global Currency
2" currencyTag="int:euro-l" />
   <UserCurrencyPreference sessionVarValue="gc3" displayText="Global Currency
3" currencyTag="loc:ja-JP" />
   <UserCurrencyPreference sessionVarValue="orgc" displayText="Org Currency"
currencyTag="loc:en-BZ" />
</UserCurrencyPreferences>
```

The figure below shows how these values from the userpref_currencies.xml file are displayed in a drop-down list of currency options for a prompt on a dashboard page. The drop-down list is similar to what is displayed for the **Currency** box on the My Account dialog: Preferences tab.



Define User-Preferred Currency Options Using a Dynamic Mapping

You can use a mapping to define a dynamic list of options that users see for selecting currency.

The list changes dynamically based on a logical SQL statement that you specify. This is useful, for example, to dynamically change the currency options based on the user.

1. Use a text editor to open the userpref_currencies.xml file that is located in the following directory:

   BI_DOMAIN/config/fmwconfig/biconfig/OBIPS

2. Add a UserCurrencyPreferences element as follows:

   ```
   <UserCurrencyPreferences currencyTagMappingType="dynamic">

   </UserCurrencyPreferences>
   ```

3. Add a UserPrefCurrencyLogicalSQL element between the <UserCurrencyPreferences> tags using this format:

   ```
   <UserPrefCurrencyLogicalSQL>
   SELECT column1, column2, column3 FROM userCurrencyPreference
   </UserPrefCurrencyLogicalSQL>
   ```

   In this format:

   - *column1* contains the values that are used to set the session variable PREFERRED_CURRENCY. Each value in this column is a string that uniquely identifies the currency, for example, `gc1`.
   - *column2* contains the currency tags in the currencies.xml file whose displayMessage values are to be used to populate the Currency box and currency prompts, for example, `int:euro-1`. (The currencies.xml file, which is located in `ORACLE_HOME/bi/bifoundation/web/display`, provides currency formats.)
   - (optional) *column3* contains the values used to set the presentation variable currency.userPreference. Each value in this column is a string that identifies the currency, such as Global Currency 2.

     > **Note:**
     >
     > If you omit *column3*, then the values for the displayMessage attributes for the corresponding currency tags in the currencies.xml file are used.

     For more information about the currency.userPreference variable, see *Visualizing Data in Oracle Analytics Server*

4. Save and close the userpref_currencies.xml file.
5. Restart Oracle Analytics Server.

Example: Dynamic Mapping to Define User-Preferred Currency Options

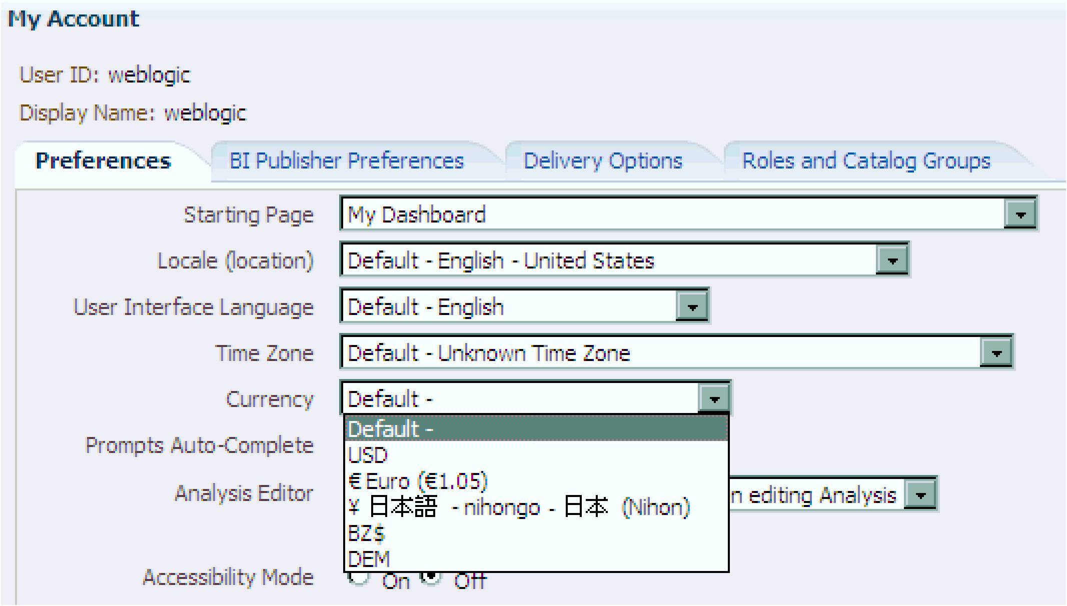
The following example shows a userpref_currencies.xml file that uses a dynamic mapping to define user-preferred currency options.

```
<UserCurrencyPreferences currencyTagMappingType="dynamic">
UserPrefCurrencyLogicalSQL>
<!-- In this SELECT statement, column1 contains the values to set the
PREFERRED_CURRENCY variable, column2 contains the currency tag values, and
column3 contains the values to set the currency.userPreference variable. -->
SELECT markets.userpreferences, markets.currencyTag,
markets.userpreferencename FROM userCurrencyPreference
</UserPrefCurrencyLogicalSQL>
</UserCurrencyPreferences>
```

The table below shows sample results from the logical SQL statement.

"Markets"."UserPreference"	"Markets"."CurrencyTag"	"Markets"."UserPreferenceName"
varchar	varchar	varchar
orgc1	loc:en-BZ	Org currency
gc2	int:euro-1	Global currency 2
lc1	int:DEM	Ledger currency
gc1	int:USD	Global Currency 1

The figure below shows how the values that are generated dynamically from the SQL statement in the userpref_currencies.xml file are displayed in a drop-down list of currency options for the **Currency** box on the Preferences tab of the My Account dialog. The drop-down list is similar to what is displayed for a prompt on a dashboard page.



User ID: weblogic

Display Name: weblogic

Starting Page: My Dashboard

Locale (location): Default - English - United States

User Interface Language: Default - English

Time Zone: Default - Unknown Time Zone

Currency: Default -

Prompts Auto-Complete

Analysis Editor: n editing Analysis

Accessibility Mode: ○ On ● Off

18

Manage How Content Is Indexed and Searched

Administrators can set up how data sources and catalog content are indexed and crawled so that users find the latest content when they search or create visualizations from the search bar on the Home page.

Topics:

- Configure Data Model Search Indexing
- Schedule Regular Content Crawls
- Monitor Search Crawl Jobs
- Certify a Dataset to Enable Users to Search It from the Home Page

Configure Data Model Search Indexing

You can configure how and when subject areas are indexed from the **Data Model** tab.

1. In the Oracle Analytics Home page, click the **Navigator**, and then click **Console**.
2. In the Console under Configuration and Settings, click **Search Index**.
3. In the Data Model tab, click **Enable Data Model Crawl**.
4. In **User to Run Crawl As**, click **Search** and enter a user that is a member of the **BIDataModelAuthor** or **BIServiceAdministrator** application roles.
5. Select a language or hold down `Ctrl` to select multiple languages.
6. Under **Select Data Models to Index** select a **Crawl Status** for your data models and columns.
7. Click **Save**.
8. Optional: Click **Run Now** to rebuild the search index with your changes.

Schedule Regular Content Crawls

The administrator selects which folders to crawl and schedules when and how often to crawl the content.

1. In the Oracle Analytics Home page, click the **Navigator**, and then click **Console**.
2. Click **Search Index**.
3. Select **Data Model** or **Catalog**.
4. Use the **Schedule** options to specify when and how often to run the crawl.

   The index updates automatically as users add or modify content in the catalog.

- **Catalog crawl frequency**: By default, a catalog crawl runs once a month. The minimum number of days you can specify between catalog crawls is 7 days.
- **Data model crawl frequency**: By default, a data model (that is, semantic model) crawl runs once a day.

Normally you don't need to change the defaults. However, in some cases you might want to schedule a crawl as needed (for example, after importing a BAR file or if automatic indexing didn't run).

5. For **Languages**, select all the languages you want to create indexes for.

   Crawl results are added to the index in the languages that you specify. For example, if your company's headquarters are in the United States, and you have offices in Italy, then you can choose English and italiano to create indexes in both English and Italian.
6. Click the **Save** icon to save your changes.

Monitor Search Crawl Jobs

Administrators can check the last time content was indexed and monitor the status of crawl jobs. You can stop any crawl job that is running, cancel the next scheduled crawl before it starts, or rerun a failed crawl.

If users report search issues, check the status of crawls to ensure that they're current. After a crawl is completed, users might have to wait a few minutes before they can locate the latest content.

1. In the Oracle Analytics Home page, click the **Navigator**, and then click **Console**.
2. Click **Search Index**.
3. Click **Monitor Crawls**.

   The Crawl Job Status page shows information about the past, current, and the next scheduled crawl. In the Progress column, XSA indicates a dataset.
4. Look at the **Status** column to find out when the content was last crawled and when the next crawl is due.
5. Click **Cancel** to stop a crawl job that is Running or Scheduled.
6. To rerun a crawl with the status of Terminated or that displays progress totals of zero:
   a. Click the **Configure Crawls** link.
   b. In the Data Model tab, deselect and then reselect the **Enable Data Model Crawl** checkbox.
   c. Click **Save**.
   d. Click the **Monitor Crawls** link and locate the scheduled job. The revised crawl runs in a few minutes time.

Certify a Dataset to Enable Users to Search It from the Home Page

You certify a dataset uploaded by a user so that other users can search it from the home page using the search bar.

As an administrator, you use certification to control how much compute time is consumed by indexing datasets, which can affect system performance. If your dataset is not already indexed, see Index a Dataset.

1. From the Home page, click **Navigator**, click **Data**, then click **Datasets**.
2. Hover over the dataset you’d like to certify, click **Options** ⋮, then click **Inspect**.

   If you can't see **Options**, expand the size of your browser or scroll to the right-hand side of your device screen.
3. On the General tab, click **Certify**.
4. Click **Save**.

19

Configure and Manage the Presentation Catalog

This chapter describes how to configure and manage the Presentation Catalog and provides information about basic maintenance procedures and configuring for full-text searching.

Topics:

- About the Presentation Catalog
- Use the Catalog Manager Command Line Interface
- Update Catalog Objects
- Deploy Catalogs to Production

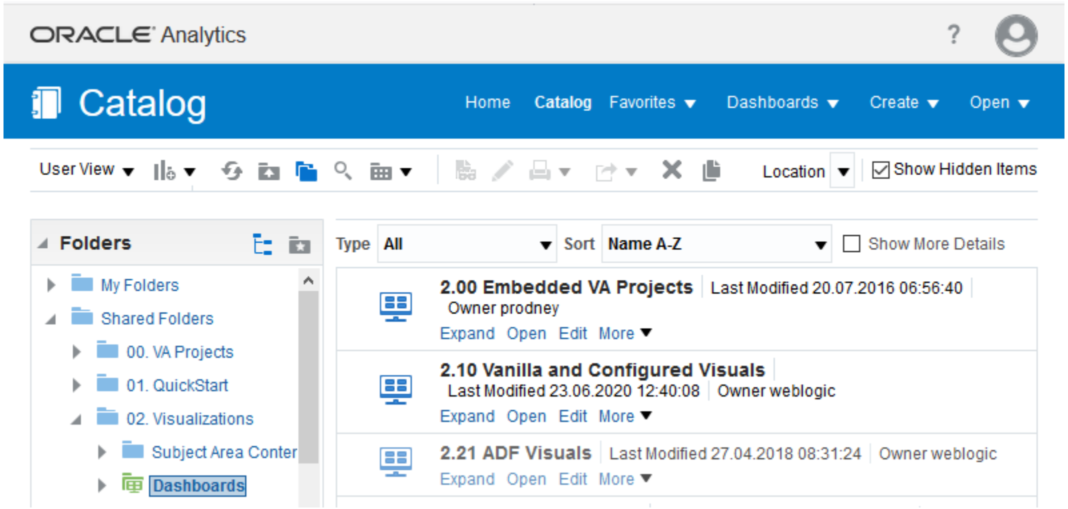
About the Presentation Catalog

The Presentation Catalog stores the content that users create.

This content includes folders, shortcuts, Oracle Analytics Server objects (such as analyses, filters, prompts, and dashboards), and Publisher objects (such as reports and templates).

Objects in the Catalog

The figure below shows sample objects in the catalog, as seen in Presentation Services.



Type All | Sort Name A-Z

2.00 Embedded VA Projects | Last Modified 20.07.2016 06:56:40 | Owner prodney

2.10 Vanilla and Configured Visuals | Last Modified 23.06.2020 12:40:08 | Owner weblogic

2.21 ADF Visuals | Last Modified 27.04.2018 08:31:24 | Owner weblogic

Guidelines for Object Names

The object name that is visible to users, such as Analysis 1, is referred to as the logical object name.

The following list provides guidelines for object names:

- No restrictions exist on which characters are allowed in the logical name of an object in the catalog, provided that the characters are valid Unicode characters. The following are valid logical names:

  ```
  Hello World
  Profit / Loss
  % Sales * $ Cost ~~ $ "Expense"?
  ```

- The length of the logical object name must not exceed 256 Unicode characters.
- The length of the logical path name for an object must not exceed 16000 Unicode characters.
- The number of directory segments in a logical path name for an object must be not exceed 255 segments.

  For example, a directory with a name such as /n1/n2/n3/n4/…./n253/n254/n255 is acceptable, while a name such as /n1/n2/n3/n4/…./n254/n255/n256 is unacceptable.

- When you pass the path name of an object using SOAP, you must escape the following characters:

  ```
  Forward slash (/)
  Backward slash (\)
  Tilde (~)
  Asterisk (*)
  Question mark (?)
  ```

  The following logical path names are all valid:

  ```
  /shared/test/Hello World
  /shared/test/Profit \/ Loss
  /shared/test/% Sales \* $ Cost \~\~ $ "Expense"\?
  ```

  Use care when building a catalog path. It is very common to see code that assumes the forward slash (/) is always a path separator. Always verify your path code with an object name such as "Profit / Loss".

- When you pass a catalog search filter using SOAP, you must escape the following characters:

  ```
  Forward slash (/)
  Backward slash (\)
  Tilde (~)
  Asterisk (*)
  Question mark (?)
  Caret (^)
  Dollar sign (?)
  ```

  The following search filters are all valid:

  ```
  Hello World
  Profit \/ Loss
  % Sales \* \$ Cost \~\~ \$ "Expense"\?
  ```

Use the Catalog Manager Command Line Interface

You use the Catalog Manager command line interface to perform advanced catalog management. For example, you can create, delete, rename, and view catalog objects.

1. On Windows, from the Start menu, select **Oracle Analytics Client Tools**, then **Catalog Manager CLI**.
2. When using the command line, change to the following directory:

   `BI_DOMAIN\bitools\bin`
3. Run the appropriate script:
   - `runcat.cmd` (on Windows)
   - `runcat.sh` (on Linux)
   - For assistance about commands run:
     `runcat.cmd - help | runcat.sh -help`
4. Enter a command.

   For example, to generate a report that lists the contents of the catalog, enter `runcat.sh -cmd report`.

Operations Available

Available operations that you can perform using the Catalog Manager command line interface are described here.

Generally, the operations available include operations and write operations that don't affect the entire catalog, such as setting permissions for an object.

You can perform the following operations:

- Cutting objects.
- Copying objects.
- Pasting objects.
- Copying objects for another catalog.
- Pasting objects from another catalog.
- Creating shortcuts of objects.
- Deleting objects.
- Renaming objects without reference updates.
- Renaming objects with reference updates (you might be able to rename certain objects, depending on your permissions.)
- Creating folders.
- Setting permissions for objects.
- Working with properties of objects.
- Searching for objects.
- Searching for and replacing catalog text (you might be able to replace certain objects, depending on your permissions).

- Creating reports to display catalog data.
- Setting browser preference.
- Exporting captions for localization purposes.

Update Catalog Objects

If you apply an update to Oracle Analytics Server or install a patch and work with objects in the catalog, then you might notice that certain objects aren't being accessed as quickly as in the previous update.

This change can occur if objects aren't updated properly. You can confirm the need to update by viewing the metrics in Fusion Middleware Control. In the Catalog folder, find a metric called "Reads Needing Upgrade" with description "The number of objects read that required upgrading." If the number is large, then you can resolve this issue by updating objects in the catalog using the Administration page in Presentation Services.

You apply an update to Oracle Analytics Server by following the instructions in *Migrating and Upgrading Oracle Analytics Server*. Oracle recommends that you update when Presentation Services isn't running. If you suspect that the updating of objects wasn't performed thoroughly, then you can update the objects yourself using the Administration page. The advantage of this approach is that Presentation Services can stay up and running during the update.

Bear the following points in mind as you prepare to update objects:

- If you're performing a rolling update of machines in a cluster, then don't use this option or the UpgradeAndExit configuration setting until all machines in the cluster are updated.
- Use this option on only one node in a cluster at a time.

1. In the global header, click **Administration**.
2. Click the **Scan and Update Catalog Objects That Require Updates** link.
3. Click **Update Catalog Objects** to begin the update process.

   Click the other links on the page to see which objects were updated and which weren't. You can view the log files for details on objects that weren't updated.

Deploy Catalogs to Production

You deploy a Catalog to production using BAR files.

See Move Oracle Analytics Server Between Environments.

Part VI

Advanced Configuration Settings

This part describes advanced configuration settings are optional and others that enable fine-tuning of a deployment..

This part includes the following chapters:

20

Configure and Manage Analyses and Dashboards

This chapter describes how to configure and manage analyses and dashboards and the objects that they contain, such as views. For information about how content designers work with analyses and dashboards, see *Visualizing Data in Oracle Analytics Server*.
End users with appropriate privileges can modify personal and shared dashboards, including the addition of pages and content. End users can't create analyses and dashboards.

Topics:

Manage Dashboards

Before you create shared dashboards, ensure that you've planned the Presentation Catalog directory or folder structure and security strategy.

In general, to create a shared dashboard, you first create the dashboard and add content using the Dashboard Builder. You can also assign permissions to access the dashboard. Users who are members of multiple application roles can select the dashboard that they display by default from all of the dashboards to which they have permissions.

The following list provides other resources with information about dashboards:

- Guidelines for creating a shared dashboard, within the broader context of the Presentation Catalog structure and security framework, are provided in Controlling Access to Saved Customization Options in Dashboards in *Managing Security for Oracle Analytics Server*.
- Information about shared folder structures in the Presentation Catalog is provided in Configure and Manage the Presentation Catalog.
- Information about permissions is provided in *Managing Security for Oracle Analytics Server*.

Perform General Configuration Tasks for Analyses

This section describes general tasks that you can perform to configure for the creation of analyses.

It includes the following sections:

- Increase Heap Size to Assist in Exports to Excel
- Manually Configure for Export
- Support Nested Folders, Navigation, and Drill-Down

Increase Heap Size to Assist in Exports to Excel

Various options are available for exporting the results of analyses, for example, exporting to Microsoft Excel.

These options are described in Exporting Results in *Visualizing Data in Oracle Analytics Server*. While users can export directly to an Excel format, for large numbers of rows you can expect similar performance if you export first to CSV, then import that file into Excel. Or you can export using the Data > Excel option to preserve the supported data types, String, Number, and Data.

If a user exports a large dataset without using the CSV format and get an out-of-memory error, they should increase the heap size for the JavaHost service. The default heap size is 1024MB. Depending on the available memory on the computer, you might want to increase the heap size for the JavaHost service. You can export using Data > Excel option which doesn't use JavaHost.

1. Open the obijh.properties file for editing. You can find the file at:

   `ORACLE_HOME/bi/modules/oracle.bi.cam.obijh/env/obijh.properties`

2. Change the existing -Xmx1024M entry (in the line starting OBIJH_ARGS=).

   set the -Xmx parameter to 2048M (or higher as necessary, depending on the available memory in the system and the size of the Excel export that you require).

3. Save and close the file.

   This affects all JavaHosts.

4. If you see an error message about a SocketTimeoutException from the com.siebel.analytics.javahost.io.ChannelWithTimeout class, then update the SocketTimeout parameter for the JavaHost service.

   Open the config.xml file for the JavaHost system component in:

   `BI_DOMAIN/config/fmwconfig/biconfig/OBIJH/config.xml`.

   Locate the MessageProcessor section and the SocketTimeout parameter, which might be commented out. Uncomment SocketTimeout if necessary and specify a higher value. For example, specify at least 300000 milliseconds.

   The config.xml file and its settings are described in Use the JavaHost Service for Presentation Services.

5. Save and close the file.

6. Restart Oracle Analytics Server.

Manually Configure for Export

You can configure various options that change how the results of analyses or views are exported.

1. Open the `instanceconfig.xml` file for editing, located in:

   BI_DOMAIN/config/fmwconfig/biconfig/OBIPS

2. Enter the following namespace declaration in the WebConfig element:

   ```
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   ```

   Note that the Export element includes the required attribute `xsi:type`, which specifies the type of export. Valid values are:

   - `excel` (for export to Microsoft Excel)
   - `formattedText` (for data export)
   - `pdf` (for export to PDF)
   - `ppt` (for export to Microsoft PowerPoint)
   - `all` (you can use `true` or `false` to choose whether to leave out charts, images, or formatting, for all print and export options)

3. Locate the Download section in which you must add the elements that are described in the table below.

4. Include the elements and their ancestor elements as appropriate, as shown in the following example:

   > **Note:**
   >
   > The default export value is `UseRawValue`. However, if you want to export rounded values, you must use the value `UseFormattedValue` instead.

   ```
   <ServerInstance>
     <Download>
       <Export xsi:type="excel">
           <DataValue>UseRawValue</DataValue>
           <RepeatRows>false</RepeatRows>
       </Export>

       <Export xsi:type="formattedText">
           <Delimiter char=","/>
       </Export>

       <Export xsi:type="pdf">
           <KeepRowsTogether>true</KeepRowsTogether>
           <Orientation>Landscape</Orientation>
       </Export>

       <Export xsi:type="ppt">
           <Orientation>Portrait</Orientation>
   ```

```
    </Export>

    <Export xsi:type="all">
        <ExportCharts>false</ExportCharts>
        <ExportImages>false</ExportImages>
        <ExportFormatting>false</ExportFormatting>
    </Export>


  </Download>
</ServerInstance>
```

5. Save your changes and close the file.
6. Restart Oracle Analytics Server.

Elements to Manually Configure for Export

The table describes the elements that you can manually configure for export

Element	Description	Default Value
DataValue	Specifies whether data values (that is, numbers and dates) are exported in raw format with full number precision and format mask or as a string in the data format specified when exporting to Excel. Valid values are: • UseRawValue • UseFormattedValue The export type is: `xsi:type="excel"`	UseRawValue

Element	Description	Default Value
RepeatRows	Specifies whether cells that span rows and cells that span columns are to be repeated when exporting tables and pivot tables to Excel. If set to true, then cells that span rows and cells that span columns are repeated, regardless of the **Value Suppression** setting in the Analysis editor. For example, in a table that has Year and Month values, Year is repeated for all Month values. If set to false, then the behavior is the same as that defined by the **Value Suppression** option in the Analysis editor: • If **Value Suppression** is set to **Suppress**, then cells that span rows and cells that span columns are not repeated. For example, in a table that has Year and Month values, Year is displayed only once for Month values. • If **Value Suppression** is set to **Repeat**, then cells that span rows and cells that span columns are repeated. For example, in a table that has Year and Month values, Year is repeated for all Month values. For more information on the **Value Suppression** option, see *Visualizing Data in Oracle Analytics Server* The export type is: `xsi:type="excel"`	false
Delimiter	Specifies the column delimiter character for the **CSV Format** option, for example, a semicolon (;), when exporting raw data from results and views. The export type is: `xsi:type="formattedText"`	","
KeepRowsTogether	Specifies whether rows are to be kept together at page breaks when exporting to PDF. If set to true, rows are kept together at page breaks. If set to false, rows are split across page breaks. The export type is: `xsi:type="pdf"`	false

Orientation	Specifies the orientation (either **Portrait** or **Landscape**) when exporting to PDF and to Powerpoint.
	The export type for PDF exports is:
	`xsi:type="pdf"`
	The export type for Powerpoint exports is:
	`xsi:type="ppt"`
		Landscape (for PDF exports)
		Portrait (for Powerpoint exports)

Element	Description	Default Value
QuoteTxtTab	Adds quotes for the **CSV Format** option. When set to false, no quotes are added. The export type is: `xsi:type="formattedText"`	true

Support Nested Folders, Navigation, and Drill-Down

The administrator can set up subject areas in ways that assist content designers who work with analyses.

Managing Metadata Repositories for Oracle Analytics Server provides complete information about setting up subject areas. The following list includes features of subject areas that assist content designers:

- To make selections easy for content designers to discern in the Subject Areas pane when creating analyses, the administrator can set up the Presentation layer in the Model Administration Tool to give the appearance of nested folders. For example, the administrator can make the Sales Facts folder appear as a subfolder in the Facts folder.
- When content designers create analyses, they can enable users to go to related analyses and content. If the administrator sets up dimensions and dimensional hierarchies for the subject area, then users can drill down on data results that are presented in graphs, tables, and pivot tables to obtain more detailed information.

  There are no specific privilege settings that control access to navigation and drill-down features, which are available to all users.
- Content designers can create analyses that include columns from a primary subject area and from one or more related subject areas.

Configure for Display and to Process Data in Views

You can configure various options that change the display and processing of data in views.

See also Use Fusion Middleware Control to Set Configuration Options for Data in Tables and Pivot Tables and Use Fusion Middleware Control to Set the Maximum Number of Rows Processed to Render a Table View.

This section contains the following topics:

- Manually Configure for Data in Views
- Manually Configure Graphs and Gauges
- Manually Change Alternating Bar Color
- Manually Configure for Interactions in Views

Manually Configure for Data in Views

You can configure various options that change the processing and display of data in views.

See the following sections:

- Manually Configure Cube Settings for Pivot Tables and Graphs
- Manually Configure Settings for Data in Views
- Manually Configure Settings to Fetch Data for Table Views, Pivot Table Views, and Trellis Views

Manually Configure Cube Settings for Pivot Tables and Graphs

You can use settings within the Cube element to affect the display and processing of data in pivot tables and graphs. The settings also take effect for XMLA export.

1. Open the instanceconfig.xml file for editing, located in:

   BI_DOMAIN/config/fmwconfig/biconfig/OBIPS

2. Locate the Cube section, in which you must add the following elements:

   - `CubeMaxRecords` — Specifies the maximum number of records that are returned by an analysis for the view to process. This roughly governs the maximum number of cells that can be populated in a view; unpopulated cells in a sparse view don't count. The default is 40000.
   - `CubeMaxPopulatedCells` — Specifies the maximum number of cells that are returned by an analysis for the view to process. This must be an appropriate value related to `CubeMaxRecords`.

3. Include the elements and their ancestor elements as appropriate.

   For example, if a table has 10 columns and 1,000,000 rows, the number of cells is 10,000,000:

```
<ServerInstance>
  <Views>
    <Cube>
      <CubeMaxRecords>1000000</CubeMaxRecords>
      <CubeMaxPopulatedCells>10000000</CubeMaxPopulatedCells>
    </Cube>
  </Views>

  <Views>
   <Table>
    <MaxCells>10000000</MaxCells> ===> for table
   </Table>
  </Views>

  <Views>
   <Pivot>
    <MaxCells>10000000</MaxCells> ===> for pivot
   </Pivot>
  </Views>

  <ODBC>
   <ResultRowLimit>1000000</ResultRowLimit> ==> no of rows
  </ODBC>

</ServerInstance>
```

> **Note:**
>
> Both CubeMaxRecords and ResultRowLimit limit the number of rows returned. The limit is determined by the setting with the larger value (see Use Fusion Middleware Control to Set the Maximum Number of Rows Processed to Render a Table View).

4. Save your changes and close the file.
5. Restart Oracle Analytics Server.

Manually Configure Settings for Data in Views

You can configure a similar group of settings that affects the display of data in table, pivot table, graph, trellis, narrative, ticker, and treemap views.

While the settings are often the same, you must include the element within each appropriate parent element to override the default setting that applies to that view, for example, many of the views use the MaxVisiblePages element. You must include that element within each of the Table, Pivot, Trellis, Charts, and Treemap parent elements, to override the default value of that setting for each of those view types.

1. Open the instanceconfig.xml file for editing, located in:

   BI_DOMAIN/config/fmwconfig/biconfig/OBIPS

2. Locate the Table, Pivot, Trellis, Charts, Narrative, Ticker, and Treemap parent sections, in which you must add the elements that are described in the table below.
3. Include the elements and their ancestor elements as appropriate, as shown in the following example:

```
<ServerInstance>
  <Views>
      <Table>
        <MaxCells>10000</MaxCells>
        <MaxPagesToRollOutInDelivery>1000</MaxPagesToRollOutInDelivery>
        <MaxVisiblePages>1000</MaxVisiblePages>

        <MaxVisibleSections>25</MaxVisibleSections>
        <DefaultRowsDisplayed>30</DefaultRowsDisplayed>
        <DefaultRowsDisplayedInDelivery>250</
DefaultRowsDisplayedInDelivery>
        <DefaultRowsDisplayedInDownload>65000</
DefaultRowsDisplayedInDownload>
        <DefaultRowsDisplayedInDownloadCSV>65000</
DefaultRowsDisplayedInDownloadCSV>
      </Table>
      <Pivot>
        <MaxPagesToRollOutInDelivery>1000</MaxPagesToRollOutInDelivery>
        <MaxVisibleColumns>300</MaxVisibleColumns>
        <MaxVisiblePages>1000</MaxVisiblePages>

        <MaxVisibleSections>25</MaxVisibleSections>
        <DefaultRowsDisplayed>30</DefaultRowsDisplayed>
        <DefaultRowsDisplayedInDelivery>250</
DefaultRowsDisplayedInDelivery>
```

```
        <DefaultRowsDisplayedInDownload>65000</
DefaultRowsDisplayedInDownload>
              </Pivot>
      <Trellis>
        <Simple>
            <MaxCells>1000</MaxCells>
            <MaxVisibleSections>10</MaxVisibleSections>
            <MaxVisiblePages>1000</MaxVisiblePages>

            <MaxVisibleColumns>75</MaxVisibleColumns>
            <MaxPagesToRollOutInDelivery>1000</MaxPagesToRollOutInDelivery>
            <DefaultRowsDisplayed>10</DefaultRowsDisplayed>
            <DefaultRowsDisplayedInDelivery>100</
DefaultRowsDisplayedInDelivery>
            <DefaultRowsDisplayedInDownload>6500</
DefaultRowsDisplayedInDownload>
        </Simple>
        <Advanced>
            <MaxCells>5000</MaxCells>
            <MaxVisibleSections>50</MaxVisibleSections>
            <MaxVisiblePages>1000</MaxVisiblePages>

            <MaxVisibleColumns>150</MaxVisibleColumns>
            <MaxPagesToRollOutInDelivery>1000</MaxPagesToRollOutInDelivery>
            <DefaultRowsDisplayed>25</DefaultRowsDisplayed>
            <DefaultRowsDisplayedInDelivery>250</
DefaultRowsDisplayedInDelivery>
            <DefaultRowsDisplayedInDownload>10000</
DefaultRowsDisplayedInDownload>
        </Advanced>
      </Trellis>
      <Charts>
        <MaxVisibleColumns>2000</MaxVisibleColumns>
        <MaxVisiblePages>1000</MaxVisiblePages>

        <MaxVisibleSections>25</MaxVisibleSections>
        <JavaHostReadLimitInKB>4096</JavaHostReadLimitInKB>
      </Charts>
      <Narrative>
        <MaxRecords>40000</MaxRecords>
        <DefaultRowsDisplayed>30</DefaultRowsDisplayed>
      </Narrative>
      <Ticker>
        <MaxRecords>40000</MaxRecords>
      </Ticker>
      <Treemap>
        <MaxCells>5000</MaxCells>
        <MaxVisiblePages>10000</MaxVisiblePages>

        <MaxVisibleSections>50</MaxVisibleSections>
      </Treemap>
  </Views>
  <ODBC>
    <ResultRowLimit>65000</ResultRowLimit>
  </ODBC>
```

```
</ServerInstance>
```

Note that this example doesn't include elements that might exist in the file, but that are centrally managed by Fusion Middleware Control and can't be changed manually.

4. Save your changes and close the file.
5. Restart Oracle Analytics Server.

About Common Elements that Affect the Display of Data in Views

You can use common elements to affect the display of data in views

The table describes the common elements that affect the display of data in views. If a user exceeds these values, then the Oracle BI Server returns an error message when the view is rendered.

Element	Description	Default Value	Applicable Views
DefaultRowsDisplayed	Specifies the default number of rows to display in views in analyses and dashboards. This number shouldn't exceed the number that's specified for the MaxVisibleRows element.	30 (10 for Simple Trellis)	Narrative, Pivot Table, Table, Trellis
DefaultRowsDisplayedInDelivery	Specifies the default number of rows that can be included in the view when it's displayed on a dashboard.	100 for Simple Trellis; 250 for Advanced Trellis, Table, and Pivot Table	Pivot Table, Table, Trellis
DefaultRowsDisplayedInDownload	Specifies the default number of rows that can be included in the view when it's downloaded, such as to a PDF file.	65000 (6500 for Simple Trellis; 10000 for Advanced Trellis)	Pivot Table, Table, Trellis
DefaultRowsDisplayedInDownloadCSV	Specifies the default number of rows that can be included in the view when it's downloaded to a CSV file. For this to work, ODBC `ResultRowLimit` must be set to the same value.	1100000	Table

Element	Description	Default Value	Applicable Views
MaxCells	Specifies the maximum number of cells or for a treemap, the maximum number of groups and tiles, to be displayed in a view. For pivot tables, tables, and trellises, this number shouldn't exceed the product of MaxVisibleColumns times MaxVisibleRows, which is what the system attempts to render.	50000 (1000 for Simple Trellis; 5000 for Treemap)	Pivot Table, Table, Trellis, Treemap
MaxPagesToRollOutInDelivery	Specifies the maximum number of pages that can be included in the view when it's displayed on a dashboard.	1000	Pivot Table, Table, Trellis
MaxRecords	Specifies the maximum number of records that can be included in the view.	40000	Narrative, Ticker
MaxVisibleColumns	Specifies the maximum number of columns to be displayed in a view.	2000 (75 for Simple Trellis; 150 for Advanced Trellis, 300 for Pivot Table)	Graph, Pivot Table, Trellis
MaxVisiblePages	Specifies the maximum number of view prompts (or pages in PDF) to be displayed in a view.	1000 (10000 for Treemap)	Graph, Pivot Table, Table, Trellis, Treemap
MaxVisibleSections	Specifies the maximum number of sections to be displayed in a view. This element doesn't apply when a slider is in place for a graph. The SectionSliderDefault and SectionSliderLimit elements apply to limit section values when a slider is in place. See Manually Configure Graphs and Gauges.	25 (10 for Simple Trellis; 50 for Advanced Trellis and Treemap)	Graph, Pivot Table, Table, Trellis, Treemap
JavaHostReadLimitInKB	Specifies the maximum amount of data that's sent to the browser for a single graph.	4096	Graph

Manually Configure Settings to Fetch Data for Table Views, Pivot Table Views, and Trellis Views

You can use settings within the GridViews element (such as DefaultRowFetchSlicesCount) to specify how data is fetched for table views, pivot table views, and trellis views that use scrolling as the method to browse data.

Content designers specify the method to use to browse data (either scrolling or paging controls) in a table view, pivot table view, or trellis view in the Table Properties dialog: Style tab, the Pivot Table Properties dialog, or the Trellis Properties dialog: General tab, respectively. See *Visualizing Data in Oracle Analytics Server*.

1. Open the `instanceconfig.xml` file for editing, located in:

   `BI_DOMAIN``/config/fmwconfig/biconfig/OBIPS`

2. Locate the GridViews parent section, in which you must add the elements that are described in the table below.
3. Include the elements and their ancestor elements as appropriate, as shown in the following example:

```
<ServerInstance>
  <Views>
    <GridViews>
        <DefaultRowFetchSlicesCount>1000</DefaultRowFetchSlicesCount>
        <DefaultColumnFetchSlicesCount>300</DefaultColumnFetchSlicesCount>
        <DefaultFreezeHeadersClientRowBlockSize>60</
DefaultFreezeHeadersClientRowBlockSize>
        <DefaultFreezeHeadersClientColumnBlockSize>15</
DefaultFreezeHeadersClientColumnBlockSize>
        <DefaultFreezeHeadersWidth>1000</DefaultFreezeHeadersWidth>
        <DefaultFreezeHeadersHeight>1000</DefaultFreezeHeadersHeight>
    <DefaultScrollingEnabled>false</DefaultScrollingEnabled>
    </GridViews>
  </Views>
</ServerInstance>
```

4. Save your changes and close the file.
5. Restart Oracle Analytics Server.

About Common Elements that Affect the Fetch of Data in Views

You can use common elements to affect the fetch of data in views

The table describes the common elements for fetching data.

Element	Description	Default Value
DefaultRowFetchSlicesCount	Specifies the maximum number of rows to be used in calculating the size of the scrolling view when it's initially displayed. When a user scrolls to the last row in the view, a link to fetch more rows (if there are more) is displayed.	1000

Element	Description	Default Value
DefaultColumnFetchSlicesCount	Specifies the maximum number of columns to be used in calculating the size of the scrolling view when it is initially displayed. When a user scrolls to the last column in the view, a link to fetch more columns (if there are more) is displayed.	300
DefaultFreezeHeadersClientRowBlockSize	Specifies the number of rows to be returned to the client on an AJAX request (that is, when a user scrolls such that a request needs to be made to the server to get more rows into the table view, pivot table view, or trellis view).	60
DefaultFreezeHeadersClientColumnBlockSize	Specifies the number of columns to be returned to the client on an AJAX request (that is, when a user scrolls such that a request needs to be made to the server to get more columns into the table view, pivot table view, or trellis view).	15
DefaultFreezeHeadersWidth	Specifies the default maximum width of table views, pivot table views, and trellis views in pixels. Content designers can override this value using the **Maximum Width** field in the properties dialog for the view.	700
DefaultFreezeHeadersHeight	Specifies the default maximum height of table views, pivot table views, and trellis views in pixels. Content designers can override this value using the **Maximum Height** field in the properties dialog for the view.	400
DefaultScrollingEnabled	Specifies the Data View scrolling of table views, as follows: • true: sets reports to show the output with Data View as 'Fixed headers with scrolling content' • false: sets reports to show the Data View as 'Content Paging'.	true

Manually Configure Graphs and Gauges

You can configure various options that change the display of graphs, including funnel graphs, and gauges.

These views types are also affected by the settings that are described in Manually Configure for Data in Views.

1. Open the `instanceconfig.xml` file for editing, located in:

   BI_DOMAIN/config/fmwconfig/biconfig/OBIPS

2. Search for the Charts sections, in which you must add the elements that are described in the table below.
3. Include the elements and their ancestor elements as appropriate, as shown in the following example:

```
<ServerInstance>
  <Views>
    <Charts>
```

```
        <EmbedFonts>true</EmbedFonts>
        <SectionSliderDefault>150</SectionSliderDefault>
        <SectionSliderLimit>300</SectionSliderLimit>
        <DefaultWebImageType>flash</DefaultWebImageType>
        <FlashCodeBase>\\CORPORATE\Download\Flash</FlashCodeBase>
        <FlashCLSID>E38CDB6E-BA6D-21CF-96B8-432553540000</FlashCLSID>
      </Charts>
    </Views>
  </ServerInstance>
```

4. Save your changes and close the file.
5. Restart Oracle Analytics Server.

Configuration Settings for Graphs

You can use configuration settings for graphs.

The table describes the configuration settings for graphs.

Element	Description	Default Value
EmbedFonts	See Configure Fonts for Graphs for details.	false
DefaultWebImageType	Specifies the default type for rendering an image when a format hasn't been specified in the URL or in the XML file for the view. Valid values are: • flash In a browser that doesn't support the flash format, the image doesn't render. You should use the html5 value instead. • png (W3C Portable Network Graphics) • svg (W3C Scalable Vector Graphics) The svg value isn't supported in this update, so flash is used if svg is specified. • html5 In a browser that doesn't support the html5 format, the image renders in the flash format instead. Flash, png, and html5 images provide the greatest degree of interaction because they support mouse-over behaviors (such as popup data labels), navigation, and drilling. DefaultWebImageType is honored for graphs only when `renderFormat="default"` in the report xml.	html5
SectionSliderDefault	Specifies the default number of values that can be displayed on a section slider bar. A section slider displays members of one or more attribute or hierarchical columns as values on a rectangular bar and provides mechanisms to select a value. For more information about defining section sliders in graphs, gauges, and funnels, see *User's Guide for Oracle Business Intelligence Enterprise Edition*.	5
SectionSliderLimit	Specifies the maximum number of values that can be displayed on a section slider bar.	10

Element	Description	Default Value
FlashCodeBase	Specifies the name of the source for downloading the Flash plug-in. The default download source for the Flash plug-in is the vendor's website. In some organizations, users are instructed to download the latest Flash software from a corporate location instead of the vendor's website. You can modify the setting to point to another location that holds the Flash code base. Then, when users view a graph and a newer version of the Flash software is available on the corporate server, they can be prompted to download the newer version.	*vendor's web site*
FlashCLSID	Specifies a custom global identifier (clsid) property for downloading Flash. After modifying the Flash download directory using the FlashCodeBase element, you can enable a download prompt by creating a new classID for the Flash ActiveX control to add a custom global identifier property. You can obtain the current global identifier property from any computer where Oracle BI Presentation Services graphing is being used. (The global identifier property used by Oracle Analytics Server is D27CDB6E-AE6D-11CF-96B8-444553540000.) The custom global identifier property must contain the same number of characters and dashes as the global identifier used in the default Flash ActiveX control. Test flash graphs independent of Oracle Analytics Server to ensure that they function with the custom global identifier property.	No default value

Configure Fonts for Graphs

There are two ways you can configure fonts for graphs.

- Set the embed fonts element
- Deliver font files for printing

Set the Embed Fonts Element

By default, graphs rely on users to have the appropriate device fonts installed on their system to display multilingual text in the graphs. When users enable rotation on O1 axis labels, the graphs can look unattractive at certain angles. The labels appear obscured without any anti-aliasing. You can set the EmbedFonts element to true to specify the use of embedded fonts instead of device fonts, which resolves this display issue.

Be aware that the use of embedded fonts can cause a loss of fidelity. Whenever end users select fonts, they see the Oracle-licensed Albany WT plain fonts by default. Because the graphing engine does not provide embedded fonts for Chinese, Japanese, and Korean locales, users with those locales might obtain unattractive results for label rotation.

Deliver Font Files for Printing

If you plan to print graphs in bi-directional languages to PDF or graphs in Chinese, Japanese, or Korean to PNG images, then you must deliver required font files (.TTF) as follows:

- To print graphs in bi-directional languages to PDF, you must deliver the Albany family of fonts to this Java Runtime Environment (JRE) directory:

  ```
  JAVA.HOME/lib/fonts
  ```

  where *JAVA.HOME* is the directory name as specified by the "java.home" system property.
- To print graphs in Chinese, Japanese, or Korean to PNG images, you must deliver the font file that contains all the needed glyphs to this JRE directory:

  lib/fonts/fallback

For more information about font configuration files, see your Java documentation.

Configure Graph and Gauge Rendering

You can use the DefaultWebImageType element to specify the default type for rendering an image when a format hasn't been specified in the URL or in the XML file for the view.

Valid values are:

- flash

  In browsers that don't support flash format, the graph or gauge won't render. You should use the html5 value instead.
- png (W3C Portable Network Graphics)
- svg (W3C Scalable Vector Graphics)

  The svg value isn't supported in this update, so flash is used if svg is specified.
- html5

  In browsers that support only the flash format, the graph or gauge renders in flash format.

Flash and SVG images provide the greatest degree of interaction because they support mouse-over behaviors (such as popup data labels), navigation, and drilling.

Manually Change Alternating Bar Color

Both tables and pivot tables can have colored bars on alternating lines. Such formatting is sometimes called "green bar styling," and the default color for these alternating bars is green. For pivot tables, content designers can control formatting features when editing tables and pivot tables, including whether alternating bar color is enabled.

As the administrator, you can change the default color for alternating bars, by editing a style configuration file. To change the color, edit the views.css file in the b_mozilla_4 folder, as shown in the following list. Change the six-digit hexadecimal color value to a new color value.

- Tables use the CSS selector:

  .ECell (for even-numbered rows)

  .OCell (for odd-numbered) rows.
- Pivot tables use the CSS selector:

  .PTE (for odd-numbered rows)

The option for enabling the alternating bars is in the Edit View dialog and is labeled **Enable alternating row "green bar" styling**. If you change the color of the bars, then you might also want to change the label to indicate the color that you have set.

To change the label in the dialog for both the table and pivot table, open the tableviewmessages.xml file and find this entry:

```
WebMessageName = "kmsgTableViewEnableGreenbarReporting"
```

Copy the entry and the text line under it to a custom messages file in the custom messages folder, and change the text line appropriately. For example:

```
WebMessageName = "kmsgTableViewEnableGreenbarReporting"
<TEXT>Enable alternating row "RED bar" styling</TEXT>"
```

Manually Configure for Interactions in Views

You can configure various options that change the way that right-click interactions are handled in views for an analysis at runtime.

You can also set in the Console, see View Options.

The elements in the instanceconfig.xml file specify the default settings for a new or updated analysis. You can edit the properties of an analysis in Presentation Services to modify how the analysis handles right-click interactions in views.

1. Open the instanceconfig.xml file for editing, located in:

   `BI_DOMAIN/config/fmwconfig/biconfig/OBIPS`

2. Locate the sections in which you must add the elements that are described in the table below.
3. Include the elements and their ancestor elements as appropriate, as shown in the following example:

```
<ServerInstance>
  <Analysis>
    <InteractionProperties>
      <InteractionPropertyAddRemoveValues>false</
InteractionPropertyAddRemoveValues>
      <InteractionPropertyCalcItemOperations>false</
InteractionPropertyCalcItemOperations>
      <InteractionPropertyDrill>true</InteractionPropertyDrill>
      <InteractionPropertyGroupOperations>false</
InteractionPropertyGroupOperations>
      <InteractionPropertyInclExclColumns>true</
InteractionPropertyInclExclColumns>
      <InteractionPropertyMoveColumns>true</InteractionPropertyMoveColumns>
      <InteractionPropertyRunningSum>false</InteractionPropertyRunningSum>
      <InteractionPropertyShowHideSubTotal>false</
InteractionPropertyShowHideSubTotal>
      <InteractionPropertySortColumns>true</InteractionPropertySortColumns>
      <InteractionPropertyHideColumns>false</
InteractionPropertyHideColumns>
    </InteractionProperties>
  </Analysis>
</ServerInstance>
```

4. Save your changes and close the file.

5. Restart Oracle Analytics Server.

Configuration Settings for Interactions in Views

You can use configuration settings for interactions in views.

The table describes the configuration settings for interactions in views.

Element	Description	Default Value
InteractionPropertyAddRemoveValues	Specifies whether the **Add/Remove Values** option is selected by default in the Analysis Properties dialog: Interactions tab.	false
InteractionPropertyCalcItemOperations	Specifies whether the **Create/Edit/Remove Calculated Items** option is selected by default in the Analysis Properties dialog: Interactions tab.	false
InteractionPropertyDrill	Specifies whether the **Drill (when not a primary interaction)** option is selected by default in the Analysis Properties dialog: Interactions tab.	true
InteractionPropertyGroupOperations	Specifies whether the **Create/Edit/Remove Groups** option is selected by default in the Analysis Properties dialog: Interactions tab.	false
InteractionPropertyInclExclColumns	Specifies whether the **Include/Exclude Columns** option is selected by default in the Analysis Properties dialog: Interactions tab.	true
InteractionPropertyMoveColumns	Specifies whether the **Move Columns** option is selected by default in the Analysis Properties dialog: Interactions tab.	true
InteractionPropertyRunningSum	Specifies whether the **Display/Hide Running Sum** option is selected by default in the Analysis Properties dialog: Interactions tab.	false
InteractionPropertyShowHideSubTotal	Specifies whether the **Display/Hide Sub-totals** option is selected by default in the Analysis Properties dialog: Interactions tab.	false
InteractionPropertySortColumns	Specifies whether the **Sort Columns** option is selected by default in the Analysis Properties dialog: Interactions tab.	true

Configure for Prompts

You can configure settings that affect the way that users work with prompts.

You can also set in the Console, see Prompt Options.

1. Open the instanceconfig.xml file for editing, located in:

   `BI_DOMAIN/config/fmwconfig/biconfig/OBIPS`

2. Locate the sections in which you must add the elements that are described in the table below.

3. Include the elements and their ancestor elements as appropriate, as shown in the following example:

```
<ServerInstance>
  <Prompts>
    <MaxDropDownValues>256</MaxDropDownValues>
    <ResultRowLimit>65000</ResultRowLimit>
    <AutoApplyDashboardPromptValues>true</AutoApplyDashboardPromptValues>
    <AutoSearchPromptDialogBox>true</AutoSearchPromptDialogBox>
    <AutoCompletePromptDropDowns>
      <SupportAutoComplete>true</SupportAutoComplete>
      <CaseInsensitive>true</CaseInsensitive>
      <MatchingLevel>MatchAll</MatchingLevel>
      <ResultsLimit>50</ResultsLimit>
    </AutoCompletePromptDropDowns>
    <ShowNullValueWhenColumnIsNullable>never</
ShowNullValueWhenColumnIsNullable>
  </Prompts>
</ServerInstance>
```

4. Save your changes and close the file.
5. Restart Oracle Analytics Server.

Configuration Settings for Prompts

You can use configuration settings for prompts.

The table describes the configuration settings for prompts.

Element	Description	Default Value
AutoApplyDashboardPromptValues	Specifies whether to display various fields, as described in the following list: If true, then • The **Show Apply Button** and **Show Reset Button** fields are displayed on the Edit Page Settings dialog. • The **Prompts Apply Buttons** and **Prompts Reset Buttons** fields are displayed in the Dashboard Properties dialog. • The **Prompt Buttons on Current Page** option is displayed on the Dashboard builder's Tools menu. If false, then • The **Show Apply button** and **Show Reset button** fields aren't displayed on the Edit Page Settings dialog. • The **Prompts Apply Buttons** and **Prompts Reset Buttons** fields aren't displayed in the Dashboard Properties dialog. • The **Prompt Buttons on Current Page** option isn't displayed on the Dashboard builder's Tools option.	true
AutoSearchPromptDialog	Specifies whether search results are displayed and highlighted when the user types the search parameter (without clicking the **Search** button).	true

Element	Description	Default Value
CaseInsensitive	Specifies whether the auto-complete functionality is case-insensitive. If set to true, case isn't considered when a user enters a prompt value such as "Oracle" or "oracle." If set to false, case is considered when a user enters a prompt value, so the user must enter "Oracle" and not "oracle" to find the Oracle record. The system recommends the value with the proper case.	true
Matching Level	Specifies whether the auto-complete functionality uses matching to find the prompt value that the user enters into the prompt field. These settings don't apply when the user accesses the Search dialog to locate and specify a prompt value. Use the following settings: • **StartsWith** — Searches for a match that begins with the text that the user types. For example, the user types "M" and the following stored values are displayed: "MicroPod" and "MP3 Speakers System". • **WordStartsWith** — Searches for a match at the beginning of a word or group of words. For example, the user types "C" and the following values are displayed: "ComCell", "MPEG Camcorder", and "7 Megapixel Digital Camera". • **MatchAll** — Searches for any match within the word or words.	MatchAll
MaxDropDownValues	Specifies the maximum number of choices to display in the following locations: • At runtime, in choice lists, check boxes, list boxes, and radio buttons in prompts. • At runtime, in the Values list of the Select Values dialog when the user selects the Search option from the prompt values list. • At design time in the Values list on the Filter editor and in the Available list of the Select Values dialog that's displayed when the user clicks the Search link from the Value list in the Filter editor. • At design time in the Available list of the Select Values dialog that's displayed when the user selects Specific Column Values in the Choice List Values field or Specific Values in the Default selection field and clicks the corresponding Select values button. • At design time while working with the Slider User Input type, in the list that's displayed for the Lower Limit, Upper Limit, Default Low Value, and Default High Value fields or when the users clicks the corresponding Search link within any of these fields.	256
ResultsLimit	Specifies the number of matching values that are returned when the auto-complete functionality is enabled.	50
ResultRowLimit	Specifies the number of records returned from the logical SQL for prompts (analyses and dashboard prompts).	65000

Element	Description	Default Value
ShowNullValueInPromptsWhenDatabaseColumnIsNullable	Specifies whether to show the term "NULL" at runtime in the column prompt above the column separator in the drop-down list when the database allows null values. Use the following settings: • **always** — Always shows the term "NULL" above the column separator in the drop-down list. • **never** — Never shows the term "NULL" in the drop-down list. • **asDataValue** — Displays as a data value in the drop-down list, not as the term "NULL" above the separator in the drop-down list.	always
SupportAutoComplete	Enables or disables the auto-complete functionality of prompts. A setting of true turns auto-complete on, which means that the **Prompts Auto-Complete** field is displayed and is set to On in the My Account dialog and in the Dashboard Properties dialog. A setting of false turns auto-complete off, which means that the auto-complete fields in the My Account and Dashboard Properties dialogs aren't available.	false, unless you're running Oracle Analytics Server on the Oracle Exalytics In-Memory Machine

Manually Change Presentation Settings

You can configure settings that change the display of dashboards and presentation settings.

See the following sections:

- Manually Change Presentation Setting Defaults
- Manually Create Custom Links in Presentation Services
- Enable the Ability to Create Links to Dashboard Pages
- Configure an Alternate Toolbar for Oracle Analytics Server Publisher
- Enable the Ability to Export Dashboard Pages to Oracle Analytics Server Publisher
- Modify the Table of Contents for PDF Versions of Briefing Books
- Configure a Custom Download Link for the Smart View Installer

Manually Change Presentation Setting Defaults

In addition to the presentation settings that you can change in Fusion Middleware Control, other settings can be changed manually. Use various elements in the instanceconfig.xml file to change these settings.

1. Open the instanceconfig.xml file for editing, located in:

   ```
   BI_DOMAIN/config/fmwconfig/biconfig/OBIPS
   ```

2. Locate the sections in which you must add the elements that are described in the table below.

3. Include the elements and their ancestor elements as appropriate, as shown in the following example:

```
<ServerInstance>
  <Enable508>false</Enable508>
  <Dashboard>
    <DefaultName>Templates</DefaultName>
    <EnableDelayExecution>true</EnableDelayExecution>
  </Dashboard>
  <BriefingBook>
      <MaxFollowLinks>6</MaxFollowLinks>
  </BriefingBook>
  <Formatters>
  <NumericFormatter maxSignificantDigits="16"/>
  <Formatters>
</ServerInstance>
```

> **Note:**
>
> This example doesn't include elements that might exist in the file, but that are centrally managed by Fusion Middleware Control and can't be changed manually.

4. Save your changes and close the file.
5. Restart Oracle Analytics Server.

Configuration Settings for Presentation

You can use configuration settings for presentation.

The table describes the configuration settings for presentation.

Element	Description	Default Value
DefaultName	Specifies the name to be used for dashboards that contain dashboard template pages and to override the path in which Oracle Analytics Server searches for dashboard template pages. By default, Oracle Analytics Server searches for dashboard template pages in dashboards named "default" in subfolders under /Shared Folders.	default
EnableDelayExecution	Specifies whether content designers have the ability to delay the execution of dashboard pages. When this is set to true, the **Prompt before Opening** option is displayed in the Dashboard Properties dialog. .	true

Element	Description	Default Value
MaxFollowLinks	Specifies the default value for the maximum number of navigation links to follow in a briefing book. A briefing book navigation link is a type of link that can be added to a dashboard using the Dashboard Builder. The default value for this element is 5; the minimum is 1; and the maximum is 10. If you plan to download briefing books to PDF format, then don't set the value of this element to a number greater than 9 because of the table of contents limitation of nine links. For information about the table of contents, see Modify the Table of Contents for PDF Versions of Briefing Books.	5
NumericFormatter	Specifies a value to use for consistent output at a fixed number of digits. You might notice that after a certain number of significant digits, the number isn't displayed appropriately. Use this setting to set the maximum number of significant digits, such as 16 on Linux platforms.	No default value

Note:

See Enable the Ability to Create Links to Dashboard Pages for information about the defaults for the Bookmarks, MaxAgeMinutes, EnableBookmarkURL, and EnablePromptedURL elements. See Configure an Alternate Toolbar for Oracle Analytics Server Publisher for information about the defaults for the ReportingToolbarMode element.

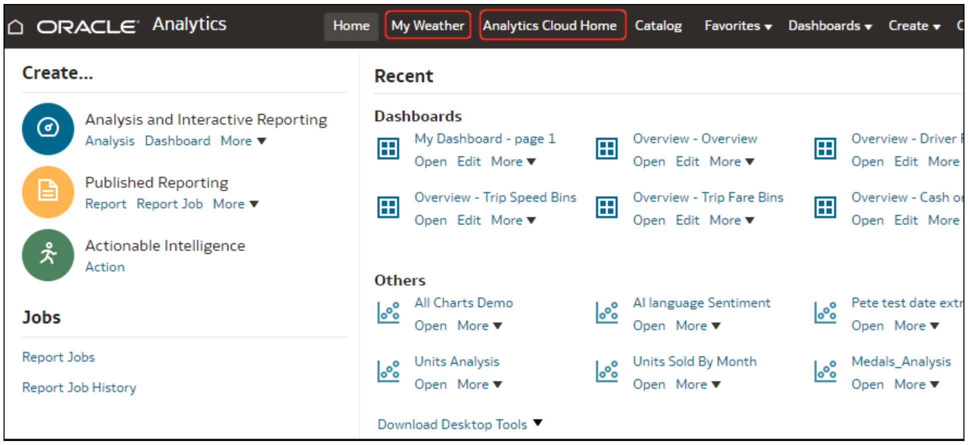
Manually Create Custom Links in Presentation Services

By default, the global header contains menus and options that enable you to navigate easily among features. You can manually customize the global header to add your own links.

Oracle recommends that you use the **Console** to make the changes to the `CustomLinks` XML file. See Customize Links on the Classic Home Page, and Analytic Content Options.

If you do use this manual method, you shouldn't enter any content into the **Custom Links XML** field in the **Analytic Content** section of the Console.

In this example, links are added for "My Weather" and "Analytics Cloud Home".



If you want to create custom links, you can specify various attributes, including the following:

- The text for the link (either a static string or a message name to use for localization).
- A URL to access.
- Whether the page from the URL replaces the current page or opens in a new tab or window that you can name.
- The relative ordering of links in the header.
- An optional icon to use with the link.

If you want to manually edit a configuration file to customize the global header, you can perform the tasks that are described in the following sections:

- Create the customlinks.xml File
- Add the CustomLinks Element
- Set the Custom Links Privilege

Create the customlinks.xml File

Create the `customlinks.xml` file to specify customizations to the global header, as described in the following sections.

Create the file

You can place the `customlinks.xml` file in any absolute path specified in the `instanceconfig.xml` file. See Add the CustomLinks Element. The recommended default location in which to create this file is the data directory for Presentation Services:

```
BI_DOMAIN/bidata/components/OBIPS/customMessages
```

Elements in the File

The following table describes the elements and attributes that you can include in the `customlinks.xml` file. If you want to hide existing links that are shown by default, then you can comment out their entries in the file. You can't modify the order of default links such as Favorites or Dashboards.

See Customize Links on the Classic Home Page for a list of the elements and attributes that you can use to specify custom links.

After you update the `customlinks.xml` file, the file is reloaded when you next restart Presentation Services, as described in Start and Stop Your System.

Sample File and Output

This example displays two custom links to the left of the **Catalog** link in the global header of the Classic home page.

```
<?xml version="1.0" encoding="utf-8"?>
<customLinks xmlns="com.siebel.analytics.web/customlinks/v1">
   <link id="1" name="My Weather" description="Local weather site"
src="https://www.example.com/weather" target="blank" >
       <locations>
           <location name="header" insertBefore="catalog"/>
       </locations>
   </link>
   <link id="1" name="Analytics Cloud Home" description="OAC Viz Home Page"
src="https://<OAC example URL>.analytics.ocp.oraclecloud.com/ui/dv/?
pageid=home" target="blank" >
       <locations>
           <location name="header" insertBefore="catalog"/>
       </locations>
   </link>
</customLinks>
```

> **Note:**
>
> To obtain the link for Oracle Analytics home page, log into Oracle Analytics, copy the URL, and paste it into the `src="<target link>"` element (as shown in the example XML code).

- The global header is modified to include the following links:
  - **My Weather** — A link to a local weather site.
  - **Analytics Cloud Home** — A link to the Oracle Analytics Cloud home page.

Add the CustomLinks Element

Before custom links are visible on the Home page, you must edit the `instanceconfig.xml` file to include the `CustomLinks` element, and the `Enabled` element (whose default is true) and `filePath` element within it. Omitting the `CustomLinks` element is the same as setting the `Enabled` element to false; no custom links are displayed.

You can also set in the Console, see Analysis, Dashboard, and Pixel-Perfect Report Options in Configure Advanced System Settings in the Console.

1. Open the `instanceconfig.xml` file for editing, located in:

   `BI_DOMAIN/config/fmwconfig/biconfig/OBIPS`

2. Search for the `ServerInstance` section, in which you must add the `CustomLinks` element.

3. Include the elements and their ancestor elements as appropriate, as shown in the following example:

   ```
   <ServerInstance>
      <CustomLinks>
         <Enabled>true</Enabled>
         <filePath>c:/mydir/bidata/components/OBIPS/customMessages/
   customlinks.xml</filePath>
      </CustomLinks>
   </ServerInstance>
   ```

4. Save your changes and close the file.
5. Restart Oracle Analytics Server.

Set the Custom Links Privilege

You must configure the software properly to make customizations visible to users.

If you want users to see the customizations that you've made, then you must ensure that the Custom Links privilege is assigned to the BI Consumer role, which occurs by default. You can't assign this privilege to individual users, groups, or roles other than BI Consumer.

To verify the role for this privilege, use the Manage Privileges page in the Administration pages of Presentation Services. See 'Manage Presentation Services Privileges' in *Managing Security for Oracle Analytics Server*.

Enable the Ability to Create Links to Dashboard Pages

Users can create links (both bookmark links and prompted links) to dashboard pages.

This enables them, for example, to save a link as a bookmark or to copy and send a link to other users in email. A bookmark is a hidden object in the Presentation Catalog (under the `/system/bookmarks` folder) that captures the state of a dashboard page. It's created when a user creates a bookmark link to the page. You can enable or disable the ability to create these links to dashboard pages. Further, in order for users to be able to create these links, you must grant the Create Bookmark Links and Create Prompted Links privileges associated with this feature.

For more information on these links, see About Creating Links to Dashboard Pages in *Visualizing Data in Oracle Analytics Server*. For more information on privileges, see Managing Presentation Service Privileges in *Managing Security for Oracle Analytics Server* .

1. Open the instanceconfig.xml file for editing, located in:

   `BI_DOMAIN/config/fmwconfig/biconfig/OBIPS`

2. Locate the Server Instance section.
3. Include the following elements and their ancestor elements as appropriate:
   - EnableBookmarkURL: Use this element to enable the ability to create bookmark links to dashboard pages:
     - **true** — Enables the ability to create bookmark links. (Default)
     - **false** — Disables the ability to create bookmark links.
   - EnablePromptedURL: Use this element to enable the ability to create prompted links to dashboard pages:

  - **true** — Enables the ability to create prompted links. (Default)
  - **false** — Disables the ability to create prompted links.

- MaxAgeMinutes: Use this element within the Bookmarks element to specify that bookmarks older than the specified number of minutes are removed. The default is 43200 minutes, which corresponds to 30 days.

  Note that every time a bookmark is accessed, the expiration timer is reset. This resetting means that if a bookmark is accessed frequently, it might never be removed. Setting the value to 0 means that the bookmark is saved for 0 minutes (it'll still expire). You can't set bookmarks to never expire. If you want bookmarks to last for a long time, then set the value to a large number of minutes and access the bookmarks within the allotted number of minutes.

The following entry is an example of these settings:

```
<ServerInstance>
  <Dashboard>
    <EnableBookmarkURL>true</EnableBookmarkURL>
    <EnablePromptedURL>true</EnablePromptedURL>
  </Dashboard>
  <Cache>
      <Bookmarks>
          <MaxAgeMinutes>43200</MaxAgeMinutes>
      </Bookmarks>
  </Cache>
</ServerInstance>
```

4. Save your changes and close the file.
5. Restart Oracle Analytics Server.

Configure an Alternate Toolbar for Oracle Analytics Server Publisher

When you include a Oracle Analytics Server Publisher report on a dashboard, you generally enable that report to participate as a recipient of the dashboard state by passing in dashboard context to that report using core dashboard prompts.

For scenarios that don't require passing of context to or from the report to the larger dashboard-based analytic application, you can display a variant of the default toolbar, which exposes the underlying parameter prompts of that report. Within that frame, a user can then pass in parameters to a single report.

This approach can be confusing to the user as any other dashboard prompts on the page don't contribute to the report, which also does not participate in passing context back to the rest of the application. Changes to the toolbar are also applied globally for all reports that are embedded in dashboards across the entire Oracle Analytics Server Publisher instance.

Use the ReportingToolbarMode element to affect how Oracle Analytics Server Publisher reports are embedded in Oracle Analytics Server. You configure the alternate Oracle Analytics Server Publisher toolbar by setting the element's value to 6. Remove the ReportingToolbarMode element to revert to the default toolbar behavior, or set it to the default value of 1.

You can also set in the Console, see Analysis, Dashboard, and Pixel-Perfect Report Options in Configure Advanced System Settings in the Console.

1. Open the instanceconfig.xml file for editing, located in:

```
BI_DOMAIN/config/fmwconfig/biconfig/OBIPS
```

2. Locate the AdvancedReporting section in which you must add the **ReportingToolbarMode** element.
3. Include the element and its ancestor elements as appropriate, as shown in the following example:

```
<ServerInstance>
  <AdvancedReporting>
      <ReportingToolbarMode>6</ReportingToolbarMode>
  </AdvancedReporting>
</ServerInstance>
```

   The following list describes the element values:

   - 1 = Does not display the toolbar.
   - 2 = Displays the URL to the report without the logo, toolbar, tabs, or navigation path.
   - 3 = Displays the URL to the report without the header or any parameter selections. Controls such as Template Selection, View, Export, and Send are still available.
   - 4 = Displays the URL to the report only. No other page information or options are displayed.
   - 6 = Displays the Oracle Analytics Server Publisher toolbar to display the parameter prompts of the Oracle Analytics Server Publisher report

4. Save your changes and close the file.
5. Restart Oracle Analytics Server.

Enable the Ability to Export Dashboard Pages to Oracle Analytics Server Publisher

Content designers can create custom layouts for printing and exporting dashboard pages.

When a content designer creates a custom layout for a dashboard page, the dashboard page is exported to Oracle Analytics Server Publisher. You can enable or disable the ability to export dashboard pages to Oracle Analytics Server Publisher by setting the EnableDashPageExport element.

See About Creating Custom Layouts for Printing and Exporting Dashboard Pages in *Visualizing Data in Oracle Analytics Server*.

1. Open the instanceconfig.xml file for editing, located in:

   BI_DOMAIN/config/fmwconfig/biconfig/OBIPS

2. Locate the AdvancedReporting section in which you must add the **EnableDashPageExport** element.
3. Include the element and its ancestor elements as appropriate, as shown in the following example:

```
<ServerInstance>
  <AdvancedReporting>
      <EnableDashPageExport>true</EnableDashPageExport>
  </AdvancedReporting>
</ServerInstance>
```

The element values are:

- **true** — Enables the ability to export dashboard pages to Oracle Analytics Server Publisher by showing the **Custom Print & Export Layouts** component in the Print & Export Options dialog. (Default)
- **false** — Disables the ability to export dashboard pages to Oracle Analytics Server Publisher by hiding the **Custom Print & Export Layouts** component in the Print & Export Options dialog.

4. Save your changes and close the file.
5. Restart Oracle Analytics Server.

Modify the Table of Contents for PDF Versions of Briefing Books

The PDF version of a briefing book contains a table of contents that is automatically generated. It contains an entry for each dashboard page, analysis, and report in the briefing book.

The default template for the table of contents, toc-template.rtf, is located in the `*ORACLE_INSTANCE*\config\OracleBIPresentationServicesComponent\coreapplication_obisn` directory. You can modify the toc-template.rtf file to accommodate the needs of your organization.

Configure a Custom Download Link for the Smart View Installer

The Smart View version packaged with Oracle Analytics Server may be out of synchronization with the server or with the latest available version of Smart View.

You can configure the **Smart View for MS Office** link that is displayed on the Download Desktop Tools list on the Oracle Analytics Server Home page to point to a custom download link for the Smart View installer. You can then ensure that the correct version of Smart View for your environment is always available to your users. You do this by adding the SmartViewInstallerURL element to instanceconfig.xml.

You can configure the download link to point to a location where the smartview.exe resides, for example:

- An external URL, such as the Smart View download page on Oracle Technology Network, where the latest version of Smart View is always available
- An internal URL, such as internal web page or intranet site where the installation can start immediately
- A folder on a local server, where the installation can start immediately

1. Open the instanceconfig.xml file for editing, located in:

   ```
   BI_DOMAIN/config/fmwconfig/biconfig/OBIPS
   ```

2. Locate the <CatalogPath> and <DSN>AnalyticsWeb</DSN> elements and add the **SmartViewInstallerURL** element after those elements.
3. Use the syntax in the following examples to add the **SmartViewInstallerURL** element:

11*g* example of download from Oracle Technology Network:

```
<CatalogPath>/example/path/work/abc/instances/instance1/bifoundation/
OracleBIPresentationServicesComponent/coreapplication_obips1/catalog/
SampleApp</CatalogPath>
<DSN>AnalyticsWeb</DSN>
<SmartViewInstallerURL>http://www.oracle.com/technetwork/middleware/epm/
downloads/smart-view-1112x-1939038.html</SmartViewInstallerURL>
```

11*g* example of download from an intranet site:

```
<CatalogPath>/example/path/work/abc/instances/instance1/bifoundation/
OracleBIPresentationServicesComponent/coreapplication_obips1/catalog/
SampleApp</CatalogPath>
<DSN>AnalyticsWeb</DSN>
<SmartViewInstallerURL>http://myserver:8080/downloads/smartview.exe</
SmartViewInstallerURL>
```

11*g* example of download from an internal server

```
<CatalogPath>/example/path/work/abc/instances/instance1/bifoundation/
OracleBIPresentationServicesComponent/coreapplication_obips1/catalog/
SampleApp</CatalogPath>
<DSN>AnalyticsWeb</DSN>
<SmartViewInstallerURL>\\myserver\downloads\smartview.exe</
SmartViewInstallerURL>
```

4. Save your changes and close the file.
5. Restart Oracle Analytics Server.

Validate and Block Queries in Analyses Using Custom JavaScript

You can develop validation scripts in JavaScript to validate analysis criteria and column formulas, and block invalid queries.

- Block Queries in Analyses
- Develop JavaScript to Block Analyses Based on Criteria
- Develop JavaScript to Block Analyses Based on Formula
- Validation Helper Functions

Block Queries in Analyses

Users working with analyses can invoke custom JavaScript to validate analysis criteria and column formulas. The validation allows for queries to be blocked when editing an analysis. The custom JavaScript must be accessible through a web server. To enable this feature, administrators specify the URL of the web server in system settings and register the web server as a safe domain.

1. Develop your custom validation scripts in JavaScript, store them in a web server, and make a note of the URL pointing to the JavaScript (*.JS) file containing the custom scripts.

For example, you might develop a blocking script that you store in `myblocking.js`, and the URL might be:`http://example.com:8080/mycustomscripts/myblocking.js`.

2. Specify the URL of your web server in system settings:
   a. Click **Console**, then click **Advanced System Settings**.
   b. In **URL for Blocking Queries in Analyses**, enter the URL that you noted in Step 1.
3. Register the web server as a safe domain:
   a. Click **Console**, then click **Safe Domains**.
   b. Add an entry for the domain in the URL you specified in Step 2.

      For example, you might add: `example.com:8080`.
   c. For options, select **Script** and **Connect**.
4. Test your validation scripts:
   a. Open an analysis.
   b. Run the analysis with both valid and invalid criteria.
   c. Verify that queries are blocked as expected.

Develop JavaScript to Block Analyses Based on Criteria

Whenever a user tries to run an analysis, Oracle Analytics invokes the function `validateAnalysisCriteria`. You can customize `validateAnalysisCriteria` to validate and block queries based on your own specific criteria. If the function returns `true`, the query runs. If the function returns `false` or displays a message, the query is blocked.

For example, the following is sample code for a JavaScript program called `myblocking.js`.

```
// This is a blocking function. It ensures that users select what
// the designer wants them to.
function validateAnalysisCriteria(analysisXml)
{
   // Create the helper object
   var tValidator = new CriteriaValidator(analysisXml);
   // Validation Logic
   if (tValidator.getSubjectArea() != "Sample Sales")
      return "Try Sample Sales?";
   if (!
tValidator.dependentColumnExists("Markets","Region","Markets","District"))
   {
      // If validation script notifies user, then return false
      alert("Region and District are well suited, do you think?");
      return false;
   }
   if (!tValidator.dependentColumnExists("Sales
Measures","","Periods","Year"))
   return "You selected a measure so pick Year!";
   if (!tValidator.filterExists("Sales Measures","Dollars"))
   return "Maybe filter on Dollars?";
   if (!tValidator.dependentFilterExists("Markets","Market","Markets"))
   return "Since you are showing specific Markets, filter the markets.";
   var n = tValidator.filterCount("Markets","Region");
   if ((n <= 0) || (n > 3))
      return "Select 3 or fewer specific Regions";
   return true;
}
```

If the function returns anything other than `false`, the criteria is considered to be valid and the analysis runs. The function is also use to validate criteria for preview and save operations.

Develop JavaScript to Block Analyses Based on Formula

Whenever a user tries to enter or modify a column formula, Oracle Analytics invokes the function `validateAnalysisFormula` to verify the operation. You can customize `validateAnalysisFormula` to validate and block formulas based on your own specific criteria. If the function returns `true`, the formula is accepted. If validation fails the function returns `false`, the formula is rejected and your custom message displays.

To display a message and allow users to continue, your function must return `true`. To block the query, your function must return `false` or display a message. You can use a JavaScript string and regular expression techniques in your function to investigate and validate the formula.

Helper functions are available so the query blocking function can check for filters, columns, and so on. See Validation Helper Functions.

For example, the following code shows how to block a query if a user enters an unacceptable formula.

```
// This is a formula blocking function. It makes sure the user doesn't enter
an unacceptable formula.
function validateAnalysisFormula(sFormula, sAggRule)
{
   // don't allow the use of concat || in our formulas
   var concatRe = /\|\|/gi;
   var nConcat = sFormula.search(concatRe);
   if (nConcat >= 0)
      return "You used concatenation (character position " + nConcat + ").
That isn't allowed.";
   // no case statements
   var caseRe = /CASE.+END/gi;
   if (sFormula.search(caseRe) >= 0)
      return "Don't use a case statement.";
   // Check for a function syntax: aggrule(formula) aggrule shouldn't contain
a '.'
   var castRe = /^\s*\w+\s*\(.+\)\s*$/gi;
   if (sFormula.search(castRe) >= 0)
      return "Don't use a function syntax such as RANK() or SUM().";
    return true;
}
```

Validation Helper Functions

Several validation helper functions are available in a JavaScript file for you to use.

These functions are defined within a JavaScript file named answers/queryblocking.js. This table contains the list of helper functions and their descriptions.

Validation Helper Function	Description
`CriteriaValidator.getSubjectArea()`	Returns the name of the subject area referenced by the analysis. It generally is used in a switch statement within the function before doing other validation. If the analysis is a set-based criteria, then it returns `null`.
`CriteriaValidator.tableExists(sTable)`	Returns `true` if the specified folder (table) has been added to the analysis by the content designer, and `false` if the folder wasn't added.
`CriteriaValidator.columnExists(sTable, sColumn)`	Returns `true` if the specified column has been added to the analysis by the content designer, and `false` if the column wasn't added.
`CriteriaValidator.dependentColumnExists(sCheckTable, sCheckColumn, sDependentTable, sDependentColumn)`	Checks to ensure that the `dependentColumn` exists if the `checkColumn` is present. It returns `true` if either the checkColumn isn't present, or the `checkColumn` and the dependent column are present. If `checkColumn` and `dependentColumn` are `null`, then the folders are validated. If any column from `checkTable` is present, then a column from `dependentTable` must be present.
`CriteriaValidator.filterExists(sFilterTable, sFilterColumn)`	Returns `true` if a filter exists on the specified column, and `false` if no filter is present.
`CriteriaValidator.dependentFilterExists(sCheckTable, sCheckColumn, sFilterTable, sFilterColumn)`	Checks to ensure that the `dependentFilter` exists if the `checkColumn` is present in the projection list. It returns `true` if either the `checkColumn` isn't present, or the `checkColumn` and the dependent filter are present.
`CriteriaValidator.filterCount(sFilterTable, sFilterColumn)`	Returns the number of filter values that are specified for the given logical column. If the filter value is "`equals`," "`null`," "`notNull`", or "`in`", then it returns the number of values chosen. If the column isn't used in a filter, then it returns zero. If the column is prompted with no default, then it returns -1. For all other filter operators (such as "`greater than`," "`begins with`," and so on) it returns `999`, because the number of values can't be determined.

Specify View Defaults for Analyses and Dashboards

You can control certain aspects of the initial state of new views that are added to an analysis and of new objects that are added to a dashboard page.

For example, you can add a default footer to new analyses and set defaults for dashboard sections. You control these aspects by customizing the appropriate XML message files to override the default values that are specified during installation.

XML Message Files for View Defaults

This section describes the XML message files to customize to override the view defaults distributed with Oracle BI Presentation Services.

For analyses, the file answerstemplates.xml includes a message named kuiCriteriaDefaultViewElementsWrapper from within kuiAnswersReportPageEditorHead. This message includes two additional messages, kuiCriteriaDefaultViewElements, in which you can define default values, and kuiCriteriaDefaultViewElementsMask, in which masks are defined. The mask XML message is protected and you can't modify its contents.

The wrapper message adds the combined XML into a JavaScript variable, *kuiDefaultViewElementsXML*, that's used to apply the new default values.

For dashboards, the file dashboardtemplates.xml includes a message named kuiDashboardDefaultElementsWrapper that adds XML into a JavaScript variable named *kuiDefaultDashboardElementsXML* for use within the dashboard editor.

Examples of Customizing Default Values for Analyses and Dashboards

The following sections provide examples of customizing default values.

- Add a Default Header or Footer to New Analyses
- Prevent Auto-Preview of Results
- Set Defaults for Analyses in the Compound Layout
- Change Dashboards Section Defaults
- Specify Dashboard Page Defaults and Include Headers and Footers

To cause these customizations to take effect, either stop and restart the server for Oracle BI Presentation Services, or click the **Reload Files and Metadata** link on the Administration page.

Add a Default Header or Footer to New Analyses

You can specify that default headers and footers are displayed on all new analyses.

For example, footers can contain messages such as a confidentiality notice, the company's name, and so on. You can specify a default header or footer by creating an XML message that specifies the text and formatting to apply.

The following XML code example creates a footer that contains the text "Acme Confidential" in bold, red letters.

```
<?xml version="1.0" encoding="utf-8"?>
<WebMessageTables xmlns:sawm="com.siebel.analytics.web.messageSystem">
   <WebMessageTable system="Answers" table="ViewDefaults">
<WebMessage name="kuiCriteriaDefaultViewElements" translate="no"><HTML>
   <view signature="compoundView" >
          <pageProps pageSize="a4">
         <pageFooter showOnDashboard="true" show="true">
           <zone position="top">
              <caption fmt="html">
                <text>[b]Acme Confidential[/b]
                </text>
              </caption>
              <displayFormat>
                <formatSpec hAlign="center" fontColor="#FF0000"/>
              </displayFormat>
          </zone>
         </pageFooter>
      </pageProps>
   </view>
</HTML>
</WebMessage>
```

```
   </WebMessageTable>
</WebMessageTables>
```

Paste the XML into a CustomMessages.xml file that you create in `oas_home/bi/bifoundation/web/msgdb/l_en/customMessages/CustomMessages.xml`, and restart obips1 to apply the changes.

Prevent Auto-Preview of Results

The results of an analysis are displayed when editing views of data. If you prefer that the content designer explicitly asks to view the results, then you can create an XML message that specifies that auto-preview is disabled when new views are created.

The content designer can still click the **Display Results** link to view the results when editing a view.

> **Note:**
>
> You can add signature entries for various views to the XML code. You can locate the signature value for a view in the XML representation of the analysis. While editing an analysis, see the analysis XML field on the Advanced tab of the Analysis editor. Look for the xsi:type attribute of the <saw:view> element. The signature value is the value without the "saw:" prefix.

The following XML code example disallows the auto-previewing of results when working with a view in Answers.

```
<?xml version="1.0" encoding="utf-8"?>
<WebMessageTables xmlns:sawm="com.siebel.analytics.web.messageSystem">
   <WebMessageTable system="Answers" table="ViewDefaults">
<WebMessage name="kuiCriteriaDefaultViewElements" translate="no"><HTML>
   <view signature="tableView" showToolBar="true" showHeading="true />
   <view signature="pivotTableView" autoPreview="false" />
   <view signature="titleView" autoPreview="false" />
   <view signature="viewSelector" autoPreview="false" />
   <view signature="htmlviewnarrativeView" autoPreview="false" />
   <view signature="tickerview" autoPreview="false" />
   <view signature="htmlview" autoPreview="false" />
   <view signature="dvtchart" autoPreview="false" />
   <view signature="dvtgauge" autoPreview="false" />
   <view signature="dvtfunnel" autoPreview="false" />
   <view signature="trellisView" autoPreview="false" />
</HTML>
</WebMessage>
   </WebMessageTable>
</WebMessageTables>
```

Set Defaults for Analyses in the Compound Layout

The results of a newly formed analysis are displayed as a title view followed by either a table or pivot table in a compound layout.

A table is created if the analysis contains only attribute columns, and a pivot table is created if the analysis contains at least one hierarchical column.

You can create an XML message that specifies that the compound view defaults to a different assemblage of views, such as a narrative followed by a graph. The content designer can still add and rearrange views within the compound layout.

The following XML code example sets the default compound layout to a narrative followed by a graph.

```
<?xml version="1.0" encoding="utf-8"?>
<WebMessageTables xmlns:sawm="com.siebel.analytics.web.messageSystem">
   <WebMessageTable system="Answers" table="ViewDefaults">
<WebMessage name="kuiCriteriaDefaultViewElements" translate="no"><HTML>
   <view signature="compoundView" >
      <cv signature="narrativeView" />
      <cv signature="dvtchart" />
   </view>
</HTML>
</WebMessage>
   </WebMessageTable>
</WebMessageTables>
```

Change Dashboards Section Defaults

By default, the results of drilling in a dashboard are displayed on a new page, section names are not displayed in the dashboard, and sections can be expanded and collapsed.

You can change these default values by creating an XML message that specifies that new default values for the dashboard section. A content designer who edits the dashboard can still modify this behavior using the options within the dashboard editor.

The following XML code example makes section heads visible, enables drilling, and does not allow sections to collapse.

```
<?xml version="1.0" encoding="utf-8"?>
<WebMessageTables xmlns:sawm="com.siebel.analytics.web.messageSystem">
   <WebMessageTable system="Answers" table="ViewDefaults">
<WebMessage name="kuiDashboardDefaultElements" translate="no"><HTML>
   <element signature="dashboardSection" drillInline="true"
showHeading="true" collapsible="false" />
</HTML>
</WebMessage>
   </WebMessageTable>
</WebMessageTables>
```

Specify Dashboard Page Defaults and Include Headers and Footers

By default, dashboards are printed without headers or footers and in a portrait orientation.

If you prefer that newly added dashboard pages default to having a custom header and footer and print in landscape orientation, then you can create an XML message that specifies these characteristics. A content designer who edits the dashboard can still modify this behavior using the options within the dashboard editor.

The following XML code example adds a custom header and footer to a dashboard page and specifies landscape orientation.

```
<?xml version="1.0" encoding="utf-8"?>
<WebMessageTables xmlns:sawm="com.siebel.analytics.web.messageSystem">
   <WebMessageTable system="Answers" table="ViewDefaults">
<WebMessage name="kuiDashboardDefaultElements" translate="no">
<HTML>
<element signature="dashboardPage" personalSelections="false">
     <pageProps orientation="landscape" printRows="all" pageSize="a4">
         <pageHeader showOnDashboard="true" show="true">
            <zone position="top">
            <caption>[b]Acme is Cool[/b]</caption>
            <displayFormat>
            <formatSpec fontSize="9pt" hAlign="center"
fontColor="#FFFFFF" backgroundColor="#000000"/>
            </displayFormat>
            </zone>
         </pageHeader>
         <pageFooter showOnDashboard="true" show="true">
            <zone position="top">
            <caption>[b]CONFIDENTIAL[/b]</caption>
            <displayFormat>
            <formatSpec fontSize="7.5pt" hAlign="center"
fontColor="#999999" borderColor="#CC99CC" fontStyle="italic"
borderPosition="all" borderStyle="single"/>
            </displayFormat>
            </zone>
         </pageFooter>
       </pageProps>
   </element>
</HTML>
</WebMessage>
   </WebMessageTable>
  </WebMessageTables>
```

Paste the XML into a CustomMessages.xml file that you create in `oas_home/bi/bifoundation/web/msgdb/1_en/customMessages/CustomMessages.xml`, and restart obips1 to apply the changes. When you create a new dashboard it displays the default header and footer.

Configure for Write-back in Analyses and Dashboards

Users of a dashboard page or an analysis might have the ability to modify the data that they see in a table view.

This ability is often referred to as "write back." As the administrator, you assist the content designer in configuring write back for users.

The following sections provide information about how to configure for write back:

- About Write-back for Administrators
- Enable Write-back in Analyses and Dashboards
- Write-Back Limitations
- Create Write-Back Template Files
- Set the LightWriteback Element

About Write-back for Administrators

Write-back enables users to update your data directly from dashboards and analyses.

Users with the **Write Back to Database** privilege see write-back fields as editable fields in analyses. The values they enter are saved to the database. Users without the **Write Back to Database** privilege, see write-back fields as read-only fields.

If a user types a value in an editable field and clicks the write-back button, then the application runs the `insert` or `update` SQL command defined in a *write-back template*. If the command succeeds, the analysis is updated with the new value. If there's an error either reading the template or running the SQL command, an error message is displayed.

The `insert` command runs when a record doesn't yet exist and the user enters new data into the table. In this case, the user typed in a table record where the original value was null. The `update` command runs when a user modifies existing data. To display a record that doesn't yet exist in the physical table, you can create another similar table. Use this similar table to display placeholder records that a user can modify.

> **Note:**
>
> When you create write-back templates, you must include both an `insert` command and an `update` command, even if they're not both used. For example, if you're only performing an `insert`, you must include an empty `update` statement `<update></update>`, as in this XML code:

Here's a sample write-back XML file that contains two `insert` commands and two empty `update` statements. To find out more about how to create and structure write-back XML files, see Create Write-Back Template Files.

```
<?xml version="1.0" encoding="utf-8" ?>
<WebMessageTables xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="oracle.bi.presentation/writebackschemas/v1">
<WebMessageTable lang="en-us" system="WriteBack" table="Messages">
   <WebMessage name="SetQuotaUseID">
      <XML>
         <writeBack connectionPool="Supplier">
            <insert>INSERT INTO regiontypequota
VALUES(@{c5f6e60e1d6eb1098},@{c5d7e483445037d9e},'@{c3a93e65731210ed1}','@{c6b
8735ea60ff3011}',@{c0432jkl53eb92cd8})</insert>
            <update></update>
         </writeBack>
      </XML>
   </WebMessage>
<WebMessage name="SetForecastUseID">
      <XML>
         <writeBack connectionPool="Supplier">
            <insert>INSERT INTO regiontypeforecast
VALUES(@{c83ebf607f3cb8320},@{cb7e2046a0fba2204},'@{c5a93e65d31f10e0}','@{c5a9
3e65d31f10e0}',@{c7322jkl93ev92cd8})</insert>
            <update></update>
         </writeBack>
      </XML>
   </WebMessage>
</WebMessageTable>
</WebMessageTables>
```

Enable Write-back in Analyses and Dashboards

Administrators can enable users to edit the data in analyses and dashboards.

1. Set up your semantic model.

   > **Note:**
   >
   > Follow these steps if you use Model Administration Tool to develop semantic models. If you use Semantic Modeler, see
   > Enable Write Back On Columns.

   a. In Model Administration Tool, open your semantic model (.rpd file).

   b. In the Physical layer, double-click the physical table that contains the column for which you want to enable write-back.

   c. On the **General** tab of the Physical Table dialog, ensure that **Cacheable** isn't selected. Deselecting this option ensures that Presentation Services users can see updates immediately.

   d. In the Business Model and Mapping layer, double-click the corresponding logical column.

   e. In the Logical Column dialog, select **Writeable**, then click **OK**.

   f. In the Presentation layer, double-click the column that corresponds to the logical column for which you enabled write-back.

   g. In the Presentation Column dialog, click **Permissions**.

   h. Select the **Read/Write** permission for the appropriate users and application roles.

   i. Save your changes.

2. Create an XML document with your write-back template (or templates). See Create Write-Back Template Files.

   Your XML document can contain multiple templates. This example shows an XML document that contains two templates (`SetQuotaUseID` and `SetForecastUseID`).

```
<?xml version="1.0" encoding="utf-8" ?>
<WebMessageTables xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="oracle.bi.presentation/writebackschemas/v1">
<WebMessageTable lang="en-us" system="WriteBack" table="Messages">
   <WebMessage name="SetQuotaUseID">
      <XML>
         <writeBack connectionPool="Supplier">
            <insert>INSERT INTO regiontypequota
VALUES(@{c5f6e60e1d6eb1098},@{c5d7e483445037d9e},'@{c3a93e65731210ed1}','@{
c6b8735ea60ff3011}',@{c0432jkl53eb92cd8})</insert>
            <update>UPDATE regiontypequota SET
Dollars=@{c0432jkl53eb92cd8} WHERE YR=@{c5f6e60e1d6eb1098} AND
Quarter=@{c5d7e483445037d9e} AND Region='@{c3a93e65731210ed1}' AND
ItemType='@{c6b8735ea60ff3011}'</update>
         </writeBack>
      </XML>
   </WebMessage>
<WebMessage name="SetForecastUseID">
      <XML>
         <writeBack connectionPool="Supplier">
            <insert>INSERT INTO regiontypeforecast
VALUES(@{c83ebf607f3cb8320},@{cb7e2046a0fba2204},'@{c5a93e65d31f10e01}','@{
c5a93e65d31f10e0}',@{c7322jkl93ev92cd8})</insert>
            <update>UPDATE regiontypeforecast SET
Dollars=@{c7322jkl93ev92cd8} WHERE YR=@{c83ebf607f3cb8320} AND
Quarter=@{cb7e2046a0fba2204} AND Region='@{c5a93e65d31f10e01}' AND
ItemType='@{c5a93e65d31f10e0}'</update>
         </writeBack>
      </XML>
   </WebMessage>
</WebMessageTable>
</WebMessageTables>
```

Note: You must include an `<insert>` element and an `<update>` element, even if both aren't used. For example, if you're only performing an `insert`, you must include the empty `update` statement `<update></update>`.

3. Copy the XML document containing your write-back templates to the clipboard.
4. Apply your write-back template in Oracle Analytics:
    a. Click **Console**, then click **Advanced System Settings**.
    b. In **Writeback Template XML**, paste the write-back template that you copied in Step 3.
5. Grant permissions to use the write-back code:
    a. Navigate to Classic home, then click **Administration**.
    b. Under **Security**, click **Manage Privileges**, and navigate to **Write Back**.
    c. Grant **Write Back to Database** to **Authenticated User**.
    d. Grant **Manage Write Back** to **BI Service Administrator**.
6. To enable write-back in columns:
    a. In the analysis editor, display the Column Properties of the column on which you want to enable write-back.
    b. In the Column Properties dialog, click the **Write Back** tab.

       If the column has been enabled for write-back in the semantic model, then the **Enable Write Back** box is available.
    c. Select the **Enable Write Back** option.
    d. Specify the value of other options if you want to change the default.
    e. Save your changes.

    The column is enabled for write-back in any analysis that includes this column.
7. To enable write-back in table views:
    a. In the analysis editor, open the table view for editing.
    b. Click **View Properties**.
    c. In the Table Properties dialog, click the **Write Back** tab.
    d. Select the **Enable Write Back** option.

e. Select the **Template Name** box, specify the value of "WebMessage name=" in the write-back template that you specified in Step 2.

   For example, the **Template Name** for the example template in Step 2 is 'SetQuotaUseID'.

f. Save your changes.

Write-Back Limitations

Users can write back to any data source (except for an ADF data source) that allows the execution of SQL queries from Oracle BI Server.

As you configure for write back, keep the following limitations in mind:

- Numeric columns must contain numbers only. They mustn't contain any data formatting characters such as dollar signs ($), pound signs or hash signs (#), percent signs (%), and so on.
- Text columns must contain string data only.
- If a logged-on user is already viewing a dashboard that contains an analysis where data has been modified using write back, the data isn't automatically refreshed in the dashboard. To see the updated data, the user must manually refresh the dashboard.
- You can use the template mechanism only with table views and only for single-value data. The template mechanism isn't supported for pivot table views or any other type of view, for multiple-value data, or for drop down columns with single-value data.
- All values in write-back columns are editable. When displayed in non printer friendly context, editable fields are displayed as if the user has the **Write Back to Database** privilege. However, when a logical column is mapped to a physical column that can change, the logical column returns values for multiple level intersections. This scenario might cause problems.
- Any field in an analysis can be flagged as a write-back field, even if it's not derived from the write-back table that you created. However you can't successfully run the write-back operation if the table isn't write-back enabled. The responsibility for correctly tagging fields lies with the content designer.
- A template can contain SQL statements other than `insert` and `update`. The write-back function passes these statements to the database. However, Oracle doesn't support or recommend the use of any statements other than `insert` or `update`.
- Presentation Services performs only minimal validation of data input. If the field is numeric and the user enters text data, then Presentation Services detects that and prevents the invalid data from going to the database. However, it doesn't detect other forms of invalid data input (values out of range, mixed text and numeric, and so on). When the user clicks the write-back button and an insert or update is run, invalid data results in an error message from the database. The user can then correct the faulty input. Content designers can include text in the write-back analysis to aid the user, for example, "Entering mixed alphanumeric values into a numeric data field isn't allowed."
- The template mechanism isn't suitable for entering arbitrary new records. In other words, don't use it as a data input tool.
- When creating a table for write back, ensure that at least one column doesn't include write-back capability but does include values that are unique for each row and are non-null.
- Write-back analyses don't support drill-down. Because drilling down modifies the table structure, the write-back template doesn't work.

> **Caution:**
>
> The template mechanism takes user input and writes it directly to the database. The security of the physical database is your own responsibility. For optimum security, store write-back database tables in a unique database instance.

Create Write-Back Template Files

A write-back template file is an XML-formatted file that contains one or more write-back templates.

A write-back template consists of a `WebMessage` element that specifies the name of the template, the connection pool, and the SQL statements that are needed to insert and update records in the write-back tables and columns that you've created. When content designers enable a table view for write back, they must specify the name of the write-back template to use to insert and update the records in the table view.

You can create multiple write-back template files. You can include multiple write-back templates in a template file, customizing each one for the fields that are used in each specific analysis. However, the best practice recommendation is to include only one template in a file.

You can also set in the Console, see Analysis, Dashboard, and Pixel-Perfect Report Options in Configure Advanced System Settings in the Console.

1. Create an XML file. The write-back template file can have any name of your choosing, because the system reads all XML files in the custommessages folder.
2. Add the appropriate elements following the requirements in “Requirements for a Write-Back Template” and examples in “Examples: Write-Back Template Files” below.
3. Store the write-back template file in the msgdb directory that the administrator has configured for static files and customer messages:

   ```
   SDD/components/OBIPS/custommessages
   ```

   Where SDD is the Singleton Data Directory for example, `DOMAIN_HOME/bidata`.

   While XML message files that affect a language-specific user interface must be localized, the XML file that's used for configuring a write-back template is usually not translated, because it's language-independent.

   In the rare cases where write-back template files must be language-dependent (for example, if a user logging in using the l_es (Spanish) locale would use a different SQL command than a user logging in using l_fr (French) locale), then the write-back template files should exist in appropriate language directories.

Requirements for a Write-Back Template

A write-back template must meet the following requirements:

- `WebMessage`: You must specify a name for the write-back template using the `name` attribute in the WebMessage element.

  For write back to work correctly, when enabling a table view for write back, a content designer must specify the name of the write-back template to be used to insert and update the records in the view.

This example shows a write-back template called `SetQuotaUseID`.

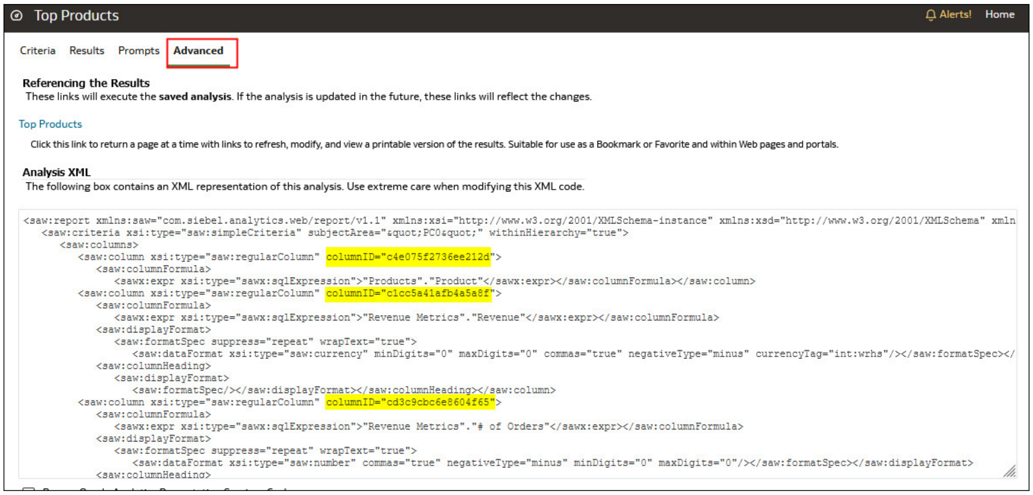
```
<WebMessage name="SetQuotaUseID">
```

- `connectionPool`: To meet security requirements, you must specify the connection pool along with the SQL commands to insert and update records. These SQL commands reference the values that are passed in the write-back schema to generate the SQL statements to modify the database table.
- `VALUES`: Column values can be referenced either by *column ID* or *column position*. The use of column ID is preferred.

  Surround string and date values with single quotes. Single quotes aren't required on numerical values.

  - **Column ID** - Each column ID is alphanumeric and randomly generated. You can find column IDs in the XML definition of the analysis that's available in the **Advanced** tab of the analysis editor. For example, column ID values such as: `@{c5f6e60e1d6eb1098}`, `@{c3a93e65731210ed1}`, `'@{c6b8735ea60ff3011}'`

    When you use column IDs, write-back continues to work even when the order of columns change.

    Referencing the Results
    These links will execute the **saved analysis**. If the analysis is updated in the future, these links will reflect the changes.

    Top Products

    Click this link to return a page at a time with links to refresh, modify, and view a printable version of the results. Suitable for use as a Bookmark or Favorite and within Web pages and portals.

    Analysis XML
    The following box contains an XML representation of this analysis. Use extreme care when modifying this XML code.

    ```
    <saw:report xmlns:saw="com.siebel.analytics.web/report/v1.1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmln
       <saw:criteria xsi:type="saw:simpleCriteria" subjectArea="&quot;PC0&quot;" withinHierarchy="true">
          <saw:columns>
             <saw:column xsi:type="saw:regularColumn" columnID="c4e075f2736ee212d">
                <saw:columnFormula>
                   <sawx:expr xsi:type="sawx:sqlExpression">"Products"."Product"</sawx:expr></saw:columnFormula></saw:column>
             <saw:column xsi:type="saw:regularColumn" columnID="c1cc5a41afb4a5a8f">
                <saw:columnFormula>
                   <sawx:expr xsi:type="sawx:sqlExpression">"Revenue Metrics"."Revenue"</sawx:expr></saw:columnFormula>
                <saw:displayFormat>
                   <saw:formatSpec suppress="repeat" wrapText="true">
                      <saw:dataFormat xsi:type="saw:currency" minDigits="0" maxDigits="0" commas="true" negativeType="minus" currencyTag="int:wrhs"/></saw:formatSpec></
                <saw:columnHeading>
                   <saw:displayFormat>
                      <saw:formatSpec/></saw:displayFormat></saw:columnHeading></saw:column>
             <saw:column xsi:type="saw:regularColumn" columnID="cd3c9cbc6e8604f65">
                <saw:columnFormula>
                   <sawx:expr xsi:type="sawx:sqlExpression">"Revenue Metrics"."# of Orders"</sawx:expr></saw:columnFormula>
                <saw:displayFormat>
                   <saw:formatSpec suppress="repeat" wrapText="true">
                      <saw:dataFormat xsi:type="saw:number" commas="true" negativeType="minus" minDigits="0" maxDigits="0"/></saw:formatSpec></saw:displayFormat>
                <saw:columnHeading>
    ```

  - **Column position** - Column positions start numbering with 1. For example, column position values such as: `@1`, `@3`, `'@5'`

    If the order of columns changes, write back no longer works and this is the reason why column IDs are preferred.

- You must include both an `<insert>` and an `<update>` element in the template. If you don't want to include SQL commands within the elements, then you must insert a blank space between the opening and closing tags. For example, you must enter the element as:

  ```
  <insert>  </insert>
  ```

  Rather than:

  ```
  <insert></insert>
  ```

  If you omit the blank space, then you see a write-back error message such as "The system can't read the Write Back Template 'my_template'".

- If a parameter's data type isn't an integer or real number, then add single quotation marks around it. If the database doesn't do Commits automatically, then add the optional `postUpdate` node after the `insert` and `update` nodes to force the commit. The `postUpdate` node typically follows this example:

  ```
  <postUpdate>COMMIT</postUpdate>
  ```

Example Write-Back Template File Using Column ID Syntax

A write-back template file that references values by **column ID** might resemble this example:

```
<?xml version="1.0" encoding="utf-8" ?>
<WebMessageTables xmlns:sawm="com.siebel.analytics.web/message/v1">
<WebMessageTable lang="en-us" system="WriteBack" table="Messages">
   <WebMessage name="SetQuotaUseID">
      <XML>
         <writeBack connectionPool="Supplier">
            <insert>INSERT INTO regiontypequota
VALUES(@{c5f6e60e1d6eb1098},@{c5d7e483445037d9e},'@{c3a93e65731210ed1}','@{c6b
8735ea60ff3011}',@{c0432jkl53eb92cd8})</insert>
            <update>UPDATE regiontypequota SET Dollars=@{c0432jkl53eb92cd8}
WHERE YR=@{c5f6e60e1d6eb1098} AND Quarter=@{c5d7e483445037d9e} AND
Region='@{c3a93e65731210ed1}' AND ItemType='@{c6b8735ea60ff3011}'</update>
         </writeBack>
      </XML>
   </WebMessage>
</WebMessageTable>
</WebMessageTables>
```

Example Write-Back Template File Using Column Position Syntax

A write-back template file that references values by **column position** might resemble this example:

```
<?xml version="1.0" encoding="utf-8" ?>
<WebMessageTables xmlns:sawm="com.siebel.analytics.web/message/v1">
<WebMessageTable lang="en-us" system="WriteBack" table="Messages">
   <WebMessage name="SetQuota">
      <XML>
         <writeBack connectionPool="Supplier">
            <insert>INSERT INTO regiontypequota VALUES(@1,@2,'@3','@4',@5)</
insert>
            <update>UPDATE regiontypequota SET Dollars=@5 WHERE YR=@1 AND
Quarter=@2 AND Region='@3' AND ItemType='@4'</update>
         </writeBack>
      </XML>
   </WebMessage>
</WebMessageTable>
</WebMessageTables>
```

Set the LightWriteback Element

For users to write back values, you must manually add the LightWriteback element in the instanceconfig.xml file.

1. Open the instanceconfig.xml file for editing, located in:

   `BI_DOMAIN/config/fmwconfig/biconfig/OBIPS`

2. Locate the ServerInstance section in which you must add the LightWriteback element.
3. Include the element and its ancestor elements as appropriate, as shown in the following example:

   ```
   <WebConfig>
       <ServerInstance>
           <LightWriteback>true</LightWriteback>
       </ServerInstance>
   </WebConfig>
   ```

   Note that this example doesn't include elements that might exist in the file, but that are centrally managed by Fusion Middleware Control and can't be changed manually.

4. Save your changes and close the file.
5. Restart the system.

Add Custom Knowledge for Data Enrichment

Add custom knowledge to Oracle Analytics to augment the system knowledge. For example, you might add a custom knowledge reference that classifies prescription medication into USP drug categories Analgesics or Opioid.

▤ Tutorial

Custom knowledge enables the Oracle Analytics semantic profiler to identify more business-specific semantic types and make more relevant and governed enrichment recommendations.

Before you start, download your custom knowledge reference files (in CSV format) and make them available locally for upload. The maximum file size you can upload is 250 MB. You can also create your own custom knowledge reference files in CSV or XLSX format. See Custom Knowledge Recommendations.

1. In the Oracle Analytics Home page, click the **Navigator**, and then click **Console**.
2. Click **Data Enrichments**.
3. Under **Custom Knowledge**, click **Add Custom Knowledge**.
4. In the Open dialog, navigate to and select your custom knowledge CSV file, then click **Open**.
5. In the Create Custom Knowledge from dialog, specify a name, verify the upload options, then click **OK**.

   The Custom Knowledge page lists the new file with the **Include** option selected. When content authors enrich datasets, Oracle Analytics presents enrichment recommendations based on this data.

Customize the User Interface

The user interface is generated by using scripts and custom interfaces, and is therefore highly customizable.

The look-and-feel is controlled by skins, styles, and themes. Skins and themes can define the user interface chrome (visible graphic features) outside the home and dashboard area.

Oracle Analytics is shipped with several styles. Saved themes are also available in style selectors.

The following sections provide information about how to customize the web user interface:

- What Are Skins, Styles, and Themes?
- General Tips to Customize the User Interface
- About Style Customizations
- Modify User Interface Styles
- Customize Your Style
- Example to Modify Master Branding Class
- Modify the User Interface With Themes

What Are Skins, Styles, and Themes?

Skins, styles, and themes change the way an interface looks.

You can control the way that the user interface is displayed to users by creating skins, styles, and themes. The primary difference between skins and styles is that styles apply to only dashboard content, whereas skins apply to every other part of the user interface. For example, skins apply to components that are used in analyses. Themes can apply to the entire interface, dashboard content, or both, depending on how they are applied.

You specify the default skin and style in the instanceconfig.xml file. The content designer can then alter certain elements to control how dashboards are formatted for display, such as the color of text and links, the font and size of text, the borders in tables, the colors and attributes of graphs, and so on. See Modify User Interface Styles. You design and apply themes using the Manage Themes page, which is enabled in the instanceconfig.xml file. You can then provide access on the Manage Privileges page.

Skins and styles are organized into folders that contain Cascading Style Sheets (CSS) and images. While skins and styles are typically used to customize the look and feel of analyses and dashboards by providing logos, color schemes, fonts, table borders, and other elements, they can also be used to control the position and justification of various elements by including specialized style tags in the relevant style sheet file. See About Style Customizations. Themes are organized using the Manage Themes page.

General Tips to Customize the User Interface

Some points to keep in mind as you plan to customize the user interface.

- In Presentation Services, you customize the user interface elements and appearance by modifying skins, and styles, not with JavaScript, and you don't modify JavaScript files because the objects and methods in scripts might change, and because these files might be replaced when upgrading.

  In a dashboard, users with the appropriate permissions can customize an individual dashboard section by adding HTML to it. This HTML can include JavaScript.

  Users with appropriate permissions can customize the entire interface and analyses output using themes. See Creating Custom Themes in *Visualizing Data in Oracle Analytics Server*.
- You can adapt the deployment to a particular language. See Localize Oracle Analytics Server.

About Style Customizations

Oracle Analytics Server has various styles. If you're going to customize the look-and-feel of Oracle Analytics Server or Oracle Analytics Server Publisher, Oracle strongly recommends that you use the custom style provided in the bicustom-template.ear as your starting point.

See Modify User Interface Styles for additional information.

Most of the common styles and image files, including the style sheet (master.css), are contained in a master directory. The topic Customize Your Style describes this directory and its structure in detail.

Within the style sheet, each element (or class) that's available for update is documented by comments.

Other style sheets are also contained within the style and skin folders. You aren't likely to need to update these files unless you're creating an advanced custom skin that provides styles for each detail of the user interface.

Modify User Interface Styles

When creating your own styles and skins, you must create CSS, graph.xml, and image files, and then make them available. You can use two approaches to make these files available:

- **Approach 1:** You use the bicustom.ear file to package your files into a single file that can then be easily deployed throughout the nodes in a cluster. This approach is useful when deploying in a clustered environment for scaled-out production systems. See Approach 1: Deploy the "bicustom.ear" File for the First Time?.

  To redeploy the bicustom.ear file, see Approach 1: Redeploy the "bicustom.ear" File.
- **Approach 2:** You use a shared file system. This approach is useful when you want to see your changes immediately or when your customizations are outside the system firewall. See Approach 2: Deploy Using Shared Folders.

  To view the modifications that you made using Approach 2, see Approach 2: View Your Modifications to a Shared Folder.

Approach 1: Deploy the "bicustom.ear" File for the First Time?

Enterprise Archive (EAR) files are archive (ZIP) files composed of a specific folder and file structure.

You can create an EAR file using any ZIP tool (for example, 7-zip) and then renaming the ZIP extension to EAR. Oracle provides the bicustom-template.ear file as a starting point.

The bicustom-template.ear file contains a bicustom.war file. Web Archive (WAR) files are also ZIP files composed of a specific folder and file structure. You must update the bicustom.war file within the bicustom-template.ear file to include your custom skin files. The bicustom.war file contains an example folder structure to help you get started.

> **Note:**
>
> This approach automatically deploys your custom skin to all nodes in your cluster.

1. Copy *ORACLE_HOME*/bi/bifoundation/jee/bicustom-template.ear to *BI_DOMAIN*/bidata/components/OBIPS/bicustom.ear.

   > **Note:**
   >
   > The patch or upgrade process may overwrite the bicustom-template.ear file, but it doesn't overwrite the bicustom.ear file.

2. Update the bicustom.ear file.
   a. Extract the bicustom.war file from the bicustom.ear file.
   b. Extract the files from the bicustom.war file.
   c. Edit the files to create your custom style and save your changes. See Customize Your Style
   d. Update the bicustom.war file with your changes.
   e. Update the bicustom.ear file with your new bicustom.war file.
3. Deploy the bicustom.ear file.
   a. Log in to Oracle Weblogic Server Administration Console.
   b. Click **Lock & Edit**.
   c. Click **Deployments** in the Domain Structure region.
   d. Click **Install** in the Deployments table.
   e. Navigate to the folder containing the bicustom.ear file (by default, this file is located in *BI_DOMAIN*/bidata/components/OBIPS/).
   f. Select the **bicustom.ear** file.
   g. Click **Next**.
   h. Select **Install this deployment as an application**.
   i. Click **Next**.
   j. Select **I will make the deployment accessible from the following location**.
   k. Click **Finish**.
   l. Click **Save**.
   m. Click **Activate Changes**.
4. Start the new application.
   a. Click **Deployments** in the Domain Structure region.
   b. Select the **bicustom** check box in the Deployments table.
   c. Click **Start**, and then select **Servicing all requests**.
5. Update the instanceconfig.xml file to specify which style and skin to use as the default value of the Styles option in the Dashboard Properties dialog.

   Styles and skins are located in the *ORACLE_HOME*\bi\bifoundation\web\appv2\res directory.

   a. Open the instanceconfig.xml file for editing, located in:

      BI_DOMAIN/config/fmwconfig/biconfig/OBIPS

   b. Include the element and its ancestor elements as appropriate, as shown in the following example:

```
<ServerInstance>
    <UI>
        <DefaultStyle>StyleExample</DefaultStyle>
        <DefaultSkin>StyleExample</DefaultSkin>
    </UI>
</ServerInstance>
```

   where DefaultStyle and DefaultSkin are the names of the custom style and skin properties, respectively.

   To use the Alta custom style and skin, replace the word `StyleExample` above with the word `Alta`.

   > **Note:**
   >
   > These names must match the names given to the folders containing the style and skin. Don't include underscores. For example: If your folder begins with the characters **s_**, such as s_Alta, then omit **s_**.

   c. Save your changes and close the file.

6. Restart Presentation Services. See About Managing Processes.

To redeploy the bicustom.ear file, see Approach 1: Redeploy the "bicustom.ear" File.

Approach 1: Redeploy the "bicustom.ear" File

Enterprise Archive (EAR) files are archive (ZIP) files composed of a specific folder and file structure.

Use this task to update your custom skin

1. Update the bicustom.ear file.
2. Update the deployment.
   a. Log in to Oracle Weblogic Server Administration Console.
   b. Click **Lock & Edit**.
   c. Click **Deployments** in the Domain Structure region.
   d. Select the **bicustom** checkbox in the Deployments table.
   e. Click **Update**.
   f. Click **Finish**.
   g. Click **Activate Changes**.
   h. Click **Release Configuration**.
3. Restart Presentation Services.

> **Note:**
>
> If you're changing component values, such as an image or font color in your customized files, you don't need to restart Presentation Services. Additionally, you do need to restart Presentation Services if you make a change to the default skin or style in the instanceconfig.xml file.

Approach 2: Deploy Using Shared Folders

Approach 2 is best suited to development or test environments where you want the changes that you make to the CSS and image files in your custom style and skin folders to be visible as soon as possible.

This approach is also used when your customizations are outside the Presentation Services firewall.

Static files are located in `*ORACLE_HOME*\bi\bifoundation\web\appv2`. Web servers have their own ways of exposing static directories that can be located anywhere within their file system, including a shared file system such as clustering. Refer to the documentation for your specific server on exposing static directories.

> **Note:**
>
> See the following documents for full information about how to configure Oracle WebLogic Server to work with web servers such as Apache HTTP Server, Microsoft Internet Information Server (Microsoft IIS), and Oracle HTTP Server:
>
> *Using Oracle WebLogic Server Proxy Plug-Ins*
>
> *Administering Oracle HTTP Server*

1. Create your custom style.
   1. Extract the bicustom.war file from the bicustom-template.ear file.
   2. Extract the contents of the bicustom.war file to a location accessible to Oracle Weblogic Server (for example, `c:\custom`).
   3. Edit the files to create your custom style and save your changes. See Customize Your Style for additional information.
2. Deploy the custom folder.
   1. Log in to Oracle Weblogic Server Administration Console.
   2. Click **Lock & Edit**.
   3. Click **Deployments** in the Domain Structure region.
   4. Click **Install** in the Deployments table.
   5. Navigate to the folder containing your custom style (for example, `c:\custom`).
   6. Click **Next**.
   7. Select **Install this deployment as an application**.
   8. Click **Next**.

   i. Select **bi_cluster** as the deployment target.
   j. Click **Next**.
   k. Set the name to **AnalyticsRes**.
   l. Select **I will make the deployment accessible from the following location**.
   m. Click **Next**.
   n. Select **Yes, take me to the deployment's configuration screen**.
   o. Click **Finish**.
   p. Click the Configuration tab.
   q. Enter **/analyticsRes** in the Context Root box.
   r. Click **Save**.
   s. Click **OK**.
   t. Click **Activate Changes**.
   u. Click **Release Configuration**.

3. Start the new application.
   a. Click **Deployments** in the Domain Structure region.
   b. Select the **analyticsRes** checkbox in the Deployments table.
   c. Click **Start**, and then select **Servicing all requests**.

4. Update the instanceconfig.xml file to specify the path that points to your customization, which can then be accessed by Presentation Services.

   Presentation Services generates the user interface for the Analysis editor and dashboards, which visualize data.

   a. Open the instanceconfig.xml file for editing, located in:

      BI_DOMAIN/config/fmwconfig/biconfig/OBIPS

   b. Locate the ServerInstance section.
   c. Include the elements, as shown in the following example:

```
<ServerInstance>
   <UI>
    <DefaultStyle>StyleExample</DefaultStyle>
    <DefaultSkin>StyleExample</DefaultSkin>
   </UI>
   <URL>
      <CustomerResourcePhysicalPath>c:\custom\res
         </CustomerResourcePhysicalPath>
      <CustomerResourceVirtualPath>/analyticsRes/res
      </CustomerResourceVirtualPath>
   </URL>
</ServerInstance>
```

      where DefaultStyle and DefaultSkin are the names of the custom style and skin properties, respectively.

      To use the Alta custom style and skin, replace the word `StyleExample` above with the word `Alta`.

> **Note:**
>
> The CustomerResourceVirtualPath points to the location entered in the Context Root box.

If you're embedding a Presentation Services' dashboard or report in an ADF application, you must include the SkinMappings element following the DefaultStyle and DefaultSkin elements, as shown in the following example:

```
<ServerInstance>
   <UI>
    <DefaultStyle>Alta</DefaultStyle>
    <DefaultSkin>Alta</DefaultSkin>
    <SkinMappings>

    <skinMapping>
       <biadfSkinFamily>SkinNewExample</biadfSkinFamily>
          <biSkin>SkinNewExample</biSkin>
     </skinMapping>
     <skinMapping>
          <biadfSkinFamily>SkinFamilyExample_1</biadfSkinFamily>
          <biSkin>SkinFamilyExample_1</biSkin>
      </skinMapping>
    </SkinMappings>
   </UI>
   <URL>
      <CustomerResourcePhysicalPath>c:\custom\res
         </CustomerResourcePhysicalPath>
      <CustomerResourceVirtualPath>/analyticsRes/res
         </CustomerResourceVirtualPath>
   </URL>
</ServerInstance>
```

d. Save your changes and close the file.

e. Restart Presentation Services.

Approach 2: View Your Modifications to a Shared Folder

Once you have configured your shared folder, you can view the changes to the CSS files and images.

1. Reload the metadata.
   a. Click the **Administration** link in the global header.
   b. On the Administration tab, click the **Reload Files and Metadata** link.
2. Clear the browser's cache.
3. Click any link on the global header (for example, **Home** or **Catalog**) to see your changes.

> **Note:**
>
> If you add files, such as an image file, to your custom style folder, you must restart Presentation Services.

Customize Your Style

To create your custom style, you must edit one or more files.

The bicustom.war folder structure from which the style directory is extracted, contains the following in the application resources (res) directory (./res/), which houses the relevant files:

- ./res/filemap.xml
- ./res/s_Custom/
- ./res/s_Custom/master/
- ./res/s_Custom/master/master.css
- ./res/s_Custom/master/graph.xml
- ./res/s_Custom/master/custom.css
- ./res/s_Custom/master/styleproperties.res
- ./res/s_Custom/master/*.png
- ./res/s_Custom/master/*.gif

The style directory, prefixed by **s_,** contains the style sheet, image files, and a graph xml file.

All classes are located within the CSS or the graph.xml file.

The table below describes the content of the "res" folder structure.

<table>
<tr><th>Folder or File Name</th><th>Description</th></tr>
<tr><td>filemap.xml[2]</td><td>The filemap.xml file enables you to specify that your style or skin extends another style or skin. By default, s_Custom[1] extends s_ExampleStyle, which is defined in the filemap.xml file. This means that if a file can't be found as part of your custom style or skin, then Presentation Services knows where to look next for that file. The content within the filemap.xml file looks similar to the following:
<pre><code><FileMap>
   <Styles Default="s_example">
      <Hierarchy>s_alta/s_Custom</Hierarchy>
   </Styles
   <Skins Default="sk_example">
      <Hierarchy>sk_alta/sk_Custom</Hierarchy
   </Skins>
</FileMap&gt;</code></pre></td></tr>
</table>

Folder or File Name	Description
s_Custom	s_Custom is a folder containing the files that define a style named **Custom**. The prefix, **s_**, indicates that the folder contains a *style*. This name displays in the Dashboard Properties dialog, Styles option. . If you want to change the default style manually by using the instanceconfig.xml file, the style name that you specify in the instanceconfig.xml file must match the style name specified as part of this folder name. You can name your style whatever you choose, but it must begin with **s_**. If you do change the name of the custom style, you must also update the filemap.xml file with the new style's name. If the files and folders that you add to the new style folder share the same names as files and folders in the base style (as defined in the filemap.xml file), then these new files and folders are used instead of the files in the base style. Use the files and folders in the base style as a guide for creating your custom style folders. You can define as many different styles as you like. Make a copy of the s_Custom folder in the same folder and name it by prefixing it with **s_** (for example, s_Corporate). If you create additional custom styles, you must also add them to a style hierarchy in the filemap.xml file.
sk_Custom	sk_Custom[1] is a folder that you create to customize the full skin and not just the style. You create sk_Custom as a sibling of the s_Custom folder. sk_Custom isn't provided as part of the bicustom-template.ear file. The prefix, **sk_**, indicates that the folder contains a *skin*. If you want to change the default skin manually by using the instanceconfig.xml file, the skin name that you specify in the instanceconfig.xml file must match the skin name specified as part of this folder name. You can name your skin whatever you choose, but it must begin with **sk_**. If you do change the name of the custom skin, you must also update the filemap.xml file with the skin's new name. If the files and folders that you add to the new skin folder share the same names as files and folders in the base skin (as defined in the filemap.xml file), then these new files and folders are used instead of the files in the base skin. Use the files and folders in the base skin as a guide for creating your custom style folders. You can define as many different skins as you like. Make a copy of the sk_Custom folder in the same folder and name it by prefixing it with **sk_** (for example, sk_Corporate). If you create additional custom skins, you must also add them to a skin hierarchy in the filemap.xml file.
master	The master folder contains all of the files that you likely need to create a custom style.
master.css	The master.css file contains all the CSS classes that are used by the style and defines the majority of the CSS classes and styles that are used throughout Presentation Services and Publisher. Changing the styles defined in the classes contained in this file widely impacts the system. Oracle does not recommend changing the CSS class selectors (CSS class names) in this file. Change the styles defined within each CSS class. The master.css file also contains comments to help you understand what classes apply and to which parts of the user interface they apply.
custom.css	The custom.css file is an empty (or blank) file that is imported by the master.css file. You can use the custom.css file to add your own CSS classes (for example, to apply styles to analyses) and override classes in the master.css file without changing the master.css file. Keeping your changes in the custom.css file, a separate file, enables you to take advantage of future improvements to the master.css file that are applied by patches and updates.

Folder or File Name	Description

graph.xml	The graph.xml file enables you to define all the default styles that are applied to various graphs within analyses. The graph.xml file is documented with comments. These comments describe the valid values for each setting and provide a description of the elements for that style.
	Note:
	When you customize a style in the graph.xml file, you must copy the updated file to:
	`ORACLE_HOME/bi/bifoundation/javahost/config/styles/<stylename>/graph.xml`
	For example, for styleexample graph customizations you copy the updated graph.xml file to:
	`ORACLE_HOME/bi/bifoundation/javahost/config/styles/styleexample/graph.xml`

styleproperties.res	The styleproperties.res file provides advanced control over some of the elements of the user interface. For example, you can specify which version of the data loading animation to use when your dashboard is rendered. Data loading animations are used in the "status indicator" area of a web page when a page is loading data from the server. There are two different versions of the data loading animation: one animation uses dark foreground colors for use on light backgrounds and the other animation uses the reverse. The styleproperties.res file is documented with comments. These comments provide a description of the elements.
.png and .gif (image files)	All required images and the most common images are located in the master directory. These images are primarily .png and .gif formats. You can replace these images with your own files, preferably with images of the same size.

1. File names, such as s_Custom, are case-sensitive and in lowercase on Linux.
2. Using the filemap.xml file to specify the style or skin that you're extending is applicable only for Presentation Services.

Example to Modify Master Branding Class

You can modify the CSS for a style to effect changes to any element contained within the CSS.

For example, if you want to change the background color of the master branding area from black to bright blue, perform the following steps:

1. Open the master.css file.

2. Scroll down and locate the class that you want to customize. In the sample code, modify the **.masterBrandingArea** class. Also note that the code contains inline comments to identify that code section.

```
.
.
.
.masterHeader
{
      font-weight: bold;
      color: #003d5b;
      text-align: left;
}
/* BRANDING AREA =========================================================
*/
/* This class applies to the branding area shown at the top of most pages.
*/
{
.masterBrandingArea
{
      background-color: #000000;
      color: #ffffff;
      padding: 5px;
}
/* This class applies to the brand name text shown at the top of most
pages. */
.masterBrandingAreaBrandName
{
      font-size: 17px;
      font-weight: bold;
.
.
.
```

Note:

To ensure that a custom.css setting is used in Dashboards, add the text `! important` just before the colon (`;`) in the custom.css file. This stops other css-files opened after custom.css from overwriting the value.

For example, if you change the background-color on the masterBrandingArea class from black to a light grey and the text color from white to black. The branding area appears light grey for login, Home, Catalog, Analysis and logoff, but for Dashboards, the branding area is still black with white text. To retain the custom settings for Dashboards you add the word`!important` before the semi colon (`;`) as follows:

```
.masterBrandingArea
{
  background-color: #ffffff !important; /* white background */
  color: #333333 !important; /* almost black text color */
}
```

3. Change the HTML background-color code value from **#000000** to **#3300ff**.
4. Save the file.
5. Reload the metadata and clear the browser's cache. See Customize the User Interface for additional information.

Modify the User Interface With Themes

To create your own themes, you must edit the instanceconfig.xml file to make themes available to the system.

1. On the Presentation Server, open the instanceconfig.xml file for editing, located in `BI_DOMAIN/config/fmwconfig/biconfig/OBIPS`.
2. Locate the ServerInstance section.
3. Add the elements shown:

```
<ServerInstance>...
    <UI>
        <EnableDynamicThemes>true</EnableDynamicThemes>
    </UI>
```

4. Save your changes and close the file.
5. Restart Presentation Services.

21

Configure and Manage Agents

This chapter describes how configure and manage agents.
If your organization licenses Delivers and if you've the appropriate privileges, then you can use the agents functionality as part of a default installation with no additional configuration. For information about using agents, see Create Agents to Deliver Content in *Visualizing Data in Oracle Analytics Server*.

Topics:

- How Are Agents Used?
- Enable Content Delivery Through Agents
- How Do Antivirus Software and Privileges Affect Agents?
- Configure Settings that Affect Agents
- Manage the Types of Devices that Deliver Content
- Monitor Active Agent Sessions

> **Note:**
>
> If you're migrating to a new system, then ensure that you also migrate the semantic model file, the Presentation Catalog, and the Scheduler tables. The Scheduler tables are required for agents.
>
> See Diagnose Issues with Agents for information about diagnostics and log files for agents.

How Are Agents Used?

Agents deliver targeted analytics to users based on a combination of schedule and trigger event. Delivery can be by a variety of routes, for example to Dashboard Alerts or to email.

To create an agent, users (with the Create Agent privilege) define the operations that the agent is to perform. Oracle Analytics Server packages information such as priority, delivery devices, and user, into a job, and schedules the job.

Enable and Customize Content Delivery Through Agents

You can use agents to deliver your content. This feature isn't enabled automatically. To display the **Create Agent** link on the Classic home page, grant the **View Delivers Full UX** privilege to the BI Content Author application role.

> **Note:**
>
> You also have to enable this feature if you import a snapshot taken from an early update of Oracle Analytics Cloud that didn’t support the **Delivers Full UX** privilege.

If you need to, you can sets some limits on emails sent by agents. For example, you can set limits for email size, email domains, and the number of recipients. By default, there aren't any limits. Can also customize whether to send emails using TO or BCC, and how to encode MIME email parameters.

1. Enable agents to deliver your content by email.
   a. On the Classic Home page, click the user profile icon and then click **Administration**.
   b. Click **Manage Privileges**.
   c. Navigate to the **Delivers** section, and grant **View Delivers Full UX** to **BI Content Author**.

   Now, users with the BI Content Author application role can see the **Create Agent** link on the Classic Home page.
2. Customize agent delivery.
   a. In the Oracle Analytics Home page, click the **Navigator**, and then click **Console**.
   b. Click **System Settings**.
   c. Click **Email Delivered by Agents**.
   d. Customize the way agents deliver email for your organization by setting a maximum email size, a maximum number of recipients, restricting email domains, whether to use BCC, how to encode MIME email parameters, and so on.

   See Email Delivered by Agents Options.

How Do Antivirus Software and Privileges Affect Agents?

This section provides the following information about how antivirus software and privileges affect agents.

How Does Antivirus Software Affect Agents?

Some antivirus software programs, such as Norton AntiVirus, enable a script-blocking feature, which tries to block all calls made by scripts to system objects (such as the Windows file system object) that the antivirus software deems unsafe.

If you start a script as part of post-agent processing, then this antivirus feature might cause unexpected results. If you run antivirus software with a script-blocking feature on the computer where Oracle BI Scheduler is installed, then disable the script-blocking feature to prevent the software from unexpectedly blocking agent script calls.

What Privileges Affect Agents?

You access the privilege settings for agents in the Delivers section of the Manage Privileges page in Presentation Services Administration.

To create an agent, users must be granted the Create Agent privilege. To enable users with the Publish Agents for Subscription privilege, which provides the ability to change or to delete an agent, you must grant them the Modify permission to the shared agent objects and child objects in the Presentation Catalog.

> **Note:**
>
> If the Presentation Services is configured to authenticate users through database logons, then impersonation is permitted until the number of associated variables exceeds one (for example, when session variables other than USER are associated with the initialization block). If the number of associated variables exceeds one, then the impersonated user does not have the password to log in to the database and to fill the other session variables. Agents work with database authentication, if only the initialization block that's set up for authentication in the Model Administration Tool uses a connection pool with pass-through login. That connection pool can't be used for any other initialization block or request.

Configure Settings that Affect Agents

You configure settings for agents by changing values for Presentation Services or Oracle BI Scheduler.

You configure delivery options for agents using the SA System subject area. This section contains the following topics:

- Manually Configure Presentation Services Settings that Affect Agents
- Manually Change Additional Scheduler Settings that Affect Agents
- What Additional Scheduler Configuration Settings Affect Agents?
- Control Delivery Options for Agents

Manually Configure Presentation Services Settings that Affect Agents

Use various elements in the instanceconfig.xml file for Presentation Services to change these settings. You must apply changes to both the primary and secondary scheduler's instanceconfig.xml in a cluster.

1. Open the instanceconfig.xml file for editing, located in:

   ```
   BI_DOMAIN/config/fmwconfig/biconfig/OBIPS
   ```

2. Locate the section in which you must add the elements that are described in the table below.
3. Include the elements and their ancestor elements as appropriate, as shown in the following example:

```
<ServerInstance>
  <Alerts>
     <Enabled>false</Enabled>
     <DefaultDeliveryFormat>pdf</DefaultDeliveryFormat>
  </Alerts>
</ServerInstance>
```

4. Save your changes and close the file.
5. Restart Presentation Services.

Configuration Settings for Presentation Services that Affect Agents

You can use configuration settings for Presentation Services that affect agents.

The table describes the configuration settings for Presentation Services that affect agents.

Element	Description	Default Value
Enabled	Specifies whether Delivers is enabled. Possible values are true or false. Delivers is an optional component of Presentation Services that is enabled by default for organizations that have purchased the appropriate license. You use the Delivers component to create agents.	true
DefaultDeliveryFormat	Specifies the default format for sending emailed reports through an agent. For example, a content designer can create an agent to send a report every day to a development team to share how many bugs have been fixed in the past day. When the content designer creates the agent, he can specify the format of the email. As the administrator, you can specify the default format that is used for such emails, using one of the following values: html pdf excel text	html

Manually Change Additional Scheduler Settings that Affect Agents

In addition to the scheduler settings that you can change in Fusion Middleware Control, you can change other settings manually. Use various elements in the `schedulerconfig.xml` file to change these settings. You must apply changes to both the primary and secondary scheduler's `schedulerconfig.xml` in a cluster.

1. Open the Oracle BI Scheduler `schedulerconfig.xml` file for editing, located in:

   BI_DOMAIN/config/fmwconfig/biconfig/OBISCH

2. Locate the sections in which you must add or update the elements described in What Additional Scheduler Configuration Settings Affect Agents?

3. Include the elements and their ancestor elements as appropriate. The entry for <Log> is shown in the following example:

   ```
   <Log>
       <Level>
           <Error>1</Error>
           <IncidentError>1</IncidentError>
           <Trace>0</Trace>
           <Warning>1</Warning>
           <Notification>1</Notification>
       </Level>
       <MaximumFileSizeKb>512</MaximumFileSizeKb>
       <MaximumLogAgeDay>3</MaximumLogAgeDay>
   </Log>
   ```

   > **Note:**
   >
   > You can't specify values for user name and password in the `schedulerconfig.xml` file. Instead you specify values in Fusion Middleware Control that are stored securely within the central credentials wallet, along with all other user names and passwords.

4. Save your changes and close the file.
5. Restart the system.

What Additional Scheduler Configuration Settings Affect Agents?

Agents can be changed by scheduler configuration settings.

You can change the following additional Oracle BI Scheduler configuration settings that affect agents:

- General Scheduler Configuration Settings that Affect Agents
- Email Scheduler Configuration Settings that Affect Agents
- Agent Scheduler Configuration Settings

General Scheduler Configuration Settings that Affect Agents

General configuration settings include access to, and configuration of, the Scheduler back-end database, some behavior settings, and settings for secure sockets and clustering configuration.

The table below describes the settings.

Element	Description	Default Value
PoolTimeout	Specifies the amount of time in minutes that a connection to the data source remains open after an operation completes. During this time, new operations use this connection rather than open a new one, up to the number specified for Maximum Connections. The time is reset after each completed connection request. Specify a value of 1 or greater.	60
NumDbConnections	Specifies the maximum number of database connections that Oracle BI Scheduler can open concurrently. Specify a value of 1 or greater. When this limit is reached, the connection request waits until a connection becomes available.	5
TABLE_JOBS	Specifies the name of a database table used to store information about scheduled jobs.	S_NQ_JOB
TABLE_INSTANCES	Specifies the name of a database table used to store information about job instances.	S_NQ_INSTANCE
TABLE_PARAMS	Specifies the name of a database table used to store information about job parameters.	S_NQ_JOB_PARAM
TABLE_ERRMSGS	Specifies the name of a database table used to store information about job instances that don't complete successfully.	S_NQ_ERR_MSG
SchedulerScriptPath	Refers to the path where Oracle BI Scheduler-created job scripts are stored. In general, don't add or remove scripts from this directory. By default, this field is set to: `BI_DOMAIN/servers/obisch1`	scripts\scheduler

Element	Description	Default Value
DefaultScriptPath	Specifies the path where user-created job scripts (not agents) are stored. If a file name is entered in the Script field when adding or modifying a job, then Oracle BI Scheduler examines the contents of this directory for the specified file. However, if a full path is given in the Script field, then this directory isn't examined. By default, this field is set to: `BI_DOMAIN/servers/obisch1`	scripts\common
TempPath	Specifies the path where temporary files are stored during Oracle BI Scheduler execution.	No default value
BulkFetchBufferSize	Used in the database gateways. Specifies the maximum size in bytes of a bulk fetch page for retrieving data from a data source.	33,792
LogAllSqlStmts	Occasionally you might set up the Scheduler to point to a database using a generic protocol like ODBC. This is usually done when the Target Type isn't specified. When this happens, and a SQL statement fails, you must be able to determine which statement failed. Turning this setting on places the SQL statements in the Scheduler log file. Don't set this to TRUE in production mode as the overhead for this is quite high.	false
ServerPortNumber	Specifies the port number for the server. Defaults to the Oracle BI Scheduler port number.	The port number set for the Scheduler.
PurgeInstDays	Specifies the number of days after which old job instances are deleted from the back-end database automatically.	7

Element	Description	Default Value
PurgeIntervalMinutes	Specifies the number of minutes in which Oracle BI Scheduler updates the tables and flags the affected rows as deleted. **Note**: Oracle BI Scheduler doesn't actually issue SQL DELETE statements when jobs or instances are removed, instead rows are flagged for deletion. After every *X* minutes (where *X* is defined as the value of this field), the actual SQL DELETE statements are issued.	60
MinExecThreads	Specifies the minimum number of multiple threads in the Oracle BI Scheduler thread pool that runs jobs at runtime.	1
MaxExecThreads	Specifies the maximum number of multiple threads in the Oracle BI Scheduler thread pool that runs jobs at runtime.	100
PauseOnStartup	Specifies that no jobs are run when Oracle BI Scheduler starts. While Oracle BI Scheduler pauses, users can add, modify, and remove jobs. However, no jobs run. From the Service Management menu, select **Continue Scheduling** to continue with regular execution.	false
CertificateFileName	Specifies the SSL Certificate File Path. This setting supports SSL.	No default value
CertPrivateKeyFileName	Specifies the SSL Certificate Private Key File. This setting supports SSL.	No default value
PassphraseFileName	Specifies the SSL File Containing Passphrase. This setting supports SSL.	No default value
PassphraseProgramName	Specifies the SSL Program Producing Passphrase. This setting supports SSL.	No default value
CertificateVerifyDepth	Specifies the SSL Certificate Verification Depth.	No default value
CACertificateDir	Specifies the CA Certificate Directory. This setting supports SSL.	No default value
CACertificateFile	Specifies the CA Certificate File. This setting supports SSL.	No default value
TrustedPeerDNs	Specifies the SSL Trusted Peer DNs.	No default value

Element	Description	Default Value
VerifyPeer	Specifies whether to verify the peer. This setting supports SSL.	false
CipherList	Specifies the Cipher List. This setting supports SSL.	No default value
ScriptRPCPort	Specifies the port used for interprocess communication between the script processes and the Oracle BI Scheduler process. This port accepts connections only from the same computer on which Oracle BI Scheduler is running.	9707

Email Scheduler Configuration Settings that Affect Agents

Configure agents by specifying the following configuration settings.

Element	Description	Default Value
SmtpCipherList	Specifies the list of ciphers that match the cipher suite name that the SMTP server supports. For example, RSA+RC4+SHA. .	No default value
UseStartTLS	Ignored unless UseSSL is true. If UseStartTls is true, then use the STARTTLS option (RFC 2487) for the SMTP session. Initial connection is through an unsecured link, usually port 25. The connection is then promoted to a secure link using the STARTTLS SMTP command. If UseStartTls is false, then a secured connection is created immediately, before the SMTP protocol is started. This is also known as SMTPS. SMTPS typically uses port 465.	true

Agent Scheduler Configuration Settings

Agents are functionally a combination of data that is stored in Oracle BI Server and Oracle BI Scheduler.

The elements in the Scheduler instanceconfig.xml file describe the behavior of all agents that run on a specific Oracle BI Scheduler. The table below describes each agent configuration element.

Element	Description	Default Value
Log_Dir	Agents can create log files if exceptional error conditions occur. Log_Dir specifies the directory where these files are saved. The directory must be accessible to the Oracle BI Scheduler server. In Windows, the default installation runs the service as a system account, which prevents Oracle BI Scheduler from writing to or reading from network directories. If you put script files on network shares, or your scripts access network shares, then Oracle BI Scheduler must be run as a network user. For information about log files, see Diagnose Issues with Agents.	ibots
LogPurgeDays	Specifies the number of days after which old agent logs are deleted automatically. To prevent old logs from being deleted automatically, set the value to 0 (zero).	7
NumGlobalRetries	A web or mail server that has too many people logged on might reject new connections, including connections from Oracle BI Scheduler. To cope with such overload, an agent retries the connection. This element sets the maximum number of retries to obtain global information about what to deliver and to whom before the agent gives up. If you set this value to 0 (zero), then no retries are attempted.	2
MinGlobalSleepSecs	Specifies the minimum number of seconds that the agent randomly sleeps after its connection is refused before it attempts to reconnect to the server to obtain global information about what to deliver and to whom.	3
MaxGlobalSleepSecs	Specifies the maximum number of seconds that the agent randomly sleeps after its connection is refused before it attempts to reconnect to the server to obtain global information about what to deliver and to whom.	10
NumRequestRetries	After an agent has received the global information, it issues a series of unique requests to the server for each user. This element specifies the number of times Oracle BI Scheduler retries its attempts to connect to the server to issue these requests. If you set this value to 0 (zero), then no retries are attempted.	3
MinRequestSleepSecs	Specifies the minimum number of seconds that the agent randomly sleeps after its connection is refused before it attempts to reconnect to the server to issue requests.	2
MaxRequestSleepSecs	Specifies the maximum number of seconds that the agent randomly sleeps after its connection is refused before it attempts to reconnect to the server to issue requests.	10
NumDeliveryRetries	After a unique request has run, the agent tries to deliver the results to specified devices. This specifies the number of times that Oracle BI Scheduler attempts to retry to connect to the server to deliver the results. If you set this value to 0 (zero), then no retries are attempted.	4
MinDeliverySleepSecs	Specifies the minimum number of seconds that the agent randomly sleeps after its connection is refused before it attempts to reconnect to the server to deliver results.	5
MaxDeliverySleepSecs	Specifies the maximum number of seconds that the agent randomly sleeps after its connection is refused before it attempts to reconnect to the server to deliver results.	10

Element	Description	Default Value
MaxRowsTimesColumns	When agents are chained, this value governs the size of filters passed between agents. When you pass a filter to another agent in a chain, Oracle BI Scheduler creates a union of the result sets for the Conditional Report for each personalized recipient. This report can grow very large in some cases (1000 users with 100 unique rows each with ten columns per report = 1,000,000 column values in the filter). The Oracle Analytics Server servers might not be able to handle such a large filter, so this element specifies the maximum number of rows*columns in the filter.	10,000
Debug	Debug Enabled. Set this element to have Oracle BI Scheduler generate a log file for each agent. This log file has useful logging messages when trying to diagnose a problem. This log file is stored in *BI_DOMAIN*/servers/obisch1/logs A new log file named Agent-<*Job number*>-<*Instance number*>.log is created for each job instance. See Diagnose Issues with Agents.	false
KeepErrorLogFiles	Set this element to true to generate an error log file for each agent. This log file is created only when an agent execution encounters errors and contains only error messages. The file is stored in: *BI_DOMAIN*/bi/servers/obisch1/logs	true

Control Delivery Options for Agents

Delivery options (that is, delivery devices and delivery profiles) determine how the contents of agents are delivered to users.

Delivery options can be configured by users, in the LDAP server (for email addresses), or in the SA System subject area. See Appendix A Setting Up the SA System Subject Area in *Scheduling Jobs Guide for Oracle Business Intelligence Enterprise Edition*.

Manage the Types of Devices that Deliver Content

Oracle Analytics can deliver content to a wide range of devices. You can add more devices for your organization, if users want to receive content on a device that’s not on the list. You can’t edit or delete default devices, such as AT&T Wireless.

1. On the Classic Home page, click the user profile icon and then click **Administration**.
2. Click **Manage Device Types**.
3. To define a new type of device:
   a. Click **Create New Device Type**.
   b. Enter information about the device, and click **OK**.
4. To edit a device that you added:
   a. Click **Edit**.
   b. Make your changes, and click **OK.**
5. To delete a device that you added:
   a. Click **Delete**.

b. Click **OK** to confirm.

Monitor Active Agent Sessions

Using the Manage Agent Sessions page in Presentation Services Administration, you monitor currently active agent sessions that are triggered by Oracle BI Scheduler. For example, you can see a list of active agents per session.

When one or more agent sessions are active, information about each agent session is displayed, such as the job identifier and the instance identifier that are assigned to the agent session by the Oracle BI Scheduler. Expanding the agent session shows the individual agents (one agent, or multiple agents if they are chained). The state of the agent is either Created, Populated, or Conditional Request Resolved.

Expanding a specific agent in a particular session shows the recipients for the agent and their type, such as the Engineering recipients defined in a group, or individual users. When the recipient is a group, the individual members of the group are not listed.

> **Note:**
>
> When agents are chained, the recipient list is depends on the parent agent. The recipients are shown for the parent agent definition only, and not for the actual execution of chained agents.

1. In the global header, click **Administration**.
2. Click the **Manage Agent Sessions** link to display the Manage Agent Sessions page.
3. The **Agent Sessions** section displays a table with information about each agent session. You can see details such as the job identifier, the instance identifier, the primary agent, and the day and time at which the agent session was last accessed. Do one of the following:
   - To sort agent sessions by their values in a particular column, click the **Sort** button for that column.

     Re-sorting the list causes the page to refresh so the number of active agent sessions might increase or decrease as a result.
   - Click the link in the Primary Agent column to display the agent in the Agent editor.
   - To view more information about an agent session or about agents within a particular session, click the **Expand** button.
4. The **Agents** section displays a table with information about each agent. You can see details such as the path for the agent and its state. The state can be either Created, Populated, or Conditional Request Resolved.Do one of the following:
   - Click the link in the **Path** column to display the agent in the Agent editor.
   - Click the Expand button to see the list of recipients for that agent.
   - To view the definition of an individual agent, click its link.

22

Configure Advanced Options for Maps and Spatial Information

This chapter describes advanced configuration options that you can set for map views in Oracle Analytics Server.

Topics:

- Configure MapViewer to Support Map Views
- Manually Configure Map Views
- Insert Text on a Map
- Configure Maps for External Consumption

See Configure Map and Spatial Information.

Configure MapViewer to Support Map Views

Oracle Fusion Middleware MapViewer is installed as part of Oracle Analytics Server and deployed in the same BI domain on the web application server.

The default context path of MapViewer in the application server is `/mapviewer`. You must use the administration console in MapViewer to configure it for use with map views. .

You can configure a separate remote instance just for MapViewer to act as a proxy that supports the heavy processing load that maps require. If performance isn't a major concern, then you can use a MapViewer instance that's co-located with Oracle Analytics Server as the rendering engine.

The MapViewer engine can serve in the following roles:

- Co-located MapViewer — Also known as nonproxy mode. If the MapViewer is located in the BI domain and used as the rendering engine, then all map resources (such as JavaScript files and images) are downloaded from that instance of MapViewer.
- Remote MapViewer — Also known as proxy mode. If a separate remote instance of MapViewer is configured as the rendering engine, then the browser can't communicate with the remote instance for resources. Browsers don't permit cross-domain AJAX calls for security reasons. To overcome this limitation, all requests are first forwarded to the co-located MapViewer, which in turn communicates with the actual remote instance.

  Complete the following steps to configure for a remote MapViewer:

  – Edit the RemoteOracleMapViewerAbsoluteURL element in the instanceconfig.xml file, as described in Manually Configure Map Views.

  – Edit the proxy_enabled_hosts element in the mapViewerConfig.xml configuration file for MapViewer to point to the MapViewer on the remote server, as shown in the following example:

```
<security_config>
  <proxy_enabled_hosts>http://remoteserver:9704/mapviewer</
```

```
proxy_enabled_hosts>
  </security_config>
```

The mapViewerConfig.xml file is located in the following directory:

```
ORACLE_BI1\bifoundation\jee\mapviewer.ear\web.war\WEB_INF\conf
```

> **Note:**
>
> If you edit the mapViewerConfig.xml file, then your edits are overwritten when you upgrade or patch Oracle Analytics Server. Before upgrading or patching, ensure that you make a backup copy of the mapViewerConfig.xml file. When the upgrade or patch operation is complete, copy the backed-up file to its original location. After copying the file, restart the server for MapViewer so that map views are displayed properly in analyses and dashboards in Oracle Analytics Server.

Manually Configure Map Views

Use various elements in the instanceconfig.xml file to configure map views.

1. Open the instanceconfig.xml file for editing, located in:

   ```
   BI_DOMAIN/config/fmwconfig/biconfig/OBIPS
   ```

2. Search for the SpatialMaps section, in which you must add the following elements:

   - ColocatedOracleMapViewerContextPath — Specifies the context path of the MapViewer instance that is co-located with Oracle Analytics Server. The default value is /mapviewer.
   - RemoteOracleMapViewerAbsoluteURL — Specifies the URL of the remote MapViewer instance. This element has no default value.

     If this element has no value, then the system assumes that the map rendering engine is the co-located MapViewer instance (such as /mapviewer). If this element has a value, then the co-located MapViewer acts as proxy for all requests for the remote server. The following example shows a sample value:

     ```
     <RemoteOracleMapViewerAbsoluteURL>http://remoteserver:9704/mapviewer </
     RemoteOracleMapViewerAbsoluteURL>
     ```

   - MaxRecords — Specifies the maximum number of records that can be included in a layer on the map. The setting applies to all layers on the map and overrides the MaxVisibleRows element that applies to data cubes. The default value is 500. If the format for a layer causes this value to be exceeded, then a warning message is displayed. The parent element is LayerDataLayout.
   - SyndicatedOracleMapViewerContextPath — Specifies the URL of the MapViewer instance for embedding maps in external pages. For details and an example, see Configure Maps for External Consumption.

3. Include the elements and their ancestor elements as appropriate, as shown in the following example.

   ```
   <ServerInstance>
      <SpatialMaps>
   ```

```
      <ColocatedOracleMapViewerContextPath>/mapviewer</
ColocatedOracleMapViewerContextPath>
      <RemoteOracleMapViewerAbsoluteURL></RemoteOracleMapViewerAbsoluteURL>
      <LayerDataLayout>
        <MaxRecords>600</MaxRecords>
      </LayerDataLayout>
   </SpatialMaps>
</ServerInstance>
```

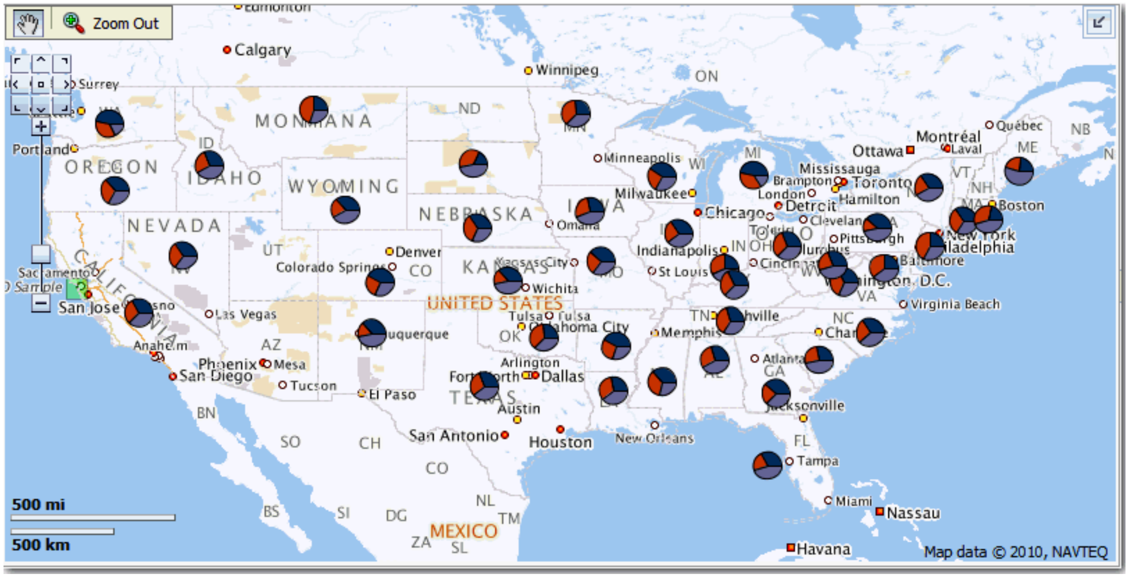
4. Save your changes and close the file.
5. Restart the system.

Insert Text on a Map

You can add any text, such as a copyright string, to the tile layer definition of a map.

The text is automatically updated on the map in Oracle Analytics Server when a tile layer is added or deleted or becomes invisible. The position of the text is also automatically adjusted when the user minimizes, restores, or removes the overview map.

The figure below shows an example of a copyright string on a map. The string is in the lower-right corner.



1. Create the tile layer.
2. Edit the tile layer definition by selecting it and clicking **Edit / View Details** .
3. On the Edit page, enter the appropriate text in the **Copyright text** field.

   For example, the following code displays the copyright line shown in the figure above.

   ```
   <copyright>Map data © 2020, NAVTEQ</copyright>
   ```

4. Click **Submit** to save your changes.
5. If you don't see the updated text on the map, then click the browser's **Refresh** button to refresh the map.

Configure Maps for External Consumption

You can embed map views in external pages, such as those from Oracle WebCenter Portal, after setting the appropriate configuration options.

To embed views that are hosted in a separate web application server, you can follow the proxy rules that are outlined in the previous sections. Because of browser restrictions, install the MapViewer instance in the same application server as external pages or portals. The proxy context path of the MapViewer instance that is installed in the web application server that hosts external pages can differ from the application server that is hosting content for Oracle Analytics Server. In this case, set the **SyndicatedOracleMapViewerContextPath** element. When the server for Presentation Services identifies a request that originated from a third-party page, the server checks the element value to determine where to pass the proxy requests.

The following example provides sample values for this element.

```
<ServerInstance>
   <SpatialMaps>
      <SyndicatedOracleMapViewerContextPath>/mapviewer</
SyndicatedOracleMapViewerContextPath>
      <RemoteOracleMapViewerAbsoluteURL>http://myserver:9704/mapviewer</
RemoteOracleMapViewerAbsoluteURL>
   </SpatialMaps>
</ServerInstance>
```

23

Configure Resource Availability and URL Generation

This chapter describes how to configure how resources are made available for HTTP access and how URLs are generated by Presentation Services.
To perform this configuration, you modify the instanceconfig.xml file to include the URL element and its interrelated subelements, as described in the following procedure.

1. Open the instanceconfig.xml file for editing, located in:

   BI_DOMAIN/config/fmwconfig/biconfig/OBIPS

2. Locate the section in which you must add the elements that are described in the table below.
3. Include the elements and their ancestor elements as appropriate, as shown in the following example:

```
<ServerInstance>
 <URL>
    <AbsoluteCommandURLPrefix>value</AbsoluteCommandURLPrefix>
    <CustomerResourcePhysicalPath>value</CustomerResourcePhysicalPath>
    <CustomerResourceVirtualPath>value</CustomerResourceVirtualPath>
    <ForceAbsoluteResourceURL>value</ForceAbsoluteResourceURL>
    <ResourcePhysicalPath>value</ResourcePhysicalPath>
    <ResourceServerPrefix>value</ResourceServerPrefix>
    <ResourceVirtualPath>value</ResourceVirtualPath>
  </URL>
 </ServerInstance>
```

4. Save your changes and close the file.
5. Restart the system.

Configuration Settings for URLs and Resource Files

You can use configuration settings for generating URLs and resource files.

The table describes the configuration settings for URLs and resource files.

Element	Description	Default Value
AbsoluteCommandURLPrefix	Specifies how Presentation Services generates command URLs. If you explicitly specify an value, then it must be of the following form: `protocol://server/virtualpath` where *virtualpath* is the complete virtual path to Presentation Services. The default is determined separately for each client, based on the URL that the client sends to Presentation Services.	Varies

Element	Description	Default Value
CustomerResourcePhysicalPath	Specifies the physical location of resource files that are not part of a default installation. Such resource files include customized styles and skins. The internal default is `ORACLE_HOME\bi\bifoundation\web\appv2\res`. You must provide a full path. Presentation Services must have read permission to this path. For example, if this is a shared network resource, then you must ensure that the user under which Presentation Services is running has read access to the shared resource and read access to the file system from which the shared resource is exported.	Varies
CustomerResourceVirtualPath	Specifies the virtual path used for resource files that are not part of a default installation as specified in the `CustomerResourcePhysicalPath` element.	No default value
ForceAbsoluteResourceURL	Specifies whether Presentation Services always generates fully qualified URLs for resource files that have fully qualified virtual paths. When set to false, resources and the Presentation Services extension are served from one server. When set to true, default resources are served from the same server as the Presentation Services extension, and customer resources are served from another server. Depending on the value of the other settings described in this table, you can also configure to have default and customer resources served from one server, and the Presentation Services extension served from another server.	false
ResourcePhysicalPath	Specifies the physical location of the primary resource files for Presentation Services. These are the resource files that are distributed with Presentation Services, not user-customized files such as custom styles or skins. The internal default is `ORACLE_HOME\bi\bifoundation\web\appv2\res`. You must provide a full path. Presentation Services must have read permission to this path. For example, if this is a shared network resource, then you must ensure that the user under which Presentation Services is running has read access to the shared resource and read access to the file system from which the shared resource is exported. If the value for this entry is different from the physical location of the DLLs for Presentation Services, then you must specify a value for the ResourceVirtualPath element.	No default value
ResourceServerPrefix	Specifies how Presentation Services generates URLs for static resources such as images, script files, style sheets, and other user-specified files. The default is *protocol*://*server* from the AbsoluteCommandURLPrefix element. If you explicitly specify a value, then it must be of this form: `protocol://server` If you specify a virtual path, then it is removed. This element designates a separate web server for delivering static resources, thereby reducing the load on the main web server. This prefix is used for the resources that have a fully qualified virtual path of the form '/Path/file'. If a resource file has a relative virtual path of the form 'Path/file', then the prefix used is the same one that is used for commands to the Presentation Services extension.	*protocol*://*server*

Element	Description	Default Value
ResourceVirtualPath	Specifies the virtual path used for the primary resource files for Presentation Services, as specified by the ResourcePhysicalPath element. These resource files and customer-defined resource files must be served from the same web server. For generating relative URLs, the virtual path defaults to res, if the resource folder is present under the same virtual directory as the Oracle BI Presentation Services DLL files. For generating absolute URLs, the value of the AbsoluteCommandURLPrefix element is used as the default. The value must be a fully qualified virtual path of this form: `'/`*`VirtualPath`*`'` If you omit the leading slash, then one is added.	res

24

Use Custom Scripts to Curate Your Data

Use custom scripts to apply your own machine logic algorithms and curate your data using data flows.

Topics:

- Upload Custom Scripts for Machine Learning and Data Curation

Upload Custom Scripts for Machine Learning and Data Curation

You can upload custom scripts to Data Visualization to use with machine learning.

1. In the Home page, click **Create**, then click **Script** to display the Add Script dialog.

   > **Note:**
   >
   > You must be an administrator to upload scripts. If you're an administrator and you can't see the **Script** button, it means that this option is currently disabled for your instance.

2. Complete one of the following tasks in the Add Script dialog to choose a custom script that you want to add to Data Visualization.

   - Click the link to browse and select a custom script.
   - Drag and drop a custom script to the Add Script dialog.

   A preview of the selected custom script is displayed in the Add Script dialog.
   You can preview custom script details and update some display information before you upload it to Data Visualization.

3. (Optional) View or edit the **Name** and **Description** of the custom script.
4. (Optional) Click **Change file** if you want to choose a different custom script to upload.
5. (Optional) Display tabs to preview additional read-only information about the custom script.

   For example:

   - Click **Parameters** to view the parameters used in this custom script.
   - Click **Inputs** to view the inputs used in this custom script.
   - Click **Outputs** to view the outputs used in this custom script.

6. Click **OK** to upload the custom script to Data Visualization.

The uploaded custom script is now available for use and is displayed in the Scripts tab on the Machine Learning page. The script is also available in the data flow editor when you add a Apply Custom Script step to a flow.

Part VII

Manage the Life Cycle

This part describes life cycle management tasks for Oracle Analytics Server. Life cycle management consists of installing, updating, patching, moving to a new environment, moving to a production environment, and backing up and recovering systems.

For information about installation, see *Installing and Configuring Oracle Analytics Server*.

For information about upgrade, see *Migrating and Upgrading Oracle Analytics Server*.

This part includes the following chapters:

- Patch Oracle Analytics Server Systems
- Move Oracle Analytics Server Between Environments
- Backup and Recovery

25

Patch Oracle Analytics Server Systems

This chapter provides information on patching Oracle Analytics Server.
See *Oracle Fusion Middleware Patching Guide*.

Topics:

About Patching Oracle Analytics Server Systems

Patching involves copying a small collection of files over an existing installation.

A patch is normally associated with a particular update of an Oracle product and involves updating from one minor update of the product to a newer minor update of the same product (for example, from update 5.5.0.0.0 to update 5.5.0.1.0). A patch set is a single patch that contains a collection of patches that are designed to be applied at the same time.

Typically you apply a patch that contains one or more bug fixes to an existing production Oracle Analytics Server system that's distributed across one or more computers. Bug fixes might affect the system components and Java components that are deployed inside the Oracle WebLogic Server. The patch might include new server executables and updated and new Java class files.

Where Do I Find Complete Information on Patching?

You use the Oracle OPatch utility to apply (and to roll back) Oracle Analytics Server platform patches. You download patches from Oracle Support Services. Consult the following sources:

- For complete information about patching in Oracle Fusion Middleware, see *Oracle Fusion Middleware Patching Guide*.
- The patch readme contains all the information necessary to understand what the patch does and the steps that you must perform to apply the patch. You access the Readme documentation from the Patches & Updates screen on the My Oracle Support site when downloading patches.

Updating an Existing Oracle Analytics Installation

This section explains how you update an existing Oracle Analytics Server installation.

Assumptions:

- For one-off patches you download <*patch-number*>.ZIP from:

  https://support.oracle.com
- For all other updates you download oa_platform-5.5.x.y.z-<platform>.<extension> from:

  https://otn.oracle.com

- Software updates only change ORACLE_HOME and PRODUCT_HOME content, DOMAIN_HOME and BI_DATA_HOME is untouched.

Pre-conditions:

All product processes have been shut down.

1. For each distinct ORACLE_HOME, run the Oracle Universal Installer.

   For information on installation, see Prepare to Install and Configure the Software in *Installing and Configuring Oracle Analytics Server*.

2. (Optional) For each distinct ORACLE_HOME, you launch OPatch (`ORACLE_HOME/OPatch/opatch`) with the zip file.

   This updates the ORACLE_HOME.

What Happens If a Patching Conflict Occurs?

If a patching conflict occurs, then the process stops. An example of a conflict is when a duplicate patch fixes the same set of bugs fixed by another patch.

For details on resolving patch conflicts, see *Oracle OPatch User's Guide*. Contact Oracle Support Services for assistance in resolving conflicts.

What Is Patched for the Oracle Analytics Server Platform?

Oracle Analytics Server platform patching applies patches for binary files with extensions such as DLL, JAR, and EXE.

Oracle Analytics Server platform patching doesn't patch the following:

- Configuration Files

  If configuration updates are required as part of a patch, then these are detailed in the accompanying README.txt file, and you must manually apply them. No automated mechanism is available for merging customer configuration and patched configuration files.

- Schema-based Metadata

  Information that's stored in database schemas (including schemas for the Scheduler, usage statistics, event polling,semantic model files, and the Oracle BI Presentation Catalog) isn't patched.

- Application Metadata

  Updates to applications metadata (for example, BAR files and semantic model files). Patching of these files is the responsibility of the application provider.

Roll Back a Platform Patch

OPatch maintains metadata for each patch that is applied to each Oracle home and keeps copies of what it replaces during a patch.

You can roll back a complete patch.

> **Note:**
>
> To confirm that an Oracle Analytics Server platform patch is no longer applied after a rollback, you must establish the patch levels before applying the rollback, then repeat the task after rollback. See Determine Current Patch Levels.

Determine Current Patch Levels

Each Oracle home must be patched to the same version as OPatch to ensure that Oracle Analytics Server functions properly.

Use the OPatch lsinventory utility to determine the current patch version for any given Oracle home in the system. You can also use the utility to retrieve a full list of patches, with their corresponding IDs, for a given Oracle home.

1. Display a command window and navigate to the location of the OPatch executable:

   ```
   ORACLE_HOME/OPatch
   ```

2. Run the lsinventory utility using the following command syntax:

   ```
   Path_to_OPatch/opatch lsinventory [-all] [-detail] [-patch] [-oh (Oracle home
   location)]
   ```

   For example:

   ```
   opatch lsinventory -patch -detail
   ```

   For information about the lsinventory options, see the user guides in the `ORACLE_HOME/OPatch/docs` directory.

3. To run the lsinventory utility against other Oracle homes, repeat the previous steps for each Oracle home.

See *Oracle Fusion Middleware Patching Guide*.

26 Move Oracle Analytics Server Between Environments

This chapter describes how to move Oracle Analytics Server between environments. You can move Oracle Analytics Server from a test to a production environment or to a new environment.

Topics:

- Take Snapshots and Restore
- Move Between Environments Using the Oracle WebLogic Scripting Tool
- Move to a New Environment
- Migrate the Whole Server

Move Between Environments Using the Oracle WebLogic Scripting Tool

Moving from a test to a production environment applies solely to metadata (content, data model and authorization).

To move from a test environment to a production environment you would typically develop and test applications in a test environment and then roll out the test applications (and optionally test data), in the production environment on the same operating system. This can also include moving from a single to a multiple computer environment.

You can also use snapshots to move between environments in Oracle Analytics Server instead of using the Oracle WebLogic Scripting Tool. See Take Snapshots and Restore.

Prerequisites:

- You must have file system (offline) permissions.
- Service instance deployments A and B must exist in different domains.
- Model Administration Tool (the data model), Answers (the Presentation Catalog), Fusion Middleware Control (security), or Oracle WebLogic Scripting Tool. You only need these tools if you want to make changes to metadata.

To move from a test to a production environment, the following options are available:

- export all

  You might use this option for example, if all users and data sources are the same between two systems.
- export without user folder content

  You might use this option as part of user acceptance testing, where test users have different content but access to the same data sources (although with different data access)
- export without connection pool credentials

You might use this option if all you need to do is to move updated metadata from one system to another after it's been tested (no users in common, different data source security).

To support these options the export command has optional parameters to export user folder content, or connection pool credentials as follows:

```
$<DOMAIN_HOME>/bitools/bin/exportarchive.sh
```

exportarchive(domainHome, serviceInstanceKey, workDir, exportDir, applicationName=None, applicationDesc=None, applicationVersion=None, includeCatalogRuntimeInfo=false, includeCredentials=None)

1. Optional: If you want to preserve existing user production service instance folders, use the runcat archive command against production to save user folders.
2. Run the exportarchive command to export the TEST service instance to a BAR file (with or without user folders and connection credentials).

   For example:

   ```
   exportarchive('/u01/Oracle/Middleware/Oracle_Home/user_projects/domains/
   bi', 'ssi', '/u01/workDir', '/u01/exportDir', applicationName=None,
   applicationDesc=None, applicationVersion=None,
   includeCatalogRuntimeInfo=false, includeCredentials=None)
   ```

   This example exports the service instance with key ssi, to /u01/exportDir/ssi.bar.
3. Run the importarchive command to import the exported TEST BAR file into the PROD service instance.

   ```
   $<DOMAIN_HOME>/bitools/bin/importarchive.sh

   importarchive('/u01/Oracle/Middleware/Oracle_Home/user_projects/domains/
   bi','ssi','/u01/exportDir/ssi.bar')
   ```

   This example imports the archive BAR file /u01/exportDir/ssi.bar into the service instance with key ssi.
4. If you used the runcat archive command in step 1, you can now use the runcat unarchive command to put production user folders back in place.

Post Conditions:

- The PROD service instance metadata is replaced with TEST metadata.
- Any content created directly in the PROD service instance is replaced or lost.
- No configuration is changed on the domain hosting the PROD service instance.
- No metadata or configuration is changed on the domain hosting the TEST service instance.

Move to a New Environment

You can move (or migrate) to a new environment by re-creating an existing system in a different location to the one in which it was originally installed. The objective is to re-create an identical deployment on different hardware.

You might want to move to a new environment for the following reasons:

- To move the system as a whole onto more powerful hardware.
- To move to a different operating system.
- To move into a different physical location.

Migrate the Whole Server

Oracle supports whole server migration, in which you migrate a WebLogic Server instance to a different physical computer upon failure, either manually or automatically.

See Whole Server Migration in *Administering Clusters for Oracle WebLogic Server*.

27 Backup and Recovery

This chapter describes general information on backup and recovery. Backup and recovery refers to the various strategies and procedures involved in guarding against hardware failures and data loss and in reconstructing data if loss occurs.
You can use Snapshots to backup and restore content. See Take Snapshots and Restore in *Administering Oracle Analytics Server*.

Backup and recovery is described in Advanced Administration: Backup and Recovery in *Administering Oracle Fusion Middleware*.

Disaster recovery is described in Recommendations for Oracle Fusion Middleware Components in *Oracle Fusion Middleware Disaster Recovery Guide12c R2 (12.2.1)*.

For more information about backing up and restoring metadata, see Manage Oracle Analytics Server Instances Using the WebLogic Server Scripting Tool.

Part VIII
Reference Information

There are several resources with reference information for managing Oracle Analytics Server.

This part contains the following appendixes:

- Certification Information
- Frequently Asked Questions
- Troubleshooting
- Advanced Configuration Reference
- Map User Interface Labels with Configuration File Elements
- Configuration File Settings

A

Certification Information

This appendix contains certification information for the latest update of Oracle Analytics Server.

Topics:

Certification - Supported Data Sources

Here are the supported data sources for the latest update of Oracle Analytics Server.

Notes on using this table

- **Version Numbers**:
  - "1.x" means any version starting with a 1 (for example, this includes version 1.4.3 but not version 2.0).
  - "2.0.x" means any version starting with 2.0 (for example, this includes version 2.0.4 but not version 2.4).
  - "1.6+" means any version starting with 1 and is greater than or equal to (>=) 1.6 (for example, this includes version 1.8 but not version 2.4).
- **Use in Datasets** - "Yes" means that you can connect to a data source for analysis by creating a connection from the Oracle Analytics home page.
- **Use in Data Models** - "Yes" means that you can connect to the semantic layer of a data source and model the data of that type. You can use the Console or Oracle Analytics Developer Client Tool to connect.
- In addition to the connections types listed on the Connections page, you can connect to other data sources using the **JDBC** connection type.

Data Sources Supported for Use with Oracle Analytics Server (sorted alphabetically by Oracle databases first, then other databases)

> **Note:**
>
> To access Cloud databases across the public internet, Oracle Analytics Server must be deployed on a server that doesn't require proxy access. If you're not sure, check with your administrator.

Database	Supported Versions	Datasets	Data Models - Semantic Modeler	Data Models - Model Administration Tool	More Information
Oracle Database	12.1+ 12.2+ 19+ 21c 23ai	Yes	Yes	Yes	Supports saving output from data flows.
Oracle Analytic Views	Oracle Database 19c and later	Yes	No	Yes	See Connect to Oracle Analytic Views.
Oracle Applications	11.1.1.9+ or Oracle Fusion Applications Release 8 and later Oracle Fusion Cloud Applications Suite, On-premises Oracle BI Enterprise Edition deployments, Another Oracle Analytics service	Yes	No	No	-
Oracle Autonomous Data Warehouse	18.2.4+ 19c and later 23ai	Yes	Yes	Yes	Supports saving output from data flows. You can also connect via Delta Sharing using the Delta Share connection type. See Connect to a Database Using Delta Sharing.
Oracle Autonomous Transaction Processing	19c and later 23ai	Yes	Yes	Yes	Supports saving output from data flows.

Database	Supported Versions	Datasets	Data Models - Semantic Modeler	Data Models - Model Administration Tool	More Information
Oracle Enterprise Performance Management Cloud	Current version	Yes	Yes	Yes	You can't blend datasets that use Oracle Enterprise Performance Management Cloud data sources. See Connect to Oracle Fusion Cloud Enterprise Performance Management (EPM).
Oracle Essbase	11.1.2.4.0+ 21c	Yes	Yes	Yes	You can't blend datasets that use Oracle Essbase data sources. Connectivity to a data source in an Essbase cluster (via an Analytic Provider Services (APS) node) isn't supported.
Oracle Hyperion Financial Management	11.1.2.4+ 11.2+	No	No	Yes	-
Oracle Hyperion Planning	11.1.2.4+ 11.2+	No	No	Yes	-
Oracle NetSuite	Release 2019.2 (JDBC Driver 8.10.85.0)	Yes	No	No	See Connect to NetSuite Data Sources.
Oracle OLAP	12+	No	No	Yes	-
Oracle Service Cloud	Current version	Yes	No	No	-
Oracle Talent Acquisition Cloud	17.4+ 15b.9.3+	Yes	No	No	-
Oracle TimesTen	18.1+* 22.1	No	No	Yes	* Requires installation and configuration of the 18.1 client software.
OCI Object Storage	-	Yes	No	No	See Create a Dataset from OCI Object Storage.
OCI Data Flow SQL Endpoints	-	Yes	No	No	Use the ODI Data Flow connection type.
Amazon EMR	4.7.2*	Yes	No	No	* Running Amazon Hadoop 2.7.2 and Hive 1.0.0 Amazon EMR (MapR) No Amazon Machine Image (AMI) 3.3.2 running MapR Hadoop M3 and Hive 0.13.1
Amazon Redshift	1.0.1036 +	Yes	Yes	Yes	-

Database	Supported Versions	Datasets	Data Models - Semantic Modeler	Data Models - Model Administration Tool	More Information
Apache Hive	2.3.0+ 3.0+	Yes	Yes	Yes	Supports saving output from data flows. Supports Kerberos
Apache Spark	1.6+ 3.0	Yes	Yes		Supports saving output from data flows. Supports Kerberos
Aster Database	5.0 6.0	No	No	Yes	-
Cloudera CDH	4.6+ 5.5+	No	Yes	No	-
Delta Share	-	Yes	No	No	Supports Oracle data sharing. Share file sizes are subject to a 250 MB limit per partition. See Connect to a Database Using Delta Sharing.
DropBox	Current version	Yes	No	No	-
Google Analytics	V4	Yes	No	No	-
Google BigQuery	8.0 8.1	Yes	Yes	Yes	See Connect to Google BigQuery.
Google Drive	Current version	Yes	No	No	-
Greenplum	4.3+	Yes	No	No	-
Hortonworks Hive	1.2+	Yes	Yes	Yes	Supports saving output from data flows. Supports Kerberos authentication for datasets
IBM BigInsights Hive	1.2+	Yes	Yes	No	Supports Kerberos authentication for datasets
IBM DB2	11.5+	Yes	Yes	Yes	-
IBM Netezza NPS	NPS 6+ NPS 7+ NPS 11+	No	No	Yes	-
Impala (Cloudera)	2.7+	Yes	Yes	Yes	-
Informix	12.10+	Yes	Yes	Yes	-
JDBC	-	Yes	Yes	No	See Connect to Data Using JDBC.
MapR Hive	1.2+	Yes	Yes	Yes	Supports saving output from data flows. Supports Kerberos authentication for datasets
Microsoft Access	2016	No	No	Yes	-

Database	Supported Versions	Datasets	Data Models - Semantic Modeler	Data Models - Model Administration Tool	More Information
Microsoft Azure SQL Database	Current version	Yes*	No	No	* Use the **SQL Server** connection type on the Create Connection page. SSL is mandatory.
Microsoft Azure Synapse Analytics	Current version	Yes	No	No	-
Microsoft Excel	2016	Yes	No	No	-
Microsoft SQL Server Analysis Services	2016 2017 2019 2022	No	No	Yes	Connectivity is through XML for Analysis (XMLA). You must configure XMLA access to Microsoft SQL Server Analysis Services.
MongoDB	3.2.5	Yes	Yes	Yes	-
MySQL	5.6+ 5.7+ 8.0+	Yes	Yes	Yes	-
MySQL Heatwave	8.0.31+	Yes	Yes	Yes	Current latest cloud version supported.

Database	Supported Versions	Datasets	Data Models - Semantic Modeler	Data Models - Model Administration Tool	More Information
OData	2.0 4.0	Yes	No	No	Oracle Analytics doesn't support the following OData features: - Columns other than the following data types: Edm.String, Edm.Int16, Edm.Int32, Edm.Int64, Edm.Double, Edm.Single, Edm.Decimal, Edm.Date, Edm.TimeOfDay, and Edm.DateTimeOffset. - Complex Types and Enums. - OData v4 Actions and Functions. - OData v4 servers with a custom $top limit. If you specify a custom limit for $top and a query exceeds the $top limit and receives a failure response from the server, then this might prevent you from loading tables in Oracle Analytics.
ODBC	-	Yes	No	Yes	-See Create Generic ODBC Connections.
Pivotal HD Hive	-	Yes	Yes	Yes	Supports Kerberos authentication for datasets.
PostgreSQL	9.0+ 13 14 15	Yes	Yes	Yes	-
Presto DB	0.149+	Yes	No	No	-

Database	Supported Versions	Datasets	Data Models - Semantic Modeler	Data Models - Model Administration Tool	More Information
REST API	-	Yes	No	No	See Create a Connection to a Data Source with REST Endpoints, and JSON Examples for Common Data Sources with REST Endpoints.
Salesforce	Current version	Yes	No	No	-
Snowflake Data Warehouse	Current version	Yes	Yes	Yes	-
Spark	1.6+ 3.0	Yes	Yes	Yes	Supports saving output from data flows.
SQL Server	2016 2017 2019 2022	Yes*	Yes*	Yes	* Supports case sensitive and case insensitive collation.
Sybase ASE (Adaptive Server Enterprise)	15.7+	Yes	Yes	Yes	-
Sybase IQ	16+	Yes	Yes	Yes	-
Teradata	16.20 17.x	Yes	Yes	Yes	-
Vertica	9.x 12.x	Yes	Yes	Yes	-

Certification - Supported Data Types

Here're the supported data types for Oracle Analytics.

Topics:

- Supported Base Data Types
- Supported Data Types by Database

Supported Base Data Types

When reading from a data source, Oracle Analytics attempts to map incoming data types to the supported data types.

For example, a database column that contains only date values is formatted as a DATE, a spreadsheet column that contains a mix of numerical and string values is formatted as a VARCHAR, and a data column that contains numerical data with fractional values uses DOUBLE or FLOAT.

In some cases Oracle Analytics can’t convert a source data type. To work around this data type issue, you can manually convert a data column to a supported type by entering SQL commands. In other cases, Oracle Analytics can't represent binary and complex data types such as BLOB, JSON, and XML.

Note that some data types aren't supported. You'll see an error message if the data source contains unsupported data types.

Oracle Analytics supports the following base data types:

- **Number Types** — SMALLINT, SMALLUNIT, TINYINT, TINYUINT, UINT, BIT, FLOAT, INT, NUMERIC, DOUBLE
- **Date Types** — DATE, DATETIME, TIMESTAMP, TIME
- **String Types** — LONGVARCHAR, CHAR, VARCHAR

Supported Data Types by Database

Oracle Analytics supports the following data types.

Database Type	Supported Data Types
Oracle	BINARY DOUBLE, BINARY FLOAT CHAR, NCHAR CLOB, NCLOB DATE FLOAT NUMBER, NUMBER (p,s), NVARCHAR2, VARCHAR2 ROWID TIMESTAMP, TIMESTAMP WITH LOCAL TIMEZONE, TIMESTAMP WITH TIMEZONE
DB2	BIGINT CHAR, CLOB DATE, DECFLOAT, DECIMAL, DOUBLE FLOAT INTEGER LONGVAR NUMERIC REAL SMALLINT TIME, TIMESTAMP VARCHAR
SQL Server	BIGINT, BIT CHAR DATE, DATETIME, DATETIME2, DATETIMEOFFSET, DECIMAL FLOAT INT MONEY NCHAR, NTEXT, NUMERIC, NVARCHAR, NVARCHAR(MAX) REAL SMALLDATETIME, SMALLINT, SMALLMONEY TEXT, TIME, TINYINT VARCHAR, VARCHAR(MAX) XML

Database Type	Supported Data Types
MySQL	BIGINT, BIGINT UNSIGNED CHAR DATE, DATETIME, DECIMAL, DECIMAL UNSIGNED, DOUBLE, DOUBLE UNSIGNED FLOAT, FLOAT UNSIGNED INTEGER, INTEGER UNSIGNED LONGTEXT MEDIUMINT, MEDIUMINT UNSIGNED, MEDIUMTEXT SMALLINT, SMALLINT UNSIGNED TEXT, TIME, TIMESTAMP, TINYINT, TINYINT UNSIGNED, TINYTEXT VARCHAR YEAR
Apache Spark	BIGINT, BOOLEAN DATE, DECIMAL, DOUBLE FLOAT INT SMALLINT, STRING TIMESTAMP, TINYINT VARCHAR
Teradata	BIGINT, BYTE, BYTEINT CHAR, CLOB DATE, DECIMAL, DOUBLE FLOAT INTEGER NUMERIC REAL SMALLINT TIME, TIMESTAMP VARCHAR

Certification - Repository Creation Utility (RCU) Databases

You can host the mid-tier schemas created by the Repository Creation Utility (RCU) on a variety of database types.

Databases supported by the latest update of Oracle Analytics Server for hosting the mid-tier schemas created by RCU (sorted alphabetically by Oracle databases first, then other databases)

Database	Supported Versions	JDBC Driver	More Information
Oracle	19+	Oracle JDBC Thin Driver 19.x+	To create or upgrade RCU schemas from an Oracle Database Standard Edition instance, apply patch 35476806, available from My Oracle Support.
Oracle	23ai	Oracle JDBC Thin Driver 19.x+	To create or upgrade RCU schemas from an Oracle Database Standard Edition instance, apply patch 35476806, available from My Oracle Support.
Oracle Autonomous Transaction Processing	19+	Oracle JDBC Thin Driver 19.x+	You must use the Oracle Autonomous Database TP consumer group. Also certified for Autonomous Database when configured for the Autonomous Transactional Processing workload type.
Microsoft SQL Server *	2016 2017 2019 2022	WebLogic JDBC Driver for Microsoft SQL Server Type 4	* You must have SQL Server Enterprise Edition. Supports case sensitive and case insensitive collation. See Recommended Configuration Settings for Usage Tracking using SQL Server.

Exceptions and Additional Information

When you migrate from BI Enterprise Edition to Oracle Analytics Server, you can't migrate scheduler data from IBM DB2 or Microsoft SQL Server. You can migrate scheduler data only from Oracle databases.

Certification - Supported Browsers

Oracle Analytics supports these web browsers:

- **Apple Safari**: Support for the most current major production release and one prior release.
- **Google Chrome**: Support for the most current major stable channel release only. Visit the Google Chrome FAQ for a description of the Chrome support policy: https://support.google.com/chrome/a/answer/188447?hl=en.
- **Microsoft Edge**: Support for the latest major version of Microsoft Edge Chromium. Beginning in Microsoft Edge v77, Microsoft has decoupled Edge from the Windows operating system and shifted to a Chromium-based browser technology and deployment model that follows a continuous 6-week update pattern, similar to Chrome and Firefox. Visit the Microsoft Edge FAQ for a description of the Edge support policy: https://docs.microsoft.com/en-au/deployedge/microsoft-edge-support-lifecycle.
- **Mozilla Firefox**: Support for the most current major Extended Support Release (ESR) version and above, in production only. Visit the Mozilla FAQ for a description of the Firefox support policy: https://www.mozilla.org/en-US/firefox/organizations/faq/.
- For mobile device operating systems, Oracle provides support for the most recent browser delivered by the device operating system only.

Certification - Clients

Here're the clients supported by the latest update of Oracle Analytics Server.

Supported Clients

Client	Processor	JDK Version	Additional information
Model Administration Tool	Microsoft Windows x64 - Windows 11 Microsoft Windows x64 - Windows 10 Microsoft Windows x64 - Windows 2016 Microsoft Windows x64 - Windows 2019	8u411+	See Download, Install, and Start the Client Tools for Oracle Analytics Server.
Oracle Analytics Publisher Desktop Tools	Microsoft Windows x64 - Windows 11 Microsoft Windows x64 - Windows 10 Microsoft Windows x64 - Windows 2016 Microsoft Windows x64 - Windows 2019	8u411+	-

Certification - Identity Servers and Access

Here're the identity servers supported by the latest update of Oracle Analytics Server.

Supported Identity Servers

Product	Version	Provider	Additional Information
Microsoft Active Directory	2016, 2019, 2022	Active Directory Authenticator	-
Microsoft Active Directory Lightweight Directory Services	2016, 2019, 2022	Active Directory Authenticator	-
Open LDAP	2.4+	Open LDAP Authenticator	-
Oracle Access Manager	11+, 12.2+	Oracle Access Manager Asserter	Oracle Access Manager is a supported SSO provider. It isn't a supported identity provider.
Oracle Identity Cloud Service	-	Oracle Identity Cloud Integrator	Oracle Identity Cloud Service is a supported identity provider and supported SSO provider.
Oracle Directory Server Enterprise Edition	11+	iPlanet Authenticator	-
Oracle Internet Directory	11+, 12+	Oracle Internet Directory Authenticator	-
Oracle Unified Directory	11+	iPlanet Authenticator	-

Certification - Interoperability

Here're the other components supported by the latest update of Oracle Analytics Server.

Supported Products and Components for Oracle Analytics Server

Product/Component	Interoperates With	Versions
Oracle Analytics Server	Oracle Weblogic Server	12.2.1.4+
Oracle Analytics Server - Delivers / Dashboards	Microsoft Office	2016, 2019, 365
Oracle Analytics Server - BI Publisher	Microsoft Office	2016, 2019, 365
Oracle Analytics Server - Answers / Dashboards	Adobe Acrobat Reader	7+
Oracle Analytics Server - BI Publisher	Adobe Acrobat Reader	7+

Certification - Systems

Here're the operating systems supported by the latest update of Oracle Analytics Server.

Supported Processors

Processor	OS Version	OS Update Level	Oracle JDK Version*	Additional information
Linux x86-64	Oracle Linux 8	0+	1.8.0_211+	Oracle Fusion Middleware installer supports JDK 1.8.0_341+.
Linux x86-64	Oracle Linux 9	0+	1.8.0_211+	Oracle Fusion Middleware installer supports JDK 1.8.0_341+.
Linux x86-64	Red Hat Enterprise Linux 8	0+	1.8.0_211+	Oracle Fusion Middleware installer supports JDK 1.8.0_341+.
Linux x86-64	Red Hat Enterprise Linux 9	0+	1.8.0_211+	Oracle Fusion Middleware installer supports JDK 1.8.0_341+.
Linux x86-64	SUSE Linux Enterprise Server 12	12.2+	1.8.0_211+	Oracle Fusion Middleware installer supports JDK 1.8.0_341+
Linux x86-64	SUSE Linux Enterprise Server 15	0+	1.8.0_211+	Oracle Fusion Middleware installer supports JDK 1.8.0_341+.
Microsoft Windows x64	2016, 2019, 2022	SP 0+	1.8.0_211+	-

* Oracle Analytics Server only supports Oracle JDK. Others, such as OpenJDK, are not supported.

Certification - Web Servers

Here're the web servers supported by the latest update of Oracle Analytics Server.

Supported Web Servers

Web Server	Release	Technology	OS	Additional information
Apache HTTP Server	2.4.x	Oracle WebLogic Server Plug-in 12.2.1.3.0.	64	-
Oracle HTTP Server	12.2.1.1+	mod_wl_ohs	64	-

B

Frequently Asked Questions

This appendix provides answers to common questions asked by administrators responsible for configuring and managing, backing up and restoring content, and indexing data in Oracle Analytics Server.

Topics:

- Top FAQs to Configure and Manage Oracle Analytics Server
- Top FAQs to Back Up and Restore User Content
- Top FAQs to Index Content and Data

Top FAQs to Configure and Manage Oracle Analytics Server

The top FAQs for configuring and managing Oracle Analytics Server are identified in this topic.

Can I see how many users are currently signed in?

Yes. Display the Home page , click **Console**, and then click **Sessions and Query Cache**. See Monitor Users Who Are Signed In.

Can I see the SQL generated by an analysis and analyze the log?

Yes. Display the Home page, click **Console**, and then click **Sessions and Query Cache**. See Analyze SQL Queries and Logs.

Can I change the default logo and dashboard style for the whole deployment?

Yes. When logged in as Administrator, navigate to the Classic Home page, click the user profile icon, click **Administration**, and then click **Manage Themes**. Create a new theme including dashboard properties such as logo, branding, page colors, and link colors, and click **Active**. This new style is applied for all new browser sessions.

Top FAQs to Back Up and Restore User Content

The top FAQs for backing up and restoring user content are identified in this topic.

What do I need to back up?

You should regularly back up the content that users create to a file called a *snapshot*. User content includes catalog content such as reports, dashboards, data visualization projects, and pixel perfect reports, datasets, data flows, data models, security roles, and so on.

If something goes wrong with your content or service, you can revert to the content you saved in a snapshot. Snapshots are also useful if you want to move or share content.

To back up user content, see Export Snapshots.

To restore user content, see Import Snapshots.

Top FAQs to Index Content and Data

The top FAQs for indexing semantic models and catalog content are identified in this topic.

What can I index?

Administrators can choose to index:

- Semantic models - Subject area, dimensions names and values, and measure names and values. You must be an administrator to modify the semantic model indexing preferences.
- Catalog content - Workbooks, analyses, dashboards, and reports. You must be an administrator to modify the catalog indexing preferences.
- File-based datasets - You can index a file-based dataset so that specified users can build visualizations with a dataset's data. Or you can certify a file-based dataset so that the specified users can search for its data from the home page. Any user can set a file-based dataset to index or certify the dataset.

See Configure Data Model Search Indexing.

What is a certified dataset?

Any user can upload a spreadsheet to create a dataset, and uploaded spreadsheets can be of varying quality. When a user certifies a shared dataset, it means that the user is confirming that the dataset contains good, reliable data that other users can search for from the home page. When you and users who've been granted access to datasets search from the home page, the data in a certified dataset is ranked high in the search results.

How often should I schedule a crawl?

The index updates automatically as users add or modify catalog content. By default, the catalog and semantic model crawl run once per day. In some cases you might want to change this default after importing a BAR file, if automatic indexing didn't run, or if your data updates occur less frequently (for example, monthly).

Can I index content in languages other than English?

Yes. You can index content in 28 languages.

- **Semantic models and catalog content** - You can generate indexes for multiple languages at the same time. Go to the **Search Index** page and **Ctl-click** to select one or more of the 28 available languages. For example, if your company's headquarters are in the United States and you have offices in Italy, you might select **English** and **italiano** to create indexes in both English and Italian. See Configure Data Model Search Indexing.
- **Datasets** - You can index a dataset for a single language at a time. Go to the **Inspect** dialog for the dataset and select one of the 28 available languages. See Index a Dataset.

> **Note:**
>
> If your data is in English and your index language is English, you can't search for the data in a different language such as French. For example, if your data includes English product names (such as *chair*, *desk*, *matches*), you can't search using French product names (such as *chaise*, *bureau*, *alumettes*).

Are there considerations when indexing subject areas with large tables?

You can index any size table, but big tables take longer to index. For large subject areas that have many tables or large tables, consider indexing only the columns your users need to search for.

Because index files are compact, it's rare to exceed the storage space that Oracle Analytics reserves for indexing.

How are search results ordered?

Search results are listed in this order:

1. Semantic model (semantic layer)
2. Certified datasets
3. Personal datasets
4. Catalog items (workbooks, analyses, dashboards, and reports)

Should I use Don't Index to secure my catalog items?

No. Oracle doesn't recommend setting the **Crawl Status** field to **Don't Index** as a way of hiding a catalog item from users. Users won't see the item in search results or on the home page, but are still able to access the item. Instead, use permissions to apply the proper security to the item.

How do I build an index most effectively?

For best results only index the subject areas, dimensions, catalog items, and certify datasets that users need to find. Indexing all items yields too many search results. Oracle recommends that you deselect all semantic model and catalog items and then select only the items that the user needs. You can then add items to the index as needed.

Why are there many select distinct queries on the database during indexing?

This is most likely because the semantic model's indexing option is set to **Index**. When you set this option to **Index**, the metadata and values are indexed, which means that during indexing the select distinct queries are run to fetch the data values for all of the columns in all of the subject areas that are configured for indexing.

If this system overhead isn't acceptable or if users don't need the additional functionality to visualize data values from the search bar on the Home page, then go to the **Console**, click **Search Index**, and set the indexing option to **Index Metadata Only**. Setting this option to **Index Metadata Only** indexes dimension and measure names, only, and doesn't run select distinct queries.

C Troubleshooting

This appendix describes common problems that you might encounter in Oracle Analytics Server and explains how to solve them.

Topics:

Troubleshoot General Issues

This topic describes common problems that you might encounter and explains how to solve them.

I can’t access certain options from the Home page

Check with your administrator to ensure that you have the correct permissions to access the options that you need.

I see a performance decrease when using Mozilla Firefox

If you use Mozilla Firefox and notice a decrease in performance, then ensure that the **Remember History** option is enabled. When Firefox is set to not remember the history of visited pages, then web content caching is also disabled, which greatly affects performance. See Firefox documentation for details on setting this option.

I'm having trouble uploading data from a spreadsheet (XLSX) exported from Microsoft Access

Open your spreadsheet in Microsoft Excel and resave it as an Excel Workbook (*.xlsx).

When you export spreadsheets from other tools the file format can vary slightly. Saving your data again from Microsoft Excel can fix this.

Users can't see the **Auto Insights** option in the Visualize canvas in the workbook editor.

In Console, navigate to **Advanced System Settings**, then **Performance and Compatibility**, and enable the **Enable Auto Insights on Datasets** option. Then ask dataset developers to select the **Enable Insights** option on the Dataset Inspect dialog for datasets where they require insights. Workbook users can then use the **Auto Insights** option in the Visualize canvas in the workbook editor.

My analysis or workbook times out

You attempt to run an analysis or workbook and find that it times out. You see a message like this:

```
[nQSError: 60009] The user request exceeded the maximum query governing execution
time.
```

This message is displayed when an Oracle Analytics query spends more than the allotted time communicating with the data source. For performance reasons, the limit for a single query to run is 11 minutes. If 11 minutes is too high for your organization, your administrator can select a lower query limit through system settings. See System Settings - Maximum Query Limit.

Try running the query again. To prevent this error, avoid long-running queries or split the query into multiple queries.

> **Note:**
>
> The query limit automatically extends to 60 minutes to accommodate occasional, longer-running queries. To avoid excessive loads on the database, Oracle Analytics restricts the number of queries that are allowed to automatically extend at any one time. Administrators can disable occasional query limit extensions for your organization through system settings. See System Settings - Query Limit Extension.

Search results on the Home page don't include the data I'm looking for

Datasets that users create from files must be indexed (and in some cases certified) to appear in search results on the Home page.

- A file-based dataset must be indexed before you can use it to build visualizations from the Home page.
- A file-based dataset must be indexed and certified before other users with permission to access the dataset can use it to build visualizations from the Home page.

See About Indexing a Dataset and Visualize Data from the Home Page.

Troubleshoot Configuration Issues

This topic describes common problems that you might encounter when configuring or managing Oracle Analytics Server and explains how to solve them.

I can’t access options in the Console

If you see an "unauthorized” message or don't see an option in the Console, you probably don’t have the BI Service Administrator application role. You must have the BI Service Administrator application role to access most Console options, for example **Roles and Permissions**, **Snapshots**, **Database Connections**, **Safe Domains**, **Session and Query Logs**, **Issue SQL**, and **Search Index**.

Ask an administrator to verify your permissions. See Assign Application Roles to Users.

I can’t upload my snapshot

You can only upload snapshots taken from Oracle BI Enterprise Edition (12c), and Oracle Analytics Server. Check where the .bar file you’re trying to upload was originally downloaded from.

I can't use Model Administration Tool in SSL mode

If the default security certificates don't work, import the server security certificates. For example, on the machine where you've installed Model Administration Tool, you might use the Key and Certificate Management Tool (keytool) to execute these commands:

```
C:\Oracle\Middleware\oracle_common\jdk\jre\bin\keytool.exe -importcert -alias
oacserver -file
C:\Oracle\Middleware\oracle_common\jdk\jre\lib\security\server.crt -keystore
C:\Oracle\Middleware\oracle_common\jdk\jre\lib\security\cacerts -storepass
thepassword
```

Troubleshoot Indexing

This topic describes common problems that you might encounter when indexing semantic models and catalog content and explains how to solve them.

A home page search returns no results

If you search on the home page and no results are returned, then check that the **Index User Folders** option is selected. If this option isn't selected then nothing in the catalog is indexed.

This option is located on the Search Index page's Catalog tab.

A home page search returns too many or duplicate items

If your search results aren't meaningful, then reduce the number of items to index. For example, if a dimension called Sales is included in 20 subject areas and all subject areas are indexed, then when you search for Sales your results will contain 20 items called Sales.

Go to the Search Index page's Data Model and Catalog tabs and reduce the number of items to index. Oracle suggests that you deselect everything and then select only the items that you need.

Expected items are missing from search results

If some items are missing from your search results, then check that the crawl job completed successfully. Sometimes a crawl was terminated or its progress totals are zero. In such cases, rerun the crawl.

1. Click **Console**.
2. Click **Search Index**.
3. Click **Monitor Crawls**.
4. Click the **Configure Crawls** link.
5. In the Data Model tab, deselect and then reselect the **Enable Data Model Crawl** checkbox.
6. Click **Save**.
7. Click the **Monitor Crawls** link and locate the scheduled job. The revised crawl will run in a few minutes.

D

Advanced Configuration Reference

This appendix describes advanced post-installation configuration and administration procedures that aren't specific to analyses, agents, dashboards, or the catalog.

Directions for configuring these components are in earlier chapters. Most administrators don't need to change the configuration settings that are described in this appendix.

Topics:

- Make Advanced Configuration Changes for Presentation Services
- Use the JavaHost Service for Presentation Services

Make Advanced Configuration Changes for Presentation Services

The Oracle Analytics Server process hosts most of the business logic of the web server and provides the framework and interface for the presentation of business intelligence data to web clients.

- Under Windows, the process is `sawserver.exe`
- Under Linux, the process is `sawserver`

The `instanceconfig.xml` file stores the configuration settings that affect Oracle Analytics Server. Many configuration settings are available in Fusion Middleware Control and that's the preferred method for making configuration changes. If a particular setting isn't available in Fusion Middleware Control, then you can change it using the `instanceconfig.xml` file. You can use the `instanceconfig.xml` file to customize various aspects of your deployment. Make changes directly in this file only to change default elements, such as the name of the Presentation Catalog, or override internal default settings, such as those related to caches.

Several entries are present in the `instanceconfig.xml` file by default, including the name of the data source used by Presentation Services to access Oracle Analytics Server.

> **Note:**
>
> If you've previously made configuration changes by modifying the Windows registry, then migrate those changes to the `instanceconfig.xml`. In the Windows registry, entries under the Common key remain valid.

The following procedure provides information about general configuration changes that you can make.

1. Open the `instanceconfig.xml` file for editing, located in:

   BI_DOMAIN`/config/fmwconfig/biconfig/OBIPS`

2. Locate the section in which you must add the elements that are described the table below.

3. Include the elements and their ancestor elements as appropriate, as shown in the following example:

```
<ServerInstance>
  <ClientStorage>
    <Enabled>true</Enabled>
    <LocalStorage>true</LocalStorage>
    <SessionStorage>true</SessionStorage>
  </ClientStorage>


  <FavoritesSyncUpIdleSeconds>300</FavoritesSyncUpIdleSeconds>
  <BIClientInstallerURL32Bit>http://myhost:7777/my32bitfile</
BIClientInstallerURL32Bit>
  <BIClientInstallerURL64Bit>http://myhost:7777/my64bitfile</
BIClientInstallerURL64Bit>
 <Security>
    <AllowRememberPassword>false</AllowRememberPassword>
    <CookieDomain>value</CookieDomain>
    <CookiePath>/analytics</CookiePath>
    <InIFrameRenderingMode>prohibit</InIFrameRenderingMode>
   <Cursors>
      <NewCursorWaitSeconds>3</NewCursorWaitSeconds>
    </Cursors>
  <LogonExpireMinutes>180</LogonExpireMinutes>
 </Security>
 <ODBC>
   <UnaccessedRunningTimeoutMinutes>5</UnaccessedRunningTimeoutMinutes>
  </ODBC>
 <UI>
    <MaxSearchResultItemsToReturn>300</MaxSearchResultItemsToReturn>
  <UserPickerDialogMaxAccounts>300</UserPickerDialogMaxAccounts>
 </UI>
 <Download>
   <Export xsi:type="all">
    <AsyncExport>false</AsyncExport>
   </Export>
 </Download>
</ServerInstance>
```

4. Save your changes and close the file.
5. Restart the system.

Advanced Configuration Settings for Presentation Services

You can use configuration settings to make advanced Presentation Services changes.

The table describes the advanced configuration settings for Presentation Services.

Element	Description	Default Value
<**ClientStorage**>	Specifies the parent element for maintaining client state across sessions and within a session. See also the Enabled, LocalStorage, and SessionStorage elements.	No default value

Element	Description	Default Value

`<ClientStorage>`
`<`**`Enabled`**`>`
`</ClientStorage>`
	Specifies whether to maintain client state across sessions and within a session. State isn't maintained across browser sessions. The following items maintain state:
	• Dashboard menu — Stores whether the menu is collapsed or expanded.
	• Folders — Stores whether folders are expanded for the Catalog page and the Open dialog.
	• Favorites menu — Stores the name of the last expanded item.
	• Catalog Page Toolbar — Stores various details about the state of the toolbar, including the settings of the **Type**, **Sort**, and **Show More Details** options.
	See also the LocalStorage and SessionStorage elements.
		true

`<ClientStorage>`
`<`**`LocalStorage`**`>`
`</ClientStorage>`
	Specifies whether the local storage of the browser is used to maintain state. If the browser doesn't support local storage, then no state is maintained.	true

`<ClientStorage>`
`<`**`SessionStorage`**`>`
`</ClientStorage>`
	Specifies whether the local storage of the browser is used to maintain state for sessions. If the browser doesn't support local storage, then no state is maintained.	true

`<`**`FavoritesSyncUpIdleSeconds`**`>`	Specifies the number of seconds of idle time before synchronizing data from a mobile application and favorites from the Presentation Catalog.	300
`<`**`BIClientInstallerURL64Bit`**`>`	Specifies that you want to override the default download location for the Installer when the user selects to download the Installer from the Oracle Analytics Server Home page. The file for the 64-bit Installer is named biee_client_install64.exe.	No default value

`<Security>`
`<`**`AllowRememberPassword`**`>`
`</Security>`
	Specifies whether to allow the browser to save the password, using browser-specific password management software. If set to true, prompts the user to specify whether to save the password for future sign-ins.	false

`<Security>`
`<`**`CookieDomain`**`>`
`</Security>`
	Specifies the domain information for a cookie that is sent to the browser.	No default value

Element	Description	Default Value

`<Security>`
`<CookiePath>`
`</Security>`
	Specifies the domain path to which cookies apply.	/analytics

`<Security>`
`<InIFrameRenderingMode>`
`</Security>`
	See Protect Pages in Oracle Analytics Server from Attack for information.	sameDomainOnly

`<Security>`
`<cursors>`
`<NewCursorWaitSeconds>`
`</cursors>`
`</Security>`
	Specifies how long the server waits for results after the initial request before returning the search page to the browser. It may be useful to set this value higher (such as 3 seconds) to avoid page refreshes if the majority of queries aren't returning in 1 second.	No default value

`<ODBC>`
`<UnaccessedRunningTimeoutMinutes>`
`</ODBC>`
	Specifies the time to elapse, in minutes, before an unattended analysis is canceled. An unattended analysis is one that has not been accessed in the number of minutes specified by this setting. The minimum value is 2.
	This element addresses the case where a user is editing an analysis and browses elsewhere, abandoning the analysis, at least temporarily. Don't set the value too small, however, as the user might return to the analysis.
	Use this element only for Presentation Services queries that run against the BI Server. The element doesn't apply to any other type of connection.
		5

`<UI>`
`<MaxSearchResultItemsToReturn>`
`</UI>`
	Specifies the maximum number of items to display within a directory listing of the catalog within Presentation Services. The minimum value is 0. Use care when setting this element to a high value as the performance of the user interface might decrease.	300

`<UI>`
`<UserPickerDialogMaxAccounts>`
`</UI>`
	Specifies the maximum number of users and roles to include in the selection list based on the specified search criteria. For example, if this element is set to 300 and more than 300 users and roles match the search criteria, then only the first 300 of those are included in the list. Use care when setting this element to a high value, because the performance of the user interface might decrease.	300

Element	Description	Default Value

`<Download>`
` <Export xsi:type="all">`
`  <AsyncExport>false</`
`AsyncExport>`
` </Export>`
`</Download>`
	Specifies which threading model is used when exporting a classic analysis. The default value is `true` which limits to four concurrent threads.	true

Protect Pages in Oracle Analytics Server from Attack

As the administrator, you must be aware of a security concern that is known as *clickjacking*. Clickjacking refers to the ability of attackers to subvert clicks and send the user's clicks to web pages that permit them to be framed with or without JavaScript.

For example, suppose an attacker develops a website that uses an inline frame for an Oracle Analytics Server Console application. When you visit this site, you're unknowingly clicking buttons on the inline-framed Console application. This vulnerability is very serious, because the attacker isn't stopped by the same origin policy principles that apply to other Oracle Analytics Server applications. You can find many examples of clickjacking documented on the Worldwide Web.

The term that describes preventing attackers from framing an application in an inline frame is *frame busting*. To affect frame busting, you use the InIFrameRenderingMode element in the instanceconfig.xml file. You can set the element to the following three values:

- prohibit = Never permit content from Presentation Services to be rendered in an inline frame.
- sameDomainOnly = (Default) Enable rendering of pages in an inline frame if the enclosing page was generated by the server in the same domain. By default, pages have the same domain if they were generated by the same server. See the Worldwide Web for information on the "same origin policy."
- allow = Always allow content from Presentation Services to be rendered in an inline frame.

Use the JavaHost Service for Presentation Services

Java library functionality can be achieved using the JavaHost service.

The JavaHost service gives Presentation Services the ability to use functionality that's provided in Java libraries to support the following components:

- Graph generation
- SVG renderer (Apache Batik)
- Actions that require Java actions, for example, calling web services
- Printing to PDF and exporting to Microsoft Excel and PowerPoint
- Advanced reporting
- URL Connect iIssues an HTTP request to another component
- Integration Service Call used by the Oracle Analytics Server to run Java code

Note:

The JavaHost service uses the core libraries of Oracle Analytics Server Publisher to export the contents of analyses into various formats such as PDF, Microsoft Excel and PowerPoint, and images. Oracle Analytics Server Publisher libraries are embedded within the JavaHost service and don't depend on Oracle Analytics Server Publisher running or being deployed in a J2EE container.
In the configuration file for the JavaHost Service, elements related to the Oracle Analytics Server Publisher libraries are located within the XMLP element.

To configure the JavaHost service, you can manually edit the configuration elements for the service in its configuration file (*config.xml*), located in the `*BI_DOMAIN*/config/fmwconfig/biconfig/OBIJH` directory. See the next table for a description of the elements. The elements are identified by their relative path starting from the JavaHost element.

The common subelements, such as InputStreamLimitInKB, don't apply to the MessageProcessor, Listener, or SSL loaders.

Element	Description
Loaders	Contains the ListOfEnabledLoaders and Loader elements. These elements specify the components for the JavaHost service. Avoid editing the elements in the Loaders section.
Loaders/ListOfEnabledLoaders	Specifies the list of components such as Oracle BI Scheduler and Oracle Analytics Server Publisher to enable. If this element is missing from the file, then all Loaders are enabled. If the element has an empty value, then all loaders are disabled. Each component has a corresponding Loader element. The name of the component listed here must match the name that's specified in the corresponding Loader/Name element.
Loaders/Loader	Contains the following elements, which specify configuration information for a specific component: • Name • Class • ConfigNodePath • ClassPath
Loaders/Loader/Name	Specifies the unique name of the component. Use this name in the ListOfEnabledLoaders element.
Loaders/Loader/Class	Specifies the main class for the component.
Loaders/Loader/ConfigNodePath	Specifies the XPath (starting from the JavaHost element) to the configuration information for the Loader.
Loaders/Loader/ClassPath	Specifies the paths for the JAR files of libraries that the JavaHost service can use.

Element	Description
InputStreamLimitInKB	A child element common to each loader that specifies, in kilobytes, the maximum input size for requests that are sent to JavaHost. A value of zero deactivates this limit. If the maximum size is exceeded, then an error message is displayed. Default: 8192 Set the InputStreamLimitInKB value to zero (the value is unlimited) only for testing purposes. Configuring the value too high allocates or consumes more resources than necessary for an individual request to the JavaHost, might cause the JavaHost to become unstable or crash, and must fit the context of all JavaHost requests, for example, graphs or export operations. Set the element to a reasonable value that works with large datasets. The default value is 8192 (8 MB), but you might need to increase it incrementally to 16384 (16 MB), 32768 (32 MB), and so on.
RequestResponseLogDirectory	A child element common to each loader that specifies the name of the directory for the response files of requests. Default: A default temp directory
LogLargeRequests	A child element common to each loader that specifies whether to create a response file when processing large requests. Default: true
ReadRequestBeforeProcessing	A child element common to each loader that specifies whether to wait to process the request until a file is completely read. If your organization uses the export feature, it's recommended that you set this child element to false. When set to false, data is streamed to JavaHost gradually rather than saved to a file first and then processed, thereby improving export performance. Default: true
LargeRequestThresholdInKB	A child element common to each loader that specifies, in kilobytes, the maximum size before using disk space for requests. For requests larger than this size, use disk space instead of memory to cache the requested data. The larger this value is the more memory that the JavaHost service might potentially use and the faster the request processing can occur. This setting also establishes the threshold for the LogLargeRequests element. Default: 200
MessageProcessor	Contains the SocketTimeout element.
MessageProcessor/SocketTimeout	Specifies the idle timeout (in milliseconds) for the socket, after which the socket is returned to the idle sockets pool. JavaHost uses a socket polling mechanism to wait for new data on the whole set of idle sockets in a single thread. Initial messages in the idle pool are handled through Java NIO channels. Default: 5000 (5 seconds)
Listener	Contains the following elements: • PermittedClientList • Port • Address • Secure
Listener/PermittedClientList	Specifies a list of IP addresses and host names from which JavaHost accepts incoming connections. Separate each client's IP address or host name by a comma. To accept all client connections, set this element to an asterisk (*). Default: *

Element	Description
Listener/Port	Identifies the JavaHost TCP/IP listening port. Default: 9810
Listener/Address	Specifies the network interface that JavaHost is to bind to. If this element has no value, then JavaHost binds to all available network interfaces.
Listener/Secure	Specifies whether to enable SSL encryption for the JavaHost service: • Yes: Enables SSL encryption • No: Disables SSL encryption Default: No
Batik	Contains only the common subelements such as InputStreamLimitInKB, as they relate to converting SVG images to rasterized image formats.
Scheduler	Contains the following elements: • Enabled • DefaultUserJarFilePath • DefaultTempFilePath • DefaultPurgingPeriod
Scheduler/Enabled	Specifies whether to enable the interaction of the JavaHost service with Oracle BI Scheduler to run Java jobs: • true: Enables interaction with Oracle BI Scheduler • false: Disables interaction with Oracle BI Scheduler Default: false
Scheduler/DefaultUserJarFilePath	Specifies the default directory for storing JAR files for the Java extension utility. The Jar file contains the implementation of the Java class to be run. When Oracle BI Scheduler is enabled, this element is required and accepts a single path.
Scheduler/DefaultTempFilePath	Specifies the default directory for storing temporary files for Oracle BI Scheduler requests. Default: the system temp directory
Scheduler/DefaultPurgingPeriod	Specifies the default period (in seconds) for Oracle BI Scheduler requests to remove failed jobs. Default: 300
XMLP	Contains only the common subelements such as `InputStreamLimitInKB` and `ReadRequestBeforeProcessing`, as they relate to Oracle Analytics Server Publisher.
URLConnect	Contains elements that relate to SSL. Avoid modifying these elements.
DVT	Contains only the common `InputStreamLimitInKB` child element by default, as they relate to graph generation. You can add other common child elements as necessary.

E

Map User Interface Labels with Configuration File Elements

This appendix maps Fusion Middleware Control User Interface (UI) labels for Oracle Analytics Server with the corresponding element names used in configuration files.
The information in the following tables is included here for completeness. You don't need this information for most operations.

- For information about elements for the Oracle Analytics Server that aren't included in the following tables, see Configuration File Settings.
- For information about the location of configuration files, see Configuration Files.

Fusion Middleware Control UI Label	Configuration Element	Configuration File (NQSConfig.INI for the BI Server and instanceconfig.xml for Presentation Services)	Related Information
Cache enabled	ENABLE	NQSConfig.INI	Use Fusion Middleware Control to Enable and Disable Query Caching
Maximum cache entry size	MAX_CACHE_ENTRY_SIZE	NQSConfig.INI	Use Fusion Middleware Control to Set Query Cache Parameters
Maximum cache entries	MAX_CACHE_ENTRIES	NQSConfig.INI	Use Fusion Middleware Control to Set Query Cache Parameters
Global cache path	GLOBAL_CACHE_STORAGE_PATH	NQSConfig.INI	Use Fusion Middleware Control to Set Global Cache Parameters
Global cache size	GLOBAL_CACHE_STORAGE_PATH	NQSConfig.INI	Use Fusion Middleware Control to Set Global Cache Parameters
Disallow RPD Updates	READ_ONLY_MODE	NQSConfig.INI	Use Fusion Middleware Control to Disallow Semantic Model Updates
User Session Expiry	ClientSessionExpireMinutes	instanceconfig.xml	Use Fusion Middleware Control to Set the User Session Log-Off Period
Maximum Number of Rows Processed when Rendering a Table View	ResultRowLimit	instanceconfig.xml	Use Fusion Middleware Control to Set the Maximum Number of Rows Processed to Render a Table View
Maximum Number of Rows to Download	DefaultRowsDisplayedInDownload	instanceconfig.xml	Use Fusion Middleware Control to Set Configuration Options for Data in Tables and Pivot Tables
Maximum Number of Rows Per Page to Include	DefaultRowsDisplayedInDelivery	instanceconfig.xml	Use Fusion Middleware Control to Set Configuration Options for Data in Tables and Pivot Tables

For information about diagnostic log configuration files (for example, logconfig.xml), see:

- What Are Diagnostic Log Configuration Files and Where Are They Located?

- What Are Log File Message Categories and Levels?
- What Is Log File Rotation?

Fusion Middleware Control UI Label Name	Configuration Element	Configuration File	Related Information
Show page tabs	ShowPageTabsAlways	instanceconfig.xml	Use Fusion Middleware Control to Change Presentation Setting Defaults
Show section headings	ShowSectionHeadingsDefault	instanceconfig.xml	Use Fusion Middleware Control to Change Presentation Setting Defaults
Allow dashboard sections to be collapsible	CollapsibleSectionsDefault	instanceconfig.xml	Use Fusion Middleware Control to Change Presentation Setting Defaults
Pivot Tables show auto-preview	DisableAutoPreview	instanceconfig.xml	Use Fusion Middleware Control to Change Presentation Setting Defaults

Fusion Middleware Control UI Label	Configuration Element	Configuration File (for BI Scheduler)	Related Information
SMTP Server	SMTP_Server	instanceconfig.xml	Use Fusion Middleware Control to Configure Scheduler Email Settings that Affect Agents
Port	SMTP_Port	instanceconfig.xml	Use Fusion Middleware Control to Configure Scheduler Email Settings that Affect Agents
Display name of sender	From	instanceconfig.xml	Use Fusion Middleware Control to Configure Scheduler Email Settings that Affect Agents
Email address of sender	Sender	instanceconfig.xml	Use Fusion Middleware Control to Configure Scheduler Email Settings that Affect Agents
Username	mail.server	Credential found in oracle.bi.enterprise credential map	Use Fusion Middleware Control to Configure Scheduler Email Settings that Affect Agents *Managing Security for Oracle Analytics Server*
Password	See Username, in the preceding row.	Not Available	Use Fusion Middleware Control to Configure Scheduler Email Settings that Affect Agents
Confirm password	See Username, in the preceding row.	Not Available	Use Fusion Middleware Control to Configure Scheduler Email Settings that Affect Agents
Number of retries upon failure	Try	instanceconfig.xml	Use Fusion Middleware Control to Configure Scheduler Email Settings that Affect Agents
Maximum recipients	MaxRecipients	instanceconfig.xml	Use Fusion Middleware Control to Configure Scheduler Email Settings that Affect Agents
Addressing method	UseBcc	instanceconfig.xml	Use Fusion Middleware Control to Configure Scheduler Email Settings that Affect Agents

Fusion Middleware Control UI Label	Configuration Element	Configuration File (for BI Scheduler)	Related Information
Use SSL to connect to mail server	UseSSL	instanceconfig.xml	Use Fusion Middleware Control to Configure Scheduler Email Settings that Affect Agents
Specify CA certificate source	This controls whether to fill in either SmtpCACertificateDir or SmtpCACertificateFile	instanceconfig.xml	Use Fusion Middleware Control to Configure Scheduler Email Settings that Affect Agents
CA certificate directory	SmtpCACertificateDir	instanceconfig.xml	Use Fusion Middleware Control to Configure Scheduler Email Settings that Affect Agents
CA certificate file	SmtpCACertificateFile	instanceconfig.xml	Use Fusion Middleware Control to Configure Scheduler Email Settings that Affect Agents
SSL certificate verification depth	SmtpCertificateVerificationDepth	instanceconfig.xml	Use Fusion Middleware Control to Configure Scheduler Email Settings that Affect Agents
SSL cipher list	SmtpCipherList	instanceconfig.xml	Use Fusion Middleware Control to Configure Scheduler Email Settings that Affect Agents

Fusion Middleware Control UI Label	Configuration Element	Configuration File	Related Information
Maximum File Size	maxFileSizeKb	instanceconfig.xml (for Presentation Services and BI Scheduler) logging-config.xml (for JavaHost) ccslogging.xml (for Cluster Controller)	Use Fusion Middleware Control to Configure Log File Rotation Policy and Specify Log Levels
Maximum Log Age	MaximumLogAgeDay	instanceconfig.xml (for Presentation Services and Scheduler) logging-config.xml (for JavaHost) ccslogging.xml (for Cluster Controller)	Use Fusion Middleware Control to Configure Log File Rotation Policy and Specify Log Levels
Maximum File Size **Note:** Field in Query Logs region.	MaximumFileSizeKb	logconfig.xml (for BI Server)	Use Fusion Middleware Control to Configure Log File Rotation Policy and Specify Log Levels
Maximum Log Age **Note:** Field in Query Logs region.	MaximumLogAgeDay	logconfig.xml (for BI Server)	Use Fusion Middleware Control to Configure Log File Rotation Policy and Specify Log Levels
Incident Error	IncidentError	instanceconfig.xml (for Presentation Services and Scheduler) logging-config.xml (for JavaHost) ccslogging.xml (for Cluster Controller)	Use Fusion Middleware Control to Configure Log File Rotation Policy and Specify Log Levels

Fusion Middleware Control UI Label	Configuration Element	Configuration File	Related Information
Error	Error	instanceconfig.xml (for Presentation Services and Scheduler) logging-config.xml (for JavaHost) ccslogging.xml (for Cluster Controller)	Use Fusion Middleware Control to Configure Log File Rotation Policy and Specify Log Levels
Warning	Warning	instanceconfig.xml (for Presentation Services and Scheduler) logging-config.xml (for JavaHost) ccslogging.xml (for Cluster Controller)	Use Fusion Middleware Control to Configure Log File Rotation Policy and Specify Log Levels
Notification	Notification	instanceconfig.xml (for Presentation Services and Scheduler) logging-config.xml (for JavaHost) ccslogging.xml (for Cluster Controller)	Use Fusion Middleware Control to Configure Log File Rotation Policy and Specify Log Levels
Trace	Trace	instanceconfig.xml (for Presentation Services and Scheduler) logging-config.xml (for JavaHost) ccslogging.xml (for Cluster Controller)	Use Fusion Middleware Control to Configure Log File Rotation Policy and Specify Log Levels

Fusion Middleware Control UI Label	Configuration Element	Configuration File	Related Information
Enable SSO	EnabledSchemas (indirectly associated)	instanceconfig.xml	*Managing Security for Oracle Analytics Server*
SSO Provider	EnabledSchemas (indirectly associated)	instanceconfig.xml	*Managing Security for Oracle Analytics Server*

F

Configuration File Settings

This appendix lists key configuration files for Oracle Analytics and provides details about parameters in the NQSConfig.INI file.

Topics:

- Configuration Files
- NQSConfig.INI File Configuration Settings

Configuration Files

Configuration files control the behavior of the system.

The following table lists key configuration files for Oracle Analytics and their locations. See What Are Diagnostic Log Configuration Files and Where Are They Located?.

BI Component	Configuration File	File Location
Oracle BI Server	NQSConfig.INI logconfig.xml	For example: *BI_DOMAIN*/config/fmwconfig/biconfig/OBIS
Oracle BI Presentation Services	instanceconfig.xml credentialstore.xml marketingwebexpressions.xml userpref_currencies.xml	For example: *BI_DOMAIN*/config/fmwconfig/biconfig/OBIPS Don't add elements to the instanceconfig.xml file unless you're overriding the stated default values. Override only those settings that are necessary for configuring the system to meet the needs of your organization.
Cluster Controller	bi-cluster-config.xml ccslogconfig.xml	For example: *BI_DOMAIN*/config/fmwconfig/biconfig/OBICCS
Oracle BI Scheduler	schedulerconfig.xml	For example: *BI_DOMAIN*/config/fmwconfig/biconfig/OBISCH
JavaHost	config.xml logging-config.xml	For example: *BI_DOMAIN*/config/fmwconfig/biconfig/OBIJH
Presentation Services Plug-in	bridgeconfig.properties	For example: *BI_DOMAIN*/config/fmwconfig/biconfig

BI Component	Configuration File	File Location
Oracle BI Environment	bi-environment.xml odbc.ini obis.properties	For example: *BI_DOMAIN*/config/fmwconfig/bienv/core bi-environment.xml contains environment configuration settings (separate from process command-specific parameters), for example: • Data directory (SDD) - path to a singleton directory (for high availability purposes). By default, SDD = $DOMAIN_HOME/bidata. It's required that the SDD is mounted to the same point for all machines in a scaled-out system. Mixed mode (where some components use the SDD, and some don't), isn't allowed. See Key Directories. • Runtime compatibility flag. • Hardware acceleration flag. • Port ranges. • SSL External/Internal Certificate Authorities (CA) and certificate paths. Although the internal certificates DNs are not verified (and thus don't display hostname), it may be required to change these during cloning operations. odbc.ini contains the single source of truth for ODBC connection endpoints: • Where the endpoint is internal (for example, BIEE cluster controller or Essbase), then the drivers (or clients of the drivers) must use the endpoint API to recover the appropriate endpoint. • Where the endpoint is external to the system, then this file might change if the domain is copied or moved. • OPSS and BIPLATFORM DSNs are provided in odbc.ini. The credentials for BIPLATFORM DSN is in the OPSS Credential Store. obis.properties contains BI Server-specific environment and configuration settings that are substituted with specified values, and is located in: *BI_DOMAIN*/config/fmwconfig/bienv/OBIS

NQSConfig.INI File Configuration Settings

This section lists the NQSConfig.INI file parameters for Oracle Analytics Server, gives a brief description, and provides any required syntax for each parameter. The Oracle Analytics Server software uses an initialization file called NQSConfig.INI to set parameters upon startup. This initialization file includes parameters to customize behavior based on the requirements of each individual installation. The parameters are generally listed in the order in which they appear in the configuration file.

> **Note:**
>
> The examples in this section assume that you're editing a Windows version of NQSConfig.INI. If you're editing this file on a LINUX system, then ensure that you use LINUX-appropriate file system paths and conventions.

This topic includes the following sections:

About Parameters in the NQSConfig.INI File

Oracle Analytics Server has one NQSConfig.INI file.

Note the following rules and guidelines for NQSConfig.INI file entries:

- Oracle Analytics Server reads the NQSConfig.INI file each time it starts.
- Each parameter entry in NQSConfig.INI must be within the section to which the parameter belongs (Repository, Cache, General, and so on).
- Each entry must be terminated with a semicolon (;).
- You can add comments anywhere in the NQSConfig.INI file. Comments must begin with either of the following:

  ```
  #
  ```

  ```
  //
  ```

  Any text following these comment characters up to the end of the line is ignored when Oracle Analytics Server reads the file.
- For parameters that require a setting in bytes, you can specify the value in either bytes, KB, MB, or GB. If you omit the size qualifier, then the value is interpreted as the number of

bytes. If you include the size qualifier, then ensure that you include a space before the qualifier. The following are examples of valid values:

```
MAX_QUERY_PLAN_CACHE_ENTRY_SIZE = 1 MB;
MAX_QUERY_PLAN_CACHE_ENTRY_SIZE = 1024 KB;
MAX_QUERY_PLAN_CACHE_ENTRY_SIZE = 1048576;
```

- Any syntax errors prevent Oracle Analytics Server from starting. The errors are logged to the obis1-diagnostic.log file, which is located in `*BI_DOMAIN*/servers/obis1/logs`

  There might also be a summary message in the system log that relates to the error.

  If you get an error, then correct the problem and start Oracle Analytics Server again. Repeat this process until the server starts with no errors.

How to Update Parameters in NQSConfig.INI

The following procedure explains how to update parameters in NQSConfig.INI.

To update parameters in NQSConfig.INI:

1. Open the NQSConfig.INI file in a text editor. You can find this file at:

   `BI_DOMAIN/config/fmwconfig/biconfig/OBIS`

   Make a backup copy of the file before editing.
2. Locate and update the parameter you want to change.
3. Save and close the file.
4. Restart the Oracle BI Server.

   See About Managing Processes.

Repository Section Parameters

The Repository section contains one entry for every semantic model that's loaded when the server starts.

> **Note:**
>
> Hosting multiple semantic models on a single Oracle BI Server isn't recommended for production systems.

Syntax: *logical_name* = *repository_name.rpd;*

Optional syntax: *logical_name* = *repository_name.rpd*, `DEFAULT;`

In this syntax:

- *logical_name*: A logical name for the semantic model. Client tools use this name to configure the ODBC data sources that connect to the semantic model. To use a reserved keyword for the name, such as OCI7 or OCI8, enclose it in single quotation marks.
- *repository_name.rpd*: The file name of the semantic model. The file name must have the .rpd file extension, and the file must reside in the semantic model subdirectory.

  The demonstration semantic model SampleAppLite.rpd is installed when selected at install time with Oracle Analytics Server.

When `DEFAULT` is specified for a semantic model, connections that don't specify a logical semantic model name in the DSN connect to the default semantic model.

Example: `Star = SampleAppLite.rpd, DEFAULT;`

Multitenancy Section Parameters

The parameters in the Multitenancy Section provide support for a configuration that includes multiple tenants. The parameters in this section are reserved for future use.

MT_ROOT_DIRECTORY

This parameter is reserved for future use.

Example: `MT_ROOT_DIRECTORY= "";`

MT_ENTRIES

This parameter is reserved for future use.

Example: `MT_ENTRIES= ;`

Query Result Cache Section Parameters

The parameters in the Query Result Cache Section provide configuration information for Oracle BI Server caching.

The query cache is enabled by default. After deciding on a strategy for flushing outdated entries, configure the cache storage parameters in Fusion Middleware Control and in the NQSConfig.INI file.

Note that query caching is primarily a runtime performance improvement capability. As the system is used over a period of time, performance tends to improve due to cache hits on previously run queries. The most effective and pervasive way to optimize query performance is to use the Aggregate Persistence wizard and aggregate navigation.

This section describes only the parameters that control query caching. For information about how to use caching in Oracle Analytics Server, including information about how to use agents to seed the Oracle BI Server cache, see Manage Performance Tuning and Query Caching .

ENABLE

> **Note:**
>
> The `ENABLE` parameter can be managed by Fusion Middleware Control or by manually editing NQSConfig.INI.
>
> The **Cache enabled** option on the Performance tab of the Configuration page in Fusion Middleware Control corresponds to the `ENABLE` parameter. See Use Fusion Middleware Control to Enable and Disable Query Caching.

Specifies whether the cache system is enabled. When set to `NO`, caching is disabled. When set to `YES`, caching is enabled. The query cache is enabled by default.

Example: `ENABLE = YES;`

DATA_STORAGE_PATHS

Specifies one or more paths for where the cached query results data is stored and are accessed when a cache hit occurs and the maximum capacity in bytes, kilobytes, megabytes, or gigabytes.

The maximum capacity for each path is 4 GB. For optimal performance, specify the paths on high-performance storage systems.

Each path listed must be an existing, writable path name, with double quotation marks (") surrounding the path name. Specify mapped directories only. UNC path names and network mapped drives are enabled only if the service runs under a qualified user account.

You can specify either fully qualified paths, or relative paths. When you specify a path that doesn't start with "/" (on Linux) or "<drive>:" (on Windows), the Oracle BI Server assumes that the path is relative to the local writable directory. For example, if you specify the path "cache," then at runtime, the Oracle BI Server uses the following:

```
BI_DOMAIN/servers/obisn/cache
```

> **Note:**
>
> Multiple Oracle BI Servers across a cluster don't share cached data. Therefore, the `DATA_STORAGE_PATHS` entry must be unique for each clustered server. To ensure this unique entry, enter a relative path so that the cache is stored in the local writable directory for each Oracle BI Server, or enter different fully qualified paths for each server.

Specify multiple directories with a comma-delimited list. When you specify multiple directories, ensure that they reside on different physical drives. If you have multiple cache directory paths that all resolve to the same physical disk, then both available and used space might be double-counted. When you specify multiple directories, ensure that the directory names aren't subsets of each other. For example, use names such as *cache1* and *cache2* rather than *cache* and *cache2*.

Syntax: `DATA_STORAGE_PATHS = "path_1" sz[, "path_2" sz{, "path_n" sz}];`

Example: `DATA_STORAGE_PATHS = "cache" 256 MB;`

> **Note:**
>
> Specifying multiple directories for each drive does not improve performance, because file input and output (I/O) occurs through the same I/O controller. In general, specify only one directory for each disk drive. Specifying multiple directories on different drives might improve the overall I/O throughput of the Oracle BI Server internally by distributing I/O across multiple devices.

The disk space requirement for the cached data depends on the number of queries that produce cached entries, and the size of the result sets for those queries. The query result set size is calculated as row size (or the sum of the maximum lengths of all columns in the result set) times the result set cardinality (that is, the number of rows in the result set). The expected maximum is the guideline for the space needed.

This calculation gives the high-end estimate, not the average size of all records in the cached result set. Therefore, if the size of a result set is dominated by variable length character strings, and if the length of those strings is distributed normally, you would expect the average record size to be about half the maximum record size.

> **Note:**
>
> It is a best practice to use values less than 4 GB on your 64-bit system. Create multiple paths if you've values in excess of 4 GB.

MAX_ROWS_PER_CACHE_ENTRY

Specifies the maximum number of rows in a query result set to qualify for storage in the query cache.

Limiting the number of rows is a useful way to avoid consuming the cache space with runaway queries that return large numbers of rows. If the number of rows a query returns is greater than the value specified in the `MAX_ROWS_PER_CACHE_ENTRY` parameter, then the query isn't cached.

When set to 0, there's no limit to the number of rows per cache entry.

Example: `MAX_ROWS_PER_CACHE_ENTRY = 100000;`

MAX_CACHE_ENTRY_SIZE

The `MAX_CACHE_ENTRY_SIZE` parameter can be managed by either Fusion Middleware Control or by editing NQSConfig.INI.

> **Note:**
>
> The **Maximum cache entry size** option on the Performance tab of the Configuration page in Fusion Middleware Control corresponds to the `MAX_CACHE_ENTRY_SIZE` parameter. See Use Fusion Middleware Control to Set Query Cache Parameters.

Specifies the maximum size for a cache entry. Potential entries that exceed this size are not cached. The default size is 20 MB.

Specify GB for gigabytes, KB for kilobytes, MB for megabytes, and no units for bytes.

Example: `MAX_CACHE_ENTRY_SIZE = 20 MB;`

MAX_CACHE_ENTRIES

The **Maximum cache entries** option on the Performance tab of the Configuration page in Fusion Middleware Control corresponds to the `MAX_CACHE_ENTRIES` parameter.

> **Note:**
>
> The `MAX_CACHE_ENTRIES` parameter can be managed by either Fusion Middleware Control or by editing NQSConfig.INI.
>
> See Use Fusion Middleware Control to Set Query Cache Parameters.

Specifies the maximum number of cache entries allowed in the query cache to help manage cache storage. The actual limit of cache entries might vary slightly depending on the number of concurrent queries. The default value is 1000.

Example: `MAX_CACHE_ENTRIES = 1000;`

POPULATE_AGGREGATE_ROLLUP_HITS

Specifies whether to aggregate data from an earlier cached query result set and create a new entry in the query cache for rollup cache hits. The default value is `NO`.

Typically, if a query gets a cache hit from a previously run query, then the new query isn't added to the cache. A user might have a cached result set that contains information at a particular level of detail (for example, sales revenue by ZIP code). A second query might ask for this same information, but at a higher level of detail (for example, sales revenue by state). The `POPULATE_AGGREGATE_ROLLUP_HITS` parameter overrides this default when the cache hit occurs by rolling up an aggregate from a previously run query (in this example, by aggregating data from the first result set stored in the cache). That is, Oracle Analytics Server sales revenue for all ZIP codes in a particular state can be added to obtain the sales revenue by state. This is referred to as a rollup cache hit.

Normally, a new cache entry isn't created for queries that result in cache hits. You can override this behavior specifically for cache rollup hits by setting `POPULATE_AGGREGATE_ROLLUP_HITS` to `YES`. Nonrollup cache hits aren't affected by this parameter. If a query result is satisfied by the cache—that is, the query gets a cache hit—then this query isn't added to the cache. When this parameter is set to `YES`, then when a query gets an aggregate rollup hit, then the result is put into the cache. Setting this parameter to `YES` might result in better performance, but results in more entries being added to the cache.

Example: `POPULATE_AGGREGATE_ROLLUP_HITS = NO;`

USE_ADVANCED_HIT_DETECTION

When caching is enabled, each query is evaluated to determine whether it qualifies for a cache hit.

A cache hit means that the server was able to use cache to answer the query and didn't go to the database at all. The Oracle BI Server can use query cache to answer queries at the same or later level of aggregation.

The parameter `USE_ADVANCED_HIT_DETECTION` enables an expanded search of the cache for hits. The expanded search has a performance impact, which isn't easily quantified because of variable customer requirements. Customers that rely heavily on query caching and are experiencing misses might want to test the trade-off between better query matching and overall performance for high user loads. Refer to the parameter MAX_SUBEXPR_SEARCH_DEPTH for related information.

Example: `USE_ADVANCED_HIT_DETECTION = NO;`

Reasons Why a Query Is Not Added to the Cache

Customers who rely on query result caching in the Oracle BI Server to meet their performance KPIs can use caching parameters to help determine why a cache hit didn't occur.

Logging facilities can help diagnose common reasons for getting a cache miss, where the logical SQL query that was supposed to seed the cache didn't get inserted into the cache. The following describes some situations when this might occur.

- Noncacheable SQL element. If a SQL request contains `CURRENT_TIMESTAMP`, `CURRENT_TIME`, `RAND`, `POPULATE`, or a parameter marker, then it's not added to the cache.
- Noncacheable table. Physical tables in the semantic model can be marked "noncacheable." If a query references any noncacheable table, then the query results aren't added to the cache.
- Cache hit. In general, if the query gets a cache hit on a previously cached query, then the results of the current query aren't added to the cache.

  The exception is query hits that are aggregate rollup hits. These are added to the cache if the NQSConfig.INI parameter `POPULATE_AGGREGATE_ROLLUP_HITS` has been set to `YES`.
- Result set is too big.

  This situation occurs when you exceed the size set in `DATA_STORAGE_PATHS`, or if you have rows in excess of the number set in `MAX_ROWS_PER_CACHE_ENTRY`. See DATA_STORAGE_PATHS and MAX_ROWS_PER_CACHE_ENTRY.
- Query is cancelled. This can happen by explicit cancellation from Oracle BI Presentation Services or the Model Administration Tool, or implicitly through timeout.
- Oracle BI Server is clustered. Queries that fall into the 'cache seeding' family are propagated throughout the cluster. Other queries continue to be stored locally. Therefore, even though a query might be put into the cache on Oracle BI Server node 1, it might not be on Oracle BI Server node 2.

Level 4 of query logging is the best tool to diagnose whether the Oracle BI Server compiler intended to add the entry into the query result cache. See Configure Query Logging.

MAX_SUBEXPR_SEARCH_DEPTH

Lets you configure how deep the hit detector looks for an inexact match in an expression of a query. The default is `5`.

For example, at level 5, a query on the expression `SIN(COS(TAN(ABS(ROUND(TRUNC(profit))))))` misses on `profit`, which is at level 7. Changing the search depth to 7 opens up `profit` for a potential hit.

Example: `MAX_SUBEXPR_SEARCH_DEPTH = 7;`

DISABLE_SUBREQUEST_CACHING

When set to `YES`, disables caching at the subrequest (subquery) level.

The default value is `NO`.

Caching subrequests improves performance and the cache hit ratio, especially for queries that combine real-time and historical data. In some cases, however, you might disable subrequest caching, such as when other methods of query optimization provide better performance.

Example: `DISABLE_SUBREQUEST_CACHING = NO;`

CACHE_FILE_BUFFER_SIZE

Specifies the amount of memory used to temporarily store the cache file when writing to the disk.

The default value is 128 KB.

Example: `CACHE_FILE_BUFFER_SIZE = 128;`

GLOBAL_CACHE_STORAGE_PATH

The `GLOBAL_CACHE_STORAGE_PATH` parameter can be managed by Fusion Middleware Control or by editing NQSConfig.INI.

> **Note:**
>
> The **Global cache path** and **Global cache size** options on the Performance tab of the Configuration page in Fusion Middleware Control correspond to the `GLOBAL_CACHE_STORAGE_PATH` parameter. See Use Fusion Middleware Control to Set Global Cache Parameters.

In a clustered environment, Oracle BI Servers can be configured to access a shared cache that is referred to as the global cache. The global cache resides on a shared file system storage device and stores seeding and purging events and the result sets that are associated with the seeding events.

This parameter specifies the physical location for storing cache entries shared across clustering. This path must point to a network share. All clustering nodes share the same location.

You can specify the size in KB, MB, or GB, or enter a number with no suffix to specify bytes.

Syntax: `GLOBAL_CACHE_STORAGE_PATH = "`*`directory name`*`" SIZE;`

Example: `GLOBAL_CACHE_STORAGE_PATH = "C:\cache" 250 MB;`

MAX_GLOBAL_CACHE_ENTRIES

The maximum number of cache entries stored in the location that is specified by `GLOBAL_CACHE_STORAGE_PATH`.

Example: `MAX_GLOBAL_CACHE_ENTRIES = 1000;`

CACHE_POLL_SECONDS

The interval in seconds that each node polls from the shared location that is specified in `GLOBAL_CACHE_STORAGE_PATH`.

Example: `CACHE_POLL_SECONDS = 300;`

CLUSTER_AWARE_CACHE_LOGGING

Turns on logging for the cluster caching feature.

Used only for troubleshooting. The default is `NO`.

Example: `CLUSTER_AWARE_CACHE_LOGGING = NO;`

General Section Parameters

The General section contains general server default parameters, including localization and internationalization, temporary space and memory allocation, and other default parameters used to determine how data is returned from the Oracle BI Server to a client.

> **Note:**
>
> The settings for the parameters `LOCALE`, `SORT_ORDER_LOCALE`, `SORT_TYPE` and `CASE_SENSITIVE_CHARACTER_COMPARISON`, described in the following topics, are interrelated. They help determine how the Oracle BI Server sorts data.

LOCALE

Specifies the locale in which data is returned from the server. This parameter also determines the localized names of days and months.

To successfully run Oracle Analytics Server, ensure that you configure the appropriate locales on the operating system for the language in which you run the applications. In some cases, you might install additional content on the system to support the locale. The Oracle BI Server sets the C-runtime locale during the server startup. Some locale- and language-related settings are interrelated and help determine how the Oracle BI Server sorts data. Ensure that the settings for the following parameters work together:

- `LOCALE`
- `SORT_ORDER_LOCALE`
- `SORT_TYPE`
- `CASE_SENSITIVE_CHARACTER_COMPARISON`

Valid platform-independent values for `LOCALE` and `SORT_ORDER_LOCALE` are:

- Arabic
- Chinese
- Chinese-traditional
- Croatian
- Czech
- Danish
- Dutch
- English-USA
- Finnish
- French
- German
- Greek
- Hebrew
- Hungarian
- Italian
- Japanese
- Korean
- Norwegian
- Polish
- Portuguese
- Portuguese-Brazilian
- Romanian
- Russian
- Slovak
- Spanish
- Swedish
- Thai
- Turkish

For information about language extensions, see Localize Oracle Analytics Server.

SORT_ORDER_LOCALE

Used to help determine whether the Oracle BI Server can function-ship (push down) an `ORDER BY` clause to a relational database.

`ORDER BY` clauses are used in sorting.

Every database that's defined in the Physical layer in the Model Administration Tool has a features table associated with it. If you want to override the default value in the Features table for a particular type of relational database, then you must do so for all occurrences of it in the Physical layer.

In the Model Administration Tool, the Features table in the Features tab of the Database dialog specifies the features and functions that the relational database supports. The settings for `SORT_ORDER_LOCALE` in the Features table and in the NQSConfig.INI file match only if the database and the Oracle BI Server sort data in the same way.

For the relational database and the Oracle BI Server to sort data the same way, they must be in agreement on the parameters that are shown in the following table.

Functional Category	Specific Parameters
Base language	`LOCALE`
Base language	`SORT_ORDER_LOCALE` The default value for `SORT_ORDER_LOCALE` in both the Features table and in the NQSConfig.INI file is `english-usa`. If the Oracle BI Server and the database sort data differently, then the Features table entry `SORT_ORDER_LOCALE` for the database must be set to a different value than `english-usa`. Otherwise, the different data sort methods clash. The `LOCALE` and `SORT_ORDER_LOCALE` parameters accept platform-independent names only. See the list provided in LOCALE for details.
Case	`CASE_SENSITIVE_CHARACTER_COMPARISON`
Binary versus linguistic comparison	`SORT_TYPE`

If the `SORT_ORDER_LOCALE` setting in the actual data source doesn't match the `SORT_ORDER_LOCALE` setting in the Features tab of the Database dialog in the semantic model, then result sets might not be correct. If the settings don't match, then incorrect answers can result when using multi-database joins, or errors can result when using the Union, Intersect, and Except operators, which all rely on consistent sorting between the back-end data source and the Oracle BI Server.

If the `SORT_ORDER_LOCALE` setting in NQSConfig.INI doesn't match the `SORT_ORDER_LOCALE` setting in the Features tab of the Database dialog in the semantic model, then query performance might be negatively impacted. However, this situation doesn't affect the correctness of the result set.

```
SORT_ORDER_LOCALE = "english-usa";
```

SORT_ORDER_LOCALE on Linux Operating Systems

The Oracle BI Server sets the C-runtime locale during server startup.

A value for the setting is specified using the `SORT_ORDER_LOCALE` entry in the NQSConfig.INI file. See Set Locale Parameters on the Oracle BI Server.

SORT_TYPE

Specifies the type of sort to perform.

The default value is `BINARY`. Binary sorts are faster than nonbinary sorts.

Valid values are `BINARY` and `DEFAULT`. If you specify `DEFAULT`, then a nonbinary sort is performed; this yields better sort results for data that contains accented characters.

Example: `SORT_TYPE = "BINARY";`

CASE_SENSITIVE_CHARACTER_COMPARISON

Specifies whether the Oracle BI Server differentiates between uppercase and lowercase characters when performing comparison operations.

Valid values are `ON` and `OFF`. When set to `OFF`, case is ignored. When set to `ON`, case is considered for comparisons. This parameter is set to `ON` by default. For binary sorts, case sensitivity for the server and for the relational database should be set the same way.

For information about how this parameter relates to the case setting in Oracle BI Server, see Make Advanced Configuration Changes for Presentation Services.

This setting only applies to the internal comparisons of the Oracle BI Server for caching and aggregation. Case sensitivity is a function of database operations and is set at the database level. The `CASE_SENSITIVE_CHARACTER_COMPARISON` parameter enables the Oracle BI Server to match the functions of the back-end database. The following operators are affected:

- Order By
- Group By
- Distinct
- Join
- comparisons (<, >, =, <=, >=, <>)

For example, consider the following three terms:

- ACME
- DELTA
- acme

An `ORDER BY` with `CASE_SENSITIVE_CHARACTER_COMPARISON` set to `ON` results in rows in the order shown in the preceding example. An ORDER BY with a case-insensitive setting results in ACME and acme appearing next to one another in the list.

If the term is case-sensitive and you perform a duplicate remove (`DISTINCT`), then the result is three rows. If the term isn't case-sensitive, then the `DISTINCT` result is two rows.

Set `CASE_SENSITIVE_CHARACTER_COMPARISON` to correspond with how the back-end database deals with case. For example, if the back-end database is case-insensitive, then configure the Oracle BI Server to be case-insensitive. If the Oracle BI Server and the back-end database aren't similarly case-sensitive, then some subtle problems can result.

For an example of `CASE_SENSITIVE_CHARACTER_COMPARISON` applied to aggregation, a case-sensitive database has the following tuples (or rows):

```
Region  Units
 WEST     1
 west     1
 West     1
```

With `CASE_SENSITIVE_CHARACTER_COMPARISON` set to `ON`, the data is returned to the client the with the same results shown in the preceding table.

With `CASE_SENSITIVE_CHARACTER_COMPARISON` set to `OFF`, the data is again returned to the client the with the same results shown in the preceding table. There is no change because the Oracle BI Server hasn't done any character comparisons.

However, if SUM_SUPPORTED is set to OFF in the features table, the Oracle BI Server is forced to do a character comparison. The results of the query in this case are as follows:

```
Region  Units
 WEST     3
```

The reason for these results is that the Oracle BI Server has case-sensitive character comparison turned off, so it now treats the three tuples as the same value and aggregates them. In this case WEST = West = west. However, if you filter on the Region column, you would still see the regions WEST, West, and west; CASE_SENSITIVE_CHARACTER_COMPARISON doesn't affect filtering on a back-end database. The logic shown in the aggregation example applies to caching as well.

Because CASE_SENSITIVE_CHARACTER_COMPARISON is set in the NQSConfig.INI file, the parameter applies to all back-end databases in a semantic model. Therefore, set the parameter to match the case sensitivity of the dominant back-end database of the semantic model.

Example: CASE_SENSITIVE_CHARACTER_COMPARISON = ON;

NULL_VALUES_SORT_FIRST

Specifies if NULL values sort before other values (ON) or after (OFF).

ON and OFF are the only valid values. Ensure that the value of NULL_VALUES_SORT_FIRST conforms to the underlying database. If there are multiple underlying databases that sort NULL values differently, then set the value to correspond to the database that is used the most in queries.

Example: NULL_VALUES_SORT_FIRST = OFF;

DATE_TIME_DISPLAY_FORMAT

Specifies the format for how date/time stamps are input to and output from the Oracle BI Server. The default value is yyyy/mm/dd hh:mi:ss.

Example: DATE_TIME_DISPLAY_FORMAT = "yyyy/mm/dd hh:mi:ss";

How are the Date and Time Display Formats Used?

The property values specified by DATE_TIME_DISPLAY_FORMAT, DATE_DISPLAY_FORMAT, and TIME_DISPLAY_FORMAT determine the default format that the BI Server uses when converting TIMESTAMP, DATE, and TIME expressions to and from character data types such as VARCHAR and CHAR.

DATE_TIME_DISPLAY_FORMAT, DATE_DISPLAY_FORMAT, and TIME_DISPLAY_FORMAT determine how date or time conversion expressions such as CAST(<*chardata*> as TIMESTAMP), CAST(<*chardata*> as DATE), CAST(<*datetimeexpr*> AS VARCHAR(20)), and CAST(<*dateexpr*> AS CHAR(10)) work when CAST_SUPPORTED isn't enabled in the database.

When the CAST_SUPPORTED feature is enabled in the database, the date and time formats are determined by the database rather than the DATE_TIME_DISPLAY_FORMAT, DATE_DISPLAY_FORMAT, and TIME_DISPLAY_FORMAT properties.

These properties don't affect the format of the timestamps written to the obis1-diagnostic.log or the nqquery.log. The format of the timestamps written in the log files is fixed according to Oracle Fusion Middleware standards and can't be changed since many tools like Fusion

Middleware Control need to be able to parse the log files. These tools rely on the fact that the timestamps in the log files have a fixed format.

See DATE_DISPLAY_FORMAT and TIME_DISPLAY_FORMAT.

DATE_DISPLAY_FORMAT

Specifies the format for how dates are input to and output from the Oracle BI Server. The default value is yyyy/mm/dd.

> **Note:**
>
> Specify the year as either 2-digit (yy) or 4-digit (yyyy). Separators can be any character except y, m, or d.

Example: `DATE_DISPLAY_FORMAT = "yyyy/mm/dd";`

See How are the Date and Time Display Formats Used?.

TIME_DISPLAY_FORMAT

You can configure the way times are displayed or entered.

Specifies the format for how times are input to and output from the Oracle BI Server. The default value is hh:mi:ss.

Example: `TIME_DISPLAY_FORMAT = "hh:mi:ss";`

See How are the Date and Time Display Formats Used?.

WORK_DIRECTORY_PATHS

Specifies one or more directories for temporary space.

Each directory listed must be an existing, writable path name, with double quotation marks (") surrounding the path name. Specify mapped directories only.

You can specify either fully qualified paths, or relative paths. When you specify a path that does not start with "/" (on Linux) or "<drive>:" (on Windows), the Oracle BI Server assumes that the path is relative to the local writable directory. For example, if you specify the path "temp," then at runtime, the Oracle BI Server uses the following:

```
BI_DOMAIN/servers/obisn/tmp/obis_temp
```

Specify multiple directories with a comma-delimited list. Valid values are any relative path, or fully qualified path to an existing, writable directory. UNC path names and network mapped drives are allowed only if the service runs under a qualified user account.

For optimum performance, temporary directories must reside on high-performance storage devices. If you specify multiple directories, then ensure that they reside on different physical drives.

Syntax: `WORK_DIRECTORY_PATHS = "path_1" [, "path_2"{, "path_n"}];`

Example 1: `WORK_DIRECTORY_PATHS = "temp" ;`

Example 2: `WORK_DIRECTORY_PATHS = "D:\temp", "F:\temp";`

> **Note:**
>
> Specifying multiple directories for each drive does not improve performance because file I/O occurs through the same I/O controller. In general, specify only one directory for each disk drive. Specifying multiple directories on different drives improves the overall I/O throughput of the Oracle BI Server because internally, the processing files are allocated using a round-robin algorithm that balances the I/O load across the given disk drives.

WORK_FILE_COMPRESSION_LEVEL

Use this parameter for Oracle BI Server internal temporary file tuning.

This parameter uses the compression library to compress the temporary working files.

For example, WORK_FILE_COMPRESSION_LEVEL = 2;

ENABLE_COLUMNAR_STORAGE_FOR_WORK_FILE

Use this parameter for Oracle BI Server internal temporarily file tuning.

This parameter applies to the temporary file created for the aggregation operator.

For example, ENABLE_COLUMNAR_STORAGE_FOR_WORK_FILE = YES;

WORK_DIRECTORY_SIZE_GLOBAL_LIMIT

Use this parameter for Oracle BI Server internal temporarily file tuning.

This parameter specifies the directory size limit and works along with MAX_WORK_FILE_SIZE_PERCENT to ensure that the temporary file does not exceed a specified percentage of the global work directory size limit.

For example, WORK_DIRECTORY_SIZE_GLOBAL_LIMIT = 100 GB;

MAX_WORK_FILE_SIZE_PERCENT

Use this parameter for Oracle BI Server internal temporarily file tuning.

You can also set in the Console, see Performance and Compatibility Options in Configure Advanced System Settings in the Console.

This parameter works with WORK_DIRECTORY_SIZE_GLOBAL_LIMIT to determine the maximum size that the temporarily file can grow to.

For example, set MAX_WORK_FILE_SIZE_PERCENT = 5;

VIRTUAL_TABLE_PAGE_SIZE

Several operations, such as sort, join, union, and database fetch, can require memory resources beyond those available to the Oracle BI Server.

To manage this condition, the server uses a virtual table management mechanism that provides a buffering scheme for processing these operations. When the amount of data exceeds the VIRTUAL_TABLE_PAGE_SIZE, the remaining data is buffered in a temporary file and placed in the virtual table as processing continues. This mechanism supports dynamic memory sizes and ensures that any row can be obtained dynamically for processing queries.

VIRTUAL_TABLE_PAGE_SIZE specifies the size of a memory page for Oracle BI Server internal processing. A larger value reduces I/O but increases memory usage, especially in a multiuser environment.

When VIRTUAL_TABLE_PAGE_SIZE is increased, I/O operations are reduced. Complex queries might use 20 to 30 virtual tables, while simple queries might not even require virtual tables. The default size of 128 KB is a reasonable size when one considers that the size for virtual paging in Windows is 64 KB. This parameter can be tuned depending on the number of concurrent users and the average query complexity. In general, setting the size larger than 256 KB does not yield a corresponding increase in throughput due to the 64 KB size limit of Windows system buffers, as each I/O still goes through the system buffers. 128 KB is also a reasonable value on Linux systems.

Example: `VIRTUAL_TABLE_PAGE_SIZE = 128 KB;`

USE_LONG_MONTH_NAMES

Specifies whether month names are returned as full names, such as JANUARY and FEBRUARY, or as three-letter abbreviations, such as JAN and FEB.

Valid values are YES and NO. Specify YES to have month names returned as full names, or NO to have months names returned as three-letter abbreviations. The default value is NO.

Example: `USE_LONG_MONTH_NAMES = NO;`

MEMORY_COMPACT_PERIOD_IN_SECONDS

Specifies the number of seconds that the Oracle BI Server waits between calls to its internal memory compaction routine.

The Oracle BI Server has a memory manager that does not return free memory to the system until the memory compaction routine is called in a background thread. Setting this parameter to a smaller value causes the Oracle BI Server to return unused memory to the system sooner at the expense of some additional CPU overhead. The default is 3600 seconds.

Example: `MEMORY_COMPACT_PERIOD_IN_SECONDS = 3600;`

USE_LONG_DAY_NAMES

Specifies whether day names are returned as full names, such as MONDAY and TUESDAY, or as three-letter abbreviations, such as MON and TUE.

Valid values are YES and NO. Specify YES to have day names returned as full names, or NO to have day names returned as three-letter abbreviations. The default value is NO.

Example: `USE_LONG_DAY_NAMES = NO;`

USE_UPPERCASE_MONTH_NAMES

Specifies whether month names are returned in mixed case, or in uppercase.

Valid values are YES and NO. Specify YES to have month names returned in uppercase, or NO to have month names returned in mixed case. The default value is NO.

Example: `USE_UPPERCASE_MONTH_NAMES = NO;`

USE_UPPERCASE_DAY_NAMES

Specifies whether day names are returned in mixed case, or in uppercase.

Valid values are `YES` and `NO`. Specify `YES` to have day names returned in uppercase, or `NO` to have day names returned in mixed case. The default value is `NO`.

Example: `USE_UPPERCASE_DAY_NAMES = NO;`

UPPERCASE_USERNAME_FOR_INITBLOCK

You can use the special syntax `:USER` in initialization blocks to pass through user names.

When this parameter is set to `YES`, then user names passed through initialization blocks using `:USER` are changed to all uppercase. Otherwise, case is maintained in the user names.

Example: `UPPERCASE_USERNAME_FOR_INITBLOCK = NO;`

Security Section Parameters

The security parameters specify default values for the Oracle BI Server security features.

See *Security Guide for Oracle Business Intelligence Enterprise Edition*.

DEFAULT_PRIVILEGES

Specifies the default semantic model object privilege granted to the AuthenticatedUser application role, which is the default application role associated with any new semantic model object.

In effect, this setting specifies the default level of object security in the Presentation layer of the semantic model for new objects that don't have other explicit security settings.

Note that the AuthenticatedUser application role means "any authenticated user." This role is internal to the semantic model.

Valid values are `NONE` and `READ`. The default value is `READ`. Note that `NONE` corresponds to the **No Access** setting in the Permissions dialog in the Model Administration Tool.

Example: `DEFAULT_PRIVILEGES = READ;`

PROJECT_INACCESSIBLE_COLUMN_AS_NULL

Controls how security-sensitive columns are displayed to unauthorized users. If this parameter is set to `YES`, then a `NULL` expression replaces the original column expression in the query and secured columns are hidden from unauthorized users in analyses.

If this parameter is set to `NO`, then when a user attempts to run a report that contains a secured column the user isn't authorized to see, an unresolved column error occurs.

The default value is `YES`.

Example: `PROJECT_INACCESSIBLE_COLUMN_AS_NULL = YES;`

IGNORE_LDAP_PWD_EXPIRY_WARNING

Determines whether users can log in even when the LDAP server issues a password expiration warning.

Valid values are `YES` and `NO`. Uncomment this parameter and specify `YES` to enable users to log in when the LDAP server issues a password expiration warning, or specify `NO` to reject user logins when the warning is issued. The default value is `NO`.

After user passwords have actually expired in the LDAP server, users can't log in, regardless of the value of this parameter.

Example: `IGNORE_LDAP_PWD_EXPIRY_WARNING = NO;`

MAX_AUTHENTICATION_TIME

Specifies the number of seconds that the Oracle BI Server is allocated to run initialization blocks before the user's login attempt times out. If a timeout happens, then the Oracle BI Server user is prompted to log in again.

This setting applies to the accumulated run time for all the initialization blocks. Suppose this value is set to ten minutes (600 seconds) and there are ten initialization blocks that the Oracle BI Server needs to run. If after running the fifth initialization block the ten minute login maximum is exceeded, then the Oracle BI Server does not run the remaining five initialization blocks and rejects the login attempt.

Example: `MAX_AUTHENTICATION_TIME = 600;`

INIT_BLOCK_LOG_TIME_THRESHOLD

Specifies a threshold in seconds for initialization block execution, which if exceeded, the Oracle BI Server logs the time of execution. This might provide a warning of possible initialization block design problems.

Example: `INIT_BLOCK_LOG_TIME_THRESHOLD = 60;`

NUM_INIT_BLOCK_THREADS_PER_USER

Specifies the number of initialization block threads that the Oracle BI Server allocates for each user.

The default is one thread.

Example: `NUM_INIT_BLOCK_THREADS_PER_USER = 1;`

SSL

This parameter, along with the remaining parameters in this section, relate to Secure Sockets Layer (SSL) communication between Oracle Analytics Server components.

The default setting for `SSL` is `NO`.

SSL_CERTIFICATE_FILE

Specifies the directory path to the certificate file.

For components acting as SSL servers, such as Oracle BI Server and Oracle BI Scheduler, this is the Server Certificate file name. For client components, such as Oracle Analytics Server ODBC Client Data Source, this is the Client Certificate file name.

Example (Server): `SSL_CERTIFICATE_FILE = "servercert.pem";`

Example (Client): `SSL_CERTIFICATE_FILE = "client-cert.pem";`

SSL_PRIVATE_KEY_FILE

Specifies the private key file.

For server components, this is the Server Private Key file name. For client components, this is the Client Private Key file name.

Example (Server): `SSL_PRIVATE_KEY_FILE = "serverkey.pem";`

Example (Client): `SSL_PRIVATE_KEY_FILE = "client-key.pem";`

SSL_PK_PASSPHRASE_FILE

Specifies the private key passphrase file name.

Example: `SSL_PK_PASSPHRASE_FILE = serverpwd.txt;`

SSL_PK_PASSPHRASE_PROGRAM

Specifies the private key passphrase program executable file name.

Example: `SSL_PK_PASSPHRASE_PROGRAM = sitepwd.exe;`

SSL_VERIFY_PEER

This parameter has been deprecated.

The SSL_VERIFY_CLIENTS and SSL_VERIFY_SERVERS parameters replace comparable functionality previously controlled by the SSL_VERIFY_PEER parameter.

SSL_VERIFY_SERVERS

Specifies whether to verify server certificates when acting as a client (that is, when the Oracle BI Server is calling the BI Security Service).

The default value is YES.

Example: `SSL_VERIFY_SERVERS = YES;`

SSL_VERIFY_CLIENTS

Specifies whether to verify client certificates when acting as a server (that is, when the Oracle BI Server is receiving calls from clients such as Presentation Services).

The default value is NO.

Example: `SSL_VERIFY_CLIENTS = NO;`

SSL_CA_CERTIFICATE_DIR

Specifies the path of the trusted CA Certificate that is used to verify the server or client certificate when Verify Peer is set to `YES`.

Takes effect only when client authentication is required.

Example: `SSL_CA_CERTIFICATE_DIR = "CACertDir";`

SSL_CA_CERTIFICATE_FILE

Specifies the name of the trusted CA Certificate that is used to verify the server or client certificate when Verify Peer is set to `YES`.

Takes effect only when client authentication is required.

Example: `SSL_CA_CERTIFICATE_FILE = "CACertFile";`

SSL_TRUSTED_PEER_DNS

Specifies individual named clients that are allowed to connect by Distinguished Name (DN).

The DN identifies the entity that holds the private key that matches the public key of the certificate.

Example: `SSL_TRUSTED_PEER_DNS = "";`

SSL_INTERNAL_CA_CERTIFICATE_FILE

Specifies the internal CA certificate file name.

Example: `SSL_INTERNAL_CA_CERTIFICATE_FILE = "InternalCACertFile";`

SSL_INTERNAL_TRUSTED_PEER_DNS

Specifies the internal trusted peer DNS name.

Example: `SSL_INTERNAL_TRUSTED_PEER_DNS = "";`

SSL_WEBSERVER_CA_CERTIFICATE_FILE

Specifies the web server CA certificate file name.

Example: `SSL_WEBSERVER_CA_CERTIFICATE_FILE = "WebServerCACertFile";`

SSL_WEBSERVER_TRUSTED_PEER_DNS

Specifies the web server trusted peer DNS name.

Example: `SSL_WEBSERVER_TRUSTED_PEER_DNS = "";`

SSL_CERT_VERIFICATION_DEPTH

The depth of the certificate chain. A depth of one means a certificate has to be signed by a trusted CA.

A depth of two means the certificate was signed by a CA that was further verified by a CA. The default value is `9`.

Example: `SSL_CERT_VERIFICATION_DEPTH = 9;`

SSL_CIPHER_LIST

A list of permitted cipher suites that the server uses.

The default is empty string, which is equivalent to "ALL."

You must set this parameter only when you want to use a cipher suite other than the default choice.

Example: `SSL_CIPHER_LIST = "EXP-RC2-CBC-MD5";`

Server Section Parameters

The parameters in the Server section define defaults and limits for the Oracle BI Server.

READ_ONLY_MODE

Permits or forbids changing semantic modeler files when using the Model Administration Tool.

> **Note:**
>
> The `READ_ONLY_MODE` parameter can be set in Fusion Middleware Control or by editing NQSConfig.INI.
> The **Disallow RPD Updates** option on the Performance tab of the Configuration page in Fusion Middleware Control corresponds to the `READ_ONLY_MODE` parameter.
> See Use Fusion Middleware Control to Disallow Semantic Model Updates.

When the `READ_ONLY_MODE` is set to `NO`, the default, you can edit the semantic model.

When this parameter is set to `YES`, it prevents the Model Administration Tool from making any changes to semantic model files. When the Model Administration Tool opens the semantic model, a message informs the user that the semantic model is read-only. If this parameter is set to `NO`, then the Model Administration Tool can make changes to the semantic model.

Even when `READ_ONLY_MODE` is set to `NO`, there are still situations when Model Administration Tool opens semantic models in read-only mode. For example, a semantic model might open as read-only if an offline Oracle BI Server held a lock on the semantic model at the time the Oracle BI Server started.

In addition, the Model Administration Tool also opens in read-only mode when Oracle Analytics Server has been clustered, and the Model Administration Tool is connected in online mode to a secondary node. This occurs because the primary node holds a lock on the semantic model. To avoid this situation when running in a clustered environment, ensure that the Oracle BI

Server ODBC DSN that is used by the Model Administration Tool has been configured to point to the cluster controller rather than to a particular Oracle BI Server.

MAX_SESSION_LIMIT

Specifies the maximum number of concurrent connections that are allowed by the server.

When this number is exceeded, the server refuses the connection request.

The limit is 65,535 connections.

Example: `MAX_SESSION_LIMIT = 2000;`

About the MAX_SESSION_LIMIT and SERVER_THREAD_RANGE Parameters

The size of the connection pool determines the number of available Oracle BI Server connections and the number of available threads for processing physical queries. A logical query might generate multiple physical queries, each of which could go to different connections.

The Oracle BI Server creates server threads up to the specified maximum using the parameter `SERVER_THREAD_RANGE`. All the threads that are available at any time are used to process queries from one or more sessions as needed.

Typically, the number of sessions that is specified by `MAX_SESSION_LIMIT` is larger than the number of available threads that is specified by `SERVER_THREAD_RANGE`.

In summary:

- `MAX_SESSION_LIMIT` specifies the number of sessions that can be connected to the Oracle BI Server, even if inactive. The sessions and the corresponding queries are queued to the threads for processing as they become available.
- The size of the connection pool specifies the number of threads and connections that process physical queries.
- `SERVER_THREAD_RANGE` specifies the number of threads that process the logical queries, or in other words, the number of queries that can be active in the Oracle BI Server at any time.

MAX_REQUEST_PER_SESSION_LIMIT

Specifies the maximum number of logical requests per session. This is how many open requests there are, per session, at the same time.

The limit is 65,535 logical requests per session.

> **Note:**
>
> Usually, individual users have only one open request for each session at the same time. Application programs and Oracle BI Presentation Services, however, typically have multiple requests open at the same time. In general, the default value of 5000 is sufficient for most environments, but tune this parameter based on the application environment and the client tools in use.

Example: `MAX_REQUEST_PER_SESSION_LIMIT = 5000;`

SERVER_THREAD_RANGE

Thread allocation configuration information is recorded for each server request.

For each Oracle BI Server request, `SERVER_THREAD_RANGE` specifies configuration information for thread allocation. The lower number in the range specifies the number of threads that is initially allocated, and the larger number in the range specifies the maximum number of threads to be allocated. The thread pool grows and shrinks in 5-thread increments until the upper or lower bound is reached. If there are fewer threads than sessions, then sessions share the available number of threads on a first come-first served basis.

Although setting both values to the same number maximizes the benefits of thread pooling, there is a cost associated with doing so. If you set the lower boundary and the upper boundary to the same number, then that number of threads is always allocated, which consumes stack space.

Example: `SERVER_THREAD_RANGE = 10-200;`

See About the MAX_SESSION_LIMIT and SERVER_THREAD_RANGE Parameters for related information.

SERVER_THREAD_STACK_SIZE

Specifies the memory stack size that is allocated for each server thread.

A value of 0 sets the stack size as 1 MB per server thread (64-bit systems).

The default value is 0. If you change this value, then ensure that the value that you provide is appropriate for the memory resources that are available on the system.

Example: `SERVER_THREAD_STACK_SIZE = 0;`

DB_GATEWAY_THREAD_RANGE

Specifies the minimum and maximum number of threads in the Oracle Analytics Server Database Gateway thread pool, according to `SERVER_THREAD_RANGE`.

The default value is 40-200.

Example: `DB_GATEWAY_THREAD_RANGE = 40-200;`

DB_GATEWAY_THREAD_STACK_SIZE

Specifies the memory stack size that is allocated for each Oracle Analytics Server Database Gateway thread. A value of 0 sets the stack size as 1 MB per server thread (64-bit systems).

The default value is 0. If you change this value, then ensure that the value that you provide is appropriate for the memory resources that are available on the system.

Example: `DB_GATEWAY_THREAD_STACK_SIZE = 0;`

HTTP_CLIENT_THREAD_RANGE

Specifies the minimum and maximum number of threads in the thread pool that the Oracle BI Server uses for reading and writing data using the HTTP client wrapper.

The default value is 0-100.

Example: `HTTP_CLIENT_THREAD_RANGE = 0-100;`

HTTP_CLIENT_THREAD_STACK_SIZE

Specifies the memory stack size that is allocated for each thread that is specified in HTTP_CLIENT_THREAD_RANGE. A value of 0 sets the stack size as 1 MB per thread (64-bit systems).

The default value is 0. If you change this value, then ensure that the value that you provide is appropriate for the memory resources that are available on the system.

Example: `HTTP_CLIENT_THREAD_STACK_SIZE = 0;`

MAX_EXPANDED_SUBQUERY_PREDICATES

Controls the maximum number of values that can be populated by a subquery when it is expanded. The default is 8,192 values. The Oracle BI Server generates an error if this limit is exceeded.

The Oracle BI Server syntax supports various kinds of subqueries, including IN and COMPARISON subqueries. In some cases, the Oracle BI Server must run the subquery and convert it into values (for example, when the database features IN_SUPPORTED/ IN_SUBQUERY_SUPPORTED and COMPARISON_SUBQUERY are turned off in the database features table). When the Oracle BI Server converts subqueries into value lists, `MAX_EXPANDED_SUBQUERY_PREDICATES` is used to monitor the maximum number of values from the result set of the subquery.

Note that there is also a database feature setting called `MAX_ENTRIES_PER_IN_LIST`. This value is set according to how many literals can be supported by the given data source. If this limit is exceeded, then the Oracle BI Server breaks the `IN` list into smaller ones and ORs them together. However, if the original `IN` list is too long, it might exceed the SQL statement length limit for that data source, resulting in a database error or failure. The `MAX_EXPANDED_SUBQUERY_PREDICATES` parameter provides a second limit to ensure that this situation does not occur.

Example: `MAX_EXPANDED_SUBQUERY_PREDICATES = 8192;`

> **Note:**
>
> If you see the error `[nQSError: 42029] Subquery contains too many values for the IN predicate`, you must increase the value of `MAX_EXPANDED_SUBQUERY_PREDICATES` (default value is 8192). Then retry your query.

MAX_QUERY_PLAN_CACHE_ENTRIES

Controls the number of cached logical query plans. The query plan cache is an internal performance feature that increases the speed of the query compilation process by caching plans for the most recently used queries.

The default value of this parameter is 1024. Don't raise this value without consulting Oracle Support Services.

Example: `MAX_QUERY_PLAN_CACHE_ENTRIES = 1024;`

MAX_QUERY_PLAN_CACHE_ENTRY_SIZE

Specifies the heap memory usage limit that is allocated for the single logical plan cache entry. The total plan cache memory usage per Oracle BI Server can be calculated by multiplying MAX_QUERY_PLAN_CACHE_ENTRY_SIZE times MAX_QUERY_PLAN_CACHE_ENTRY_SIZE.

The default value of 0 indicates the default limit of 1MB (64-bit platform). If you change this value, then ensure that the value that you provide is appropriate for the memory resources that are available on the system.

Example: `MAX_QUERY_PLAN_CACHE_ENTRY_SIZE = 0;`

MAX_DRILLDOWN_INFO_CACHE_ENTRIES

Controls the number of cached Action Link drilldown information entries per semantic model. This increases the speed of computing Action Link information by caching the Action Link information for the most recently used queries.

The default value of this parameter is 1024. Don't raise this value without consulting Oracle Support.

Example: `MAX_DRILLDOWN_INFO_CACHE_ENTRIES = 1024;`

MAX_DRILLDOWN_QUERY_CACHE_ENTRIES

Controls the number of cached Action Link query entries per semantic model. This increases the speed of drilling down by caching the Action Link drilldown results for the most recently used queries.

The default value of this parameter is 1024. Don't raise this value without consulting Oracle Support.

Example: `MAX_DRILLDOWN_QUERY_CACHE_ENTRIES = 1024;`

INIT_BLOCK_CACHE_ENTRIES

Controls the number of initialization block result sets that are cached with row-wise initialization.

The cache key is the fully instantiated initialization block SQL.

The default value is `20`. Because this parameter affects internal operations for localized versions of Oracle Analytics Server, it's recommended that you don't change this value unless instructed to do so.

Example: `INIT_BLOCK_CACHE_ENTRIES = 20;`

CLIENT_MGMT_THREADS_MAX

Specifies the number of management threads to allocate for managing Oracle BI Server client/server communications. Each client process consumes a management thread.

The client/server communication method for Oracle BI Server is TCP/IP.

Because the default value of `5` is typically sufficient for server communications with clients, don't change the value of this parameter.

Example: `CLIENT_MGMT_THREADS_MAX = 5;`

DEFAULT_JOBQUEUE_SIZE_PER_THREAD

Specifies the number of jobs that are in the queue per thread.

The default is 100 jobs. When set to 0, there is no limit to the number of jobs in the queue per thread.

Example: `DEFAULT_JOBQUEUE_SIZE_PER_THREAD = 100;`

MAX_COLUMNS_IN_SELECT

Specifies the maximum number of columns in a `SELECT` statement, including all subtotaling expressions generated by Presentation Services. This limit applies to all `SELECT` statements including derived or leaf select blocks.

Setting this value to `0` does not represent unlimited. The limit that you set in this parameter applies to all users, including administrators, and all subject areas.

Example: `MAX_COLUMNS_IN_SELECT = 50;`

MAX_LOGICAL_DIMENSION_TABLES

A single presentation column might references multiple logical tables when the corresponding logical column is derived from multiple logical tables.

Also, multiple presentation tables might reference the same logical table. For example, suppose a query requests multiple logical tables such as EmployeeCity, EmployeeRegion, and EmployeeCountry. In this example, the table count is three even though all tables reference the same dimension.

Hidden dimension attributes are include in the total number of logical dimension tables.

Setting this value to `0` does not represent unlimited. The limit that you set in this parameter applies to all users, including administrators, and all subject areas.

Example: `MAX_LOGICAL_DIMENSION_TABLES = 30;`

MAX_LOGICAL_FACT_TABLES

Specifies the maximum number of logical fact tables that display in a single leaf logical request.

This parameter also applies to implicit fact measures added by the Oracle BI Server. Suppose this parameter is set to `0` and the query requests two dimensions which invokes the implicit fact measure. The query fails because the logical fact table limit was exceeded.

Hidden fact attributes are include in the total number of logical fact tables.

Note that setting this value to `0` does not represent unlimited. The limit that you set in this parameter applies to all users, including administrators, and all subject areas.

Example: `MAX_LOGICAL_FACT_TABLES = 5;`

MAX_LOGICAL_MEASURES

Specifies the maximum number of unique logical measure columns, that is the unique dimension aggregations defined in the logical layer in a single logical request.

Some measures might be referenced multiple times in a single query, but are counted once. Measures that are based on the same physical attribute and aggregation rules but with different level-based setup are counted as different measures. For example, EmployeeCountry.Revenue is derived from Sales.Revenue with its level set to COUNTRY on the Product-Region dimension, but it is counted as a measure different from Sales.Revenue.

Hidden fact attributes are included in the total number of logical measures.

Note that setting this value to `0` does not represent unlimited. The limit that you set in this parameter applies to all users, including administrators, and all subject areas.

Example: `MAX_LOGICAL_MEASURES = 15;`

MAX_SET_OPERATION_BLOCKS

Specifies the maximum number of union, intersect, or minus blocks that display in an incoming SQL query. A query with a set operator contains at least two query blocks.

Every query must have at least one query block. If you specify `0` in this parameter, then the Oracle BI Server doesn't run a query. If you specify `1` in this parameter, then only queries that don't use set operators, and therefore contain only one query block, are included in the query.

The limit that you set in this parameter applies to all users, including administrators, and all subject areas.

Example: `MAX_SET_OPERATION_BLOCKS = 15;`

DEFAULT_DB_MAX_EXEC_TIME

Determines the maximum time in seconds that a database query runs.

If the execution time of a database query exceeds the number of seconds set for this parameter, the error message `[nQSError: 60009] The user request exceeded the maximum query governing execution time.` is displayed, and the query is terminated. If query execution time is less than the number of seconds set for this parameter, then the query completes and no error message is displayed. The default value is 0, and the maximum value is unlimited.

Example: `DEFAULT_DB_MAX_EXEC_TIME = 60;`

QUERY_LIMIT_WARNING_INSTEAD_OF_ERROR

Determines if an error message displays when the logical query limits are exceeded.

If this parameter is set to `OFF` and the logical query limits are exceeded, then the Oracle Analytics Server displays an error message and terminates the remainder of the query. If this parameter is set to `ON` and the logical query limits are exceeded, then the query completes and no error message displays, but a warning message indicating that the threshold was exceeded is logged in the obis1-diagnostic.log file.

Example: `QUERY_LIMIT_WARNING_INSTEAD_OF_ERROR = OFF;`

RPC_SERVICE_OR_PORT

Specifies the IP address and port number on which the Oracle BI Server listens.

> **Note:**
>
> The `RPC_SERVICE_OR_PORT` parameter can be set by editing NQSConfig.INI. Setting the port range overrides the `RPC_SERVICE_OR_PORT` parameter.

You can specify an IP address and port number in the form *ip_address*:*port*, or you can specify a port number only.

When you specify an IP address and port number, the Oracle BI Server binds to the specified IP address.

When you specify a port number only, the IP address is set by default to 0.0.0.0, which causes the Oracle BI Server to listen on all IP addresses on that computer.

When you specify an IP address only, the port value defaults to `9703`.

When using the Oracle Analytics Server ODBC wizard to configure ODBC data sources for the Oracle BI Server, ensure that the port number that is specified in the **Port** field on the Enter Logon Information screen matches the port number that is specified here. If you change the port number in the configuration file, then ensure that you reconfigure any affected ODBC data sources to use the new port number.

Example1: `RPC_SERVICE_OR_PORT = 9703;`

Example2: `RPC_SERVICE_OR_PORT = 127.0.0.1:9703;`

LISTEN_ADDRESS

This parameter is reserved for a future update.

LISTEN_PORT

This parameter is reserved for a future update.

ENABLE_DB_HINTS

Enables optional hints to be passed along with a SQL statement to an Oracle Database.

The default value is `YES`.

Example: `ENABLE_DB_HINTS = YES;`

PREVENT_DIVIDE_BY_ZERO

Controls the behavior for when a division by zero occurs.

When set to `YES`, then a `NULL` value is returned. When set to `NO`, then the query is terminated and an appropriate error is returned to the user.

Example: PREVENT_DIVIDE_BY_ZERO = YES;

CLUSTER_PARTICIPANT

Specifies whether the Oracle BI Server that is using this configuration file is a member of an Oracle BI Server cluster.

> **Note:**
>
> The CLUSTER_PARTICIPANT parameter can be set by editing NQSConfig.INI.
>
> All Oracle Analytics Server deployments are designed to run the Cluster Controller, even if they are single-node deployments. Because of this design, always set CLUSTER_PARTICIPANT to YES.

Valid values are YES and NO. The default value is YES.

When CLUSTER_PARTICIPANT is set to YES, this server must have a valid, configured ClusterConfig.xml file in the following location:

```
BI_DOMAIN/config/fmwconfig/biconfig/core
```

For more information, see the information about the ClusterConfig.xml file in Deploy for High Availability.

Example: CLUSTER_PARTICIPANT = YES;

DISCONNECTED

This parameter has been deprecated and is no longer used.

VARIABLE_VALUE_LIMIT

Variables can be truncated to a specific length.

Specifies the maximum length of returned session variable values when client tools call the NQSGetSessionValues() function.

Example: VARIABLE_VALUE LIMIT= 10;

For example, suppose VARIABLE_VALUE_LIMIT is set to 10 and the NQSGetSessionValues() function is called on a variable whose value is "1234567890ABCDE." The value is truncated to "1234567890".

EVALUATE_SUPPORT_LEVEL

Specifies whether the database functions EVALUATE, EVALUATE_ANALYTIC, EVALUATE_AGGR, and EVALUATE_PREDICATE can be issued by users.

You can also set in the Console, see Performance and Compatibility Options in Configure Advanced System Settings in the Console.

See Evaluate Functions in *Managing Metadata Repositories for Oracle Analytics Server* for more information.

By default, this parameter is set to 0, which means that all support for the `EVALUATE` family of functions is disabled. Set this parameter to 1 to enable users with the oracle.bi.server.manageRepositories permission to issue `EVALUATE` functions. Set this parameter to 2 to enable all users to issue `EVALUATE` functions.

Note the following:

- The EVALUATE_SUPPORT_LEVEL parameter controls the use of the EVALUATE family of database functions within analyses. Oracle recommends leaving EVALUATE_SUPPORT_LEVEL set to its default value of 0 to prevent the use of these functions within analyses. Setting EVALUATE_SUPPORT_LEVEL to a value of 1 or 2 enables users to insert arbitrary SQL expressions into an analysis using the Analysis editor, which potentially compromises data access security.
- The EVALUATE_SUPPORT_LEVEL parameter does not control use of the EVALUATE family of database functions within the semantic model.

Example: `EVALUATE_SUPPORT_LEVEL = 1;`

FMW_SECURITY_SERVICE_URL

Specifies the location where Oracle WebLogic Server is running so that the Oracle BI Server can locate the Oracle Fusion Middleware security service.

Example: `FMW_SECURITY_SERVICE_URL = "http://localhost:9704";`

FMW_SECURITY_SERVICE_MAX_NUMBER_OF_CONNECTIONS

Limits the number of connections from the Oracle BI Server to the Oracle Fusion Middleware security service to avoid overloading the Oracle WebLogic Server with too many connections. Don't change.

Example: `FMW_SECURITY_SERVICE_MAX_NUMBER_OF_CONNECTIONS = 2000;`

FMW_SECURITY_SERVICE_MAX_NUMBER_OF_RETRIES

Specifies the maximum number of times to attempt to connect to the Oracle Fusion Middleware security service.

Example: `FMW_SECURITY_SERVICE_MAX_NUMBER_OF_RETRIES = 0;`

ENABLE_NUMERIC_DATA_TYPE

Specifies whether to import decimal/numeric data from Oracle Database and TimesTen as DOUBLE (the default) or NUMERIC, which provides greater precision.

Set this parameter to `YES` to enable numeric support for Oracle Database and TimesTen data sources. Data imported into the semantic model from Oracle Database and TimesTen has decimal/numeric dataset to NUMERIC, and decimal/numeric SQL code that's entered by users is treated as NUMERIC. The data type of physical columns imported prior to changing this setting remains the same.

To leverage this configuration for queries run by the Oracle BI Server, enable the NUMERIC_SUPPORTED database feature in the Physical layer database object. See SQL

Features Supported by a Data Source in *Managing Metadata Repositories for Oracle Analytics Server* .

Note that decimal/numeric data from other database types is still mapped as DOUBLE, even when this parameter is set to `YES`. Also, a column in Oracle Database or TimesTen that is declared as DOUBLE instead of NUMBER is still imported as DOUBLE in Oracle Analytics Server, regardless of how this parameter is set.

Note the following:

- Numeric data types can be cast to other Number data types, and vice versa.
- Numeric data type support isn't available through the Oracle BI Server JDBC driver.
- There might be a performance overhead of enabling the numeric data type because of the higher number of bits for numeric data.

Example: `ENABLE_NUMERIC_DATA_TYPE = NO;`

ENDECA_SERVLET_URL

This parameter is reserved for a future update.

Example: `ENDECA_SERVLET_URL = "http://localhost:9500/EndecaIntegration/EndecaServlet"`

High Availability Parameters

The parameters in the High Availability section define defaults and limits use in a highly available configuration.

HA_DB_PING_PERIOD_MILLISECS

Specifies the number of milliseconds between two consecutive polls of every TimesTen database performed by the BI Server to ensure high availability.

Through this polling, the Oracle BI Server determines which TimesTen schemas are inactive, so that the Oracle BI Server can select which TimesTen aggregate tables to use for a query.

Example: `HA_DB_PING_PERIOD_MILLISECS = 60000;`

Dynamic Library Section Parameters

This section contains one entry for each dynamic link library (DLL) or set of shared objects that is used to make connections to the Oracle BI Server, for both Windows and Linux systems.

Syntax: *`logical_name`* = *`dynamic_library`*;

In this syntax:

- *`logical_name`*: A logical name for the dynamic link library. These logical names also appear in the Connection Pool dialog.
- *`dynamic_library`*: The name of the associated dynamic library. These libraries are located in:

  `ORACLE_HOME``/bi/bifoundation/server/bin`

> **Note:**
>
> Don't make any changes to this section unless instructed to do so by Oracle Support Services.

The following are the dynamic link libraries that are shipped with this update:

- ODBC200 = nqsdbgatewayodbc;
- ODBC350 = nqsdbgatewayodbc35;
- OCI8 = nqsdbgatewayoci8;
- OCI8i = nqsdbgatewayoci8i;
- OCI10g = nqsdbgatewayoci10g;
- DB2CLI = nqsdbgatewaydb2cli;
- DB2CLI35 = nqsdbgatewaydb2cli35;
- NQSXML = nqsdbgatewayxml;
- XMLA = nqsdbgatewayxmla;
- ESSBASE = nqsdbgatewayessbasecapi;
- OracleADF = nqsdbgatewayoracleadf;
- OracleADF_HTTP = nqsdbgatewayoracleadf;
- OracleCEP_HTTP = nqsdbgatewayoraclecep;
- HyperionADM = nqsdbgatewayadm;
- OracleWS = nqsdbgatewayoraclews;
- hadoop = nqsdbgatewayhadoop;
- timesten = nqsdbgatewaytimesten;
- timesten35 = nqsdbgatewaytimesten35;
- JAVADS = nqsdbgatewayjava
- CSV = nqsdbgatewaycsv

Usage Tracking Section Parameters

The usage tracking parameters define default values for the collection of usage tracking statistics on each logical query submitted to the Oracle BI Server.

The following table shows the names and descriptions of columns that are added to the usage tracking table and to the standalone usage tracking semantic model.

Name	Data Type	Description	Notes
SAW_DASHBOARD_PG	Varchar(150)	Page within Oracle BI Presentation Services dashboard	Null if not a dashboard request.
PRESENTATION_NAME	Varchar(128)	Name of the Presentation Catalog in Oracle BI Presentation Services	NA

Name	Data Type	Description	Notes
ERROR_TEXT	Varchar(250)	Error flag and reason text for queries that don't generate a cache entry, from back-end databases	Only applicable if `SUCCESS_FLG` is nonzero. Concatenates multiple messages; the application must parse the column contents.
RUNAS_USER_NAME	Varchar(128)	Impersonated User (the Proxy User that ran the query)	Null if the request isn't run as an impersonated user.

For more information about usage tracking, see Set Direct Insertion Parameters.

ENABLE

Enables or disables the collection of usage tracking statistics.

> **Note:**
>
> For new (non-upgraded) installations, the `ENABLE` parameter in the [USAGE_TRACKING] section can be changed by manually editing NQSConfig.INI
>
> See Set Direct Insertion Parameters.

Valid values are `YES` and `NO`. The default value is `NO`. When set to `NO`, statistics are not accumulated. When set to `YES`, statistics are accumulated for each logical query.

Example: `ENABLE = NO ;`

DIRECT_INSERT

Specifies whether statistics are inserted directly into a database table or written to a local file.

> **Note:**
>
> For new (non-upgraded) installations, the `DIRECT_INSERT` parameter can be changed by editing NQSConfig.INI.
>
> See Set Direct Insertion Parameters.

- When `DIRECT_INSERT` is set to `NO`, data is written to a flat file.
- When `DIRECT_INSERT` is set to `YES`, data is inserted into a table.

> **Note:**
>
> This parameter is operative only if the usage tracking parameter `ENABLE` is set to `YES`.

Because direct insertion into a database table is recommended, the default value is `YES`.

Certain other parameters become valid, depending whether `DIRECT_INSERT` is set to `YES` or to `NO`. These parameters are summarized in the table below and described in the following sections.

DIRECT_INSERT Setting	Parameters Used	Parameter Setting
NO	CHECKPOINT_INTERVAL_MINUTES	5
NO	FILE_ROLLOVER_INTERVAL_MINUTES	30
NO	CODE_PAGE	"ANSI"
YES	PHYSICAL_TABLE_NAME	"*Database*"."*Catalog*"."*Schema*"."*Table*" or "*Database*"."*Schema*"."*Table*"
YES	CONNECTION_POOL	"*Database*"."*Connection_Pool*"
YES	BUFFER_SIZE	10 MB
YES	BUFFER_TIME_LIMIT_SECONDS	5
YES	NUM_INSERT_THREADS	5
YES	MAX_INSERTS_PER_TRANSACTION	1
YES	JOBQUEUE_SIZE_PER_INSERT_THREADPOOL_THREAD	100
YES	THROW_INSERT_WHEN_JOBQUEUE_FULL	NO

STORAGE_DIRECTORY

Specifies the full path to the directory that is used to store usage tracking log files.

The directory listed must be a valid fully qualified, writable directory path name, with double quotation marks (") surrounding the path name. Specify mapped directories only.

Valid values are any fully qualified path name to an existing, writable directory.

Example: `STORAGE_DIRECTORY = "C:\Temp\UsageTracking";`

CHECKPOINT_INTERVAL_MINUTES

Specifies how often the usage tracking data is flushed to disk.

Setting this interval to a larger number increases the amount of data that might be lost if the server shuts down abnormally. Setting this interval lower incurs additional overhead.

The default is `5` minutes.

> **Note:**
>
> When the interval is set to `0`, the Oracle BI Server attempts to write usage tracking data to disk with minimal time between attempts. This can negatively affect server performance and is strongly discouraged.

Example: `CHECKPOINT_INTERVAL_MINUTES = 5;`

FILE_ROLLOVER_INTERVAL_MINUTES

Specifies the time, in minutes, before the current usage tracking log file is closed and a new file is created. For example, if this entry is set to 60 minutes, then 24 usage tracking log files are created each day.

The default is 30 minutes.

When the checkpoint interval equals or exceeds the rollover interval, only the rollover occurs explicitly; the checkpoint occurs implicitly only when the old usage tracking log file is closed.

> **Note:**
>
> When the checkpoint interval is set to `0`, the Oracle BI Presentation Services attempts to close current usage tracking log files and open new log files with minimal time between attempts. This can negatively affect server performance and result in a large number of usage tracking log files in the storage directory. Setting this interval to `0` is strongly discouraged.

Example: `FILE_ROLLOVER_INTERVAL_MINUTES = 240;`

CODE_PAGE

For multilingual semantic models, this specifies the type of output code page to use when writing statistics to disk.

Valid values include any valid code page number (such as 1252), and other globally recognized output code page types.

The default value is `ANSI`. The type depends upon the database loader being used. For example, to support multilingual semantic models for database loaders that are used by Oracle Database and DB2, specify `UTF8`. Enclose the value in double quotation marks. USC-2 is currently not supported.

Example: `CODE_PAGE = "ANSI";`

PHYSICAL_TABLE_NAME

Specifies the table in which to insert records that correspond to the query statistics. The table name is the fully qualified name as it appears in the Physical layer of the Model Administration Tool.

> **Note:**
>
> For new (non-upgraded) installations, the `PHYSICAL_TABLE_NAME` parameter can be updated by editing NQSConfig.INI.
>
> See Set Direct Insertion Parameters.

The general structure of this parameter depends on the type of database being used:

- For SQL Server, use the following general structure:

  ```
  PHYSICAL_TABLE_NAME = "Database"."Catalog"."Schema"."Table";
  ```

  Example:

  ```
  PHYSICAL_TABLE_NAME = "OracleBI Usage"."Catalog"."dbo"."S_NQ_ACCT";
  ```

  In the preceding example, the structure is as follows:

  - "Oracle BI Usage" represents the database component
  - "Catalog" represents the catalog component
  - "dbo" represents the schema component
  - "S_NQ_ACCT" represents the table name

- For Oracle Database, use the following general structure:

  ```
  PHYSICAL_TABLE_NAME = "Database"."Schema"."Table";
  ```

  Examples:

  ```
  PHYSICAL_TABLE_NAME = "OracleBI Usage"."DEV_BIPLATFORM"."S_NQ_ACCT";
  ```

  In the preceding example, the structure is as follows:

  - "Oracle BI Usage" represents the database component
  - "DEV_BIPLATFORM" represents the schema component
  - "S_NQ_ACCT" represents the table name

CONNECTION_POOL

Specifies the connection pool to use for inserting records into the usage tracking table.

This is the fully qualified name as it appears in the Physical layer of the Model Administration Tool.

Example: `CONNECTION_POOL = "OracleBI Usage"."Connection Pool";`

> **Note:**
>
> For new (non-upgraded) installations, the `CONNECTION_POOL` parameter can be changed by editing NQSConfig.INI.
>
> See Set Direct Insertion Parameters.

INIT_BLOCK_TABLE_NAME

Specifies the table in which to insert records that correspond to the initialization block statistics.

The table name is the fully qualified name as it appears in the Physical layer of the Model Administration Tool. The default table, S_NQ_INITBLOCK, is defined in the RCU schema.

Example: `INIT_BLOCK_TABLE_NAME = Database"."Catalog"."Schema"."Table";`

INIT_BLOCK_CONNECTION_POOL

Specifies the connection pool to use for inserting records into the initialization block usage tracking table.

The connection pool name is the fully qualified name as it appears in the Physical layer of the Model Administration Tool.

Example: `INIT_BLOCK_CONNECTION_POOL = Database"."Connection_Pool";`

BUFFER_SIZE

Specifies the amount of memory that is used to temporarily store insert statements.

The buffer enables the insert statements to be issued to the usage tracking table independently of the query that produced the statistics to be inserted. When the buffer fills up, then the statistics of subsequent queries are discarded until the insert threads service the buffer entries.

You can specify the size in KB or MB, or enter a number with no suffix to specify bytes.

Example: `BUFFER_SIZE = 10 MB;`

BUFFER_TIME_LIMIT_SECONDS

Specifies the maximum amount of time that an insert statement remains in the buffer before it is issued to the usage tracking table. This time limit ensures that the Oracle BI Presentation Services issues the insert statements quickly even during periods of extended quiescence.

Example: `BUFFER_TIME_LIMIT_SECONDS = 5;`

NUM_INSERT_THREADS

Specifies the number of threads that remove insert statements from the buffer and issue them to the usage tracking table. The number of threads must not exceed the total number of threads that are assigned to the connection pool.

Example: `NUM_INSERT_THREADS = 5;`

MAX_INSERTS_PER_TRANSACTION

Specifies the number of records to group as a single transaction when inserting into the usage tracking table, using the bulk insert API of the database where this is supported.

Increasing the number might slightly increase performance, but also increases the possibility of inserts being rejected due to deadlocks in the database.

Example: `MAX_INSERTS_PER_TRANSACTION = 1;`

JOBQUEUE_SIZE_PER_INSERT_THREADPOOL_THREAD

Specifies the maximum number of insert jobs that may be put into the job queue of a thread.

Example: `JOBQUEUE_SIZE_PER_INSERT_THREADPOOL_THREAD = 100;`

THROW_INSERT_WHEN_JOBQUEUE_FULL

You can configure the system to wait until there is space in the thread job queue to complete or run a job.

Specifies that the thread running the job stops and waits until the thread job queue is no longer full (when set to NO) or reject the new insert job (when set to YES).

Example: `THROW_INSERT_WHEN_JOBQUEUE_FULL = NO;`

SUMMARY_STATISTICS_LOGGING

You can enable or disable the logging statistics

> **Note:**
>
> For new (non-upgraded) installations, the `SUMMARY_STATISTICS_LOGGING` parameter can be changed by editing NQSConfig.INI.
>
> See Turning On Summary Advisor Logging in *Managing Metadata Repositories for Oracle Analytics Server* .

Enables or disables the collection of Summary Advisor logging statistics, as follows:

- Set this parameter to `YES` to enable Summary Advisor logging.
- Set this parameter to `LOG_OUTER_JOINT_QUERIES_ONLY` to enable Summary Advisor logging only for logical queries that contain outer joins. Consider using this option when the minor performance impact of enabling full Summary Advisor logging is a concern.
- Set this parameter to `NO` (the default) to disable Summary Advisor logging.

The Oracle BI Summary Advisor feature is only available when you're running Oracle Analytics Server on the Oracle Exalytics Machine. See Using Oracle BI Summary Advisor to Identify Query Candidates for Aggregation in *Managing Metadata Repositories for Oracle Analytics Server* .

Example: `SUMMARY_STATISTICS_LOGGING = YES;`

SUMMARY_ADVISOR_TABLE_NAME

You can specify the table where logging statistic records are stored.

> **Note:**
>
> For new (non-upgraded) installations, the `SUMMARY_ADVISOR_TABLE_NAME` parameter can be changed by manually editing NQSConfig.INI.
>
> See Turning On Summary Advisor Logging in *Managing Metadata Repositories for Oracle Analytics Server* .

Specifies the table in which to insert records that correspond to the Oracle BI Summary Advisor logging statistics. The table name is the fully qualified name as it appears in the Physical layer of the Model Administration Tool.

Example: `SUMMARY_ADVISOR_TABLE_NAME = "Orcl"."DEV_BIPLATFORM".`

`"S_NQ_SUMMARY_ADVISOR"`

Query Optimization Flags Section Parameters

Parameters in the Query Optimization Flags section can override the behavior of Oracle BI Server.

STRONG_DATETIME_TYPE_CHECKING

Use this parameter to relax strong type checking to prevent some date/time data type incompatibilities in queries from being rejected.

You can also set in the Console, see Performance and Compatibility Options in Configure Advanced System Settings in the Console.

For example, a query of the form "date/time op string-literal" technically contains a date/time data type incompatibility and would normally be rejected by the Oracle BI Server.

Valid values are `ON` and `OFF`. The default value is `ON`, which means that strong type checking is enabled and queries containing date/time data type incompatibilities are rejected. This is the recommended setting.

To relax the strong type checking, set the value to `NO`. Note that invalid queries or queries with severe date/time incompatibilities are still rejected. Note also that the query could still fail, for example, if the relational database implements a similar strong type checking.

Example: `STRONG_DATETIME_TYPE_CHECKING = ON;`

Query Optimizer Thread Pool Parameters

The Query Optimizer Thread Pool handles the complex queries where the query execution plan (rqTree) is too deep to put into one thread, which causes the stack to overflow.

The Query Optimizer Thread Pool uses parameters that enable you to divide the execution plan, letting each thread in that pool handle a part of the plan.

QUERY_OPTIMIZER_THREAD_RANGE

Use this parameter to specify the minimum and maximum number of threads in the Query Optimizer Thread Pool.

Example: `QUERY_OPTIMIZER_THREAD_RANGE = 0 - 100;`

QUERY_OPTIMIZER_THREAD_STACK_SIZE

Use this parameter to specify the stack size of the Query Optimizer Thread.

When you use the default value (0), the stack size is 256KB (in 32-bit mode), and 1MB (in 64-bit mode).

Example: `QUERY_OPTIMIZER_THREAD_STACK_SIZE = 0;`

QUERY_OPTIMIZER_MAX_THREAD_PER_QUERY

Use this parameter to specify the maximum number of Query Optimizer threads used per query.

The default value is 5.

Example: `QUERY_OPTIMIZER_MAX_THREAD_PER_QUERY = 5;`

Aggregate Persistence Section Parameters

Oracle Business Intelligence provides an aggregate persistence feature that automates the creation and loading of the aggregate tables and their corresponding Oracle Business Intelligence metadata mappings.

The parameters in this section relate to configuring and using the aggregate persistence feature.

AGGREGATE_PREFIX

Specifies the Domain Server Name for aggregate persistence.

The prefix must be between 1 and 8 characters long and must not have any special characters ('_' is allowed).

Example: `AGGREGATE_PREFIX = "SA_";`

AGGREGATE_THREAD_POOL_SIZE

Specifies the number of threads to be started for aggregate persistence.

Within each phase, relational loads are run in separate threads to improve the load performance. The default value is 5.

Example: `AGGREGATE_THREAD_POOL_SIZE = 5;`

AGGREGATE_AW_NAME

Specifies the name of the Analytic Workspace object that is created in the target Oracle Database.

The aggregate AW cubes and dimensions are created under this container.

Example: `AGGREGATE_AW_NAME = "OBI_AW";`

PREAGGREGATE_AW_CUBE

Specifies whether the system-generated AW cube for aggregate persistence must be fully solved.

The default value is `YES`. Note that a `YES` value significantly increases storage space usage.

Example: `PREAGGREGATE_AW_CUBE = YES;`

SUPPORT_ANALYTICAL_WORKSPACE_TARGETS

Specifies whether to turn on support for persisting aggregates in Oracle Analytic Workspaces.

The default is NO.

Example: `SUPPORT_ANALYTICAL_WORKSPACE_TARGETS = NO;`

JavaHost Section Parameters

The parameters in this section provide information about the computers on which the JavaHost process is running.

JAVAHOST_HOSTNAME_OR_IP_ADDRESSES

This parameter provides information about JavaHost connectivity. The default port value is `9810`.

> **Note:**
>
> The `JAVAHOST_HOSTNAME_OR_IP_ADDRESS` parameter can be updated by editing NQSConfig.INI.

Syntax: `JAVAHOST_HOSTNAME_OR_IP_ADDRESS = "`*`host_name1`*`:`*`port1`*`",`*`host_name2`*`:`*`port2`*`;`

Example: `JAVAHOST_HOSTNAME_OR_IP_ADDRESS = "MYHOST:9810";`

JAVAHOST_HOSTNAME_OR_IP_ADDRESSES_OVERRIDE

Specifies whether to override the JavaHost host name or IP address for connecting to data sources for Hyperion Financial Management when Oracle Analytics Server is installed on a non-Windows platform.

Hyperion Financial Management provides a client driver for only the Windows platform. You must have a JavaHost process for Oracle Analytics Server running on a Windows computer to access Hyperion Financial Management even if the main instance of Oracle Analytics Server is running on a non-Windows platform. In this case, the JAVAHOST_HOSTNAME_OR_IP_ADDRESSES_OVERRIDE parameter must be configured to indicate the JavaHost instance running on the Windows computer where the Hyperion Financial Management client is installed.

Syntax:
`JAVAHOST_HOSTNAME_OR_IP_ADDRESS_OVERRIDE = "`*`host_name1`*`:`*`port1`*`",`*`host_name2`*`:`*`port2`*`;`

Example: `JAVAHOST_HOSTNAME_OR_IP_ADDRESS_OVERRIDE = "MYHOST:9810";`

Datamart Automation Section Parameters

The parameters in this section are reserved for a future update.

ESSBASE_SERVER

This parameter is reserved for a future update.

DMA_DATABASE

This parameter is reserved for a future update.